

# CA Virtual Assurance for Infrastructure Managers

参考指南

版本 12.7.1



本文档包括内嵌帮助系统和以电子形式分发的材料（以下简称“文档”），其仅供参考，CA 随时可对其进行更改或撤销。

未经 CA 事先书面同意，不得擅自复制、转让、翻印、透露、修改或转录本文档的全部或部分内容。本文档属于 CA 的机密和专有信息，不得擅自透露，或除以下协议中所允许的用途，不得用于其他任何用途：(i) 您与 CA 之间关于使用与本文档相关的 CA 软件的单独协议；或者 (ii) 您与 CA 之间单独的保密协议。

尽管有上述规定，但如果您为本文档中所指的软件产品的授权用户，则您可打印或提供合理数量的本文档副本，供您及您的雇员内部用于与该软件相关的用途，前提是所有 CA 版权声明和标识必须附在每一份副本上。

打印或提供本文档副本的权利仅限于此类软件所适用的许可协议的有效期内。如果该许可因任何原因而终止，您应负责向 CA 书面证明已将本文档的所有副本和部分副本已退还给 CA 或被销毁。

在所适用的法律允许的范围内，CA 按照“现状”提供本文档，不附带任何保证，包括但不限于商品适销性、适用于特定目的或不侵权的默示保证。CA 在任何情况下对您或其他第三方由于使用本文档所造成的直接或间接的损失或损害都不负任何责任，包括但不限于利润损失、投资受损、业务中断、信誉损失或数据丢失，即使 CA 已经被提前明确告知这种损失或损害的可能性。

本文档中涉及的任何软件产品的使用均应遵照有关许可协议的规定且根据本声明中的条款不得以任何方式修改此许可协议。

本文档由 CA 制作。

仅提供“有限权利”。美国政府使用、复制或透露本系统受 FAR Sections 12.212、52.227-14 和 52.227-19(c)(1) - (2) 以及 DFARS Section 252.227-7014(b)(3) 的相关条款或其后续条款的限制。

版权所有 © 2012 CA。保留所有权利。此处涉及的所有商标、商品名称、服务标识和徽标均归其各自公司所有。

## CA Technologies 产品引用

本文档引用以下 CA Technologies 产品：

- CA eHealth®
- CA Embedded Entitlements Manager (CA EEM)
- CA IT Asset Manager (CA ITAM)
- CA IT Client Manager (CA ITCM)
- CA Network and Systems Management (CA NSM)
- CA Patch Manager
- CA Server Automation
- CA Service Desk Manager (CA SDM)
- CA Spectrum®
- CA SystemEDGE
- CA Systems Performance for Infrastructure Managers
- CA Virtual Assurance for Infrastructure Managers
- CA Software Delivery

## 联系技术支持

要获取在线技术帮助以及办公地址、主要服务时间和电话号码的完整列表，请联系技术支持：<http://www.ca.com/worldwide>。



# 目录

---

|                                         |               |
|-----------------------------------------|---------------|
| <b>第 1 章：简介</b>                         | <b>15</b>     |
| 本指南涵盖的内容.....                           | 15            |
| 读者.....                                 | 15            |
| 相关出版物.....                              | 15            |
| 约定.....                                 | 16            |
| <br><b>第 2 章： AutoShell</b>             | <br><b>19</b> |
| 关于 AutoShell .....                      | 19            |
| 管理器 Shell 和客户端 Shell.....               | 20            |
| 有效 AutoShell 用户 .....                   | 21            |
| 访问 AutoShell .....                      | 21            |
| 通用信息模型 (CIM) 对象 .....                   | 22            |
| 检索 CIM 类 .....                          | 22            |
| 从 CIM 对象检索祖先项和子项 .....                  | 22            |
| 检索 CIM 对象 .....                         | 23            |
| 调用 AutoShell .....                      | 25            |
| 使用交互模式.....                             | 27            |
| 使用远程交互模式.....                           | 28            |
| 批处理模式.....                              | 28            |
| 远程批处理模式.....                            | 29            |
| 字符串化.....                               | 29            |
| 将数据写入控制台 (stdout) .....                 | 30            |
| 从键盘读取数据（标准输入） .....                     | 31            |
| 访问操作系统.....                             | 32            |
| 使用 OSRedirect 类.....                    | 32            |
| 运行脚本.....                               | 33            |
| 使用 run-local 命令 .....                   | 33            |
| 使用 run-remote 命令 .....                  | 35            |
| AutoShell 核心参考 .....                    | 40            |
| 根据类别列出的命令、函数和类.....                     | 41            |
| 常规 AutoShell 命令 .....                   | 43            |
| AutoShell 功能 .....                      | 102           |
| AutoShell 类 .....                       | 125           |
| AutoShell 可加载模块的命令参考 .....              | 149           |
| CA VMware vCenter 服务器 AutoShell 命令..... | 149           |
| CA Microsoft Hyper-V AutoShell 命令 ..... | 188           |
| CA IBM LPAR AutoShell 命令 .....          | 283           |

|                                            |     |
|--------------------------------------------|-----|
| CA Oracle Solaris Zones AutoShell 命令 ..... | 307 |
| 部署和策略配置 AutoShell 命令 .....                 | 326 |

## 第 3 章： 命令行脚本

361

|                                                         |     |
|---------------------------------------------------------|-----|
| 命令行指令 (CLI) .....                                       | 361 |
| CA VMware vCenter Server CLI 命令 .....                   | 361 |
| dpmvc addesxhost 命令—添加 ESX 主机 .....                     | 362 |
| dpmvc addvmvdisk 命令—添加虚拟磁盘 .....                        | 364 |
| dpmvc addvmvnic 命令—添加虚拟 NIC .....                       | 366 |
| dpmvc clone 命令—克隆 VM .....                              | 369 |
| dpmvc createdatastore 命令—创建数据存储 .....                   | 377 |
| dpmvc cycle 命令—循环 VM .....                              | 379 |
| dpmvc datastore 命令—获取数据存储属性 .....                       | 382 |
| dpmvc delete 命令—销毁 VM .....                             | 384 |
| dpmvc distributedswitch 命令—管理虚拟分布式交换机 .....             | 387 |
| dpmvc entermaintenancemode 命令-设置为维护模式 .....             | 391 |
| dpmvc exitmaintenancemode 命令-退出维护模式 .....               | 393 |
| dpmvc faulttolerance 命令—指定容错操作 .....                    | 395 |
| dpmvc getavailablesids 命令—获取可用的 SCSI 磁盘 .....           | 397 |
| dpmvc gethosthba 命令—获取主机总线适配器 .....                     | 398 |
| dpmvc getresources 命令-获取 VM 资源 .....                    | 399 |
| dpmvc getshares 命令-获取 VM 份额 .....                       | 401 |
| dpmvc getsnapshots 命令-获取 VM 快照 .....                    | 403 |
| dpmvc imgjobcheck 命令-获取克隆作业状态 .....                     | 405 |
| dpmvc migrate 命令-迁移 VM .....                            | 407 |
| dpmvc removedatastore 命令—删除数据存储 .....                   | 410 |
| dpmvc removevmvdisk 命令-删除虚拟磁盘 .....                     | 411 |
| dpmvc removevmvnic 命令-删除虚拟 NIC .....                    | 413 |
| dpmvc setcpu 命令-设置 VM 的 CPU 数目 .....                    | 414 |
| dpmvc setmemory 命令-设置 VM 的内存大小 .....                    | 416 |
| dpmvc setresources 命令-设置 VM 资源 .....                    | 418 |
| dpmvc setshares 命令-设置 VM 份额 .....                       | 421 |
| dpmvc snapshot 命令-管理快照 .....                            | 425 |
| dpmvc templatetovm 命令-将模板转换成 VM .....                   | 430 |
| dpmvc vapp 命令—管理 vApp .....                             | 433 |
| dpmvc virtualswitch 命令—管理虚拟交换机 .....                    | 437 |
| dpmvc vmtotemplate 命令-将 VM 转换成模板 .....                  | 440 |
| CA Hyper-V CLI 命令 .....                                 | 442 |
| dpmhyperv AddVMNic 命令-将网络适配器控制器添加到 VM 中 .....           | 443 |
| dpmhyperv AddVMSCSIController 命令-为 VM 创建 SCSI 控制器 ..... | 444 |
| dpmhyperv AgentGet 命令-返回代理值 .....                       | 445 |
| dpmhyperv AgentSet 命令-设置代理值 .....                       | 447 |

---

|                                                                |     |
|----------------------------------------------------------------|-----|
| dpmhyperv ChangeVMState 命令-更改 VM 的状态 .....                     | 448 |
| dpmhyperv CreateTemplateFromVM 命令-从 VM 创建模板 .....              | 450 |
| dpmhyperv CreateVirtDisk 命令-创建虚拟磁盘映像 .....                     | 451 |
| dpmhyperv CreateVirtFloppy 命令-创建虚拟软盘映像 .....                   | 453 |
| dpmhyperv CreateVM 命令-创建 VM .....                              | 454 |
| dpmhyperv CreateVMFromTemplate 命令-通过 SCVMM 服务器模板创建 VM .....    | 457 |
| dpmhyperv CreateVMFromTemplate 命令-利用模板创建 VM .....              | 461 |
| dpmhyperv DeleteVM 命令-删除 VM .....                              | 467 |
| dpmhyperv ExpandVirtDisk 命令-扩展虚拟磁盘的大小 .....                    | 469 |
| dpmhyperv ExportVM 命令-导出 VM .....                              | 470 |
| dpmhyperv ExportVMEx 命令-导出 VM .....                            | 471 |
| dpmhyperv GetHostSwitches 命令-获取主机的交换机 .....                    | 473 |
| dpmhyperv GetJobErrorInfo 命令-获取作业的错误信息 .....                   | 474 |
| dpmhyperv GetJobInfo 命令-获取作业的信息 .....                          | 475 |
| dpmhyperv GetJobState 命令-获取作业的状态 .....                         | 476 |
| dpmhyperv GetServerInfo 命令-获取所有受管服务器 .....                     | 478 |
| dpmhyperv GetVersion 命令-获取 PMM 版本 .....                        | 479 |
| dpmhyperv GetVMProperties 命令-获取 VM 属性 .....                    | 480 |
| dpmhyperv GetVMState 命令-获取 VM 状态 .....                         | 481 |
| dpmhyperv ImportVM 命令-导入 VM .....                              | 482 |
| dpmhyperv ImportVMEx 命令-复制并导入 VM .....                         | 484 |
| dpmhyperv SetClustered 命令-在群集中添加或删除 VM .....                   | 485 |
| dpmhyperv SetSysPrepProperties 命令-设置 Sysprep 属性 .....          | 487 |
| dpmhyperv SetVMDisk 命令-将驱动器或磁盘分配给驱动器控制器 .....                  | 490 |
| dpmhyperv SetVMProperties 命令-设置 VM 的属性 .....                   | 492 |
| dpmhyperv SetVMVirtFloppy 命令-将虚拟软盘映像分配给 VM 软盘驱动器 .....         | 495 |
| dpmhyperv ShowClusterSharedVolumes 命令-显示所有群集共享卷 .....          | 496 |
| dpmhyperv ShowDirectories 命令-显示 Hyper-V 服务器上的目录属性 .....        | 497 |
| dpmhyperv ShowLogicalDisks 命令-显示所有逻辑磁盘 .....                   | 498 |
| dpmhyperv ShowPhysDisks 命令-显示主机的物理磁盘 .....                     | 499 |
| dpmhyperv ShowSCVMMHardwareProfiles 命令-显示所有 SCVMM 硬件配置文件 ..... | 500 |
| dpmhyperv ShowSCVMMGuestOSProfiles 命令-显示来宾操作系统配置文件 .....       | 501 |
| dpmhyperv ShowTemplates 命令-显示所有 VM 模板 .....                    | 502 |
| dpmhyperv ShowVirtFloppy 命令-显示虚拟软盘驱动器列表 .....                  | 503 |
| dpmhyperv ShowVMDisks 命令-显示所有虚拟磁盘 .....                        | 504 |
| dpmhyperv ShowVMNics 命令-显示所有网络接口卡 .....                        | 505 |
| dpmhyperv ShowVMs 命令-显示所有 VM .....                             | 506 |
| dpmhyperv ShowVMSCSIControllers 命令-显示所有 SCSI 控制器 .....         | 507 |
| dpmhyperv ValidateAgentSNMPAccess 命令-验证对代理的 SNMP 访问 .....      | 508 |
| dpmhyperv ValidateServerAccess 命令-检查服务器访问权限 .....              | 509 |
| dpmhyperv DeleteTemplate 命令-删除模板 .....                         | 510 |
| CA IBM LPAR CLI 命令 .....                                       | 511 |
| dpmlpar cycle 命令-循环逻辑分区 .....                                  | 512 |

---

---

|                                                          |     |
|----------------------------------------------------------|-----|
| dpmlpar delete 命令-删除逻辑分区 .....                           | 515 |
| dpmlpar getresources 命令-获取 LPAR 资源 .....                 | 517 |
| dpmlpar image 命令-创建 IBM AIX LPAR .....                   | 519 |
| dpmlpar imgjobcheck 命令-检索 IBM AIX 映像作业的状态 .....          | 530 |
| dpmlpar list 命令-列出 HMC 资源 .....                          | 532 |
| dpmlpar setresources 命令-添加内存资源 .....                     | 534 |
| dpmlpar setresources 命令-添加处理器资源 .....                    | 536 |
| dpmlpar setresources 命令-减少内存资源 .....                     | 539 |
| dpmlpar setresources 命令-减少处理器资源 .....                    | 541 |
| dpmlpar setresources 命令-更新内存资源 .....                     | 543 |
| dpmlpar setresources 命令-更新处理器资源 .....                    | 544 |
| CA Microsoft Cluster Server CLI 命令 .....                 | 547 |
| dpmmcs getclusterinfo 命令-获取群集的信息 .....                   | 547 |
| dpmmcs getgroupresourcesinfo 命令-获取群集组的资源信息 .....         | 548 |
| dpmmcs getnetworkinterfacesinfo 命令-获取网络接口信息 .....        | 549 |
| dpmmcs getnetworkinfo 命令-获取网络信息 .....                    | 550 |
| dpmmcs getnodesinfo 命令-获取群集节点信息 .....                    | 551 |
| dpmmcs getresourcegroupsinfo 命令-获取群集资源组信息 .....          | 552 |
| dpmmcs getresourcesdependenciesinfo 命令-获取资源的依存关系信息 ..... | 553 |
| dpmmcs getresourcetypesinfo 命令-获取群集资源类型信息 .....          | 554 |
| dpmmcs getservicesinfo 命令-获取群集服务信息 .....                 | 555 |
| dpmmcs movegroup 命令-将资源组移动到其他节点 .....                    | 556 |
| dpmmcs pauseservice 命令-暂停群集服务 .....                      | 557 |
| dpmmcs bringgrouponline 命令-使资源组联机 .....                  | 558 |
| dpmmcs bringresourceonline 命令-使资源联机 .....                | 559 |
| dpmmcs resumeservice 命令-恢复群集服务 .....                     | 560 |
| dpmmcs startservice 命令-启动群集服务 .....                      | 561 |
| dpmmcs stopservice 命令-停止群集服务 .....                       | 562 |
| dpmmcs takegroupoffline 命令-使资源组脱机 .....                  | 563 |
| dpmmcs takeresourceoffline 命令-使资源脱机 .....                | 564 |
| CA Solaris Zones CLI 命令 .....                            | 565 |
| dpmzone associateproject 命令-将任务与项目关联 .....               | 565 |
| dpmzone clonezone 命令-克隆区域 .....                          | 566 |
| dpmzone createandinstallzone 命令--创建和安装区域 .....           | 567 |
| dpmzone createpool 命令-创建资源池 .....                        | 570 |
| dpmzone createproject 命令-创建项目 .....                      | 571 |
| dpmzone createzone 命令-创建区域 .....                         | 572 |
| dpmzone customcommand 命令-运行自定义命令 .....                   | 575 |
| dpmzone deleteproject 命令-删除项目 .....                      | 575 |
| dpmzone deletezone 命令-删除区域 .....                         | 576 |
| dpmzone installzone 命令-安装区域 .....                        | 577 |
| dpmzone movezone 命令-移动区域 .....                           | 578 |
| dpmzone rebootzone 命令-重新启动区域 .....                       | 579 |



|                                                               |     |
|---------------------------------------------------------------|-----|
| dpmzone renamezone 命令—重命名区域.....                              | 580 |
| dpmzone setpoolscheduler 命令—设置池排定程序.....                      | 581 |
| dpmzone setprocessorset 命令—设置处理器集的最大 CPU 数目.....              | 583 |
| dpmzone showinterfaces 命令—显示网络接口.....                         | 584 |
| dpmzone showpools 命令—显示资源池.....                               | 584 |
| dpmzone showprojects 命令—列出项目的属性.....                          | 585 |
| dpmzone showresourceset 命令—显示资源集.....                         | 586 |
| dpmzone showtasks 命令—列出项目中的任务.....                            | 586 |
| dpmzone showzoneinterfaces 命令—显示区域的网络接口.....                  | 587 |
| dpmzone showzoneprocess 命令—显示区域中的进程.....                      | 588 |
| dpmzone showzoneprojects 命令—显示区域中的项目.....                     | 588 |
| dpmzone showzones 命令—显示区域.....                                | 589 |
| dpmzone startzone 命令—启动区域.....                                | 590 |
| dpmzone stopzone 命令—停止区域.....                                 | 591 |
| dpmzone uninstallzone 命令—卸载区域.....                            | 592 |
| CA Cisco UCS 命令.....                                          | 592 |
| dpmucs associateserviceprofile—将服务配置文件与刀片服务器相关联.....          | 593 |
| dpmucs cloneserviceprofile—克隆服务配置文件.....                      | 595 |
| dpmucs createmacpool 命令—创建 MAC 地址池.....                       | 596 |
| dpmucs createorg 命令—创建组织.....                                 | 597 |
| dpmucs createserverpool 命令—创建服务器池.....                        | 599 |
| dpmucs createserviceprofile 命令—创建服务配置文件.....                  | 600 |
| dpmucs createtemplatefromserviceprofile 命令—从服务配置文件创建模板.....   | 602 |
| dpmucs createuuidpool 命令—创建 UUID 池.....                       | 603 |
| dpmucs createwwpool 命令—创建 WWNN 池.....                         | 604 |
| dpmucs deletemacpool 命令—删除 MAC 地址池.....                       | 606 |
| dpmucs deleteorg 命令—删除组织.....                                 | 607 |
| dpmucs deleteserverpool 命令—删除服务器池.....                        | 608 |
| dpmucs deleteserviceprofile—删除服务配置文件.....                     | 609 |
| dpmucs deleteuuidpool 命令—删除 UUID 池.....                       | 610 |
| dpmucs deletewwpool 命令—删除 WWNN 池.....                         | 612 |
| dpmucs deployimage 命令—部署 Cisco UCS 映像.....                    | 613 |
| dpmucs disassociateserviceprofile 命令—解除关联服务配置文件.....          | 615 |
| dpmucs failoverserviceprofile 命令—将服务配置文件与用于故障转移的刀片服务器相关联..... | 616 |
| dpmucs getblades 命令—获取刀片服务器信息.....                            | 617 |
| dpmucs getchassis 命令—获取机箱信息.....                              | 619 |
| dpmucs getpolicy 命令—从 UCS 管理器获取策略.....                        | 620 |
| dpmucs getserviceprofilestatus 命令—获取服务配置文件状态.....             | 621 |
| dpmucs getswitches 命令—获取交换机信息.....                            | 622 |
| dpmucs gettopsystem 命令—获取系统信息.....                            | 623 |
| dpmucs listserviceprofiles 命令—列出服务配置文件.....                   | 624 |
| dpmucs powercycle 命令—重新启动刀片服务器.....                           | 625 |
| dpmucs renamepool 命令—重命名池.....                                | 627 |

|                                                                     |     |
|---------------------------------------------------------------------|-----|
| dpmucs serviceprofileupdate 命令—执行服务配置文件操作 .....                     | 628 |
| 管理命令行实用工具 .....                                                     | 630 |
| dpmutil 语法 .....                                                    | 630 |
| dpmutil -amq 命令—配置 ActiveMQ 消息代理 .....                              | 631 |
| dpmutil ecs 命令—配置 Windows 排定程序 .....                                | 632 |
| dpmutil -hd 命令—配置帮助台 .....                                          | 632 |
| dpmutil -hmc 命令—配置 HMC 或 IVM 服务器 .....                              | 633 |
| dpmutil -scvmmserver 命令—配置 Hyper-V 服务器 .....                        | 634 |
| dpmutil mgmtldb 命令—配置管理数据库 .....                                    | 634 |
| dpmutil -perfdb 命令—配置性能数据库 .....                                    | 635 |
| dpmutil remotevcaim 命令—配置远程 VC AIM .....                            | 636 |
| dpmutil -sc 命令—配置服务控制器 .....                                        | 636 |
| dpmutil -scvmmserver 命令—配置 SCVMM 服务器 .....                          | 637 |
| dpmutil -snmp 命令—配置 SNMP .....                                      | 638 |
| dpmutil -systemcredential 命令—配置外部系统凭据 .....                         | 639 |
| dpmutil -sysuser 命令—配置在服务启动时使用的本地凭据 .....                           | 640 |
| dpmutil vcdefault 命令—配置默认 VMware vCenter 管理员用户 .....                | 641 |
| dpmutil vcserver 命令—配置 VMware vCenter .....                         | 641 |
| dpmutil -vioserver 命令—配置 VIO 服务器 .....                              | 642 |
| 通用发现命令 .....                                                        | 642 |
| dpmcda cancelnetworkscan 命令—取消网络发现 .....                            | 643 |
| dpmcda deletenetwork 命令—删除网络 .....                                  | 643 |
| dpmcda discoverhost 命令—发现主机 .....                                   | 644 |
| dpmcda discovernetwork 命令—发现网络 .....                                | 645 |
| 收集引擎命令 .....                                                        | 647 |
| dpmce ceconfig -configuremetric 命令—配置度量标准 .....                     | 647 |
| dpmce ceconfig -configuremetricfilter 命令—配置度量标准筛选 .....             | 649 |
| dpmce ceconfig -disablemetric 命令—禁用度量标准 .....                       | 651 |
| dpmce ceconfig -disablemetricforoverall 命令—禁用度量标准并将其排除在总体计算之外 ..... | 652 |
| dpmce ceconfig -enablemetric 命令—启用度量标准 .....                        | 653 |
| dpmce ceconfig -enablemetricforoverall 命令—启用度量标准并将其包括在总体计算中 .....   | 655 |
| dpmce ceconfig -getglobal 命令—检索数据中心间隔 .....                         | 656 |
| dpmce ceconfig -getnode 命令—检索主机收集间隔 .....                           | 656 |
| dpmce ceconfig -getmetriclist 命令—检索度量标准列表 .....                     | 657 |
| dpmce ceconfig -refreshmetriclist 命令—重试度量标准收集 .....                 | 658 |
| dpmce ceconfig -setalias 命令—设置别名 .....                              | 659 |
| dpmce ceconfig -setglobal 命令—设置数据中心间隔 .....                         | 661 |
| dpmce ceconfig -setnode 命令—设置主机间隔 .....                             | 662 |
| dpmce ceconfig -updatemetric 命令—更新度量标准 .....                        | 664 |
| 事件命令 .....                                                          | 665 |
| dpmeventmanager create_event 命令—创建事件 .....                          | 665 |
| dpmeventmanager get_events 命令—检索事件 .....                            | 667 |
| 常规外壳命令 .....                                                        | 670 |

---

|                                                         |     |
|---------------------------------------------------------|-----|
| caaipsecurity 命令--指定 Web 服务安全检査的凭据 .....                | 670 |
| createdpmdefaultgroup 命令—创建或更新服务 .....                  | 670 |
| dpmcmd run 命令—运行脚本 .....                                | 671 |
| dpmix export import verify 命令—迁移系统数据 .....              | 672 |
| dpmrpt schedulereport 命令—排定报告 .....                     | 677 |
| dpmssh run 命令—运行远程外壳命令或脚本 .....                         | 680 |
| 帮助中心命令 .....                                            | 680 |
| dpmhd addcomment 命令-使用注释更新票单 .....                      | 681 |
| dpmhd close 命令-关闭帮助台票单 .....                            | 681 |
| dpmhd getattribute 命令-获取票单的属性值 .....                    | 682 |
| dpmhd getstatus 命令-获取票单的状态 .....                        | 683 |
| dpmhd open 命令—打开帮助台票单 .....                             | 684 |
| dpmhd setattribute 命令—设置票单的属性 .....                     | 686 |
| dpmhd setstatus 命令—设置票单的状态 .....                        | 687 |
| 映像命令 .....                                              | 688 |
| dpmimg image 命令-部署 Solaris 映像 .....                     | 689 |
| dpmimg imgjobcheck 命令-获取 Solaris 映像作业状态 .....           | 693 |
| 策略命令 .....                                              | 695 |
| dmpolicy createdefaultactions 命令—创建默认操作 .....           | 695 |
| dmpolicy exportaction 命令-将操作导出到文件 .....                 | 696 |
| dmpolicy exportrule 命令—将规则导出到文件 .....                   | 697 |
| dmpolicy findbestmachine 命令-查找最佳服务器 .....               | 698 |
| dmpolicy findcandidatemachine 命令-查找候选服务器 .....          | 699 |
| dmpolicy getruleexeccount 命令-检索规则执行计数 .....             | 701 |
| dmpolicy importaction 命令-从文件导入操作 .....                  | 702 |
| dmpolicy importaction 命令—从文件导入规则 .....                  | 703 |
| dmpolicy resetruleexeccount 命令-重置运行规则的重试次数 .....        | 704 |
| dmpolicy runaction 命令-运行操作 .....                        | 705 |
| dmpolicy setruleexeclimit 命令-设置规则的限制 .....              | 707 |
| 策略配置 .....                                              | 709 |
| caismutility 命令 -- 将功能添加到策略配置 UI 中 .....                | 709 |
| 资源管理器命令 .....                                           | 711 |
| dpmresourcemgr addbaremetal 命令-添加物理计算机 .....            | 712 |
| dpmresourcemgr addcomputertonetwork 命令-向网络中添加服务器 .....  | 713 |
| dpmresourcemgr addipaddresspool 命令-添加 IP 地址池 .....      | 714 |
| dpmresourcemgr addmachinestosome 命令-向服务中添加服务器 .....     | 715 |
| dpmresourcemgr addtocontainer 命令—将内容添加到容器中 .....        | 716 |
| dpmresourcemgr createcontainer 命令—创建容器 .....            | 717 |
| dpmresourcemgr createdefaultservice 命令-创建默认服务 .....     | 718 |
| dpmresourcemgr createipnetwork 命令-创建网络 .....            | 719 |
| dpmresourcemgr createresourcegroup 命令—创建资源组 .....       | 721 |
| dpmresourcemgr createresourcetemplate 命令—创建服务器组模板 ..... | 722 |
| dpmresourcemgr createservice 命令-创建服务 .....              | 724 |

|                                                               |     |
|---------------------------------------------------------------|-----|
| dpmresourcemgr deletebaremetal 命令-删除物理计算机 .....               | 725 |
| dpmresourcemgr deletecontainer 命令-删除容器 .....                  | 726 |
| dpmresourcemgr deleteipnetworks 命令-删除网络 .....                 | 727 |
| dpmresourcemgr deletesystem 命令-删除系统 .....                     | 728 |
| dpmresourcemgr getdiscoverednetworks 命令-获取发现的网络 .....         | 729 |
| dpmresourcemgr getipaddresspools 命令-列出 IP 地址池 .....           | 730 |
| dpmresourcemgr getipnetworks 命令-列出网络 .....                    | 731 |
| dpmresourcemgr getserverlist 命令-获取服务器的列表 .....                | 732 |
| dpmresourcemgr getservicelist 命令-获取服务的列表 .....                | 733 |
| dpmresourcemgr getsubnetdefaults 命令-列出子网默认值 .....             | 734 |
| dpmresourcemgr removecomputerfromipnetwork 命令-从网络中删除服务器 ..... | 735 |
| dpmresourcemgr removefromcontainer 命令-从容器中删除内容 .....          | 736 |
| dpmresourcemgr removeipaddresspool 命令-删除 IP 地址池 .....         | 737 |
| dpmresourcemgr removemachinesfromservice 命令-从服务中删除服务器 .....   | 738 |
| dpmresourcemgr removeresourcegroup 命令-删除资源组 .....             | 739 |
| dpmresourcemgr removeservice 命令-删除服务 .....                    | 740 |
| dpmresourcemgr removesourcetemplate 命令-删除资源模板 .....           | 741 |
| dpmresourcemgr retrieveresourcegroup 命令-检索资源组 .....           | 742 |
| dpmresourcemgr retrieveresourcetemplate 命令-检索资源模板 .....       | 743 |
| dpmresourcemgr setcontainername 命令-设置容器名称 .....               | 744 |
| dpmresourcemgr setcontainerservice 命令-将容器设置为服务 .....          | 745 |
| dpmresourcemgr setmanagementip 命令-设置主机的管理 IP .....            | 746 |
| dpmresourcemgr setmanagementstatus 命令-设置主机的管理状态 .....         | 747 |
| dpmresourcemgr setserviceproperties 命令-设置服务属性 .....           | 748 |
| dpmresourcemgr updateresourcegroup 命令-更新资源组 .....             | 749 |
| dpmresourcemgr updateresourcetemplate 命令-更新资源模板 .....         | 750 |
| dpmresourcemgr updateservice 命令-更新服务 .....                    | 752 |
| dpmresourcemgr updateipnetwork 命令-更新网络 .....                  | 753 |
| dpmresourcemgr updatesubnetdefaults 命令-更新网络默认值 .....          | 754 |
| dpmresourcemgr updateipaddresspool 命令-更新 IP 地址池 .....         | 756 |
| 远程监控命令 .....                                                  | 757 |
| rmonwatch add 命令-将行添加到 MIB 表中 .....                           | 757 |
| rmonwatch del 命令-从 MIB 表中删除行 .....                            | 759 |
| rmonwatch getTable 命令-查看 rmonWbem MIB 表 .....                 | 761 |
| rmonwatch getVersion 命令-查看版本信息 .....                          | 763 |
| 高级 SNMP 参数 .....                                              | 764 |
| 日志文件 .....                                                    | 765 |
| 可用的日志文件 .....                                                 | 766 |
| 日志文件设置 .....                                                  | 768 |

## 第 4 章：性能度量标准

769

|                         |     |
|-------------------------|-----|
| CA HP UNIX 性能度量标准 ..... | 769 |
|-------------------------|-----|

---

|                                      |     |
|--------------------------------------|-----|
| CA IBM AIX 性能度量标准 .....              | 770 |
| CA IBM LPAR 性能度量标准 .....             | 771 |
| CA IBM pSeries 逻辑分区性能度量标准 .....      | 771 |
| CA IBM pSeries 性能度量标准 .....          | 772 |
| 处理器池支持 .....                         | 773 |
| CA IBM HACMP 性能度量标准 .....            | 776 |
| IBM HACMP 群集度量标准 .....               | 776 |
| IBM HACMP 节点度量标准 .....               | 776 |
| IBM HACMP 资源组度量标准 .....              | 776 |
| IBM HACMP 网络度量标准 .....               | 777 |
| CA Linux 性能度量标准 .....                | 777 |
| CA Microsoft Hyper-V 性能指标 .....      | 778 |
| CA Oracle Solaris 性能度量标准 .....       | 778 |
| CA Oracle Solaris Zones 性能度量标准 ..... | 779 |
| Solaris Zones 服务器度量标准 .....          | 779 |
| Solaris Zones 服务器磁盘度量标准 .....        | 780 |
| Solaris Zones 服务器网络度量标准 .....        | 780 |
| Solaris Zones 服务器内存度量标准 .....        | 781 |
| Solaris Zones 服务器 CPU 度量标准 .....     | 781 |
| Solaris Zones 度量标准 .....             | 781 |
| Solaris Zones CPU 度量标准 .....         | 782 |
| Solaris Zones 内存度量标准 .....           | 782 |
| Solaris Zones 最大交换度量标准 .....         | 783 |
| Solaris Zones 共享内存度量标准 .....         | 783 |
| Solaris Zones 资源池度量标准 .....          | 783 |
| Solaris Zones 项目度量标准 .....           | 783 |
| CA SystemEDGE 性能指标 .....             | 784 |
| CA SystemEDGE 增强的度量标准 .....          | 787 |
| CA VMware vCenter 服务器性能指标 .....      | 794 |
| VMware VC ESX 主机 .....               | 794 |
| VMware VC ESX 主机 CPU .....           | 795 |
| VMware VC ESX 主机磁盘 .....             | 795 |
| VMware VC ESX 主机内存 .....             | 796 |
| VMware VC ESX 主机网络 .....             | 797 |
| VMware VC ESX 主机每个 CPU .....         | 797 |
| VMware VC VM .....                   | 798 |
| VMware VC VM CPU .....               | 798 |
| VMware VC VM 磁盘 .....                | 799 |
| VMware VC VM 内存 .....                | 800 |
| VMware VC VM 网络 .....                | 802 |
| VMware VC 群集 .....                   | 802 |
| VMware VC 数据存储 .....                 | 803 |
| VMware VC 计算资源 .....                 | 803 |

---

|                                                   |     |
|---------------------------------------------------|-----|
| VMware VC 资源池.....                                | 804 |
| VMware VC Server .....                            | 805 |
| CA Citrix XenServer 性能度量标准.....                   | 805 |
| Citrix Xen 主机 .....                               | 806 |
| Citrix Xen 主机 pNic .....                          | 806 |
| Citrix Xen 主机存储 .....                             | 806 |
| Citrix Xen VM .....                               | 807 |
| Citrix Xen VM vDisk .....                         | 807 |
| Citrix Xen VM vNic.....                           | 808 |
| CA Windows Microsoft 群集性能度量标准 .....               | 808 |
| CA Windows 性能度量标准 .....                           | 809 |
| Cisco Unified Computing System (UCS) 性能度量标准 ..... | 810 |
| 远程监控性能指标.....                                     | 812 |

## 索引

817

# 第 1 章： 简介

---

此部分包含以下主题：

[本指南涵盖的内容](#) (p. 15)

[读者](#) (p. 15)

[相关出版物](#) (p. 15)

[约定](#) (p. 16)

## 本指南涵盖的内容

本指南介绍了 AutoShell、CA Virtual Assurance CLI 命令和性能度量标准。词汇表解释了在虚拟化技术中使用的特定术语。

AutoShell 是 CA Virtual Assurance 的中央命令行界面，可用于使复杂的周期性任务和管理任务实现自动化。AutoShell 为面向对象、XML 和正则表达式处理提供了一种独立的 JavaScript 实施方案。AutoShell 使用 Mozilla Spidermonkey JavaScript 解析程序的开箱即用版本（它还向 Mozilla Firefox Web 浏览器提供 JavaScript 功能）。解释器允许您在 AutoShell 中直接使用 JavaScript 语法，例如，用于显示目录列表。

本指南为您提供了该用户界面的所有 AutoShell 命令、示例和最佳实践。

## 读者

本指南适用于安装、配置和使用 CA Virtual Assurance 来管理虚拟环境的管理员。本指南假定您熟悉环境中所使用的操作系统、虚拟化技术和 SNMP。

## 相关出版物

CA Virtual Assurance 文档包括以下交付成果：

### 管理指南

了解如何管理和使用 CA Virtual Assurance 以在您的环境中管理虚拟资源。

### 安装指南

包含简短的体系结构信息、各种安装方法、后继安装配置信息以及入门说明。

### 联机帮助

提供有关窗口详细信息和程序说明，以便使用 CA Virtual Assurance 用户界面。

### 参考指南

提供有关 AutoShell、CLI 命令、MIB 属性以及性能度量标准的详细信息。

### 版本说明

提供有关操作系统支持、系统要求、已发布的修正、国际化支持、已知问题以及文档路线图。

### 服务响应监控用户指南

提供 SRM 的安装和配置详细信息。

### SystemEDGE 用户指南

提供 SystemEDGE 的安装和配置详细信息。

### SystemEDGE 版本说明

提供有关操作系统支持、系统要求和功能的信息。

## 约定

本指南使用以下约定：

### 区分大小写

在本指南中提到的所有类、命令、指令、环境参数、函数和属性的名称均区分大小写，您必须按显示的形式正确拼写它们。- 系统命令和环境变量名称 *可能* 区分大小写，视操作系统的要求而定。-

### 交叉引用

对其他指南或本指南其他部分中信息的引用采用以下格式：

#### 指南名称

指示其他指南的名称。

#### “章名称”

表示本指南或其他指南中的章节名称。

### 同义词

属性、对象和对象标识符 (OID) 等术语与本文档中的术语“变量”是同义的。

SystemEDGE 代理和 CA SystemEDGE 等术语与本文档中的 SystemEDGE 是同义的。



## 语法

语法和用户输入使用以下格式：

### *斜体*

表示必须为其提供实际值的变量名或占位符。

### **{a|b}**

表示选择强制性操作数（a 或 b）。

### **[ ] 或 [[]]**

表示可选操作数。

## 语法示例

以下示例使用这些约定：

```
modify -t ZONE [-m zoneserver] -p psetname {-min mincpu|-max maxcpu} pset  
-session ssh
```

操作数 **-min** 和 **-max** 是强制性的，但您可以只使用二者之一，具体视希望定义处理器集中 CPU 的最小数量还是最大数量而定。操作数 **-m** 不是该命令运行所必需的。该命令的所有其他部分必须按显示的形式进行输入。

## 默认目录

在路径语句中使用的 *CASYSEDGE* 表示在其中安装 SystemEDGE 的目录。**默认值：** C:\Program Files\CA\SystemEDGE。

## 安装路径

在路径语句中使用的 *Install\_Path* 表示在其中安装 CA Virtual Assurance 或 CA Virtual Assurance 组件的目录。

### **默认值：**

- Windows x86: C:\Program Files\CA
- Windows x64: C:\CA、C:\Program Files (x86)\CA 或 C:\Program Files\CA
- UNIX、Linux: /opt/CA



## 第 2 章： AutoShell

---

AutoShell 是可用于使复杂的周期性任务和管理任务实现自动化的命令行和脚本环境。

本章详细介绍了 AutoShell 并提供了有关该用户界面的许多示例和最佳实践。

此部分包含以下主题：

- [关于 AutoShell](#) (p. 19)
- [管理器 Shell 和客户端 Shell](#) (p. 20)
- [有效 AutoShell 用户](#) (p. 21)
- [访问 AutoShell](#) (p. 21)
- [通用信息模型 \(CIM\) 对象](#) (p. 22)
- [调用 AutoShell](#) (p. 25)
- [字符串化](#) (p. 29)
- [将数据写入控制台 \(stdout\)](#) (p. 30)
- [从键盘读取数据 \(标准输入\)](#) (p. 31)
- [访问操作系统](#) (p. 32)
- [运行脚本](#) (p. 33)
- [AutoShell 核心参考](#) (p. 40)
- [AutoShell 可加载模块的命令参考](#) (p. 149)

### 关于 AutoShell

AutoShell 是脚本语言和命令行 shell 的组合。它基于标准的脚本语言 ECMA-Script（通常称为 JavaScript）。虽然 JavaScript 通常被认为是在网页上使用的脚本语言，但它不需要在浏览器中运行。它是实现面向对象、XML 和正则表达式处理支持的独立脚本语言。AutoShell 使用 Mozilla Spidermonkey JavaScript 解释器的开箱即用版本（它还向 Mozilla Firefox Web 浏览器提供 JavaScript 功能）。这允许您在 AutoShell 中直接使用 JavaScript 语法，例如，用于显示目录列表。

通常，字符串必须包含在半角引号内，特殊字符必须进行转义，而且函数调用必须包含在半角圆括号内。以该方式输入指令效率较低，因此 AutoShell 在 JavaScript 语言之上实现了一种特有的命令翻译层，该层可以接受带有强制和可选参数的典型 shell 类型命令。这些命令不要求将字符串放在半角引号中，而是将它们翻译为基础的 JavaScript 语法。

关键 AutoShell 功能如下：

- 操作系统访问，以运行子进程并捕获它们的输出。
- 并行远程脚本执行，以实现高效处理无法集中运行的任务。

- 通过在计算机之间直接复制文件以进行远程文件系统访问，从而实现高级开通。
- 支持管理 API（如库调用、Java 和 COM）通常使用的常见应用程序编程接口 (API) 机制，以允许从脚本访问这些 API。
- 远程注册表访问（特定于 Windows）。
- 可将自身的客户端 shell 和任意安装软件包推送至远程 Windows 系统并进行安装。

**注意：**有关 JavaScript 语言参考和示例的信息，请参阅网站 <https://developer.mozilla.org/en/JavaScript>。AutoShell 使用 JavaScript 版本 1.7。

#### 详细信息

[管理器 Shell 和客户端 Shell](#) (p. 20)

[调用 AutoShell](#) (p. 25)

[字符串化](#) (p. 29)

[运行脚本](#) (p. 33)

[AutoShell 核心参考](#) (p. 40)

## 管理器 Shell 和客户端 Shell

AutoShell 包括两部分：管理器 shell 和客户端 shell。要输入命令或开始执行脚本，请使用管理器 shell 在管理器系统上执行本地任务。客户端 shell 被认为是 AutoShell 远程执行基础架构的一部分。要启用远程执行和远程文件系统访问，必须在所有受管的远程节点上安装客户端 shell。除此之外，您不需要使用客户端 shell。管理器 shell 通过安全 SSH 连接调用客户端 shell 以执行远程操作。因此，AutoShell 一词始终指的是管理器 shell。

### 管理器 Shell

允许您在管理器计算机上交互式输入命令、启动脚本以及执行本地任务。

### 客户端 Shell

使用安全外壳 (SSH) 连接实现远程执行和远程文件系统访问，并且必须在所有受管服务器远程节点上安装。

AutoShell 可以在交互模式或脚本（批处理）模式中调用。

## 有效 AutoShell 用户

在 CA Virtual Assurance 安装期间，您在安装向导的“本地安全性用户信息”屏幕中定义 CA Embedded Entitlements Manager (Embedded Entitlements Manager) 用户身份和密码。凭据存储在 Embedded Entitlements Manager 数据库中。用户被分配给 CA Virtual Assurance 管理员组，并且可用于登录到 CA Virtual Assurance 用户界面和 AutoShell 管理器。

如果使用 Embedded Entitlements Manager 的 CA Virtual Assurance 组件安装在本地或远程系统上，AutoShell 管理器将始终对比 Embedded Entitlements Manager 数据验证登录凭据。否则，AutoShell 管理器对比 Windows 身份验证验证登录凭据。

## 访问 AutoShell

您可以从 CA Virtual Assurance 服务器访问 AutoShell。

### 访问 AutoShell

1. 打开 Windows 资源管理器并导航到以下目录：  
C:\Program Files\CA\SC\AutoShellClient
2. 双击 caaipaomautoshellclient.exe。  
此时将打开 AutoShell 命令提示符窗口。
3. 输入有效 CA Virtual Assurance 用户的用户名和密码。  
此时将显示 AutoShell 命令提示符。

### 详细信息

[有效 AutoShell 用户](#) (p. 21)

## 通用信息模型 (CIM) 对象

CA Virtual Assurance 使用基于 CIM 的对象模型来存储有关受管对象的信息。受管对象可以作为 AutoShell 脚本环境内的本地对象而被检索、查询和操纵。有关 CIM 详细信息，请参阅由分布式管理任务组 (DMTF) 发布的 CIM 规格：

<http://www.dmtf.org/standards/cim/>。有关 DMTF CIM 文档格式和功能规格，请参阅 [http://www.dmtf.org/standards/published\\_documents/DSP0202\\_1.0.0.pdf](http://www.dmtf.org/standards/published_documents/DSP0202_1.0.0.pdf)。

要在安装期间启用可选的 AutoShell CIM 功能，请选择 AutoShell CIM ALM (AutoShell Loadable Module)。

**重要信息！** AutoShell 提供对 CA Virtual Assurance 对象存储的完全写入访问权限。使用不恰当的 AutoShell 命令或脚本可能使此存储的完整性无效，导致其他 CA Virtual Assurance 组件出现故障。始终从 AutoShell 内部针对对象存储执行只读操作。

## 检索 CIM 类

您可以在 AutoShell 中检索 CIM 类。

要检索 DMTF CIM 类，请使用以下命令：

```
get-autoShellClasses CIM_*
```

要检索 CA 定义的 CIM 扩展类，请使用以下命令：

```
get-autoShellClasses CA_*
```

## 从 CIM 对象检索祖先项和子项

您可以使用 `getAntecedents()` 和 `getDependents()` 方法来检索 CIM 对象祖先项和子项。

要检索 CIM 对象祖先项，请调用 `getAntecedents()` 方法：

```
// 检索所有计算机系统
arr=CA_ComputerSystem.getInstances()
// 继续使用返回的第一个计算机系统
cs=arr[0]
// 检索祖先项
arrAnt = cs.getAntecedents()
for(i=0; i<arrAnt.length; i++)
{
    show-cimObject arrAnt[i]
}
```

要检索某个 CIM 类的祖先项实例，请调用 `getAntecedents()` 方法：

```
os = cs.getAntecedents("CIM_OperatingSystem");
show-cimObject os
```

要检索 CIM 对象子项，请调用 `getDependents()` 方法：

```
// 检索所有计算机系统，
arr=CA_ComputerSystem.getInstances()
// 继续使用返回的第一个计算机系统
cs=arr[0]
// 检索子项
arrAnt = cs.getDependents()
for(i=0; i<arrAnt.length; i++)
{
    show-cimObject arrAnt[i]
}
```

要检索某个 CIM 类的子项实例，请调用 `getDependents()` 方法：

```
fs = cs.getDependents("CIM_FileSystem");
show-cimObject fs
```

## 检索 CIM 对象

每个 CIM 类都有一个静态 `getInstances()` 方法，该方法返回类对象的数组。

要检索受 CA Virtual Assurance 管理的所有系统，请使用以下命令：

```
aObjs = CA_ComputerSystem.getInstances()
? "Query returned", aObjs.length
```

要检索受 CA Virtual Assurance 管理的部分系统，请使用以下筛选表达式之一：

```
aObjs = CA_ComputerSystem.getInstances("ElementName=' Everest.local'")
```

或

```
aObjs = CA_ComputerSystem.getInstances("ElementName like 'Ev*')")
```

要检索系统名称和说明，请使用以下命令：

```
aObjs = CA_ComputerSystem.getInstances();
for(i=0; i<aObjs.length; i++)
    ? aObjs[i].getElementName(),aObjs[i].getDescription()
```

要检索 CIM 对象属性和值，请使用以下命令：

```
aObjs = CA_ComputerSystem.getInstances("ElementName='Debian1'")
show-cimObject aObjs[0]
```

**示例: show-cimObject 命令的输出**

```
CA_ComputerSystem
=====
Caption: null
Description: null
ElementName: Debian1
Facets: 44
InstallDate: null
Name: 47332da1-e782-0308-d614-c2e99de989bc
OperationalStatus: undefined
StatusDescriptions: undefined
Status: null
HealthState: 5
PropagatedHealthState: null
IsPropagated: true
PrimaryStatus: null
DetailedStatus: null
OperatingStatus: null
CommunicationStatus: null
ManagementStatus: null
EnabledState: null
OtherEnabledState: null
RequestedState: null
EnabledDefault: null
TimeOfLastStateChange: null
AvailableRequestedStates: undefined
TransitioningToState: null
CreationClassName: CA_ComputerSystem
NameFormat: null
PrimaryOwnerName: null
PrimaryOwnerContact: null
Roles: undefined
OtherIdentifyingInfo: undefined
IdentifyingDescriptions: undefined
Dedicated: undefined
OtherDedicatedDescriptions: undefined
ResetCapability: null
PowerManagementCapabilities: undefined
HostName: 192.168.1.119
IpAddress: null
SerialNumber: null
Version: null
CpuCount: 1
TotalVisibleMemorySize: 256
PatchLevel: null
BiosVersion: null
Fans: null
Chassis: null
DrCapable: null
VendorId: null
MacAddress: null
ArchTypeUID: null
Available: null
```



```
LastUpdate: null
SnapshotType: null
TotalStorageCapacity: null
Model: null
CpuSpeed: null
SystemImageId: null
HardwareClassId: null
```

## 调用 AutoShell

通过使用以下命令，您可以从命令行接口调用 AutoShell：

```
caaipaomautoshell.exe [-U username] [-P password] [-h host] [[-f] scriptfile]
[scriptargs] [-e expr] [-i]
```

### **-U *username***

指定有效的 CA Virtual Assurance 用户。您还可以使用 CAASUSER 环境变量来设置用户名。

### **-P *password***

指定该用户的密码。您还可以使用 CAASSPASS 环境变量来设置密码。

### **-f *scriptfile* [*scriptargs*]**

（可选）使用可选的脚本参数加载指定的脚本文件。此参数可以多次使用。

### **-e *expr***

（可选）评估指定的表达式。

### **-i**

（可选）在加载脚本之后进入交互模式。

### **-locale=*locale***

（可选）指定 ISO 639\_3166 组合（例如：法语为 fr\_FR）以覆盖默认的英语输出。要使用命令提示符的区域设置，请指定“native”。

## 示例

运行 hello.js 脚本:

```
caaipaomautoshell.exe -U vaimuser -P topsecret -f hello.js
```

计算  $4*3$  的值:

```
caaipaomautoshell.exe -U vaimuser -P topsecret -e 4*3
```

为用户名和密码设置环境变量并且计算  $\sin(\pi/2)$  的值:

```
set CAASUSER=vaimuser  
set CAASSPASS=topsecret  
caaipaomautoshell.exe -e Math.sin(Math.PI/2)
```

## 详细信息

[有效 AutoShell 用户](#) (p. 21)

[使用交互模式](#) (p. 27)

[使用远程交互模式](#) (p. 28)

[批处理模式](#) (p. 28)

[远程批处理模式](#) (p. 29)

## 使用交互模式

使用交互模式可以在本地或远程主机上执行简单操作，或者在创建脚本化命令序列时用于测试小的代码片段。

### 使用交互模式

1. 以下列方法之一运行以下 **AutoShell** 可执行文件：

```
caaipaomautoshell.exe
```

**AutoShell** 会提示您输入用户名和密码。然后，**AutoShell** 执行身份验证检查或存储凭据以供以后使用，如验证到远程客户端。该决定取决于产品集成的类型。

```
caaipaomautoshell.exe -U bob -P mypassword
```

您可以使用 **-U** 和 **-P** 开关指定用户名和密码。

2. （可选）通过按照以下方式设置环境变量 **CAASUSER** 和 **CAASSPASS** 来指定凭据：

```
set CAASUSER=bob
set CAASSPASS=mypassword
```

成功登录后将出现 **AutoShell** 提示符。

```
asmgr::->
```

**asmgr**

指定本地主机名。

3. 在提示符位置输入任何有效的 **JavaScript** 表达式，因为 **AutoShell** 使用 **JavaScript** 作为基础脚本语言。

示例：

如果输入 **2+3**，则会显示 **5**

**AutoShell** 会自动显示结果，这使得对于简单表达式而言，无需在命令提示符位置使用输出命令。但是，在脚本中显示表达式的结果则需要您使用输出命令。另请参阅 [run-local 命令](#) (p. 33) 部分。

4. 输入以下命令之一来终止 **AutoShell** 交互式会话：

```
quit
exit
```

## 使用远程交互模式

您可以对安装有 AutoShell 客户端的受管系统启动远程交互式会话。AutoShell 使用 SSH 实现管理器 shell 与客户端 shell 之间的通信，以使用用户名和密码作为登录凭据连接到任何 SSH 服务器系统。在这种情况下，可用的命令不是 AutoShell 命令，而是在远程系统上配置的 SSH 的命令。

### 使用远程交互模式

1. 使用 -h 开关指定远程主机名：

```
caaipaomautoshell.exe -h ascll
```

交互式命令提示符将显示，并且包含指定的远程主机的名称。在提示符位置输入的命令在远程系统上的客户端 shell 中执行。结果返回到管理器 shell 并显示在交互式会话中。

**注意：**在交互式远程会话中指定要执行的脚本时，这些脚本必须位于远程系统上。

2. 输入以下选项之一来终止 AutoShell 交互式会话：

```
quit  
exit
```

## 批处理模式

可以从交互模式中调用 AutoShell 脚本，但是这些脚本的运行通常无需任何手工干预。

### 使用批处理模式

使用 -f 开关以及要运行的脚本的路径名调用 AutoShell：

```
caaipaomautoshell.exe -f hello.js
```

AutoShell 运行该脚本并在脚本执行完成时终止。交互模式在脚本完成之后开始。在命令的结尾指定 -i（顺序很重要），以便在处理脚本之后进入交互模式：

```
caaipaomautoshell.exe -f hello.js -i
```

**注意：**在这些示例中，请使用凭据添加 -U/-P 开关或设置 CAASUSER/CAASSPASS 环境变量。如果未定义凭据，AutoShell 会在开始脚本执行之前提示输入凭据。

您还可以向该脚本传递参数：

```
caaipaomautoshell.exe -f hello.js 1 abc "x y z"
```

将 1、“abc”和“x y z”传递给该脚本。这些参数可以通过参数数组进行访问，这是标准的 JavaScript 机制。

多次使用 `-f` 开关指定多个脚本：

```
caaipaomautoshell.exe -f hello.js -f world.js
```

脚本按照它们被指定的顺序执行。在指定参数时，每个脚本都会接收它自己的一组参数，例如：

```
caaipaomautoshell.exe -f hello.js 1 -f world.js 2
```

在这种情况下，`hello.js` 脚本接收实际参数 `1`，`world.js` 接收参数 `2`。

表达式求值是另一种方式。不是将简单的表达式添加到脚本然后使用 `-f` 执行这些表达式，而是使用 `-e` 开关在命令行中指定它们：

```
caaipaomautoshell.exe -e Math.sqrt(2)
caaipaomautoshell.exe -e "for(i=1;i<11;i++)qout(i)"
```

## 远程批处理模式

您可以在带有远程批处理模式的远程系统上运行一个或多个脚本。

### 使用远程批处理模式

在指定脚本文件之前使用 `-h` 开关指定主机名。例如：输入以下命令：

```
caaipaomautoshell.exe -h ascll -f hello.js -f world.js
```

脚本文件位于管理器系统上，并且被传输到远程客户端以执行。

## 字符串化

字符串化接受一连串字符并且将其转化为 JavaScript 文字串。例如，AutoShell `get-help` 命令将筛选字符串作为可选参数：

```
get-help help
```

因为 AutoShell 使用开箱即用型 JavaScript 解释器，在提示符或脚本文件中输入的任何命令均被翻译成有效的 JavaScript 语法。AutoShell 命令翻译层将该命令转化为 JavaScript 函数调用：

```
help( "help" );
```

JavaScript 中的文字串必须始终用引号括起来，因此命令翻译层会自动用引号括起命令中指定的字符串。如果字符串用引号括起，则不会再对其进行字符串化。

必须对 JavaScript 字符串内的特殊字符进行转义。尤其是引号分隔的字符串中的反斜杠和引号。通过将反斜杠作为这些特殊字符的前缀完成转义。可参考翻译为函数调用 (2) 的 Autoshell dir 命令 (1):

```
(1) dir c:\Program Files\CA\*.*  
(2) ca.aip.direct.directory("c:\\Program Files\\CA\\*.*", "", false, false, false,  
false);
```

在处理一个选项中包含空格且采用多个输入参数的字符串时，可能需要使用引号。如果未使用引号指定包含空格的字符串，则空格将被解释为参数分隔符，直到所有的输入参数匹配。为了防止该行为，请使用引号。

不使用不带引号的命令 (1)，而是对反斜杠进行转义，并使用引号括起路径，使 AutoShell 可以正确标识参数 (2):

```
(1) copy c:\Program Files\CA\*.* c:\temp  
(2) copy "c:\\Program Files\\CA\\*.*" c:\temp
```

**注意：**由于命令翻译层不会对路径进行字符串化，因此必须手工对字符串内的特殊字符进行转义。

要使字符串包含空格，且仍自动对特殊字符进行转义，还可以使用 `_S()` 宏：

```
copy _S(c:\Program Files\CA\*.*) c:\temp
```

当使用 AutoShell 命令自动化任务时，命令的一部分必须经常接受来自 JavaScript 变量的输入。在这种情况下，要防止字符串化。否则，变量名会转换为文字串，且命令不会生成预期的结果。为防止表达式的字符串化，请将表达式放到括号中：

```
var topic = "help";  
get-help (topic)
```

## 将数据写入控制台 (stdout)

脚本常常会显示结果或诊断消息。使用 “?” 和 “??” 命令将任意类型的输出写入到 `stdout`，在没有指定重定向时，`stdout` 将指定控制台屏幕。要将表达式内的输出写入到 `stdout`，请使用 `qout()` 和 `qqout()` 函数。函数 `qout()` 相当于 “?”，`qqout()` 相当于 “??”。命令和函数均接受参数的列表。

### ?、qout

将列表中每个参数的字符串表示写入到 `stdout`。会在每个显示的参数之间自动添加一个空格。该输出以换行符结束。

### ??、qqout

将列表中每个参数的字符串表示写入到 `stdout`。会在每个显示的参数之间自动添加一个空格。

**注意：**“?” 和 “??” 输出命令只能在行的开头处使用。

### 示例

“?” 命令打印所指定的值，紧接着是换行符：

```
? "Hello World"
Hello World
```

“?” 命令还接受逗号分隔的参数列表。显示的参数在输出中用一个空格分隔：

```
? "Hello", "World"
Hello World
```

“??” 命令的工作方式与 “?” 类似，但是不在输出的末尾添加换行符。可以在脚本中使用多个 “??” 命令来构建单行输出：

```
?? "Hello"
?? " "
?? "World"
```

因此，这两个命令接受任何数据类型和组合：

```
? "Today is", new Date
? "The square root of 2 is", Math.sqrt(2)
```

显示数字 1 到 10：

```
for(i=1;i<11;i++)qout(i);
```

## 从键盘读取数据（标准输入）

如果某个脚本在运行时需要输入，则使用以下命令之一：

### **wait**

等待按键，并可选择将键入的字符存储在变量中。

### **accept**

从键盘读取字符串（标准输入），直到按下 Enter 键：

### 示例

等到任意键被按下：

```
wait "Press any key..."
```

等到某个键被按下，然后将结果字符存储在变量 ch 中。

```
wait ""Press any key" to ch
```

读取字符串，直到按下 Enter 键，然后将该字符串存储到变量 `str` 中。

```
accept to str
```

显示提示消息 “Enter amount:”，读取输入字符串，直到按下 Enter 键，然后将该字符串存储到变量 `str` 中。

```
accept "Enter amount: " to str
```

## 访问操作系统

管理 `shell` 经常调用操作系统命令或其他本地实用工具来完成与系统相关的任务。从交互式 `AutoShell` 提示符或从脚本内部，通过使用感叹号作为它们的前缀来执行 OS 命令。感叹号必须始终是输入行的第一个非空格字符。

默认情况下，`AutoShell` 直接将 OS 命令的输出写入到 `AutoShell` 控制台。但 `AutoShell` 不能直接访问写入 `AutoShell` 控制台的输出来做进一步处理。处理子进程输出的应用程序通常将输出重定向到文件，并在子进程结束后读取该文件。`AutoShell` 允许您直接将子进程输出分配给 JavaScript 对象。请参阅以下示例。

您可以使用 OS 访问权从 `AutoShell` 启动任何应用程序。在调用 OS 命令时，`AutoShell` 总是先等待命令完成，然后再返回到命令提示符或继续执行脚本。

### 示例

列出当前目录中的文件，打印所运行 OS 的版本并启动记事本。这些示例命令无需任何 `AutoShell` 交互即可运行。以下命令中的第一个命令使用操作系统 `dir` 命令，而不是内置的 `Autoshell dir` 命令。`AutoShell` 一直等到记事本应用程序关闭。

```
! dir
! ver
! notepad
```

通过使用与号 “&” 连接命令，并使用单个子进程执行它们：

```
! dir >t.txt & type t.txt
```

## 使用 OSRedirect 类

调用子命令时，`AutoShell` 使用预定义的 `OSRedirect` 类进行输出重定向。要重定向输出并捕获子进程返回代码，请创建 `OSRedirect` 对象，并在使用可选 `-output` 子句的命令调用中传递此对象：

```
out = new OSRedirect();
! dir -output out
```



当您在 AutoShell 控制台中交互执行该命令时，它不在控制台中显示任何信息。它将输出重定向到指定的对象。完成之后，您可以在指定的对象上调用一系列方法，以获得有关命令执行的信息：

```
? "Error occurred during command execution:", out.errorOccurred()
? "Did command complete:", out.hasCompleted()
? "Output produced by command:", out.output()
? "Command return code (errorlevel):", out.result()
```

方法 `output()` 和 `result()` 会返回必需的信息。`output()` 方法可返回带有完整目录清单的字符串。您可以使用标准 JavaScript 字符串函数来进一步处理该输出。例如，将行解析为阵列，并逐行显示它们：

```
lines=out.output().split("\n");
for(i=0;i<lines.length;i++)qout(lines[i]);
```

如果要使用 `! AutoShell` 命令调用 `dir` 操作系统命令，则先分配 `OSRedirect` 对象，然后再发出 `!` 相匹配。要保存该步骤，您也可以使用 `!!` 命令调用 `OS` 相匹配。`!!` 命令会创建 `OSRedirect` 对象并将其分配给指定的变量（如果该对象不存在）。

即使没有明确分配 `OSRedirect` 对象，以下命令也会将当前目录清单输出到变量 `out` 中：

```
!! dir -output out
```

如果发出 不带可选 `-output` 子句的 `!!` 命令，则它会自动将目录信息写入到变量 `$stdout` 中：

```
!! dir
? $stdout.output()
```

## 运行脚本

本节介绍本地主机和多个远程主机上的脚本执行。

### 详细信息

[使用 `run-local` 命令](#) (p. 33)

[使用 `run-remote` 命令](#) (p. 35)

## 使用 `run-local` 命令

使用 `run-local` 命令对脚本表达式求值或在本地运行脚本文件。`run-local` 命令允许将参数传递到脚本代码中。

要对表达式求值，请指定表达式，例如：

```
run-local 1+1
```

表达式参数已字符串化，因此您可以使用（或不使用）引号指定表达式。

要执行脚本，请在可选的 `-file` 子句中指定脚本的路径名：

```
run-local -file scripts\primes.js
```

运行的脚本要调用其他脚本，只能使用带有 `-file` 选项的 `run-local` 命令。指定要调用脚本的 JavaScript 文件的路径名只能在交互提示符下工作。脚本文件内不支持路径名规格。如果要从脚本中执行脚本，请考虑以下示例：

```
// 脚本文件 hello.js 开始
function hello()
{
    ? "Hello World!"
}
// 脚本文件结束
```

使用以下命令调用该脚本：

```
run-local -file hello.js
```

该命令使 `hello()` 函数在当前脚本上下文中可用，但是该上下文不执行此函数，因此控制台上看不到任何内容。在 `run-local` 命令返回之后，调用脚本可以调用新定义的函数来输出预期的问候语：

```
hello();
```

调用脚本文件时，在 `run-local` 命令运行期间仅执行不包含在任何周边范围（如函数）内的代码。将脚本更改为以下示例，在 `run-local` 命令返回之前输出问候语。

```
// 脚本文件 hello2.js 开始
? "Hello World!"
// 脚本文件结束
```

要使该函数可用于调用并仍执行代码，脚本文件可以包含对函数本身的调用：

```
// 脚本文件 hello3.js 开始
hello();
function hello()
{
    ? "Hello World!"
}
// 脚本文件结束
```

可在脚本文件中直接执行的代码可以在函数范围以外的任何位置出现。如上例所示，不必将可执行代码置于第一个函数声明的前面。

使用可选的 `-with` 子句，您可以传递使用 `run-local` 命令运行的表达式或脚本中的参数。通过标准 JavaScript 参数数组可以访问表达式或脚本中的参数：

```
// 脚本文件 args.js 开始
var i, l;
l = arguments.length;
for(i=0; i < l; i++)
{
    ? arguments[i]
}
// 脚本文件结束
```

脚本会输出所有传递给它的参数。例如，以下命令输出“1”、“abc”及当前日期和时间。

```
run-local -file args.js -with 1, "abc", new Date()
```

### 详细信息

[访问操作系统](#) (p. 32)

[使用 OSRedirect 类](#) (p. 32)

[run-local 命令—在本地系统上执行脚本 \(Funclet\)](#) (p. 85)

## 使用 run-remote 命令

`run-remote` 命令在一个或多个远程 AutoShell 客户端系统上执行表达式或脚本文件。要使远程执行成功，远程目标节点必须已安装并配置 AutoShell 客户端。

`run-remote` 命令支持像 `run-local` 命令一样指定表达式、脚本和参数。脚本文件必须位于管理器系统上并传输到目标系统来执行。此外，使用 `on` 子句指定远程客户端。指定服务器名称是调用 `run-remote` 命令的最简单方法：

```
run-remote "1+2" on "ascl1"
```

**注意：**主机名不支持字符串化。

您可以使用列表为远程执行指定多个目标：

```
run-remote "1+2" on "ascl1", "ascl2"
```

使用 `run-remote` 这种方法不能获取执行的远程作业的相关信息。作业运行或失败，无需任何进一步指示。远程命令异步执行，这意味着在实际的远程命令仍在执行时，`run-remote` 命令就可返回。

要等到远程命令开始处理或发生错误，请使用 `-wait` 选项调用 `run-remote`：

```
run-remote "1+2" on "ascl1" -wait
```

对远程执行的更多控制提供了预定义的 `RemoteTarget` AutoShell 类。

## 详细信息

[使用 RemoteTarget 类 \(p. 36\)](#)

[run-remote 命令—在远程系统上执行脚本 \(Funclet\) \(p. 86\)](#)

## 使用 RemoteTarget 类

在以下方法中，将 run-remote 命令与 RemoteTarget 类结合使用：

- 与 run-remote 命令返回的 RemoteTarget 对象协同工作。
- 创建 RemoteTarget 对象并将其传递给 run-remote 命令。

run-remote 命令始终返回 RemoteTarget 对象的数组。该数组的长度等于 on 子句中指定的远程目标的数目。数组元素按位置直接对应于远程目标：

```
arrRT = run-remote -file hello.js on "ascl1", "ascl2"
```

run-remote 命令返回由两个 RemoteTarget 对象组成的数组。

- arrRT[0] 包含对应于远程主机 ascl1 上脚本 hello.js 的执行的 RemoteTarget 对象。
- arrRT[1] 对应于 ascl2 上相同脚本的执行。

RemoteTarget 对象提供脚本的状态信息，包括其执行过程、输出或结果。以下 RemoteTarget 方法检索该状态信息：

### hasCompleted()

指出命令处理是否已完成（true 或 false）。如果不使用可选 -wait 开关调用 run-remote，命令执行会异步运行。调用此方法以验证命令是否已完成及结果是否可用。当命令成功完成时，方法 hasCompleted() 返回 true；当发生错误，阻止进一步处理时，返回 false。

返回值：true 或 false

### errorOccurred()

指出是已成功完成命令且能够验证输出和结果 (false)，还是发生错误 (true)。

返回值：true 或 false

**output()**

返回一个字符串，其中包含在相应远程系统上运行指定的脚本时生成的捕获输出。

返回值: *string*

**result()**

返回在相应的远程系统上执行的指定脚本的 XML 编码值。

返回值: *XML*

有关 RemoteTarget 类的其他方法，请参阅 [AutoShell 类](#) (p. 125) 一节。

**示例**

假定先前介绍的 hello.js 脚本生成 “Hello World!” 问候语，则两个 RemoteTarget 对象的 output() 方法都返回该字符串。以下脚本会将这种处理放入更大的上下文中：

```
var t;
var arrRT;
var i, l;

arrRT = run-remote -file hello.js on "ascl1", "ascl2"
l = arrRT.length;

// 等待完成，也可以使用上面的 -wait 子句
// 但通过这种方式，我们可以解决自己的超时
t = 0;

while(t<60)
{
    for(i=0; i<l; i++)
    {
        if(!arrRT[i].hasCompleted())
            break;
    }
    // 如果所有调用都完成则中断
    if(i==l)
        break;
    sleep(500);
    t++;
}
```

```

for(i=0; i<l; i++)
{
    if(arrRT[i].hasCompleted())
    {
        if(arrRT[i].errorOccurred())
        {
            ? "Error occurred on ", arrRT[i].getHostName()
        }
        else
        {
            ? arrRT[i].getHostName(),"returned output:"
            ? arrRT[i].output()
        }
    }
    else
    {
        arrRT[i].abort();
    }
}

```

通过将远程执行与 OS 命令调用相结合，可访问远程系统上的 OS 服务或数据。例如，以下代码查询并返回在服务器 ascl1 上运行的操作系统的 OS 版本字符串。

```

arrRT = run-remote ! ver on "ascl1" -wait
? arrRT[0].output()

```

AutoShell 仅在上例中显示远程命令输出。远程处理的表达式或脚本也能返回值：

```

arrRT = run-remote 2*21 on "ascl1" -wait

```

在这种情况下，将表达式“2\*21”发送到服务器 ascl1 进行远程计算。在命令完成时，通过相应 RemoteTarget 对象的 result() 方法可以获取结果。

result() 方法不是直接返回值，而是返回值的 XML 表示：

```

? arrRT[0].result()
<value type="number">42</value>

```

要将该 XML 字符串转换为常规 JavaScript 值，请使用 get-remoteResult()：

```

v = get-remoteResult(arrRT[0])
? typeof v, v

```

AutoShell 序列化也适用于返回复杂类型（如数组或对象）的远程脚本。序列化使大量结构化数据可在远程服务器之间传输，而不会丢失结构数据。以下示例显示了返回包含三个元素的数组的远程执行：

```

arrRT = run-remote [1,"abc",new Date()] on "ascl1" -wait
v = get-remoteResult(arrRT[0])
? typeof v, v

```

处理 `RemoteTarget` 对象的另一种方法是在调用实际的远程命令之前创建 `RemoteTarget` 对象。通过在对象创建中指定目标主机的名称，构造 `RemoteTarget` 对象：

```
var rt1 = new RemoteTarget("ascl1");
var rt2 = new RemoteTarget("ascl2");
```

为同一远程主机执行不止一个命令或脚本时，显式创建 `RemoteTarget` 对象很有用。在这种情况下，每个 `run-remote` 命令可以重用同一 `RemoteTarget` 对象。否则，`run-remote` 会为每个调用创建一个 `RemoteTarget` 对象。

要将 `RemoteTarget` 对象与 `run-remote` 结合使用，请指定它们，而不是主机名：

```
run-remote -file hello.js on rt1, rt2 -wait
? rt1.output()
? rt2.output()
run-remote Math.log(Math.E) on rt1, rt2
? get-remoteResult(rt1)
? get-remoteResult(rt2)
```

使用 `RemoteTarget` 对象时不需要存储 `run-remote` 返回的数组。此数组仅包含对最初传递到命令的 `RemoteTarget` 对象的引用。因此可以直接访问显式创建的 `RemoteTarget` 对象来查询执行的结果。

您可以在列表中指定远程目标，如在之前的示例中或作为数组。

```
arrRT = [new RemoteTarget("ascl1"), new RemoteTarget("ascl2")];
run-remote -file hello.js on arrRT
```

`RemoteTarget` 对象允许您在后续的 `run-remote` 命令调用中重用对象，还提供一种管理远程客户端会话创建的方法。

`AutoShell` 执行 `run-remote` 命令时，它会通过 SSH 连接到指定的远程系统。默认情况下，远程系统上的 SSH 服务器会启动新的客户端 `AutoShell` 实例。新创建的 `AutoShell` 进程评估所传递的脚本并在评估完成后被销毁。对于随后的多个命令调用，将在远程系统上创建和销毁多个客户端 `AutoShell` 进程。另外，在运行脚本时创建的任何上下文信息都将丢失，无法用于随后的远程脚本。

`RemoteTarget` 类实施一种易于使用的机制来控制远程主机上的客户端 `AutoShell` 会话创建。在 `RemoteTarget` 对象上调用 `createRemoteContext()`，以在远程主机上为该 `RemoteTarget` 对象创建永久客户端 `AutoShell` 会话：

```
var rt = new RemoteTarget("ascl1");
rt.createRemoteContext()
```

随后的 `run-remote` 调用不会在远程主机上分配临时的客户端 `AutoShell` 会话，而是使用 `createRemoteContext()` 创建的永久会话。另外，在运行脚本时创建的任何上下文信息可用于稍后执行的任何远程脚本。在收集 `RemoteTarget` 对象时，将自动销毁远程主机上的永久客户端 `AutoShell` 会话。也可通过在对应的对象上调用 `destroyRemoteContext()` 来显式销毁会话。

使用以下示例可轻松验证使用 `createRemoteContext()` 的效果:

```
var rt = new RemoteTarget("ascll");
rt.createRemoteContext()
run-remote X=5 on rt -wait
run-remote X on rt -wait
? get-remoteResult(rt)
rt.destroyRemoteContext()
run-remote X=5 on rt -wait
run-remote X on rt -wait
? get-remoteResult(rt)
```

第二个命令调用返回 5, 因为在第一个调用中创建的变量 `X` 仍然存在于永久客户端 `AutoShell` 会话中。销毁远程客户端会话后, 剩下的两个调用将触发创建两个临时客户端 `AutoShell` 会话的操作。第二个命令返回空结果, 因为在其他会话中创建的变量 `X` 已被销毁。

### 详细信息

[RemoteTarget 类](#) (p. 135)

## AutoShell 核心参考

本节介绍了特定于 `AutoShell` 核心的命令、函数和类。`ALM` 命令参考一节介绍了属于特定于平台的扩展的其他命令。

`AutoShell` 命令分为 `cmdlet` 和 `funclet` 两组:

### Cmdlet

指定命令定义, 这些命令定义必须在一行中以第一个非空格字符开头。因为该限制, 它们只能单独使用, 不能作为范围更广的 `JavaScript` 表达式的一部分进行使用。需要特别指出的是, 它们不能用作 `rvalue` (赋值运算符右边的运算对象)。? 是一个 `AutoShell cmdlet` 示例。

### Funclet

使用可选子句、字符串化等维护详细的类似命令的语法, 也可以返回值。通常, `funclet` 的用法类似于 `cmdlet`, 即, 单独位于一行中。但是, 它们能够返回可作为更广泛表达式的一部分进行处理的值。



## 根据类别列出的命令、函数和类

AutoShell 提供了以下特定于 CA Virtual Assurance 的命令（cmdlet、funclet）、函数和类。使用这些命令可创建适当的 AutoShell 应用程序来管理您的虚拟环境。

### Cmdlet

!、!!、?、??、accept、arrdump、cat、cd、DBG\_PROMPT、DBG\_PROMPT\_LINE、exit、external、get-autoShellClassInfo、get-webServiceInfo、help、mkdir、objdump、pwd、quit、set alternate to、set console、set result disp、start-java、type、wait、weak external

### Funclet

add、ASSERT、associate、chdir、clone、copy、cp、create、custom、del、dir、get-help、get-remoteResult、install、ls、modify、move、mv、new-SSHSession、new-ZONESession、new-comObject、new-webService、push-client、push-winRemote、push-winRemote、PW\_GET、query-service、reboot、rebuild、rem-client、remove、ren、rename、reset、rm、rmdir、run-SSHCommand、run-SSHShell、run-ZONECommand、run-client、run-local、run-remote、run-winRemote、show、shutdown、start、start-service、status、stop、stop-service、uninstall

### 函数

base64Decode、base64Encode、curDir、delete、disable、enable、gete、memoRead、memoWrit、platform、pute、qout、qqout、regCreateKey、regCreateSubkeys、regDeleteKey、regDeleteVal、regGetKeyValues、regGetSubKey、regGetVal、regIsKey、regIsVal、reSetKeyValues、regSetVal、setProcExitCode、shellType、typeName

### 类

OSRedirect、RemoteTarget

可根据下列条件对 AutoShell 命令、函数和类进行分类：

### 数组

arrdump

### AutoShell 部署

exit、quit、push-client、rem-client、run-client、setProcExitCode、shellType

## 类和对象

new-comObject、get-autoShellClassInfo、get-remoteResult、objdump、OSRedirect、RemoteTarget、run-local、run-remote、typeName

## 命令执行

!、!!、get-remoteResult、OSRedirect、push-winRemote、RemoteTarget、run-local、run-remote、run-SSHCommand、run-SSHShell、run-winRemote、run-ZONECommand

## 数据类型

typeName

## 加密

base64Decode、base64Encode

## 环境变量

gete、pute

## 文件和目录

cat、cd、chdir、copy、cp、curDir、del、dir、ls、mkdir、mv、pwd、ren、rm、rmdir、type

## 帮助

get-help、help

## 输入和输出

?、??、accept、DBG\_PROMPT、DBG\_PROMPT\_LINE、PW\_GET、qout、qqout、set alternate to、set console、set result disp、wait

## 数学表达式

ASSERT

## 杂项

external、start-java、weak external

## 平台

push-winRemote、run-LPARCommand、run-winRemote、run-ZONECommand、platform

## 注册表

regCreateKey、regCreateSubkeys、regDeleteKey、regDeleteVal、regGetKeyValues、regGetSubKey、regGetVal、regIsKey、regIsVal、regSetKeyValues、regSetVal

## 远程访问

get-remoteResult、push-winRemote、RemoteTarget、run-remote、run-SSHCommand、run-SSHShell、run-winRemote

## 服务

get-webServiceInfo、new-webService、query-service、start-service、stop-service

## 会话

new-SSHSession、new-ZONESession

## Solaris Zones

associate、clone、create、custom、delete、disable、enable、install、modify、move、reboot、rename、show、start、status、stop、uninstall

## 字符串

accept、PW\_GET、memoRead、memoWrit、typeName

## 常规 AutoShell 命令

本节按字母顺序详细介绍了常规 AutoShell 核心命令（cmdlet、funclet）。这些常规命令不特定于 LPAR 或 Solaris Zones 平台。AutoShell 定义了用于简化相应任务的调用过程的一系列命令。

以下常规 JavaScript 函数调用可获取 C:\ 的详细目录列表：

```
ca.aip.direct.directory("C:\\*.*" , "", false, false, false, true)
```

相应的 AutoShell 命令类似如下：

```
dir C:\\*.* -l
```

### 示例

列出控制台中当前目录的内容：

```
dir *.*
```

将目录列表分配给数组：

```
arr = dir *.* -retval
```

## !! 命令—调用命令或子进程并自动捕获输出 (Cmdlet)

运行操作系统命令，或启动应用程序作为子进程。用户指定的 `OSRedirect` 对象或默认系统变量 (`$Stdout`) 会自动捕获子进程输出。

命令的语法如下：

```
!! cmd [-output osRedirect]
```

### **cmd**

要执行的命令或要运行的应用程序。可使用 `&&` 序列连结多个命令。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

### **osRedirect**

（可选）要接收子进程输出并在指定变量中将其捕获的现有 `OSRedirect` 对象。

**默认值：** `$Stdout`

### 示例

使用重定向到未在调用之前初始化的变量 `out` 的输出，获取有关 Windows 操作系统的 DOS 内存信息：

```
!! mem -output out
// 分行输出
aLines = out.output().split(String.fromCharCode(10));
arrdump aLines
```

执行 DNS 查找并自动捕获 `$Stdout` 的输出：

```
!! nslookup ca.com
? $Stdout.output()
```

**另请参阅：**

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[OSRedirect 类](#) (p. 126)

## ! 命令—调用命令或子进程 (Cmdlet)

运行操作系统命令，或启动应用程序作为子进程。默认情况下，子进程产生的输出将写入到 AutoShell 控制台中。可以将输出重定向到 AutoShell OSRedirect 对象。

命令的语法如下：

```
! cmd [-output osRedirect]
```

### cmd

指定要执行的命令或要运行的应用程序。可使用 && 序列连结多个命令。未加引号的参数令牌会自动字符串化。通过将表达式代码放到括号中来防止自动引用表达式。

### osRedirect

（可选）要接收子进程输出的现有 OSRedirect 对象。

默认值：""

### 示例

显示 Windows OS 的版本字符串：

```
! ver
```

显示 Windows OS 的版本字符串并显示当前目录：

```
! ver && cd
```

对网络上的系统执行 Ping 操作并捕获输出：

```
out=new OSRedirect();
! ping 192.168.0.100 -output out
if(out.result()==0)
{
    ? out.output();
}
else
{
    ? "Ping failed"
}
```

另请参阅：

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect 类](#) (p. 126)

## ?? 命令—将列表中的输出写入到 `stdout` (Cmdlet)

将列表中每个参数的字符串表示写入到 `stdout`。会在每个显示的参数之间自动添加一个空格。使用 `?`、`??`、`qout()` 或 `qqout()` 的后续输出调用将其输出紧接着放在原始输出的后面，因此 `??` 或 `qqout()` 通常用于构造使用多个调用的输出行。

命令的语法如下：

```
?? [x]
```

**x**

(可选) 要输出的参数列表。

### 示例

输出以空格分隔的三个列标题：

```
?? "Col1"
?? " "
?? "Col2"
?? " "
? "Col3"
```

使用 `?` 命令输出第三个标签， 以将光标移至下一行的开头。

另请参阅：

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)  
[OSRedirect 类](#) (p. 126)

## ? 命令—将输出写入到 `stdout` (Cmdlet)

该命令 (`cmdlet`) 将列表中每个参数的字符串表示写入到 `stdout`。在每个参数之间自动放置一个空格。`set console` 和 `set alternate` 命令可重定向或停止输出。输出以换行符终止。

命令的语法如下：

```
? [x]
```

**x**

(可选) 要输出的参数列表。如果列表为空，则仅输出一个换行符。

### 示例

输出 “Hello World!” :

```
? "Hello World!"
```

输出数字列表:

```
? 1,2,3
```

输出空行:

```
?
```

输出数字 1 到 10。? 命令只能在行的开头使用，要输出表达式内部的文本，必须使用相应的 `qout()` 函数:

```
for(i=1;i<11;i++) qout(i);
```

另请参阅:

[?? 命令—将列表中的输出写入到 stdout \(Cmdlet\)](#) (p. 46)

[qqout—将输出写入 stdout \(函数\)](#) (p. 109)

[qout—将后跟换行符的输出写入 stdout \(函数\)](#) (p. 108)

[set alternate to 命令—设置备用输出文件 \(Cmdlet\)](#) (p. 97)

[set console 命令—停止控制台输出 \(Cmdlet\)](#) (p. 98)

## accept 命令—从 stdin 读取字符串并将其分配给变量 (Cmdlet)

可以提示用户输入、读取字符直到遇到回车并将不带回车的字符串分配给变量。

命令的语法如下:

```
accept [prompt] to var [-hidden]
```

### **prompt**

(可选) 指定要显示的字符串。

### **var**

要将包含读取字符的字符串分配到的变量的名称。

### **-hidden**

(可选) 遮蔽输入字符串以输入密码。

### 示例

读取字符串并将其分配给变量名:

```
accept "Enter your name: " to name
```

遮蔽输入内容以输入密码:

```
accept "Enter password:" to password -hidden
```

另请参阅:

[wait 命令—等待按键 \(Cmdlet\)](#) (p. 101)

[PW GET 命令-输入隐藏数据 \(Funclet\)](#) (p. 78)

## arrdump 命令—显示数组 (Cmdlet)

该命令可列出一维数组的所有元素。

命令的语法如下:

```
arrdump arr
```

***arr***

定义要显示的数组。

### 示例

显示数组:

```
a = [1,2,"a", true, new Date()];  
arrdump a
```

另请参阅:

[objdump 命令-显示对象的可枚举属性 \(Cmdlet\)](#) (p. 73)

## ASSERT 命令—比较数学表达式 (Funclet)

该命令 (funclet) 检查包含两个数学表达式和一个运算的语句或断言是 **true** 还是 **false**。数学表达式可以包括以下元素:

- 函数
- 操作
- 小数
- 十进制、八进制或十六进制格式的数值

如果语句或断言为 **false**, AutoShell 会上报可自定义的错误消息。



要写入八进制格式的数字，请在值前面加上一个 0，例如 0123。要写入十六进制格式的数字，请在值前面加上一个 0x，例如 0x123。

命令的语法如下：

```
ASSERT x relation v [-msg msg]
```

**x**

定义与表达式 *v* 进行比较的数学表达式。

**relation**

指定用于比较的关系。选项包括：

- eq（等于）
- ne（不等于）
- gt（大于）
- lt（小于）
- ge（大于或等于）
- le（小于或等于）

**v**

定义数学表达式。

**-msg *msg***

（可选）定义在比较返回 **false** 时上报备用错误消息而不是默认错误消息。

**默认值：** 空

## 示例

显示语句 6>7 错误：

```
ASSERT 6 gt 7 -msg "This is wrong."
这是错误的。
```

确认八进制值 0123 是否等于 0x9e：

```
ASSERT 0123 eq 0x9e
Failed: 0123==158, actual=83
```

检查 PI 是否大于 4.5\*sin(PI/2)：

```
ASSERT Math.PI gt 4.5*Math.sin(Math.PI/2)
Failed: Math.PI>4.5, actual=3.141592653589793
```

## cat、type 命令—显示文本文件 (Cmdlet)

这些命令 (cmdlet) 可将一个或多个文本文件的内容写入标准输出流。使用 `set alternate` 连结多个文件。

这些命令具有以下语法：

```
{cat|type} fileList
```

### **fileList**

定义要显示的绝对文件名或相对文件名的逗号分隔列表。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

### 示例

显示 Windows 引导选项：

```
cat c:\boot.ini
```

连结两个文件：

```
set alternate to output.txt  
cat input1.txt, input2.txt  
set alternate to
```

另请参阅：

[memoRead—将文本文件读入字符串（函数）](#) (p. 105)

## cd、chdir 命令—更改目录 (cmdlet、funclet)

这些命令在本地系统上更改当前目录。如果提供 `p`，将定义新的目录。如果 `p` 是相对路径，该命令将使用当前路径作为目录的前缀。

命令的语法如下：

```
cd [p] (Cmdlet)  
chdir [p] (Funclet)
```

### **p**

（可选）定义要更改到的目录。Windows 上的分隔符可以是反斜杠 (\) 或正斜杠 (/)。对于 UNIX 类系统，分隔符必须是正斜杠。为实现跨平台脚本兼容性，仅在路径规格中使用正斜杠。

**默认值：** ""

### 示例

更改到根目录:

```
cd /
```

更改到目录 `c:/test`

```
cd c:/test
```

更改到父目录:

```
cd ..
```

另请参阅:

[dir、ls 命令—获取文件和目录信息 \(Funclet\)](#) (p. 58)

[mkdir 命令-创建目录 \(Cmdlet\)](#) (p. 65)

[pwd 命令-输出工作目录的名称 \(Cmdlet\)](#) (p. 79)

[curDir—检索当前目录（函数）](#) (p. 103)

## copy、cp 命令—复制文件和目录 (Funclet)

这些命令 (funclet) 将文件或目录复制到其他位置。源位置或目标位置必须位于本地服务器上。两个远程系统之间的操作不受支持。要指定远程主机，需使用主机名后接两个冒号 (::) 作为路径名的前缀。通过 SFTP 执行该操作。因此，远程系统不一定要安装客户端 AutoShell，具有 SFTP 访问权的 SSH 服务器已足够。可以使用 `set alternate` 命令将输出写入文件。

命令的语法如下：

```
{copy|cp} src trg [-silent] [-retval] [-R] [-user username] [-pass password] [-key key phrase] [-port portnumber]
```

### src

定义用于指定要复制的目录或文件的搜索掩码。Windows 上的分隔符可以是反斜杠 (\) 或正斜杠 (/)。对于 UNIX 类系统，分隔符必须是正斜杠。为实现跨平台脚本兼容性，仅在路径规格中使用正斜杠。

搜索掩码使用与 DOS 通配符匹配兼容的简化正则表达式语法：

- \* 匹配任何字符序列，包括零个字符。
- ? 精确匹配一个字符。
- [abc] 精确匹配 a、b 或 c 中的一个字符。
- [a-f] 匹配从 a 至 f 的任何字符。
- [^a-f] 匹配除 a 至 f 以外的任何字符。
- [-\_] 匹配 - 或 \_；[^\_] 匹配除 - 和 \_ 以外的字符。短划线 “-” 紧跟着左括号或 ^ 出现时，不是特殊字符。
- [a^] 匹配 a 或 ^。（脱字号 “^” 没有紧跟着左括号出现时，不是特殊字符。
- \\*、\?、\|、\、\\ 匹配单个字符 \*、?、|、\。

其他所有字符都不是特殊字符且与自己匹配。

### trg

定义目标位置的路径。

### -silent

（可选）禁止输出至活动的输出数据流。

### -retval

（可选）创建表示返回值的数组。数组包含有关新文件名的信息。每个元素都包含一个表示目标文件的完全限定名称的字符串。

### -R

（可选）递归复制目录。

**-user username**

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass password**

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-key key**

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，AutoShell 将首先尝试公钥登录，如果失败，则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

**默认值：** ""

**短语**

指定私钥的密码。如果密钥没有加密，则不需要该密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** ""

**-port portnumber**

（可选）连接到目标系统的端口。

**默认值：** SSH 标准端口 22。

### 示例

将文件 `c:/test/test1.log` 复制到目录 `c:/test1`

```
copy c:/test/test1.log c:/test1
```

将整个 `c:/test` 目录复制到目录 `c:/test1`

```
cp c:/test c:/test1 -R
```

将远程主机 `remotesys` 的整个 `c:/test` 目录复制到目录 `c:/test1` 中

```
cp c:/test1/ remotesys::c:/test1/ -R
```

另请参阅：

[mv、ren 命令-移动文件和目录 \(Funclet\)](#) (p. 67)

## DBG\_PROMPT 命令—中断脚本执行以进行检查 (Cmdlet)

**DBG\_PROMPT** 命令 (`cmdlet`) 可中断脚本执行，并显示交互式提示以在当前脚本上下文中评估表达式。通常用于检查当前变量值，调试提示会重复出现，直到输入空输入行为止。

命令的语法如下：

```
DBG_PROMPT [prompt]
```

### **prompt**

指定带有要显示的可选提示的字符串。在使用多个提示时，提示可用于指示触发中断的位置。

**默认值：** “`Dbg>`”

### 示例

在最后一次循环迭代期间中断：

```
for(i=0; i<11; i++)
{
    if(i==10)
        DBG_PROMPT
        ? i
}
```

使用自定义提示中断：

```
for(i=0; i<11; i++)
{
    if(i==10)
        DBG_PROMPT "i==10> "
        ? i
}
```

另请参阅：

[DBG\\_PROMPT\\_LINE 命令—中断脚本执行以进行带行号的检查 \(Cmdlet\)](#) (p. 55)

## DBG\_PROMPT\_LINE 命令—中断脚本执行以进行带行号的检查 (Cmdlet)

DBG\_PROMPT\_LINE 命令 (cmdlet) 中断脚本执行，并显示交互式提示以在当前脚本上下文中评估表达式。提示会自动包括当前的行号。通常用于检查当前变量值，调试提示会重复出现，直到输入空输入行为止。

命令的语法如下：

DBG\_PROMPT\_LINE [*prompt*]

### prompt

指定带有要显示的可选提示的字符串。在使用多个提示时，提示可用于指示触发中断的位置。如果指定了提示，将显示 “prompt(n)>”

**默认值：** “Dbg(n)>”

**否**

显示行号作为输入提示。

### 示例

在最后一次循环迭代期间中断：

```
for(i=0; i<11; i++)
{
    if(i==10)
        DBG_PROMPT_LINE
        ? i
}
```

使用自定义提示中断：

```
for(i=0; i<11; i++)
{
    if(i==10)
        DBG_PROMPT_LINE "i==10> "
        ? i
}
```

另请参阅：

[DBG PROMPT 命令—中断脚本执行以进行检查 \(Cmdlet\)](#) (p. 54)

## del、rm 命令—删除文件 (Funclet)

这些命令 (funclet) 删除文件或目录。要指定远程主机，需使用主机名后接两个冒号 (::) 作为路径名的前缀。通过 **SFTP** 执行该操作。因此，远程系统不一定要安装客户端 **AutoShell**，具有 **SFTP** 访问权的 **SSH** 服务器已足够。

命令的语法如下：

```
{del|rm} src [-silent] [-retval] [-R] [-user username] [-pass password] [-key key phrase] [-port portnumber]
```

### **src**

定义要删除的文件或目录。

### **-silent**

（可选）禁止输出至活动的输出数据流。

### **-retval**

（可选）创建表示返回值的数组。数组包含有关新文件名的信息。每个元素都包含一个表示目标文件的完全限定名称的字符串。

### **-R**

（可选）递归复制目录。

### **-user *username***

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 **AutoShell** 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

### **-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 **AutoShell** 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 **PW\_GET()**。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass



**-key *key***

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，AutoShell 将首先尝试公钥登录，如果失败，则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

**默认值：** ""

**短语**

指定私钥的密码。如果密钥没有加密，则不需要该密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** ""

**-port *portnumber***

（可选）连接到目标系统的端口。

**默认值：** SSH 标准端口 22。

**示例**

删除文件 `c:/test/test1.log`：

```
del c:/test/test1.log
```

删除整个 `c:/test` 目录：

```
rm c:/test -R
```

删除远程主机 `remotesys` 上的整个 `c:/test` 目录：

```
del remotesys::c:/test1/ -R
```

## dir、ls 命令—获取文件和目录信息 (Funclet)

这些命令 (funclet) 列出有关本地或远程系统上指定目录中内容的信息。可以使用 `set alternate to` 命令将输出写入文件中。

这些命令具有以下语法：

```
{dir|ls} mask [-silent] [-retval] [-l] [-user username] [-pass password] [-key key phrase] [-port portnumber]
```

### mask

定义带有搜索掩码的字符串，掩码可以选择包含绝对或相对目录规格。**Windows** 上的分隔符可以是反斜杠 (\) 或正斜杠 (/)。对于 **UNIX** 类系统，分隔符必须是正斜杠。为实现跨平台脚本兼容性，仅在路径规格中使用正斜杠。

搜索掩码使用与 **DOS** 通配符匹配兼容的简化正则表达式语法：

- \* 匹配任何字符序列，包括零个字符。
- ? 精确匹配一个字符。
- [abc] 精确匹配 a、b 或 c 中的一个字符。
- [a-f] 匹配从 a 至 f 的任何字符。
- [^a-f] 匹配除 a 至 f 以外的任何字符。
- [-\_] 匹配 - 或 \_；[^\_] 匹配除 - 和 \_ 以外的字符。短划线 “-” 紧跟着左括号或 ^ 出现时，不是特殊字符。
- [a^] 匹配 a 或 ^。（脱字号 “^” 没有紧跟着左括号出现时，不是特殊字符。
- \\*、\?、\[ \]、\\ 匹配单个字符 \*、?、[ ]、\。

其他所有字符都不是特殊字符且与自己匹配。

**默认值：** ""

### -silent

（可选）禁止输出至活动的输出数据流。

### -retval

（可选）创建表示返回值的数组。数组包含文件信息。每个元素均包含表示文件名的字符串。如果指定了参数 -l，则每个元素均由包含以下各项的数组构成：

- item[0]—表示文件名的字符串
- item[1]—表示文件大小的数字
- item[2]—表示上次修改时间的日期
- item[3]—表示目录标志的布尔值
- item[4]—表示完全限定文件名（包括路径）的字符串

**-l**

（可选）为输出使用长列表格式。

**-user *username***

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-key *key***

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，AutoShell 将首先尝试公钥登录，如果失败，则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

**默认值：** ""

### 短语

指定私钥的密码。如果密钥没有加密，则不需要该密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** ""

**-port *portnumber***

（可选）连接到目标系统的端口。

**默认值：** SSH 标准端口 22。

### 示例

获取当前目录的文件名:

```
dir
```

获取目录 c:\test 的文件名:

```
dir c:/test
```

获取目录 c:\test 和 c:\test\more 的文件名作为文件 test.list 的内容:

```
set alternate to test.list
dir c:\test
dir c:\test\more
set alternate to
```

列出远程服务器 ascli1 上 c:\temp 的目录内容:

```
dir ascli1::c:/temp
```

获取目录 c:/test/more 的文件名作为数组以进行进一步处理并停止输出到 stdout:

```
set result disp off
aRet = dir c:/test/more -silent -retval
```

获取目录 c:/test/more 中文件的扩展信息作为数组以进行进一步处理:

```
aRet = dir c:/test/more -silent -retval -l
```

另请参阅:

[cd、chdir 命令—更改目录 \(cmdlet、funclet\)](#) (p. 50)

[pwd 命令-输出工作目录的名称 \(Cmdlet\)](#) (p. 79)

[set alternate to 命令—设置备用输出文件 \(Cmdlet\)](#) (p. 97)

## exit、quit 命令—终止 AutoShell 会话 (Cmdlet)

这些命令 (cmdlet) 可终止当前的 Autoshell 会话。在交互式命令提示中或在脚本内发出命令。

命令的语法如下:

```
exit
quit
```

另请参阅:

[setProcExitCode—设置 AutoShell 退出代码 \(函数\)](#) (p. 122)

## external 命令—声明本地外部函数 (Cmdlet)

声明本地外部函数。

命令的语法如下：

```
external decl
```

**decl**

指定本地外部函数。

## get-autoShellClassInfo 命令—显示本地 AutoShell 类的方法和属性 (Cmdlet)

此命令 (cmdlet) 显示本地 AutoShell 类的方法和属性。AutoShell 定义了大量用于扩展标准 Spidermonkey/JavaScript 脚本环境的类。特定于产品的 AutoShell 可加载模块 (ALM) 可以注册其他类。这些类包含有关其构造函数、方法、变量和枚举的类型信息。可以使用 `get-autoShellClassInfo cmdlet` 显示这些信息。

**注意：** 这些信息不同于帮助信息。AutoShell 仅基于内部类定义生成这些信息。

命令的语法如下：

```
get-autoShellClassInfo class [-super]
```

**类**

指定要显示信息的 AutoShell 类。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。可以传递对象名，而不是类名。在这种情况下，`get-autoShellClassInfo` 命令会从对象检索类名并尝试查询类信息。为防止字符串化，必须将对象表达式放在括号内进行传递。

### 示例

显示 RemoteTarget 类的详细信息：

```
get-autoShellClassInfo RemoteTarget
```

显示对象中 OSRedirect 类的详细信息：

```
os = new OSRedirect();  
get-autoShellClassInfo (os)
```

另请参阅：

[help 命令-请求帮助信息 \(Cmdlet\)](#) (p. 64)

[get-help 命令-获取帮助信息 \(Funclet\)](#) (p. 62)

## get-help 命令-获取帮助信息 (Funclet)

该命令 (Funclet) 会显示 AutoShell 命令、功能和类的帮助信息，并可将其重定向到文件。

命令的语法如下：

```
get-help [mask] [-output outFile]
```

### **mask**

（可选）指定掩码，用于筛选出需要帮助信息的命令。您可以使用星号 “\*” 和问号 “?” 作为通配符。

**默认值：** "help"

### **outFile**

（可选）指定文件的绝对路径或相对路径以包含帮助信息。将覆盖具有相同名称和路径的任何现有文件。

**默认值：** ""

## 示例

显示 run-remote funclet 的帮助：

```
help run-remote
```

显示 RemoteTarget 类（包括所有构造函数和方法）的帮助：

```
help RemoteTarget*
```

列出所有帮助主题并将它们写入文件：

```
help * -output help.txt
```

**另请参阅：**

[help 命令-请求帮助信息 \(Cmdlet\)](#) (p. 64)

[get-autoShellClassInfo 命令-显示本地 AutoShell 类的方法和属性 \(Cmdlet\)](#) (p. 61)

## get-remoteResult 命令-从远程目标获取结果 (Funclet)

此 Funclet 从 RemoteTarget 对象检索远程执行结果。远程执行的返回值必须通过网络传输回发起该请求的系统。在远程系统上将该结果序列化为 XML 表示，RemoteTarget.result() 方法将会以序列化表示返回结果值。

使用 get-remoteResult() funclet（它将 RemoteTarget 对象本身作为参数）将序列化表示形式转换为 JavaScript 正则值。通常脚本不需要直接调用 result()。序列化会保留类型信息，因此远程节点返回数字时，get-remoteResult() 会返回一个数字类型的值。如果远程系统返回“日期”对象，那么该对象在原始系统上同样为日期。

远程脚本的返回值并不仅限于简单数据类型。也可以返回复杂数组或仅包含数据的对象，以允许远程执行在服务器之间传输大量数据。唯一的限制是数据结构不能包含任何循环引用。

命令的语法如下：

```
get-remoteResult remoteTarget
```

funclet 从远程执行进程返回一个多态值。如果执行尚未完成，则返回值属于未定义类型。如果发生其他错误，则会创建异常。

### remoteTarget

指定远程目标系统的名称。

### 示例

获取远程执行的序列化表示形式和实际值：

```
rt = new RemoteTarget("ascli1", "bob", "ca123456");
run-remote "new Date()" on rt -wait
? rt.result()           // XML 字符串
v = get-remoteResult(rt) // 内部调用 result()
? v
? typeName(v)           // 日期
```

另请参阅：

[RemoteTarget 类](#) (p. 135)

[RemoteTarget.output 方法](#) (p. 145)

[RemoteTarget.result 方法](#) (p. 147)

## get-webServiceInfo 命令-显示 WSDL 信息 (Cmdlet)

get-webServiceInfo 会显示 WSDL 信息。

```
get-webServiceInfo w [-detail]
```

**w**

指定 Web 服务的名称。

## help 命令-请求帮助信息 (Cmdlet)

该命令 (Cmdlet) 会显示 AutoShell 命令、函数和类的帮助信息，并可将其重定向到文件。

命令的语法如下：

```
help [mask] [-output outFile]
```

**mask**

(可选) 指定掩码，用于筛选出需要帮助信息的命令。您可以使用星号 “\*” 和问号 “?” 作为通配符。

**默认值:** "help"

**outFile**

(可选) 指定文件的绝对路径或相对路径以包含帮助信息。将覆盖具有相同名称和路径的任何现有文件。

**默认值:** ""

### 示例

显示 run-remote funclet 的帮助：

```
help run-remote
```

显示 RemoteTarget 类（包括所有构造函数和方法）的帮助：

```
help RemoteTarget*
```

列出所有帮助主题并将它们写入文件：

```
help * -output help.txt
```

另请参阅：

[get-autoShellClassInfo 命令-显示本地 Autoshell 类的方法和属性 \(Cmdlet\)](#) (p. 61)

[get-help 命令-获取帮助信息 \(Funclet\)](#) (p. 62)



## mkdir 命令-创建目录 (Cmdlet)

如果目录不存在，该命令会在本地系统或远程系统上创建一个目录。

该函数会自动创建指向指定目录位置的路径元素。要指定远程主机，需使用主机名后接两个冒号 (:) 作为路径名的前缀。通过 SFTP 执行该操作。因此，远程系统不一定要安装客户端 AutoShell，具有 SFTP 访问权的 SSH 服务器已足够。

命令的语法如下：

```
mkdir d [-silent] [-user username] [-pass password] [-key key phrase] [-port portnumber]
```

**d**

定义新目录的名称。

**-silent**

（可选）该命令会停止任何对 `stdout` 的输出。

**-user username**

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass password**

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-key *key***

(可选) 指定包含 RSA 私钥的文件的绝对路径或相对路径, 用于登录到远程节点。如果指定了私钥和密码, AutoShell 将首先尝试公钥登录, 如果失败, 则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值: ""

**短语**

指定私钥的密码。如果密钥没有加密, 则不需要该密码。要在发出该命令之后输入隐藏的密码, 请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: ""

**-port *portnumber***

(可选) 连接到目标系统的端口。

默认值: SSH 标准端口 22。

**示例**

创建目录 `c:\temp\xyz`:

```
mkdir c:\temp\xyz
```

在远程服务器 `ascl1` 上创建一个目录:

```
mkdir ascl1:~/home/alice/data
```

创建一个临时目录作为当前工作目录的子目录, 但是停止任何对 `stdout` 的输出:

```
mkdir temp -silent
```

另请参阅:

[cd、chdir 命令—更改目录 \(cmdlet、funclet\)](#) (p. 50)

[dir、ls 命令—获取文件和目录信息 \(Funclet\)](#) (p. 58)

[rmdir 命令—删除目录 \(Funclet\)](#) (p. 82)

[pwd 命令—输出工作目录的名称 \(Cmdlet\)](#) (p. 79)

## mv、ren 命令-移动文件和目录 (Funclet)

这些命令将文件或目录移动到其他位置。源或目标位置必须在本地服务器上。两个远程系统之间的操作不受支持。要指定远程主机，需使用主机名后接两个冒号 (:) 作为路径名的前缀。通过 SFTP 执行该操作。因此，远程系统不一定要安装客户端 AutoShell，具有 SFTP 访问权的 SSH 服务器已足够。可以使用 set alternate 命令将输出写入文件。

在这种情况下，移动或重命名是等效的。

这些命令具有以下语法：

```
{mv|ren} src trg [-silent] [-retval] [-R] [-user username] [-pass password] [-key key phrase] [-port portnumber]
```

### src

定义一个用于指定要移动的目录或文件的搜索掩码。Windows 上的分隔符可以是反斜杠 (\) 或正斜杠 (/)。对于 UNIX 类系统，分隔符必须是正斜杠。为实现跨平台脚本兼容性，仅在路径规格中使用正斜杠。

搜索掩码使用与 DOS 通配符匹配兼容的简化正则表达式语法：

- \* 匹配任何字符序列，包括零个字符。
- ? 精确匹配一个字符。
- [abc] 精确匹配 a、b 或 c 中的一个字符。
- [a-f] 匹配从 a 至 f 的任何字符。
- [^a-f] 匹配除 a 至 f 以外的任何字符。
- [-\_] 匹配 - 或 \_；[^\_] 匹配除 - 和 \_ 以外的字符。短划线 “-” 紧跟着左括号或 ^ 出现时，不是特殊字符。
- [a^] 匹配 a 或 ^。（脱字号 “^” 没有紧跟着左括号出现时，不是特殊字符。
- \\*、\?、\[、\]、\\ 匹配单个字符 \*、?、[、]、\。

其他所有字符都不是特殊字符且与自己匹配。

### trg

指定目标位置的路径。

### -silent

（可选）禁止输出至活动的输出数据流。

### -retval

（可选）创建表示返回值的数组。数组包含有关新文件名的信息。每个元素都包含一个表示目标文件的完全限定名称的字符串。

### -R

（可选）递归复制目录。

**-user *username***

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-key *key***

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，AutoShell 将首先尝试公钥登录，如果失败，则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

**默认值：** ""

**短语**

指定私钥的密码。如果密钥没有加密，则不需要该密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** ""

**-port *portnumber***

（可选）连接到目标系统的端口。

**默认值：** SSH 标准端口 22。

### 示例

将文件 `c:/test/test1.log` 移动到目录 `c:/test1`:

```
mv c:/test/test1.log c:/test1
```

将整个 `c:/test` 目录移动到目录 `c:/test1`:

```
mv c:/test c:/test1 -R
```

将整个 `c:/test` 目录移动到远程主机 `remotesys` 的目录 `c:/test1` 中:

```
mv c:/test1/ remotesys::c:/test1/ -R
```

另请参阅:

[copy、cp 命令—复制文件和目录 \(Funclet\)](#) (p. 52)

## new-comObject 命令-实例化 COM 对象 (Funclet)

COM 是一个 Windows 特定的标准化 API，允许应用程序显示供 COM 客户端使用的服务。COM 客户端支持允许访问各种 COM 服务器，如 Internet Explorer、Microsoft Office 或与管理相关的各种操作系统服务，尤其是 WMI。

`new-comObject` 命令可创建代理客户端对象，这些对象使 COM 服务器显示的服务可用于编写脚本。

命令的语法如下:

```
new-comObject progid [-locale lcid]
```

### progid

指定一个用于标识要实例化的 COM 对象的字符串。该字符串可以是用户可读的程序 ID (PROGID)，也可以是内部 CLSID。指定 CLSID 时，必须将其置于大括号内。

### lcid

指定一个用于定义 COM 服务器（支持多个区域设置）所需的区域设置的整数。

**默认:** 0

### 返回值

`new-comObject` 返回一个表示 AutoShell 和 COM 服务器之间代理的对象。该对象对 COM 服务器显示的方法和属性进行镜像。有关某个 COM 服务器的公共接口的详细信息，请参阅该服务器的程序员文档。使用返回的 `AutoObject` 的正则 JavaScript 语法访问显示的方法和属性。

## 示例

使用 Internet Explorer 打开一个网页并查询其 URL:

```
o = new-comObject InternetExplorer.Application
// 使 Internet Explorer 可见
o.Visible = true;
o.navigate("www.ca.com");
// 获取活动文档 (另一个 COM 对象)
doc = o.document
? doc.Url      // 例如 http://www.ca.com/us
```

使用脚本化对象获取计算机名称:

```
o = new-comObject WScript.Network
? o.ComputerName
```

使用 WMI 检索有关操作系统和逻辑驱动器的信息:

```
loc=new-comObject WbemScripting.SWbemLocator
// 对于 localhost, 实际上并不需要安全性
loc.Security_.AuthenticationLevel=0; // wbemAuthenticationLevelDefault
loc.Security_.ImpersonationLevel=3; // wbemImpersonationLevelImpersonate
wmisrv=loc.connectServer("localhost", "root\cimv2");
items = wmisrv.ExecQuery("Select * from Win32_OperatingSystem");
// 返回的集合只包含一项,
// 因为只有一个运行的 OS
os = items[0];
? "OS Type:", os.Caption;
? "Service Pack:", os.CSDVersion;
? "Computer name:", os.CSName;
? "FreePhysicalMemory:", os.FreePhysicalMemory;
? "RegisteredUser:", os.RegisteredUser;
? "TotalVisibleMemorySize:", os.TotalVisibleMemorySize;
? "Version:", os.Version;
? "WindowsDirectory:", os.WindowsDirectory;
// 现在对于磁盘
items = wmisrv.ExecQuery("Select * from Win32_LogicalDisk");
// 针对返回的磁盘迭代
for(i in items)
{
    disk = items[i];
    ? "Description:", disk.Description;
    ? "DeviceID:", disk.DeviceID;
    ? "FileSystem:", disk.FileSystem;
    ? "FreeSpace:", disk.FreeSpace;
    ? "Name:", disk.Name;
    ? "Size:", disk.Size;
}
```

## new-SSHSession 命令-创建一个新的安全 SSH 会话 (Funclet)

`new-SSHSession` 命令在 SSH 服务器上创建会话，并返回一个表示该会话的 CASSHELL 对象。必须保存该对象以便今后对该会话执行操作。通常和 `run-SSHCommand` 或 `run-SSHShell` 命令一起使用。

命令的语法如下：

```
new-SSHSession -host hostname [-user username] [-pass password] [-key key]  
[-prompt prompt][-port portnumber]
```

### **-host *hostname***

SSH 服务器的名称。

### **-user *username***

（可选）指定用于登录到远程节点的用户名。使用密码或公共密钥身份验证时，需要用户名。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：\$\$User

### **-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：\$\$Pass

### **-key *key***

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，则 AutoShell 会首先尝试公钥登录，如果登录失败，则使用密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值：""

**-prompt *prompt***

(可选) 指定 SSH 服务器显示的提示字符串结尾的固定部分, 以指示其可用于输入 (例如: 用于 AutoShell 的 “::->”)。需要该信息以确定在 SSH 会话期间执行的命令的完成情况。指定尽可能多的固定字符。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值:** “::->”

**-port *portnumber***

(可选) 连接到目标系统的端口。

**默认值:** SSH 标准端口 22。

**返回值**

返回一个 CASSHELL 对象。该对象可以传递到 run-SSHCommand funclet 以执行命令, 或者传递到 run-SSHSession 以在 SSH 服务器上运行交互式对话。

**示例**

创建 SSH 会话, 假设服务器已将指定帐户的 AutoShell 配置为 SSH shell。运行命令并断开:

```
ssh=new-SSHSession -host client8 -user bob -pass xyz42 -prompt ::->
? run-SSHCommand -session ssh -command "! ver"
ssh.disconnect();
```

**另请参阅:**

[run-SSHCommand 命令—通过 SSH 运行远程命令 \(Funclet\)](#) (p. 90)

[run-SSHShell 命令—运行交互式 SSH 命令行 \(Funclet\)](#) (p. 92)

**new-webService 命令—创建 Web 服务对象 (Funclet)**

该命令会创建一个 Web 服务对象。

命令的语法如下:

```
new-webService WSDLURI [-endpoint endpoint] [-user username] [-pass password]
[-proxy proxyname] [-proxyUser username] [-proxyPass password]
```

**WSDLURI**

指定 WSDLURI。

**-endpointt *endpoint***

(可选) 指定端点。

**默认值:** ""



**-user *username***

（可选）指定用于登录到远程节点的用户名。使用密码身份验证时，需要用户名。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-proxy *proxyname***

（可选）指定代理名称。

**默认值：** ""

**-proxyUser *username***

（可选）指定用于登录到代理的用户名。

**默认值：** \$\$User

**-proxyPass *password***

（可选）指定用于登录到代理的密码。

**默认值：** \$\$Pass

## objdump 命令-显示对象的可枚举属性 (Cmdlet)

该命令 (cmdlet) 会列出指定对象的可枚举属性、其类型并可列出当前值。

命令的语法如下：

```
objdump obj [-detail]
```

***obj***

指定要显示的对象。

**-detail**

（可选）指定该选项后，该命令还会显示列出的属性的当前值。

### 示例

显示对象的属性:

```
o = new Object();  
o.num = 42;  
o.str = "Hello";  
objdump o
```

显示当前值:

```
objdump o -detail
```

另请参阅:

[arrdump 命令—显示数组 \(Cmdlet\)](#) (p. 48)

## push-client 命令-在远程 Windows 系统上安装 AutoShell 客户端 (Funclet)

该命令在远程 Windows 系统上安装 AutoShell 客户端。

该命令在一个或多个远程 Windows 系统上进行复制和执行。可用版本如下:

- 单一目标版本 (1)
- 多目标版本 (2)

命令的语法如下:

```
(1) push-client -host hostname [-localPath path] [-user username>] [-pass  
password]  
(2) push-client [-localPath path] [-user username>] [-pass password] on  
remoteTargets
```

单一目标命令成功时将返回 true。

多目标命令返回一个 WinRemote 对象数组。该数组的每个元素与 *remoteTargets* 列表列出的远程 Windows 节点的位置一致。WinRemote 对象可以验证执行是否完成, 是否出现错误, 并在执行结束时查询输出和返回的结果。如果已使用 WinRemote 对象指定远程节点, 则数组会包含对原始 RemoteTarget 对象的引用。如果已使用字符串指定目标系统, 则将在内部创建新的 RemoteTarget 对象并将其分配给数组。

**-host *hostname***

（可选）指定远程 **Windows** 主机的名称。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**-localPath *path***

（可选）指定指向要复制的 **AutoShell** 客户端的路径。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 **SSH** 会话时需要此参数。

**默认值：** ""

**-user *username***

（可选）指定要登录到远程 **Windows** 服务器的用户名。用户必须具有连接到 **\$ADMIN** 共享的权限。如果没有指定用户名，则使用在 **AutoShell** 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 **AutoShell** 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 **PW\_GET()**。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**on *remoteTargets***

指定要运行指定脚本的远程系统的逗号分隔列表。可以通过两种方式标识目标节点：按主机名/地址或按 **WinRemote** 对象。按名称或地址指定目标系统，同时使用 **WinRemote** 对象可以更灵活地控制执行进程，例如，不同的凭据用于不同的主机。**WinRemote** 对象和包含主机名/地址的字符串可以在远程目标列表中同时使用。目标列表的元素不是自动字符串化的，因此在主机名中使用文本字符串时必须将其置于引号内。

## 示例

在 host1 上安装 AutoShell 客户端:

```
push-client -host host1
```

在 host1 和 host2 上安装 AutoShell 客户端:

```
x = push-client on "host1", "host2";  
? x[1].hasCompleted();  
? x[1].result();  
? x[1].output();
```

另请参阅:

[rem-client 命令—从远程 Windows 系统中删除 Autoshell 客户端 \(Funclet\)](#) (p. 80)

[run-client 命令—在远程 Windows 系统上运行 Autoshell 客户端 \(Funclet\)](#) (p. 84)

[push-winRemote 命令—在多个远程 Windows 系统上进行复制并执行 \(Funclet\)](#) (p. 76)

[run-winRemote 命令—在多个远程 Windows 系统上执行命令 \(Funclet\)](#) (p. 94)

## push-winRemote 命令—在多个远程 Windows 系统上进行复制并执行 (Funclet)

该命令在一个或多个远程 Windows 系统上进行复制和执行。可用版本如下:

- 单一目标版本 (1)
- 多目标版本 (2)

命令的语法如下:

```
(1) push-winRemote -host hostname -localPath path [-cmd cmd] [-user username]  
[-pass password]  
(2) push-winRemote -localPath path [-cmd cmd] [-user username] [-pass password]  
on remoteTargets
```

单一目标版本返回已执行命令的结果代码。

多目标版本返回一个 WinRemote 对象数组。该数组的每个元素与 *remoteTargets* 列表列出的远程 Windows 节点的位置一致。WinRemote 对象可以验证执行是否完成, 是否出现错误, 并在执行结束时查询输出和返回的结果。如果已使用 WinRemote 对象指定远程节点, 则数组会包含对原始 RemoteTarget 对象的引用。如果已使用字符串指定目标系统, 则将在内部创建新的 RemoteTarget 对象并将其分配给数组。

**-host *hostname***

指定远程 Windows 主机的名称。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**-localPath *path***

指定指向要复制的 AutoShell 客户端的路径。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值: ""

**-cmd *cmd***

(可选) 指定要在远程 Windows 系统上执行的命令。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值: ""

**-user *username***

(可选) 指定要登录到远程 Windows 系统的用户名。用户必须具有连接到 \$ADMIN 共享的权限。如果没有指定用户名, 则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: \$\$User

**-pass *password***

(可选) 指定用于登录到远程节点的密码。如果没有指定密码, 则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码, 请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。防止表达式自动引用将表达式代码置于括号中。

默认值: \$\$Pass

**on *remoteTargets***

指定要运行指定脚本的远程系统的逗号分隔列表。可以通过两种方式标识目标节点: 按主机名/地址或按 WinRemote 对象。按名称或地址指定目标系统, 同时使用 WinRemote 对象可以更灵活地控制执行进程, 例如, 不同的凭据用于不同的主机。WinRemote 对象和包含主机名/地址的字符串可以在远程目标列表中同时使用。目标列表的元素不是自动字符串化的, 因此在主机名中使用文本字符串时必须将其置于引号内。

### 示例

在 host1 上安装 myinstall.exe:

```
push-winRemote -host host1 -localpath c:\installations\myinstall.exe -cmd myinstall.exe
```

在 host1 和 host2 上安装 myinstall.exe 并验证 host1 的结果:

```
push-winRemote -cmd myinstall.exe -localpath c:\installations\myinstall.exe on "host1", "host2";  
? x[0].hasCompleted();  
? x[0].result();  
? x[0].output();
```

另请参阅:

[run-winRemote 命令—在多个远程 Windows 系统上执行命令 \(Funclet\)](#) (p. 94)

[push-client 命令—在远程 Windows 系统上安装 Autoshell 客户端 \(Funclet\)](#) (p. 74)

[run-client 命令—在远程 Windows 系统上运行 Autoshell 客户端 \(Funclet\)](#) (p. 84)

[rem-client 命令—从远程 Windows 系统中删除 Autoshell 客户端 \(Funclet\)](#) (p. 80)

## PW\_GET 命令-输入隐藏数据 (Funclet)

该命令 (Funclet) 可提示用户输入并读取字符直到遇到回车符。对于每个输入的字符，该命令会显示一个星号 (\*) 以隐藏输入内容。

命令的语法如下:

```
PW_GET(prompt)
```

该命令返回一个包含以明文形式输入的字符的字符串。

### prompt

字符串显示为输入提示符。未加引号的参数令牌会自动字符串化。通过将表达式代码放到括号中来防止自动引用表达式。

默认值: ""

### 示例

如果无法以明文形式输入作为命令的一部分的密码，请使用该命令隐藏密码:

```
run-remote Math.PI on "ascll" -user alice -pass PW_GET(Enter password:)
```

另请参阅:

[accept 命令—从 stdin 读取字符串并将其分配给变量 \(Cmdlet\)](#) (p. 47)

## pwd 命令-输出工作目录的名称 (Cmdlet)

该命令会输出当前工作目录的名称。

命令的语法如下：

```
pwd
```

### 示例

如果您已在默认位置安装 AutoShell，pwd 命令会显示 C:\Program Files\CA\SC\AutoShellManager。

## query-service 命令—查询 Windows 服务的状态 (Funclet)

query-service 命令用于查询本地系统或远程系统上 Windows 服务的状态。query-service 命令以字符串形式返回服务的状态。可能的返回值为：

- stopped
- 启动
- stopping
- 运行
- continue pending
- pause pending
- paused On error
- query failed

命令的语法如下：

```
query-service -svcName sName [-host hostname] [-user username] [-pass password]
```

**-svcName *sName***

指定服务名称。

**-host *hostname***

（可选）远程 Windows 系统的名称。

默认值：""

**-user *username***

(可选) 登录到远程 Windows 系统所用的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: ""

**-pass *password***

登录到远程 Windows 系统所用的密码。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: ""

**示例**

查询本地 Apache 服务

```
? query-service -svcName Apache2.2
```

查询远程服务器 REMOTESYS 上的 Microsoft SQL Server 服务

```
? query-service -svcName MSSQLSERVER -host REMOTESYS -user admin -pass notsecret
```

另请参阅:

[start-service 命令—在远程或本地系统上启动 Windows 服务 \(Funclet\)](#) (p. 99)

[stop-service 命令—在远程或本地系统上停止 Windows 服务 \(Funclet\)](#) (p. 100)

**rem-client 命令—从远程 Windows 系统中删除 Autoshell 客户端 (Funclet)**

rem-client 命令用于从远程 Windows 系统中删除 Autoshell 客户端。

命令的语法如下:

```
rem-client -host hostname [-user username] [-pass password]
```

该命令在成功后返回 true。

**-host *hostname***

指定远程 Windows 主机。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。



**-user *username***

（可选）指定要登录到远程 Windows 系统的用户名。用户必须具有连接到 \$ADMIN 共享的权限。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass *password***

（可选）指定登录到远程节点时使用的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**示例**

卸载 host1 上的 AutoShell 客户端。

```
rem-client -host host1
```

**另请参阅：**

[push-client 命令-在远程 Windows 系统上安装 Autoshell 客户端 \(Funclet\)](#) (p. 74)

[push-winRemote 命令-在多个远程 Windows 系统上进行复制并执行 \(Funclet\)](#) (p. 76)

[run-client 命令-在远程 Windows 系统上运行 Autoshell 客户端 \(Funclet\)](#) (p. 84)

[run-winRemote 命令-在多个远程 Windows 系统上执行命令 \(Funclet\)](#) (p. 94)

## rmdir 命令—删除目录 (Funclet)

该命令删除目录。指定的目录必须为空，且不得包含文件或子目录。要指定远程主机，需使用主机名后接两个冒号 (::) 作为路径名的前缀。通过 SFTP 执行该操作。因此，远程系统不一定要安装客户端 AutoShell，具有 SFTP 访问权的 SSH 服务器已足够。

命令的语法如下：

```
rmdir d [-silent] [-user username] [-pass password] [-key key phrase] [-port portnumber]
```

### **d**

定义要删除的目录的名称。要指定远程主机，需使用主机名后接两个冒号 (::) 作为路径名的前缀。通过 SFTP 执行该操作。因此，远程系统不一定要安装客户端 AutoShell，具有 SFTP 访问权的 SSH 服务器已足够。可以使用 `set alternate` 命令将输出写入文件。

### **-silent**

（可选）该命令会停止任何对 `stdout` 的输出。

### **-user username**

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

### **-pass password**

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-key *key***

(可选) 指定包含 RSA 私钥的文件的绝对路径或相对路径, 用于登录到远程节点。如果指定了私钥和密码, AutoShell 将首先尝试公钥登录, 如果失败, 则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值: ""

**短语**

指定私钥的密码。如果密钥没有加密, 则不需要该密码。要在发出该命令之后输入隐藏的密码, 请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: ""

**-port *portnumber***

(可选) 连接到目标系统的端口。

默认值: SSH 标准端口 22。

**示例**

删除目录 `c:\temp`:

```
rmmdir c:/temp
```

删除作为当前工作目录子目录的 `temp` 目录, 但禁止输出到 `stdout`:

```
rmmdir temp -silent
```

从远程系统 `ascl1` 中删除目录:

```
rmmdir ascl1:/home/alice/data
```

另请参阅:

[dir、ls 命令—获取文件和目录信息 \(Funclet\)](#) (p. 58)

[cd、chdir 命令—更改目录 \(cmdlet、funclet\)](#) (p. 50)

[mkdir 命令—创建目录 \(Cmdlet\)](#) (p. 65)

[del、rm 命令—删除文件 \(Funclet\)](#) (p. 56)

[pwd 命令—输出工作目录的名称 \(Cmdlet\)](#) (p. 79)

## run-client 命令—在远程 Windows 系统上运行 Autoshell 客户端 (Funclet)

run-client 命令 (Funclet) 用于在远程 Windows 系统上运行 Autoshell 客户端。输入和输出重定向到正在调用的 AutoShell。

命令的语法如下：

```
run-client -host hostname [-user username>] [-pass password]
```

该命令返回 true 或 false。

### **-host *hostname***

指定远程 Windows 主机的名称。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

### **-user *username***

（可选）指定要登录到远程 Windows 系统的用户名。用户必须具有连接到 \$ADMIN 共享的权限。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

### **-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

### 示例

在 host1 上运行 AutoShell 客户端。

```
run-client -host host1
```

另请参阅：

[push-client 命令—在远程 Windows 系统上安装 Autoshell 客户端 \(Funclet\)](#) (p. 74)

[push-winRemote 命令—在多个远程 Windows 系统上进行复制并执行 \(Funclet\)](#) (p. 76)

[rem-client 命令—从远程 Windows 系统中删除 Autoshell 客户端 \(Funclet\)](#) (p. 80)

[run-winRemote 命令—在多个远程 Windows 系统上执行命令 \(Funclet\)](#) (p. 94)

## run-local 命令—在本地系统上执行脚本 (Funclet)

`run-local funclet` 执行包含在当前 AutoShell 会话中本地计算机上的字符串或磁盘文件中的 JavaScript 代码并返回结果。直接执行不包含在包围函数声明内的代码。函数声明已处理，以便函数可供以后执行，但不进行调用。尽管通过简单地输入 JavaScript 文件的路径名就可以从交互式命令提示符执行该文件，但 `run-local` 是从脚本文件执行外部脚本的唯一方法，也是从交互式命令提示符调用脚本文件时将参数传递到该脚本文件的唯一方法。

命令的语法如下：

```
run-local [script] [-file file] [-with args,...]
```

返回值取决于指定的脚本。

### **script**

（可选）指定包含要执行的 JavaScript 代码的字符串。此 `funclet` 从字符串或文件执行脚本。因此，`script` 和 `file` 参数是互不相容的。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

### **-file *file***

（可选）指定要执行的脚本文件的绝对路径或相对路径。此 `funclet` 从字符串或文件执行脚本。因此，`script` 和 `file` 参数是互不相容的。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

### **-with *args*,...**

指定要传递到脚本的实际参数的逗号分隔列表。脚本代码可以使用标准的 JavaScript 参数数组访问这些参数。

## 示例

评估简单表达式：

```
run-local 1+2
```

评估从内存传递参数的表达式：

```
run-local "arguments[0]*arguments[0]" -with 2
```

编写脚本文件并执行该脚本文件：

```
s = "var i;"
s += "for(i=0;i<arguments.length;i++)"
s += "qout(arguments[i]);"
s += "true;"
memoWrit("script.js", s);
run-local -file script.js -with 1, "abc", new Date()
```

**注意：**仅当脚本代码不包含在函数定义中时，才能直接执行脚本代码。脚本不使用返回语句返回结果（在这种情况下，值为 **true**），而是通过在最后执行的表达式中指定结果。

另请参阅：

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

## run-remote 命令—在远程系统上执行脚本 (Funclet)

run-remote 命令执行 JavaScript 代码，这些代码包含在一个或多个远程系统上的字符串或磁盘文件内。要使远程执行起作用，远程目标节点必须安装 AutoShell 客户端并对其进行适当配置。脚本文件必须位于管理器系统并传输到目标系统才可执行。默认情况下，脚本执行采用异步方式，这意味着在远程执行完成之前，本地处理一直持续进行。

命令的语法如下：

```
run-remote [script] [-file file] [-wait] [-with args,...] [-user username]
[-pass password] [-key key phrase] [-port portnumber] on remoteTargets
```

run-remote 命令返回 RemoteTarget 对象数组，表示指定服务器上脚本的执行状态。该数组中每个元素的位置与 *remoteTargets* 列表中的远程节点的位置相对应。RemoteTarget 对象可以验证多个结果，例如，执行是否完成、是否有错误发生或执行是否成功完成。如果已使用 RemoteTarget 对象指定了远程节点，则数组中会包含原始 RemoteTarget 对象的引用。如果已使用字符串指定目标系统，则将在内部创建新的 RemoteTarget 对象并将其分配给数组。

### script

（可选）指定包含要执行的 JavaScript 代码的字符串。此 funclet 从字符串或文件执行脚本。因此，*script* 和 *file* 参数是互不相容的。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** ""

**-file file**

（可选）指定要执行的脚本文件的绝对路径或相对路径。此 `funclet` 从字符串或文件执行脚本。因此，`script` 和 `file` 参数是互不相容的。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：**""

**-wait**

（可选）默认情况下，`run-remote` 在开始远程执行之后返回，且远程处理采用异步方式执行。当指定 `-wait` 选项后，`run-remote` 会等待直到所有远程任务成功完成或因错误而中止。

**-with args,...**

（可选）指定要传递到脚本的实际参数的逗号分隔列表。脚本代码可以使用标准的 JavaScript 参数数组访问这些参数。

**-user username**

（可选）指定用于登录到远程节点的用户名。要使用密码或公钥身份验证，需要输入用户名。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。指定的用户名仅用于由 `remoteTargets` 列表中的主机名标识的远程系统。使用 `RemoteTarget` 对象指定主机时，每个主机都可以通过 `RemoteTarget` 对象设置自己的用户名。

**默认值：**\$\$User

**-pass password**

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。指定的密码仅用于由 `remoteTargets` 列表中的主机名标识的远程系统。使用 `RemoteTarget` 对象指定主机时，每个主机都可以通过 `RemoteTarget` 对象设置自己的用户名。

**默认值：**\$\$Pass

**-key *key***

(可选) 指定包含 RSA 私钥的文件的绝对路径或相对路径, 用于登录到远程节点。如果指定了私钥和密码, AutoShell 将首先尝试公钥登录, 如果失败, 则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

**默认值:** ""

**短语**

(可选) 为私钥指定密码。如果未加密密钥, 则不需要该密码。要在发出该命令之后输入隐藏的密码, 请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值:** ""

**-port *portnumber***

(可选) 指定连接到目标系统的端口。指定的端口仅用于由 *remoteTargets* 列表中的主机名标识的远程系统。使用 `RemoteTarget` 对象指定主机时, 每个主机都可以通过 `RemoteTarget` 对象设置自己的端口。

**默认值:** SSH 标准端口 22。

**on *remoteTargets***

指定要运行指定脚本的远程系统的逗号分隔列表。可以通过两种方式标识目标节点: 按主机名/地址或按 `WinRemote` 对象。按名称或地址指定目标系统, 同时使用 `WinRemote` 对象可以更灵活地控制执行进程, 例如, 不同的凭据用于不同的主机。`WinRemote` 对象和包含主机名/地址的字符串可以在远程目标列表中同时使用。目标列表的元素不是自动字符串化的, 因此在主机名中使用文本字符串时必须将其置于引号内。

**示例**

评估简单表达式并等待直到执行完成:

```
aRT = run-remote "1+2" on "ascli1" -wait
? "Error occurred:", aRT[0].errorOccurred()
? "Result:", get-remoteResult(aRT[0])
```



在两个远程系统上从文件评估脚本，等待直到执行完成并显示输出（“Hello World!”）和结果 (42):

```
s="? 'Hello World';42;"
// 仅为说明目的创建脚本文件
// 也可以调用:
// 针对“ascli1”、“ascli2”调用 aRT=run-remote (s)
memoWrit("hello.js", s);
aRT=run-remote -file hello.js on "ascli1", "ascli2"
for(i=0; i<aRT.length; i++)
{
    ? "Target:", aRT[i].getHostName()
    ? "Error occurred:", aRT[i].errorOccurred()
    ? "Output:", aRT[i].errorOccurred()
    ? "Result:", get-remoteResult(aRT[i])
}
```

使用 RemoteTarget 对象在两个远程系统上评估脚本，为目标系统指定不同的凭据:

```
rt1 = new RemoteTarget("ascli1", "alice", "casogood42");
rt2 = new RemoteTarget("ascli2", "bob", "!secret!7");
// 从远程系统获取 OS 版本字符串
aRT = run-remote "! ver" on rt1, rt2
// 自行执行等待
while(true)
{
    // 如果发生错误, hasCompleted() 也将
    // 返回 true
    if(rt1.hasCompleted() && rt2.hasCompleted())
        break;
    sleep(500);
}
// 我们可以使用原始 RemoteTarget 对象
// (参阅上文)或返回数组中的引用
// (下文)。
for(i=0; i<aRT.length; i++)
{
    ? "Target:", aRT[i].getHostName()
    ? "Error occurred:", aRT[i].errorOccurred()
    ? "Output:", aRT[i].errorOccurred()
}
```

另请参阅:

[get-remoteResult 命令-从远程目标获取结果 \(Funclet\)](#) (p. 63)

[RemoteTarget 类](#) (p. 135)

[run-local 命令-在本地系统上执行脚本 \(Funclet\)](#) (p. 85)

## run-SSHCommand 命令—通过 SSH 运行远程命令 (Funclet)

run-SSHCommand 在现有 SSH 会话内执行命令，或创建会话以便执行命令。执行指定的命令之后销毁该会话。虽然 AutoShell 使用 SSH 作为其主远程通信管道，但是该命令的远程节点不需要运行 AutoShell 客户端。run-remote 命令更适用于在运行 AutoShell 客户端的远程节点上执行命令，因为该命令可返回有关命令执行方面的更多详细信息。

可以在任何运行 SSH 服务器的系统上执行命令。可用命令及其选项取决于在远程节点上配置的 shell。

命令的语法如下：

```
run-SSHCommand [-host hostname] [-user username] [-pass password] [-key key phrase]  
[-prompt prompt] -command cmd [-port portnumber] [-session ssh]
```

### **-host *hostname***

SSH 服务器的名称。

### **-user *username***

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：**\$\$User

### **-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 PW\_GET()。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：**\$\$Pass

**-key *key***

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，AutoShell 将首先尝试公钥登录，如果失败，则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值：""

**-短语**

指定私钥的密码。如果未加密密钥，则不需要该密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

**-prompt *prompt***

（可选）指定 SSH 服务器显示的提示字符串结尾的固定部分，以指示其可用于输入（例如：用于 AutoShell 的 “::->”）。需要该信息以确定在 SSH 会话期间执行的命令的完成情况。指定尽可能多的固定字符。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：“::->”

**-port *portnumber***

（可选）连接到目标系统的端口。

默认值：SSH 标准端口 22。

**-cmd *cmd***

（可选）指定要在远程 Windows 系统上执行的命令。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值：cmd

**-session *ssh***

`new-SSHSession` 返回的 `CASSHELL` 对象。在传递现有 SSH 会话对象时，`run-SSHCommand` 不需要创建 SSH 会话来执行指定的命令。当执行多个命令时，仅执行一次会话协商和登录通常会提高多次会话分配后的性能。如果指定该参数，则会忽略为 `host`、`user`、`pass`、`key`、`phrase` 和 `port` 传递的值。创建会话时已经指定该信息。

默认值：空

### 示例

使用明确指定用户名和密码的 SSH 会话，在远程节点上运行目录列表：

```
s=run-SSHCommand -host srv14 -user bob -pass xyz42 -prompt :> -command ls
```

使用 AutoShell 登录所用的用户名和密码的 SSH 会话，在远程节点上运行目录列表：

```
s = run-SSHCommand -host srv14 -prompt ":>" -command "ls"
```

在明确分配的 SSH 会话中执行两个命令：

```
ssh=new-SSHSession -host client8 -user bob -pass xyz42 -prompt ::->
? run-SSHCommand -session ssh -command "! ver"
? run-SSHCommand -session ssh -command "ls *.* -l"
ssh.disconnect();
```

另请参阅：

[new-SSHSession 命令—创建一个新的安全 SSH 会话 \(Funclet\)](#) (p. 71)

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

## run-SSHShell 命令—运行交互式 SSH 命令行 (Funclet)

run-SSHCommand 命令接受输入行，通过 SSH 将其发送到远程系统，并从远程节点显示输出。可以在现有 SSH 会话或新的 SSH 会话中运行命令行。在输入结束时销毁该会话。虽然 AutoShell 使用 SSH 作为其主远程通信管道，但是该命令的远程节点不需要运行 AutoShell 客户端。可以在任何运行 SSH 服务器的系统上执行命令。可用命令、其选项和结束输入循环的方式取决于远程节点上配置的 shell。通常可以通过输入“quit”或“exit”来终止输入循环。

命令的语法如下：

```
run-SSHShell [-host hostname] [-user username] [-pass password] [-key key phrase]
[-prompt prompt] [-port portnumber] [-session ssh]
```

**-host *hostname***

SSH 服务器的名称。

**-user *username***

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$User

**-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** \$\$Pass

**-key *key***

（可选）指定包含 RSA 私钥的文件的绝对路径或相对路径，用于登录到远程节点。如果指定了私钥和密码，AutoShell 将首先尝试公钥登录，如果失败，则尝试密码登录。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

**默认值：** ""

**短语**

指定私钥的密码。如果未加密密钥，则不需要该密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** ""

**-prompt *prompt***

（可选）指定 SSH 服务器显示的提示字符串结尾的固定部分，以指示其可用于输入（例如：用于 AutoShell 的 “`::->`”）。需要该信息以确定在 SSH 会话期间执行的命令的完成情况。指定尽可能多的固定字符。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**默认值：** “`::->`”

**-port *portnumber***

（可选）连接到目标系统的端口。

**默认值：** SSH 标准端口 22。

**-session ssh**

指定 new-SSHSession 返回的 CASSHELL 对象。在传递现有 SSH 会话对象时，run-SSHShell 不需要创建 SSH 会话来从命令行执行命令。如果指定该参数，则会忽略为 *host*、*user*、*pass*、*key*、*phrase* 和 *port* 传递的值。创建该会话时已指定该信息。

**默认值：**空

**示例**

使用具有密码身份验证的 SSH 会话在主机上运行交互式命令行 shell：

```
run-SSHShell -host svr14 -user bob -pass xyz42 -prompt :>
```

明确分配 SSH 会话，首先运行命令，然后放到命令行输入循环：

```
ssh=new-SSHSession -host client8 -user bob -pass xyz42 -prompt ::->
run-SSHCommand -session ssh -command "cd /"
run-SSHShell -session ssh
ssh.disconnect();
```

**另请参阅：**

[new-SSHSession 命令—创建一个新的安全 SSH 会话 \(Funclet\)](#) (p. 71)

[run-SSHCommand 命令—通过 SSH 运行远程命令 \(Funclet\)](#) (p. 90)

**run-winRemote 命令—在多个远程 Windows 系统上执行命令 (Funclet)**

run-winRemote 命令在一个或多个远程 Windows 系统上执行命令。该命令具有两种风格：

- 单目标版本
- 多目标版本

单目标版本将输入和输出重定向到调用 AutoShell。

run-winRemote 的多目标版本采用异步方式。请参阅示例代码以了解如何检索结果。

命令的语法如下：

```
(1) run-winRemote -host hostname [-cmd cmd] [-user username] [-pass password]
    [-workDir workDir]
```

```
(2) run-winRemote [-cmd cmd] [-user username] [-pass password] [-workDir workDir]
    on remoteTargets
```

`run-winRemote` 的单目标版本返回已执行命令的结果代码。

多目标版本返回一个 `WinRemote` 对象数组。该数组的每个元素与 `remoteTargets` 列表列出的远程 Windows 节点的位置一致。`WinRemote` 对象可以验证多个结果，例如，执行是否完成、是否有错误发生或执行是否成功完成。如果已使用 `WinRemote` 对象指定远程节点，则数组会包含对原始 `RemoteTarget` 对象的引用。如果已使用字符串指定目标系统，则将在内部创建新的 `RemoteTarget` 对象并将其分配给数组。

**-host *hostname***

（可选）指定远程 Windows 主机的名称。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**-cmd *cmd***

（可选）指定要在远程 Windows 系统上执行的命令。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。仅在使用公钥身份验证联机创建 SSH 会话时需要此参数。

默认值: `cmd`

**-user *username***

（可选）指定要登录到远程 Windows 系统的用户名。用户必须具有连接到 `$ADMIN` 共享的权限。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: `$$User`

**-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。要在发出该命令之后输入隐藏的密码，请指定使用可选提示作为参数的 `PW_GET()`。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值: `$$Pass`

**-workDir workdir**

指定实际工作目录。

默认值: ""

**on remoteTargets**

指定要运行指定脚本的远程系统的逗号分隔列表。可以通过两种方式标识目标节点：按主机名/地址或按 WinRemote 对象。按名称或地址指定目标系统，同时使用 WinRemote 对象可以更灵活地控制执行进程，例如，不同的凭据用于不同的主机。WinRemote 对象和包含主机名/地址的字符串可以在远程目标列表中同时使用。目标列表的元素不是自动字符串化的，因此在主机名中使用文本字符串时必须将其置于引号内。

**示例**

在 host1 上运行 cmd.exe（命令 shell）：

```
run-winRemote -host host1 -cmd cmd.exe
```

在 host1 和 host2 上运行 mybatch.bat 并验证 host2 的结果

```
x = run-winRemote -cmd mybatch.bat -localpath c:\installations\myinstall.exe on  
"host1", "host2";  
? x[1].hasCompleted();  
? x[1].result();  
? x[1].output();
```

另请参阅：

[push-winRemote 命令-在多个远程 Windows 系统上进行复制并执行 \(Funclet\)](#) (p. 76)

[push-client 命令-在远程 Windows 系统上安装 Autoshell 客户端 \(Funclet\)](#) (p. 74)

[run-client 命令-在远程 Windows 系统上运行 Autoshell 客户端 \(Funclet\)](#) (p. 84)

[rem-client 命令-从远程 Windows 系统中删除 Autoshell 客户端 \(Funclet\)](#) (p. 80)



## set alternate to 命令—设置备用输出文件 (Cmdlet)

该命令 (cmdlet) 可以为 ?、??、qout() 或 qqout() 等设置或重置备用输出文件。设置备用输出文件时，来自命令或函数的任何输出不仅写入 `stdout`，而且写入指定的文件。要仅写入备用输出文件，请使用“set console off”命令停止 `stdout`。

命令的语法如下：

```
set alternate to outFile [-append]
```

### outFile

指定字符串，其中包含备用输出文件的绝对路径或相对路径。如果未指定文件名，该命令会将备用输出流还原到前一个设置。

### -append

（可选）将输出流附加到指定的文件。

### 示例

将数字 1 到 10 写入 output.txt 和控制台：

```
set alternate to output.txt
for(i=1; i<11; i++)
{
    ? i
}
set alternate to
```

将数字 11 到 20 附加到 output.txt 和控制台：

```
set alternate to output.txt -append
for(i=11; i<21; i++)
{
    ? i
}
set alternate to
```

将目录列表写入文件：

```
set alternate to dir.txt
dir -l
set alternate to
```

另请参阅：

[? 命令—将输出写入到 stdout \(Cmdlet\)](#) (p. 46)

[qout—将后跟换行符的输出写入 stdout（函数）](#) (p. 108)

[?? 命令—将列表中的输出写入到 stdout \(Cmdlet\)](#) (p. 46)

[qqout—将输出写入 stdout（函数）](#) (p. 109)

[set console 命令—停止控制台输出 \(Cmdlet\)](#) (p. 98)

## set console 命令—停止控制台输出 (Cmdlet)

该命令 (cmdlet) 用于停止到的 `stdout` 任何输出, `stdout` 是交互模式下 `?.??\qout()` 或 `qqout()` 生成的控制台。写入文件时通常与 `set alternate` 命令配合使用。

命令的语法如下:

```
set console {on|off}
```

开

将输出写入 `stdout`。

关

停止控制台输出。

### 示例

停止控制台输出并将数字 1 到 10 写入 `output.txt`:

```
set console off
set alternate to output.txt
for(i=1; i<11; i++)
{
    ? i
}
set alternate to
set console on
```

另请参阅:

[set alternate to 命令—设置备用输出文件 \(Cmdlet\)](#) (p. 97)

## set result display on/off 命令—打开或关闭自动结果显示 (Cmdlet)

该命令用于打开或关闭结果显示。

命令的语法如下:

```
set result disp {on|off}
```

另请参阅:

[dir、ls 命令—获取文件和目录信息 \(Funclet\)](#) (p. 58)

## start-service 命令—在远程或本地系统上启动 Windows 服务 (Funclet)

start-service 命令用于在本地或远程系统上启动 Windows 服务。

命令的语法如下：

```
start-service -svcName sName [-host hostname] [-user username] [-pass password]
```

该命令在成功时返回 1，失败时返回 0。

### **-svcName *sName***

指定服务名称。

### **-host *hostname***

（可选）远程服务器的名称。

默认值：""

### **-user *username***

（可选）指定用于登录到远程节点的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

### **-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

## 示例

启动本地 Apache 服务：

```
? start-service -svcName Apache2.2
```

在远程服务器 REMOTESRV 上启动 Microsoft SQL Server 服务

```
? start-service -svcName MSSQLSERVER -host REMOTESRV -user admin -pass notsecret
```

## stop-service 命令—在远程或本地系统上停止 Windows 服务 (Funclet)

stop-service 命令用于在本地或远程系统上停止 Windows 服务。

命令的语法如下：

```
stop-service -svcName sName [-host hostname] [-user username] [-pass password]
```

该命令在成功时返回 1，失败时返回 0。

### **-svcName *sName***

指定服务名称。

### **-host *hostname***

（可选）远程服务器的名称。

默认值：""

### **-user *username***

（可选）指定用于登录到远程节点的用户名。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

### **-pass *password***

（可选）指定用于登录到远程节点的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

默认值：""

## 示例

停止本地 Apache 服务：

```
? stop-service -svcName Apache2.2
```

在远程服务器 REMOTESRV 上停止 Microsoft SQL Server 服务：

```
? stop-service -svcName MSSQLSERVER -host REMOTESRV -user admin -pass notsecret
```

## start-java 命令—启动 Java 虚拟机 (Cmdlet)

该命令用于启动 Java 虚拟机。

命令的语法如下：

```
start-java [-classpath cp]
```

**cp**

（可选）指定类路径。

**默认值：** ""

## wait 命令—等待按键 (Cmdlet)

该命令具有以下版本：

- 显示提示（可选），等待按键并将按键等效字符分配给字符串变量。
- 显示提示（可选），等待按键。

命令的语法如下：

```
wait [prompt] to var
```

```
wait [prompt]
```

**prompt**

（可选）显示为输入提示的字符串。如果未指定提示，该命令将自动显示“Press any key...”。要不显示任何提示，请传递空字符串 ("")。未加引号的参数令牌会自动字符串化。通过用括号括起表达式代码来阻止自动为表达式加引号。

**var**

将按键字符分配给的变量的名称。

### 示例

等待按键并显示自定义提示消息：

```
do
{
    wait "Enter a number between 1 and 5:" to x
    x = parseInt(x);
}
while(x<1 || x>5)
```

**另请参阅：**

[accept 命令—从 stdin 读取字符串并将其分配给变量 \(Cmdlet\)](#) (p. 47)

## weak external 命令—声明本地外部函数 (Cmdlet)

该命令用于声明本地外部函数。

命令的语法如下：

```
weak external decl
```

**decl**

指定声明。

## AutoShell 功能

本节详细描述 AutoShell 核心函数。函数定义一组处理指令，用于接收数字零或更多参数，不返回值或只返回一个值。函数参数在括号内进行传递，且必须符合 JavaScript 语言语法。必须将文字字符串放在引号内，且必须转义字符串内的特殊字符（如，反斜杠）。

### 示例

```
// 将 s 重复连接 n 次
function repeat(s,n)
{
    var i;
    var ret="";
    for(i=0; i < n; i++)
        ret += s;
    return(ret);
}
? repeat("x", 10);
? repeat("\n", 25);
```

## base64Decode—解码 base64 编码的字符串（函数）

该函数用于解码 base64 编码的字符串。输入字符串不需要 base64Encode()。可以传递任何有效的 base64 编码字符串来进行解码。

该函数具有以下语法：

```
base64Decode(sEncoded)
```

该函数返回字符串，其中包含解码数据。如果发生错误（例如，输入数据格式不正确），该函数将引发异常。

**sEncoded**

指定要解码的字符串。

### 示例

解码字符串并进行错误处理：

```
try
{
    ? base64Decode("SGVsbG8gV29ybGQh");
}
catch(ex)
{
    ? "Error decoding string"
}
```

另请参阅：

[base64Encode—对字符串执行 base64 编码（函数）](#) (p. 103)

## base64Encode—对字符串执行 base64 编码（函数）

该函数对输入数据执行 base64 编码。

该函数具有以下语法：

**base64Encode(sData)**

该函数返回字符串，其中包含传入 sData 的 base64 编码数据。结果字符串的格式为 76 个字符的行长度。

### sData

定义要编码的字符串。

### 示例

对 “Hello World” 进行编码：

```
? base64Encode("Hello World!")
```

另请参阅：

[base64Decode—解码 base64 编码的字符串（函数）](#) (p. 102)

## curDir—检索当前目录（函数）

该函数用于检索当前目录。

该函数具有以下语法：

**curDir()**

该函数返回字符串，其中包含当前工作目录的绝对路径。

### 示例

更改为父目录并检索其路径：

```
chdir ..  
? curDir()
```

另请参阅：

[cd、chdir 命令—更改目录（cmdlet、funclet）](#) (p. 50)

## gete—获取环境变量（函数）

检索环境变量的值。该函数是获取除标准 JavaScript 环境变量之外的其他环境变量的另一种备用方法。

该函数具有以下语法：

```
gete(sVar)
```

该函数返回字符串，其中包含由 **sVar** 指定的环境变量的当前值。如果变量不存在，将返回空字符串。

**sVar**

指定环境变量。

### 示例

检索当前路径设置：

```
? gete("PATH");
```

另请参阅：

[pute—设置环境变量（函数）](#) (p. 107)



## memoRead—将文本文件读入字符串（函数）

memoRead() 函数用于将文本文件读入字符串。

**注意：**对包含二进制信息的文件使用 memoRead() 将产生不可预测的结果。

该函数具有以下语法：

```
memoRead(sFileName [,nOffset] [,nCount])
```

该函数以 JavaScript 字符串返回文本文件的内容。

### **sFilename**

指定文本文件的名称。如果指定的文件不存在，该函数将引发异常。

### **nOffset**

（可选）指定从文本文件开头到读取进程起始点之间的字符数。

### **nCount**

（可选）指定读入字符串的字符数。

### 示例

读取 Windows 引导配置：

```
s=memoRead("c:\boot.ini");  
? s
```

读取文本文件中偏移量 10 到文件结尾之间的内容：

```
s = memoRead("file.txt", 10);
```

从偏移量 10 开始读取文本文件中的五个字符：

```
s = memoRead("file.txt", 10, 5);
```

另请参阅：

[memoWrite—将字符串写入文件（函数）](#) (p. 106)

[cat、type 命令—显示文本文件 \(Cmdlet\)](#) (p. 50)

## memoWrit—将字符串写入文件（函数）

memoWrit() 函数用于将 JavaScript 字符串的内容写入文件。

**注意：**如果文件存在，它会替换现有文件或将字符串附加到现有的字符串内容。

该函数具有以下语法：

```
memoWrit(sFileName, sString [, bAppend])
```

### **sString**

定义字符串。

### **sFileName**

定义文本文件的名称。如果指定的文件不存在，该函数将引发异常。

### **, bAppend**

如果设置为 **true**，该函数会将字符串添加到给定文件。

该函数返回写入指定文件的字节数。

**注意：**如果访问文件时出现问题，函数将引发异常。

### 示例

写入单行文本文件：

```
memoWrit("file.txt", "First line\");
```

将第二行附加到现有文件：

```
memoWrit("file.txt", "Second line", true);
```

另请参阅：

[memoRead—将文本文件读入字符串（函数）](#) (p. 105)

## platform—查询操作系统类型（函数）

该函数返回当前正在运行脚本的操作系统的类型。

该函数具有以下语法：

```
platform()
```

该函数以字符串的形式返回系统类型名称。

### 示例

验证脚本是否正在 Windows 上运行：

```
if(platform()!="Windows")
{
    ? "This script uses Windows specific features!"
    ? "Please run on Windows only."
    quit
}
start-service -svcName Apache2.2
```

另请参阅：

[shellType—查询 AutoShell 类型（函数）](#) (p. 123)

## pute—设置环境变量（函数）

设置现有环境变量的值，或使用指定的值创建一个变量（如果变量不存在）。该函数是设置除标准 JavaScript 环境变量之外的其他环境变量的一种备用方法。

该函数具有以下语法：

```
pute(sVar, sVal)
```

该函数在成功时返回 `true`，失败时返回 `false`。

### sVar

指定环境变量的名称。

### sVal

定义环境变量的值。

### 示例

将当前工作目录附加到路径：

```
pute(gete("PATH")+";"+curDir());
```

另请参阅：

[gete—获取环境变量（函数）](#) (p. 104)

## qout—将后跟换行符的输出写入 stdout（函数）

`qout` 函数用于将后跟换行符的输出写入 `stdout`。该函数将列表中每个参数的字符串表示写入 `stdout`。在每个参数之间自动放置一个空格。`set console` 和 `set alternate` 命令可以停止或重定向输出。输出以换行符终止。

该函数具有以下语法：

```
qout([x,...])
```

**x**

（可选）定义要显示的参数列表。如果列表为空，将输出换行符。

该函数不返回值。

### 示例

输出 “Hello World!”：

```
? "Hello World!"
```

输出数字列表：

```
? 1,2,3
```

输出空行：

```
?
```

输出数字 1 到 10。

**注意：**? 命令只能在行的开头使用。要从表达式内输出文本，请使用相应的 `qout()` 函数：

```
for(i=1;i<11;i++) qout(i);
```

另请参阅：

[?? 命令—将列表中的输出写入到 stdout \(Cmdlet\)](#) (p. 46)

[qqout—将输出写入 stdout（函数）](#) (p. 109)

## qqout—将输出写入 stdout（函数）

qqout 函数用于将列表中每个参数的字符串表示写入 stdout。在每个参数之间自动放置一个空格。set console 和 set alternate 命令可重定向或停止输出。使用 ?、??、qout() 或 qqout() 进行的后续输出调用将紧接原始输出之后放置其输出，因此 ?? 或 qqout() 通常用于构造使用多个调用的输出行。

该函数具有以下语法：

```
qqout([x,...])
```

**x**

（可选）定义要显示的参数列表。如果列表为空，将输出换行符。

该函数不返回值。

### 示例

输出以空格分隔的三个列标题：

```
?? "Col1"
?? "  "
?? "Col2"
?? "  "
? "Col3"
```

使用 ? 函数输出第三个标签，因此光标将移到下一行的开头。

另请参阅：

[? 命令—将输出写入到 stdout \(Cmdlet\)](#) (p. 46)

[qout—将后跟换行符的输出写入 stdout（函数）](#) (p. 108)

[?? 命令—将列表中的输出写入到 stdout \(Cmdlet\)](#) (p. 46)

[set alternate to 命令—设置备用输出文件 \(Cmdlet\)](#) (p. 97)

[set console 命令—停止控制台输出 \(Cmdlet\)](#) (p. 98)

## regCreateKey—创建注册表项（函数）

`regCreateKey` 函数用于在本地系统或远程系统上 *sKeyParent* 指定的注册表项下创建注册表项或注册表项层次结构。

该函数具有以下语法：

```
regCreateKey(sKeyParent, sKey)
```

### **sKeyParent**

指定父注册表项。

### **sKey**

定义新注册表项

如果该函数失败，将引发异常。否则，将返回 `true`。如果指定的注册表项或注册表项层次结构的一部分存在，则表示该函数成功。

## 示例

在本地计算机 `SOFTWARE` 注册表项下创建 `ACEM` 注册表项，并进行错误处理：

```
try {  
    regCreateKey("HKLM\\SOFTWARE", "ACME");  
    ? "Success"  
}  
catch(e) {  
    ? "Failure", e  
}
```

创建注册表项层次结构：

```
regCreateKey("HKLM\\SOFTWARE", "ACME\\HOME");
```

在远程计算机 `client8` 上创建注册表项 `ACEM`：

```
regCreateKey("client8::HKLM\\SOFTWARE", "ACME");
```

另请参阅：

[regSetValue—设置注册表值（函数）](#) (p. 121)

[regDeleteKey—删除注册表项或注册表项层次结构（函数）](#) (p. 112)

## regCreateSubkeys—从数组创建子注册表项（函数）

`regCreateSubKeys` 函数用于在本地系统或远程系统上，从数组的给定注册表项下创建注册表项子树。该数组必须包含将创建为以 `sKey` 开头的相对路径字符串的注册表项名称。`regGetSubKeys()` 将返回此类型的数组。

该函数具有以下语法：

```
regCreateSubKeys(sKey, aSubKeys)
```

### **sKey**

指定注册表项。

### **aSubKeys**

指定注册表项子树。

如果该函数失败，将引发异常。否则，将返回 `true`。

### 示例

在 `HKLM\Software\ACME` 下创建注册表项 “x”、“y” 和 “z”：

```
regCreateSubKeys("HKLM\\Software\\ACME", ["x", "y", "z"]);
```

创建嵌套的子注册表项树：

```
regCreateSubKeys("HKLM\\Software\\ACME", ["a\b\c", "x\y\z"]);
```

从远程系统 `client8` 获取子注册表项树，并将其应用于远程系统 `client9`：

```
arr = regGetSubKeys("client8::HKLM\\Software\\ACME", true);  
regSetSubKeys("client9::HKLM\\Software\\ACME", arr);
```

**注意：**必须对注册表项值进行单独处理。

另请参阅：

[regCreateKey—创建注册表项（函数）](#) (p. 110)

[regGetSubKey—检索注册表项的子注册表项（函数）](#) (p. 115)

## regDeleteKey—删除注册表项或注册表项层次结构（函数）

**regDeleteKey** 函数用于在本地系统或远程系统的注册表中删除注册表项和所有相关值。如果指定的注册表项包含子注册表项，默认情况下删除过程将失败。但是，通过为 *fRecurse* 传递 **true**，该函数将删除 *sKey*（包括所有嵌套的子注册表项）。

该函数具有以下语法：

```
regDeleteKey(sKey [, fRecurse])
```

**sKey**

指定注册表项。

**, fRecurse**

删除嵌套的子注册表项。

如果该函数失败，将引发异常。否则，将返回 **true**。

### 示例

从 HKLM\Software\ACME 中删除名为 “Settings” 的注册表项并进行错误处理：

```
try {  
    regDeleteKey("HKLM\\Software\\ACME\\Settings");  
    ? "Success"  
}  
catch(e) {  
    ? "Failure:", e  
}
```

删除强制递归删除子注册表项的 “Settings” 注册表项：

```
regDeleteKey("HKLM\\Software\\ACME\\Settings", true);
```

删除远程计算机 client8 上的 “Settings” 注册表项：

```
regDeleteKey("client8::HKLM\\Software\\ACME\\Settings");
```

另请参见

[regCreateKey—创建注册表项（函数）](#) (p. 110)

[regDeleteVal—删除注册表值（函数）](#) (p. 113)



## regDeleteVal—删除注册表值（函数）

regSetVal 函数用于在本地系统或远程系统上从 *sKey* 指定的注册表项中删除值。

该函数具有以下语法：

```
regDeleteVal(sKey, sValName)
```

### **sKey**

指定注册表项。

### **sValName**

指定值。

如果该函数失败，将引发异常。否则，将返回 **true**。

### 示例

从 HKLM\Software\ACME 中删除名为 “Label” 的值并进行错误处理：

```
try {  
    regDeleteVal("HKLM\\Software\\ACME", "Label");  
    ? "Success"  
}  
catch(e) {  
    ? "Failure:", e  
}
```

删除远程计算机 client8 上的值：

```
regDeleteVal("client8::HKLM\\Software\\ACME", "Name");
```

另请参阅：

[regDeleteKey—删除注册表项或注册表项层次结构（函数）](#) (p. 112)

[regSetVal—设置注册表值（函数）](#) (p. 121)

## regGetKeyValues—获取注册表项值信息（函数）

regGetKeyValues 函数返回一个二维数组，其中包含本地系统或远程系统上某注册表项的名称、实际值和注册表值的类型。返回的数组可以传递到 regSetKeyValues()，以将值设置从一个系统复制到另一个系统。

该函数具有以下语法：

```
regGetKeyValues(sKey)
```

### **sKey**

指定注册表项。

该函数将返回一个二维数组。每个顶级元素包含一个三元素子数组，其中包含表示值名称的字符串、设置的多态值和值类型的整数。注册表值的类型决定了返回值的类型。

- REG\_DWORD 值返回为数字
- REG\_SZ 和 REG\_EXPAND\_SZ 返回为字符串
- REG\_MULTI\_SZ 返回为字符串数组
- REG\_EXPAND\_SZ 值已使用当前环境变量进行了扩展
- REG\_BINARY 值返回为 AutoShell CharArr 对象，因为 JavaScript 字符串变量无法正确处理二进制数据。

如果该函数失败，将引发异常。

### 示例

检索并显示 HKLM\Software\ACME\Settings 注册表项的值信息：

```
arr = regGetKeyValues("HKLM\\Software\\ACME\\Settings");
l = arr.length;
for(i=0; i < l; i++)
{
    ? "Name: ", arr[i][0]
    ? "Value:", arr[i][1]
    ?? "Type: "
    switch(arr[i][2])
    {
        case REG_SZ:
            ? "REG_SZ"
            break;
        case REG_EXPAND_SZ:
            ? "REG_EXPAND_SZ"
            break;
        case REG_BINARY:
            ? "REG_BINARY"
            break;
        case REG_DWORD:
            ? "REG_DWORD"
            break;
        case REG_MULTI_SZ:
            ? "REG_MULTI_SZ"
            break;
        默认:
            ? "Unknown"
    }
}
```

从远程系统 client8 获取值，并将其应用于远程系统 client9：

```
arr = regGetKeyValues("client8::HKLM\\Software\\ACME\\Settings");
regSetKeyValues("client9::HKLM\\Software\\ACME\\Settings", arr);
```

另请参阅：

[regGetValue—获取注册表值（函数）](#) (p. 116)

[regSetKeyValue—从数组设置注册表项值（函数）](#) (p. 120)

## regGetSubKey—检索注册表项的子注册表项（函数）

`regGetSubKeys` 函数在本地系统或远程系统上以数组形式返回指定注册表项的子注册表项。默认情况下，仅返回直接子注册表项。（可选）可以对所有子注册表项进行递归检索。

该函数具有以下语法：

`regGetSubKeys(sKey, fRecurse)`

**sKey**

指定注册表项。

**, fRecurse**

检索嵌套子注册表项。

该函数返回一个数组，其中包含 `sKey` 的子注册表项的名称。单个子注册表项名称返回为以 `sKey` 开头的相对路径。返回的数组可以传递到 `regCreateSubKeys()`，以将子注册表项树从一个系统复制到另一个系统。

如果该函数失败，将引发异常。

### 示例

获取 `HKLM\Software\CA` 的子注册表项并进行错误处理：

```
try {
    var arr, i, l;
    arr = regGetSubKeys("HKLM\\Software\\CA");
    l = arr.length;
    for(i=0; i < l; i++)
    {
        ? arr[i]
    }
}
catch(e) {
    ? "Failure:", e
}
```

以递归方式获取 HKLM\Software\CA 的所有子注册表项:

```
arr = regGetSubKeys("HKLM\\Software\\CA", true);
l = arr.length;
for(i=0; i < l; i++)
{
    ? arr[i]
}
```

从远程系统 client8 获取子注册表项树, 并将其应用于远程系统 client9:

```
arr = regGetSubKeys("client8::HKLM\\Software\\ACME", true);
regSetSubKeys("client9::HKLM\\Software\\ACME", arr);
```

**注意:** 必须对注册表项值进行单独处理。

另请参阅:

[regCreateKey—创建注册表项（函数）](#) (p. 110)

[regCreateSubkeys—从数组创建子注册表项（函数）](#) (p. 111)

## regGetVal—获取注册表值（函数）

regGetVal 函数用于在本地系统或远程系统上检索 *key* 指定的注册表项下的 *valName* 值的设置。

该函数具有以下语法:

```
regGetVal(sKey, sValName)
```

### sKey

指定注册表项。

### sValName

指定值。

如果指定的值存在, 该函数会返回一个表示指定值设置的多态值。注册表值的类型决定了返回值的类型。

- REG\_DWORD 值返回为数字
- REG\_SZ 和 REG\_EXPAND\_SZ 返回为字符串
- REG\_MULTI\_SZ 返回为字符串数组。
- REG\_EXPAND\_SZ 值已使用当前环境变量进行了扩展。
- REG\_BINARY 值返回为 AutoShell CharArr 对象, 因为 JavaScript 字符串变量无法正确处理二进制数据。

如果指定的注册表项或值不存在, 该函数将引发异常。

### 示例

从项 HKLM\Software\ACME 获取名为 “Number” 的值：

```
regGetVal("HKLM\Software\ACME", "Number");
```

从注册表项 HKLM\Software\ACME 获取名为 “Label” 的值并进行错误处理：

```
try {  
    regGetVal("HKLM\Software\ACME", "Label");  
    ? "Success"  
}  
catch(e) {  
    ? "Failure:", e  
}
```

从远程计算机 client8 获取值：

```
regGetVal("client8::HKLM\Software\ACME", "Name");
```

另请参阅：

[regSetVal—设置注册表值（函数）](#) (p. 121)

[regGetKeyValues—获取注册表项值信息（函数）](#) (p. 113)

## regIsKey—检查注册表项是否存在（函数）

regIsKey 函数用于检查本地系统或远程系统上是否存在指定的注册表项。

该函数具有以下语法：

```
regIsKey(sKey)
```

**sKey**

指定注册表项。

该函数返回一个布尔值，指示 *sKey* 是否存在：

**真**

指示该注册表项存在。

**假**

指示该注册表项不存在。

### 示例

在本地系统的 HKLM\SOFTWARE 下查找 ACME 注册表项:

```
if(regIsKey("HKLM\SOFTWARE\ACME"))  
? "Key was found!"  
else  
? "Key does not exist"
```

在远程系统 ascl1 上执行同样的检查:

```
if(regIsKey("ascl1::HKLM\SOFTWARE\ACME"))  
? "Key was found"  
else  
? "Key does not exist"
```

另请参阅:

[regCreateKey—创建注册表项（函数）](#) (p. 110)

[regIsVal—检查注册表值是否存在（函数）](#) (p. 119)

## regIsVal—检查注册表值是否存在（函数）

regIsVal 函数用于检查本地系统或远程系统上是否存在指定的注册表值。

该函数具有以下语法：

regIsVal(sKey, sValName)

### sKey

指定注册表项。

### sValName

指定值的名称。

该函数将返回一个布尔值，指示 *sValName* 是否存在：

### 真

指示该值存在。

### 假

指示该指不存在。

### 示例

验证 HKLM\SOFTWARE\ACME 的值是否有 “home”：

```
if(regIsVal("HKLM\SOFTWARE\ACME", "home"))
{
    ? "Value was found"
}
else
{
    ? "Value does not exist"
}
```

另请参阅：

[regGetVal—获取注册表值（函数）](#) (p. 116)

[regIsKey—检查注册表项是否存在（函数）](#) (p. 117)

[regSetVal—设置注册表值（函数）](#) (p. 121)

## regSetKeyValues—从数组设置注册表项值（函数）

**regSetKeyValues()** 函数用于在本地系统或远程系统上从二维数组为注册表项创建或设置值。该数组中的每个顶级元素必须包含一个子数组，其中包含表示值名称的字符串、要设置的多态 JavaScript 值和（可选）用于指定要创建或设置的值的类型的整数。**regGetKeyValues()** 将返回该类型的数组。

该函数具有以下语法：

```
regSetKeyValues(sKey, aVals)
```

### **sKey**

指定注册表项。

### **sVals**

定义包含值的数组。

如果该函数失败，将引发异常。否则，将返回 **true**。

### 示例

为 HKLM\Software\ACME 设置值 “Number” 和 “Label”：

```
vals = [{"Number", 42}, {"Label", "Hello"}];  
regSetKeyValues("HKLM\\Software\\ACME\\Settings", vals);
```

设置明确指定类型的值：

```
vals = new Array(2);  
val[0] = ["MyPath", "%Path%;c:\\MyDir", REG_EXPAND_SZ];  
val[1] = ["Number", 42, REG_DWORD];  
regSetKeyValues("HKLM\\Software\\ACME\\Settings", vals);
```

从远程系统 client8 获取值，并将其应用于远程系统 client9：

```
arr = regGetKeyValues("client8:HKLM\\Software\\ACME\\Settings");  
regSetKeyValues("client9:HKLM\\Software\\ACME\\Settings", arr);
```

另请参阅：

[regGetKeyValues—获取注册表项值信息（函数）](#) (p. 113)

[regSetVal—设置注册表值（函数）](#) (p. 121)



## regSetVal—设置注册表值（函数）

`regSetVal` 用于在本地系统或远程系统上为 *sKey* 指定的注册表项创建或设置值。如果 *sValName* 标识的值不存在，将创建该值，否则将使用 *value* 覆盖现有设置。

该函数具有以下语法：

```
regSetVal(sKey, sValName, value, [type])
```

### **sKey**

指定注册表项。

### **sValName**

指定值的名称。

### **值**

定义值。

### **类型**

指定值的类型。

如果该函数失败，将引发异常。否则，将返回 `true`。

## 示例

将注册表项 `HKLM\Software\ACME` 的名为 “Number” 的 `REG_DWORD` 值设置为 42：

```
regSetVal("HKLM\Software\ACME", "Number", 42);
```

将注册表项 `HKLM\Software\ACME` 的名为 “Label” 的 `REG_SZ` 值设置为 “Hello” 并进行错误处理：

```
try {  
    regSetVal("HKLM\Software\ACME", "Label", "Hello");  
    ? "Success"  
}  
catch(e) {  
    ? "Failure:", e  
}
```

设置 REG\_EXPAND\_SZ 值:

```
regSetVal("HKLM\\Software\\ACME", "MyPath", "%Path%;c:\\MyDir", REG_EXPAND_SZ);
```

设置 REG\_MULTI\_SZ 值:

```
arr=["Alice", "Bob", "Carol"];  
regSetVal("HKLM\\Software\\ACME", "People", arr);
```

在远程计算机 client8 上设置字符串值:

```
regSetVal("client8::HKLM\\Software\\ACME", "Name", "Bob");
```

另请参阅:

[regCreateKey—创建注册表项（函数）](#) (p. 110)

[regDeleteVal—删除注册表值（函数）](#) (p. 113)

[regGetVal—获取注册表值（函数）](#) (p. 116)

[regSetKeyValues—从数组设置注册表项值（函数）](#) (p. 120)

## setProcExitCode—设置 AutoShell 退出代码（函数）

该函数用于从覆盖 AutoShell 定义的可能退出代码的用户脚本设置 AutoShell 退出。在 Windows 系统上，可以在 AutoShell 终止之后通过 %errorlevel% 变量检索退出代码。

该函数具有以下语法:

```
setProcExitCode(iExitCode)
```

**iExitCode**

指定退出代码。

### 示例

将 AutoShell 退出代码设置为 -42 并退出 AutoShell:

```
setProcExitCode(-42);  
quit
```

另请参阅:

[exit、quit 命令—终止 AutoShell 会话 \(Cmdlet\)](#) (p. 60)

## shellType—查询 AutoShell 类型（函数）

该函数用于确定脚本当前是在管理器 AutoShell 中执行还是在客户端 AutoShell 中执行。客户端 AutoShell 不提供管理器 AutoShell 的所有功能（例如，不提供远程执行）。shellType() 可以用于验证脚本是否正在预期的 AutoShell 版本内执行。

该函数具有以下语法：

```
shellType()
```

该函数返回字符串，指示当前处于活动状态的 AutoShell 的类型：

- 管理器
- client

### 示例

验证 AutoShell 类型，并在客户端 shell 内运行时退出：

```
if(shellType()!="manager")
{
    ? "This script requires AutoShell manager!"
    quit
}
```

另请参阅：

[platform—查询操作系统类型（函数）](#) (p. 106)

## typeName—获取表达式的类型名称（函数）

该函数用于检索任意表达式的类型名称。尤其对于对象，它返回定义类的名称。

该函数具有以下语法：

```
typeName(val)
```

该函数将返回字符串，其中包含传递值的类型名称。对于简单的类型，将返回下列字符串之一：

- 布尔值
- function
- 空
- number
- 字符串
- void
- xml

对于对象，将返回定义类的名称。

**val**

定义表达式。

### 示例

获取一些表达式的类型：

```
? typeName(19) // 数字
d = new Date;
? typeName(d)  // 日期
a = ["a","b","c"];
? typeName(a)  // 数组
? typeName(a[0])// 字符串
```

另请参阅：

[gete—获取环境变量（函数）](#) (p. 104)

[platform—查询操作系统类型（函数）](#) (p. 106)

## AutoShell 类

本节详细描述 AutoShell 类定义。AutoShell 是基于 JavaScript 的，因此与在 JavaScript 中一样，面向对象是通过函数对象实现的。函数可以用作常规函数、类、构造函数或方法。

可使用新的运算符将对象创建为函数实例，例如：

```
now = new Date();
```

AutoShell 也可实现其自己的类，这些类使用新的运算符进行实例化。在许多情况下，无需直接使用 AutoShell 类，因为它们通过 AutoShell 命令定义被封装。

构造函数是创建对象时调用的方法。一个类可能有多个构造函数。在这种情况下，要调用哪个构造函数取决于对象实例化中的实际参数。

方法是在对象上使用 “.” 运算符进行调用的一组关联函数，并使用 “this” 属性访问或操纵对象数据。

另请参阅 JavaScript 语言参考。

### 示例

计算自当前时间十个小时后的日期/时间。正确处理了经过午夜十二时后的日期/时间。

```
d = new Date();  
d.setHours(d.getHours()+10)
```

### 详细信息

[OSRedirect 类](#) (p. 126)

[RemoteTarget 类](#) (p. 135)

## OSRedirect 类

OSRedirect 是一个实用工具类，与用于在本地运行脚本的 `run-local` 命令配合使用。主要用于将子进程的输出重定向到 JavaScript 字符串。`!!` 命令使用的系统变量 `$$stdout` 也是 OSRedirect 类的一个实例。

### 构造器

`OSRedirect()`

### 方法

`bool clear()`

`bool errorOccurred()`

`string errout()`

`bool hasCompleted()`

`string output()`

`int result()`

### 事件处理程序

`void onCompleted(fCompleted)`

`void onError(fError)`

`void receivedErrOutput(s)`

`void receivedOutput(s)`

`void receivedResult(rc)`

另请参阅：

[OSRedirect.OSRedirect 构造函数](#) (p. 127)

[OSRedirect.clear 方法](#) (p. 127)

[OSRedirect.errorOccurred 方法](#) (p. 128)

[OSRedirect.errout 方法](#) (p. 129)

[OSRedirect.hasCompleted 方法](#) (p. 130)

[OSRedirect.onCompleted 方法](#) (p. 130)

[OSRedirect.onError 方法](#) (p. 131)

[OSRedirect.output 方法](#) (p. 131)

[OSRedirect.receivedErrOutput 方法](#) (p. 132)

[OSRedirect.receivedOutput 方法](#) (p. 133)

[OSRedirect.receivedResult 方法](#) (p. 133)

[OSRedirect.result 方法](#) (p. 134)

[run-local 命令—在本地系统上执行脚本 \(Funclet\)](#) (p. 85)

## OSRedirect.OSRedirect 构造函数

创建对象时调用的构造函数。

该构造函数具有以下语法：

```
OSRedirect()
```

构造函数不返回值。如果出现错误，它会引发异常。

### 示例

创建 OSRedirect 对象并在调用时使用。

```
out = new OSRedirect();  
! ping ascli1 -output out  
? out.output();
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

## OSRedirect.clear 方法

该方法用于重置 OSRedirect() 对象的输出和错误输出缓冲区。该方法在内部由 ! 和 !! 命令进行调用，因此当多个 OS 调用中使用同一个 OSRedirect() 实例时无需对其显式调用。

该方法具有以下语法：

```
clear()
```

该方法不返回值。

### 示例

```
out = new OSRedirect();  
! ping ascli1 -output out  
? out.output();  
out.clear();  
? out.output(); // 空输出
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

## OSRedirect.errorOccurred 方法

该方法用于检查在远程执行期间是否发生错误。如果发生错误，`errout()` 可能会提供错误说明。

该方法具有以下语法：

```
errorOccurred()
```

该方法返回以下值：

**真**

指示发生了错误。

**假**

指示未发生错误。

### 示例

执行无效命令并检查错误：

```
!! asdfghjk
if($$stdout.errorOccurred())
    ? "Error:", $$stdout.errout()
else
    ? $$stdout.output()
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.errout 方法](#) (p. 129)



## OSRedirect.errout 方法

子进程生成以下类型的结果信息。

- 写入屏幕的常规输出
- 写入屏幕的错误输出
- 最终退出代码（“错误级别”）

该方法具有以下语法：

```
errout()
```

该方法返回一个字符串，其中包含由子进程写入 `stderr` 的输出。

### 示例

调用 `nslookup` 并显示常规输出和错误输出：

```
!! nslookup ascli1
? $$stdout.output()
? $$stdout.errout()
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.errorOccurred 方法](#) (p. 128)

[OSRedirect.output 方法](#) (p. 131)

## OSRedirect.hasCompleted 方法

该方法检查是否已完成执行远程命令。如果调用 `run-remote` 时未使用可选的 `-wait` 开关，则执行将异步运行。调用此方法可验证命令是否已完成且包含可用结果。当命令成功完成时，`hasCompleted()` 方法返回 `true`；当发生错误以阻止进一步处理时，返回 `false`。

该方法具有以下语法：

```
hasCompleted()
```

该方法返回以下值：

**真**

指示命令处理已完成。

**假**

指示命令处理未完成，例如，找不到主机。

### 示例

```
!! dir c:\*.* /s
? $$stdout.hasCompleted()    // true
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

## OSRedirect.onCompleted 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则当子进程完成时调用该事件处理程序。

该方法具有以下语法：

```
onCompleted(fCompleted)
```

**注意：**AutoShell 忽略事件处理程序返回值。

### 示例

指定完成事件处理程序：

```
out = new OSRedirect();
out.onCompleted = function(f){if(f)qout("ret=", this.result());};
! ver -output out
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.result 方法](#) (p. 134)

## OSRedirect.onError 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则当 OSRedirect 对象的错误状态更改时，调用该事件处理程序。

该方法具有以下语法：

```
onError(fError)
```

**注意：**AutoShell 忽略事件处理程序返回值。

### 示例

指定完成事件处理程序：

```
out = new OSRedirect();
out.onError = function(f){if(f)qout("An error occurred!");};
! ver -output out
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.errorOccurred 方法](#) (p. 128)

## OSRedirect.output 方法

子进程生成三种类型的结果信息：通常写入屏幕的常规输出、通常写入屏幕的错误输出和最终退出代码（“错误级别”）。该方法返回一个字符串，其中包含由子进程写入 `stdout` 的输出。

该方法具有以下语法：

```
output()
```

该方法返回包含命令输出的字符串。

### 示例

获取 Windows OS 版本字符串：

```
!! ver
? $$stdout.output()
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.errout 方法](#) (p. 129)

[OSRedirect.result 方法](#) (p. 134)

## OSRedirect.receiveErrOutput 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则当子进程写入 `stderr` 时调用该事件处理程序。可以多次调用该处理程序。

该方法具有以下语法：

```
receiveErrOutput(s)
```

**注意：**AutoShell 忽略事件处理程序返回值。

### 示例

接收时显示错误输出：

```
out = new OSRedirect();
out.receiveErrOutput = function(s){qout(s)};
! ver -output out
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.receiveOutput 方法](#) (p. 133)

## OSRedirect.receivedOutput 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则当子进程写入 `stdout` 时调用该事件处理程序。可以多次调用该处理程序。

该方法具有以下语法：

```
receivedOutput(s)
```

**注意：** AutoShell 忽略事件处理程序返回值。

### 示例

接收时显示输出：

```
out = new OSRedirect();
out.receivedOutput = function(s){qout(s)};
! ver -output out
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.receivedErrOutput 方法](#) (p. 132)

[OSRedirect.receivedResult 方法](#) (p. 133)

## OSRedirect.receivedResult 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则当设置子进程的退出代码时调用该事件处理程序。

该方法具有以下语法：

```
receivedResult(rc)
```

**注意：** AutoShell 忽略事件处理程序返回值。

### 示例

显示事件处理程序的退出代码：

```
out = new OSRedirect();
out.receivedResult = function(rc){qout("Exit code", rc)};
! ver -output out
```

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.receivedOutput 方法](#) (p. 133)

## OSRedirect.result 方法

子进程生成三种类型的结果信息：通常写入屏幕的常规输出、通常写入屏幕的错误输出和最终退出代码（“错误级别”）。

该方法具有以下语法：

```
result()
```

该方法返回子进程退出代码。

### 示例

在 Windows 上，如果指定的主机进行响应，则 ping 操作将退出代码设置为 0，否则设置为 1。检查并显示结果：

```
!! ping asclil
if($?stdout.result()==0)
    ? "Ping successful"
else
    ? "Ping failed"
```

**注意：**应用程序特定的故障不会导致设置 OSRedirect() 的 errorOccurred() 标志。

另请参阅：

[! 命令—调用命令或子进程 \(Cmdlet\)](#) (p. 45)

[!! 命令—调用命令或子进程并自动捕获输出 \(Cmdlet\)](#) (p. 44)

[OSRedirect.output 方法](#) (p. 131)

## RemoteTarget 类

RemoteTarget 是一个实用工具类，用于在远程脚本执行中与 run-remote 命令配合使用。RemoteTarget 指定目标系统的详细信息（例如，主机名和地址、用于身份验证的凭据）或检索状态和结果信息（例如，错误情况、输出和最终结果）。

### 构造器

```
RemoteTarget(host [,user] [,pass] [,key] [,phrase] [, port])
```

### 方法

```
void abort()
int createRemoteContext()
int destroyRemoteContext()
bool errorOccurred()
bool execute(strScript)
int getError()
public String getHostName() {
bool hasCompleted()
string output()
string result()
bool wasAborted()
```

### 事件处理程序

```
int onError(e)
int receivedOutput(s)
int receivedResult(s)
```

### 变量

```
REM_ERR_NONE（静态、只读）
REM_ERR_CON（静态、只读）
REM_ERR_AUTH（静态、只读）
REM_ERR_CHN（静态、只读）
REM_ERR_READ（静态、只读）
REM_ERR_EXEC（静态、只读）
REM_ERR_ABORT（静态、只读）
REM_ERR_DISC（静态、只读）
```

另请参阅：

[RemoteTarget.RemoteTarget 构造函数](#) (p. 136)  
[RemoteTarget.abort 方法](#) (p. 138)  
[RemoteTarget.createRemoteContext 方法](#) (p. 138)  
[RemoteTarget.destroyRemoteContext 方法](#) (p. 139)  
[RemoteTarget.errorOccurred 方法](#) (p. 140)  
[RemoteTarget.execute 方法](#) (p. 141)  
[RemoteTarget.getError 方法](#) (p. 142)  
[RemoteTarget.getHostNames 方法](#) (p. 143)  
[RemoteTarget.hasCompleted 方法](#) (p. 144)  
[RemoteTarget.onError 方法](#) (p. 145)  
[RemoteTarget.output 方法](#) (p. 145)  
[RemoteTarget.receivedOutput 方法](#) (p. 146)  
[RemoteTarget.receivedResult 方法](#) (p. 147)  
[RemoteTarget.result 方法](#) (p. 147)  
[RemoteTarget.wasAborted 方法](#) (p. 148)  
[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

## RemoteTarget.RemoteTarget 构造函数

创建对象时调用的构造函数。构造函数不返回值。如果出现错误，它会引发异常。

该构造函数具有以下语法：

```
RemoteTarget(host [,user] [,pass] [,key] [,phrase] [, port])
```

### host

指定远程主机的名称。

### user

（可选）指定用于登录到远程节点的用户名。用户名是密码身份验证或公钥身份验证所必需的。如果没有指定用户名，则使用在 AutoShell 登录过程中输入的用户名。

**默认值：** \$\$User

### pass

（可选）指定用于登录到远程主机的密码。如果没有指定密码，则使用在 AutoShell 登录过程中输入的密码。

**默认值：** \$\$Pass



**key**

(可选) 对于包含用于登录到远程主机的 RSA 私钥的文件, 指定该文件的绝对路径或相对路径。如果指定了私钥和密码, AutoShell 将首先尝试公钥登录, 如果失败, 则尝试密码登录。

**默认值:** ""

**短语**

(可选) 为私钥指定密码。如果未加密密钥, 则不需要该密码。

**默认值:** ""

**port**

(可选) 指定连接到远程主机的端口。

**默认值:** SSH 标准端口 22。

**示例**

使用密码身份验证创建 RemoteTarget 对象:

```
rt = new RemoteTarget("ascli1", "bob", "ca123456");
```

如果公钥身份验证失败, 尝试使用后跟密码的公钥身份验证来创建 RemoteTarget 对象:

```
rt = new RemoteTarget("ascli1", "bob", "ca123456", "bob_rsa.key");
```

在这种情况下, 文件 bob\_rsa.key 包含 RSA 私钥。

**另请参阅:**

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

## RemoteTarget.abort 方法

默认情况下异步执行远程脚本。如果必须中止未完成的操作，请对与远程操作相关的 RemoteTarget 对象调用 abort()。

该方法具有以下语法：

```
abort()
```

该方法不返回值。

### 示例

中止远程操作：

```
rt = new RemoteTarget("ascli1");
run-remote Math.sin(Math.PI/2) on rt
if(!rt.hasCompleted())
{
    rt.abort()
    ? "Operation aborted:", rt.wasAborted()
}
```

另请参阅：

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

[RemoteTarget.wasAborted 方法](#) (p. 148)

## RemoteTarget.createRemoteContext 方法

默认情况下，为在 RemoteTarget 对象上运行的每个命令均创建一个新的 Autoshell 会话。在同一目标上运行多个命令时，使用单个永久会话来执行多个脚本更可取。另外，永久会话可维持状态。例如，在一次执行期间创建的变量在随后评估期间仍然可用。

如果对 RemoteTarget 对象调用 createRemoteContext()，将在远程系统上创建新的 Autoshell 会话，并且该会话保持打开状态直到调用 destroyRemoteContext() 或收集 RemoteTarget 对象。如果存在上下文时调用 createRemoteContext()，将销毁现有上下文并创建一个新上下文。

该方法具有以下语法：

```
createRemoteContext()
```

该方法返回以下错误代码之一：

**RemoteTarget.REM\_ERR\_NONE**

指示成功。

**RemoteTarget.REM\_ERR\_CON**

指示连接错误，例如，主机未知。

**RemoteTarget.REM\_ERR\_AUTH**

指示身份验证错误。

### 示例

创建永久会话，在首次评估中定义变量并在第二次评估中检索该变量：

```
rt=new RemoteTarget("ascl11");
if(rt.createRemoteContext()==RemoteTarget.REM_ERR_NONE)
{
    run-remote x=42 on rt
    run-remote x on rt -wait
    ? get-remoteResult(rt) // 42
    rt.destroyRemoteContext();
    ? get-remoteResult(rt) // 空
}
else
{
    ? "Error occurred!";
}
```

另请参阅：

[RemoteTarget.destroyRemoteContext 方法](#) (p. 139)

## RemoteTarget.destroyRemoteContext 方法

该方法销毁 createRemoteContext() 之前创建的上下文。在销毁永久上下文之后，RemoteTarget 对象仍然可以用于远程执行。该方法在内部针对每个脚本评估创建并销毁上下文，直到创建其他上下文为止。

该方法具有以下语法：

```
destroyRemoteContext()
```

该方法返回以下错误代码之一：

**RemoteTarget.REM\_ERR\_NONE**

指示成功。

**RemoteTarget.REM\_ERR\_DISC**

指示断开与远程系统的连接。

### 示例

创建永久会话，在首次评估中定义变量并在第二次评估中检索该变量：

```
rt=new RemoteTarget("ascli1");
if(rt.createRemoteContext()==RemoteTarget.REM_ERR_NONE)
{
    run-remote x=42 on rt
    run-remote x on rt -wait
    ? get-remoteResult(rt) // 42
    rt.destroyRemoteContext();
    ? get-remoteResult(rt) // 空
}
else
{
    ? "Error occurred!";
}
```

另请参阅：

[RemoteTarget.createRemoteContext 方法](#) (p. 138)

## RemoteTarget.errorOccurred 方法

该方法用于检查在远程执行期间是否发生错误。如果发生错误，`getError()` 会提供更多信息。

该方法具有以下语法：

```
errorOccurred()
```

该方法返回以下值：

**真**

指示发生了错误。

**假**

指示未发生错误。

### 示例

在远程执行完成之后检查错误:

```
rt = new RemoteTarget("ascli1");
run-remote Math.cos(0) on rt -wait
if(rt.errorOccurred())
{
    ? "Error occurred!"
}
else
{
    ? get-remoteResult(rt)
}
```

另请参阅:

[RemoteTarget.getError 方法](#) (p. 142)

## RemoteTarget.execute 方法

该方法是 `run-remote` 命令调用的核心方法。不应直接调用该方法，因为在将特定脚本传递到该方法之前 `run-remote` 会对其执行特定处理。

该方法具有以下语法:

```
execute(strScript)
```

该方法返回以下值:

**真**

指示异步执行成功开始。

**假**

指示开始远程执行时发生故障。

### 示例

评估表达式 (建议):

```
rt = new RemoteTarget("ascli1");
run-remote Math.sqrt(1) on rt
? get-remoteResult(rt);
```

评估表达式 (不建议):

```
rt = new RemoteTarget("ascli1");
rt.execute("Math.sqrt(1)");
? get-remoteResult(rt);
```

另请参阅:

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

## RemoteTarget.getError 方法

该方法在远程执行失败之后返回详细的错误信息。

该方法具有以下语法:

```
getError()
```

该方法返回以下错误代码之一:

**RemoteTarget.REM\_ERR\_NONE**

指示成功。

**RemoteTarget.REM\_ERR\_CON**

指示连接错误, 例如, 主机未知。

**RemoteTarget.REM\_ERR\_AUTH**

指示身份验证错误。

**RemoteTarget.REM\_ERR\_CHN**

指示在建立通道时发生错误。

**RemoteTarget.REM\_ERR\_READ**

指示从远程系统读取时发生错误。

**RemoteTarget.REM\_ERR\_EXEC**

指示评估表达式时发生错误。

**RemoteTarget.REM\_ERR\_ABORT**

指示用户中止操作的时间。

**RemoteTarget.REM\_ERR\_DISC**

指示断开与远程系统的连接时发生错误。

## 示例

显示详细错误信息:

```
rt = new RemoteTarget("ascli1");
run-remote Math.cos(1) on rt -wait
switch(rt.getError())
{
    case RemoteTarget.REM_ERR_NONE:
        ? "Success"
        break;
    case RemoteTarget.REM_ERR_CON:
        ? "Error connecting"
        break;
    case RemoteTarget.REM_ERR_AUTH:
        ? "Error authenticating"
        break;
    case RemoteTarget.REM_ERR_CHN:
        ? "Error establishing channel"
        break;
    case RemoteTarget.REM_ERR_READ:
        ? "Error reading from remote system"
        break;
    case RemoteTarget.REM_ERR_EXEC:
        ? "Error evaluating expression"
        break;
    case RemoteTarget.REM_ERR_ABORT:
        ? "Operation aborted by user"
        break;
    case RemoteTarget.REM_ERR_DISC:
        ? "Error disconnecting from remote system"
        break;
    默认:
        ? "Unknown error"
}
```

另请参阅:

[RemoteTarget.errorOccurred 方法](#) (p. 140)

[RemoteTarget.onError 方法](#) (p. 145)

## RemoteTarget.getHostName 方法

该方法返回创建 RemoteTarget 对象时指定的主机名或地址。

该方法具有以下语法:

```
getHostName()
```

该方法返回包含目标节点的主机名或地址的字符串。

### 示例

隐式创建 RemoteTarget 对象并检索主机名:

```
aRT = run-remote "! ver" on "ascli1"
? aRT[0].getHostName()
```

另请参阅:

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

[RemoteTarget.RemoteTarget 构造函数](#) (p. 136)

## RemoteTarget.hasCompleted 方法

该方法检查远程脚本评估的执行状态。如果 `abort()` 调用结束执行操作、处理期间发生错误或者在目标节点上成功运行脚本，则将执行视为完成。使用 `errorOccurred()` 和 `getError()` 方法确认执行是成功完成还是发生错误。

该方法具有以下语法:

```
hasCompleted()
```

该方法返回以下值:

**真**

执行完成。

**假**

执行未完成。

### 示例

手工实施等待完成所需要的时间。

```
aRT = run-remote "! ver" on "ascli1"
while(!aRT[0].hasCompleted())
{
    sleep(500);
}
if(aRT[0].errorOccurred())
{
    ? "Error"
}
else
{
    ? "Result:", get-remoteResult(aRT[0])
}
```

通常，脚本仅使用 `run-remote` 命令的 `-wait` 选项。



另请参阅：

[run-remote 命令—在远程系统上执行脚本 \(Funclet\)](#) (p. 86)

[RemoteTarget.abort 方法](#) (p. 138)

[RemoteTarget.errorOccurred 方法](#) (p. 140)

[RemoteTarget.getError 方法](#) (p. 142)

## RemoteTarget.onError 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则在远程执行期间发生错误时调用该事件处理程序。

该方法具有以下语法：

```
onError(e)
```

**注意：**AutoShell 忽略事件处理程序返回值。

### 示例

为 RemoteTarget 指定错误处理程序：

```
rt = new RemoteTarget("ascli1");  
// 分配事件处理程序  
rt.onError = function(e){qout("Error occurred, code=", e);};  
run-remote "2*2" on rt
```

另请参阅：

[RemoteTarget.errorOccurred 方法](#) (p. 140)

[RemoteTarget.getError 方法](#) (p. 142)

## RemoteTarget.output 方法

脚本生成两种类型的结果信息：通常写入屏幕的输出和最终返回值。该方法返回字符串，其中包含在远程系统上运行的脚本生成的输出。脚本正在运行时会计累输出，因此在脚本执行完成之前，该方法会返回非空结果。

该方法具有以下语法：

```
output()
```

该方法返回字符串，其中包含脚本写入 stdout 的日期。

### 示例

在 Windows 主机上远程查询正在运行中的服务列表:

```
rt = new RemoteTarget("ascli1", "bob", "casogood42");
run-remote "! sc query" on rt -wait
? rt.output()
```

另请参阅:

[RemoteTarget.result 方法](#) (p. 147)

## RemoteTarget.receivedOutput 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则在接收来自远程目标的输出时调用该事件处理程序。可以多次调用该处理程序。

该方法具有以下语法:

```
receivedOutput(s)
```

**注意:** AutoShell 忽略事件处理程序返回值。

### 示例

为 RemoteTarget 指定输出处理程序，并在事件处理程序中访问关联对象:

```
rt = new RemoteTarget("ascli1");
// 分配输出处理程序
rt.receivedOutput = function(s){
    qout("Output received=", s);
    qout("Originating host=", this.getHostName());
};
run-remote "! ver" on rt
```

另请参阅:

[RemoteTarget.output 方法](#) (p. 145)

## RemoteTarget.receiveResult 方法

事件处理程序是一种可选函数，当发生特定事件时，可以在用户脚本中实现并由 AutoShell 进行调用。如果存在事件处理程序，则在接收来自远程目标的最终结果值时调用该事件处理程序。

该方法具有以下语法：

```
receiveResult(s)
```

**注意：**AutoShell 忽略事件处理程序返回值。

### 示例

为 RemoteTarget 指定结果处理程序，并且在事件处理程序中访问关联对象：

```
rt = new RemoteTarget("ascli1");
// 分配输出处理程序
rt.receiveResult = function(s){
    qout("Result received=", s);
    qout("Originating host=", this.getHostName());
};
run-remote "Math.SQRT2" on rt
```

另请参阅：

[RemoteTarget.result 方法](#) (p. 147)

## RemoteTarget.result 方法

脚本生成两种类型的结果信息：通常写入屏幕的输出和最终返回值。该方法检索远程执行进程的返回值的表示。由于返回结果是脚本评估的最后一步，因此该方法仅在脚本执行成功完成时返回有效值。**hasCompleted()** 返回 **true**，**errorOccurred()** 返回 **false**。由于远程执行的返回值必须通过网络传输到发起请求的系统，因此会在远程系统上将该返回值序列化到 XML 表示中，然后 **RemoteTarget.result()** 方法返回其序列化表示的结果值。

使用 **get-remoteResult()** funclet（它将 **RemoteTarget** 对象本身作为参数）将序列化表示形式转换为 JavaScript 正则值。通常不需要脚本直接调用 **result()**。序列化可保持类型信息。例如，如果远程节点返回一个数字，那么 **get-remoteResult()** 会返回数字类型的值。如果远程系统返回“日期”对象，那么该对象在原始系统上同样为日期。

远程脚本的返回值并不仅限于简单数据类型。也可以返回复杂数组或仅包含数据的对象，以允许远程执行在服务器之间传输大量数据。唯一的限制是数据结构不能包含任何循环引用。

该方法具有以下语法：

```
result()
```

远程脚本返回包含值的序列化 XML 表示的字符串。

### 示例

获取远程执行的序列化表示形式和实际值：

```
rt = new RemoteTarget("ascii1", "bob", "ca123456");
run-remote "[1,2,3]" on rt -wait
? rt.result()
v = get-remoteResult(rt) // 内部调用 result()
? typeof v              // 数组
? typeof v[0]           // 数字
arrdump v
```

另请参阅：

[get-remoteResult 命令-从远程目标获取结果 \(Funclet\)](#) (p. 63)

[RemoteTarget.output 方法](#) (p. 145)

## RemoteTarget.wasAborted 方法

该方法检查是否通过调用 `abort()` 以编程方法中止了远程脚本执行。

该方法具有以下语法：

```
wasAborted()
```

该方法返回以下值：

**真**

指示执行已中止。

**假**

指示执行尚未中止。

### 示例

检查远程操作是否已中止:

```
rt = new RemoteTarget("ascli1");
run-remote Math.sin(3/2*Math.PI) on rt
if(!rt.hasCompleted())
{
    rt.abort()
    ? "Operation aborted:", rt.wasAborted()
}
```

另请参阅:

[RemoteTarget.abort 方法](#) (p. 138)

## AutoShell 可加载模块的命令参考

本节介绍了属于以下特定于平台的可选 AutoShell 可加载模块的命令:

- Microsoft Hyper-V
- Oracle Solaris Zones
- VMware vCenter Server

详细信息

[CA IBM LPAR AutoShell 命令](#) (p. 283)

[CA Microsoft Hyper-V AutoShell 命令](#) (p. 188)

[CA Oracle Solaris Zones AutoShell 命令](#) (p. 307)

[CA VMware vCenter 服务器 AutoShell 命令](#) (p. 149)

### CA VMware vCenter 服务器 AutoShell 命令

可以使用 AutoShell 编写 CA VMware vCenter 服务器命令的脚本并使其自动化，然后根据命令的结果运行操作。CLI 中也提供了相应的命令。

## dpmvc-addesxhost 命令—添加 ESX 主机 (Funclet)

dpmvc-addesxhost 命令用于将 ESX 主机添加到 vCenter 服务器中。

此命令具有以下格式：

```
dpmvc-addesxhost
-esx_host_name esxhostname
-vc_server vcservername
-esx_host_user esxhostuser
-esx_host_password esxhostpassword
[-management_ip managementip]
[-host_connect_port hostconnectport]
[-folder_name foldername]
[-datacenter_name datacentername]
[-cluster_name clustername]
[-vim_account_user vimaccountuser]
[-vim_account_password vimaccountpassword]
```

### **-esx\_host\_name esxhostname**

指定托管 VM 的 ESX 服务器的名称。

### **-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-esx\_host\_user esxhostuser**

（可选）指定用于访问 ESX 主机的用户名。

### **-esx\_host\_password esxhostpassword**

指定用于访问 ESX 主机的密码。

### **-management\_ip managementip**

（可选）指定管理 IP。

### **-host\_connect\_port hostconnectport**

（可选）指定 ESX 主机用于与 vCenter 服务器通信的端口。

### **-folder\_name foldername**

（可选）指定可以添加 ESX 主机的文件夹的名称。

### **-datacenter\_name datacentername**

（可选）指定 VM 所在的数据中心。

### **-cluster\_name clustername**

（可选）指定可以添加 ESX 主机的群集的名称。

### **-vim\_account\_user vimaccountuser**

（可选）指定 VIM 帐户的用户名。

**-vim\_account\_password *vimaccountpassword***

（可选）指定用于访问 VIM 帐户的密码。

#### 示例：添加 ESX 主机

本示例将 ESX 主机 “esx1” 添加到 vCenter 服务器 “myvcenter” 中。

```
dpmvc-addesxhost -esx_host_name esx1 -vc_server myvcenter -esx_host_user admin
-esx_host_password topsecret
```

### dpmvc-addvmvdisk 命令—添加虚拟磁盘 (Funclet)

dpmvc-addvmvdisk 命令用于将虚拟磁盘添加到 VM 中。

此命令具有以下格式：

```
dpmvc-addvmvdisk
-datastore_name datastorename
[-datacenter_name datacentername]
-vm_name vmname
-vc_server vcservername
-capacity_in_mb disk_capacity
[-controller_key controllerkey]
[-disk_mode
{append|independent_nonpersistent|independent_persistent|nonpersistent|persistent|undoable}]
-thin_provisioning {no|yes}
[-unit_number unitnumber]
```

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-capacity\_in\_mb *disk\_capacity***

指定虚拟磁盘的容量 (MB)。

**-controller\_key *controllerkey***

（可选）指定磁盘的控制器密钥。

**-disk\_mode {append | independent\_nonpersistent | independent\_persistent | nonpersistent | persistent | undoable}**

（可选）指定磁盘的模式。

**-thin\_provisioning {no | yes}**

（可选）指示磁盘是否必须精简开通。

**-unit *disk\_unitnumber***

（可选）指定磁盘的单元编号。

#### 示例：添加虚拟磁盘

本示例将虚拟磁盘添加到 VM “myvm” 中。

```
dpmvc-addvmvdisk -datastore_name disk1 -vm_name myvm -vc_server myvcenterserver  
-capacity_in_mb 20000 -thin_provisioning no
```

### dpmvc-addvmvnic 命令—添加虚拟 NIC (Funclet)

dpmvc-addvmvnic 命令用于将虚拟 NIC 添加到 VM 中。

此命令具有以下格式：

```
dpmvc-addvmvnic  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
-devicetype {e1000 | vmxnet}  
-network_name networkname  
[-mac_address macaddress]  
-wake_on_lan_enabled {no | yes}
```

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-devicetype {e1000 | vmxnet}**

表示网络设备的类型。



**-network\_name *networkname***

指定 NIC 必须连接到的虚拟网络。您可以根据以下命名约定区分标准交换机和分布式虚拟交换机的名称：

- 对于标准交换机，其名称为网络名称。
- 对于分布式虚拟交换机，其名称为 **dvPort** 组名称后跟用括号括起来的分布式虚拟交换机名称：**dvPortGroupName (dvSwitchName)**

**-mac\_address *macaddress***

（可选）指定网络设备的 MAC（介质访问控制）地址。

**-wake\_on\_lan\_enabled {no | yes}**

表示是否在启用 LAN 时启动 VM。

**示例：添加 vNIC**

本示例将虚拟 NIC 添加到 VM 中。

```
dpmvc-addvmvnic -vm_name myvm -vc_server myvcenterserver -devicetype vmxnet  
-network_name net1 -wake_on_lan_enabled no
```

## dpmvc-clone 命令—克隆 VM (Funclet)

dpmvc-clone 命令用于克隆 VM。

此命令具有以下格式：

```
dpmvc-clone
{-template_name templatename | -cloned_from vmname}
{-datacenter_name datacentername | -vc vcenterservername}
-vm_name vmname
-esx_host_name esxhostname
-datacenter_name datacentername
-data_store_name datastorename
-compute_resource_name name
-resource_pool_name resourcepool
-spec_name specificationname
[-auto_deploy value]
[-deploy_template templatename]
[-network_connection
"nic=sequence_number,network_name=name,ip_address=ip_address,def_gateway=defa
ult_gateway,alt_gateway=alt_gateway,subnet_mask=subnet_mask[,win_dns_server=w
ins_dns_server,win_alt_dns=wins_alt_dns,wins_primary=wins_primary,wins_second
ary=wins_secondary"]
[-global_dns_search_suffix suffix1[,suffix2,suffix3,...]]
[-linux_domain_name domain_name]
[-linux_dns_servers primary_dns=value,secondary_dns=value,tertiary_dns=value]
[-create_disk MB,datastorename,controller]
[-modify_disk key, datastorename]
[-set_memory number]
[-set_cpu number]
-vm_os_username username
[-vm_os_password password]
[-auth_file authorizationfilename]
[-auth_comp componentID]
[-scalability_server scalabilityservername]
```

### **-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

### **-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

### **-compute\_resource\_name *name***

指定在其中创建 VM 的群集或 VMware ESX 主机。

### **-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

### **-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-vm\_name *vmname***

指定 VM。

**-resource\_pool\_name *resourcepool***

指定您希望从中选择用于克隆的 VM 的资源池的名称。

**-spec\_name *specificationname***

(可选) 指定要用于所克隆虚拟机的规格的名称。

**-auto\_deploy {yes|no}**

指定是否自动部署 CA Virtual Assurance 代理。选项包括：

**是**

自动部署 CA Virtual Assurance 代理。

**否**

防止自动部署 CA Virtual Assurance 代理。

**默认值:** no

**-deploy\_template *templatename***

(可选) 指定在 CA Virtual Assurance 中创建的通用模板的名称。

**注意:** 请勿将该模板与由 VMware vCenter 创建并管理的模板相混淆。

**-network\_connection**

**"*nic=sequence\_number,network\_name=name,ip\_address=ip\_address,def\_gateway=default\_gateway,alt\_gateway=alt\_gateway,subnet\_mask=subnet\_mask,win\_dns\_server=wins\_dns\_server,win\_alt\_dns=wins\_alt\_dns,wins\_primary=wins\_primary,wins\_secondary=wins\_secondary*"**

**Windows:**

(可选) 为 VM 创建网络连接。该参数可以多次指定。序列号是以 (1) 开头的整数值，用于标识网络连接。如果参数包含任何空格，则必须用括号将整组参数括起来。通过输入没有值的名称（例如，

“*nic=123,alt\_gateway=*”）可以重置除 *nic*、*network\_name*、*ip\_address*、*def\_gateway* 和 *subnet\_mask* 以外的所有值。

**-network\_connection**

**"*nic=sequence\_number,network\_name=name,ip\_address=ip\_address,def\_gateway=default\_gateway,alt\_gateway=alt\_gateway,subnet\_mask=subnet\_mask*"**

**Linux:**

(可选) 创建一个或多个网络连接。该参数可以多次指定。序列号是以 (1) 开头的整数值，用于标识网络连接。如果参数包含任何空格，则必须用括号将整组参数括起来。通过输入没有值的名称（例如，

“*nic=123,alt\_gateway=*”）只能重置 *alt\_gateway*。

**-global\_dns\_search\_suffix *suffix1{,suffix2,suffix3,...}***

**Linux:**

指定一个或多个 DNS 搜索后缀。

**Windows:**

(可选) 指定一个或多个 DNS 搜索后缀。

**-linux\_domain\_name *domain\_name***

**Linux:**

(可选) 指定 Linux VM 的域名。

**-linux\_dns\_servers**

***primary\_dns=value{,secondary\_dns=value,tertiary\_dns=value}***

**Linux:**

(可选) 指定 Linux VM 的 DNS 服务器的 IP 地址。可以省略一个或多个值 (例如,

`primary_dns=123.123.123.13,secondary_dns=,tertiary_dns=123.123.123.15`), 以便取消设置该值。

**-create\_disk *MB,datastore\_name,controller***

(可选) 创建一个或多个附加的硬盘驱动器。该参数可以多次指定。

**MB**

定义硬盘驱动器的大小 (MB)。

**Datastore\_name**

指定该硬盘驱动器的数据存储的名称。当前, 每个 VM 仅支持一个数据存储。

**控制器**

指定该硬盘驱动器的控制器密钥。

**-modify\_disk {*key,datastore\_name*}**

(可选) 指定硬盘驱动器驻留的数据存储。可以多次指定。

**key**

指定现有磁盘。

**datastore\_name**

指定该磁盘移动到的数据存储的名称。

**-set\_memory *number***

(可选) 定义用于克隆的 VM 的内存量 (MB)。该值会覆盖在模板中指定的值。

**-set\_cpu *number***

(可选) 定义克隆的 VM 的 CPU 的数量。该值会覆盖在模板中指定的值。

**-vm\_os\_username *user name***

指定所克隆 VM 的用户。自动部署映像时，还可以使用该用户名进行身份验证。

**Windows:** 必须在自定义规格中定义用户名。

**Linux:** 必须与模板中定义的用户名相同。

**-vm\_os\_password *password***

(可选) 指定所克隆 VM 的用户密码。自动部署映像时，还可以使用该密码进行身份验证。

**Windows:** 必须与自定义规格中定义的密码相同。

**Linux:** 必须与模板中定义的密码相同。

**-auth\_file *authorizationfilename***

(可选) 指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭证。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

(可选) 指定可用于对主机和用户进行分组的组件 ID。

**-template\_name *templatename***

指定要用于所克隆虚拟机的模板的名称。该参数取代了 `-vm_name_cloned_from`，不能两者一起指定。

**-vm\_name\_cloned\_from *vmname***

指定将用作克隆的虚拟机的模板的虚拟机的名称。该参数用于替换 `-template_name`，您无法同时指定两者。

**-resource\_pool\_name *resourcepool***

指定您希望从中选择用于克隆的 VM 的资源池的名称。

**-scalability\_server *servername***

(可选) 指定要用于软件分发的 Unicenter DSM 可扩展性服务器。

**-spec\_name *specificationname***

(可选) 指定要用于所克隆虚拟机的规格的名称。

**-compute\_resource\_name *name***

指定在其中创建 VM 的群集或 VMware ESX 主机。

### 示例：使用全局凭据克隆 VM

本示例使用数据中心 Specifies VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01”。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name VAS/MyCity -datastore_name storage1
-compute_resource_name VPMcluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword
```

### 示例：使用授权文件和组件克隆 VM

本示例使用数据中心 VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01”。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name Specifies VAS/MyCity -datastore_name
storage1
-compute_resource_name VPMcluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -auth_file c:\localauth.dat -auth_comp Imaging
```

### 示例：使用全局凭据克隆 VM 并创建两个 CPU

本示例使用数据中心 VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01”，并为该 VM 创建两个 CPU。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name VAS/MyCity -datastore_name storage1
-compute_resource_name VPMcluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -set_cpu 2
```

### 示例：使用默认授权文件克隆 VM 并将内存设置为 4096 MB

本示例使用数据中心 VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01”，并将内存设置为 4096。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name VAS/MyCity -datastore_name storage1
-compute_resource_name VPMcluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -set_memory 4096
```

**示例：使用默认授权文件克隆 VM 并创建两个硬盘**

本示例使用数据中心 VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01”，并为数据存储 storage1 创建两个硬盘。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name VAS/MyCity -datastore_name storage1
-compute_resource_name VAScluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -create_disk 10000,storage1,1000
-create_disk 10000,storage1,2001
```

**示例：克隆 VM 并创建网络连接**

本示例使用数据中心 VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01” 和网络连接，并设置初始值。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name VAS/MyCity -datastore_name storage1
-compute_resource_name VAScluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -network_connection "nic=1,network_name=VM
Network,ip_address=123.321.5.22,def_gateway=123.321.1.108,subnet_mask=255.255
.255.0,win_dns_server=123.321.42.1,win_alt_dns=123.321.3.101,wins_primary=123
.321.3.3"
```

**示例：使用默认授权文件从已关闭电源的 VM 克隆 VM**

本示例使用 VM “testvm02” 替代数据中心 VAS/MyCity 上的模板来创建 VM “testvm01”。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -datacenter_name "VAS/MyCity" -vm_name testvm01 -datastore_name
storage1
-esx_host_name vc1.ca.com -resource_pool_name Resources/DPMTest
-compute_resource_name DPMcluster/ussdCluster -vm_name_cloned_from testvm02
-auto_deploy yes -spec_name w2k3_spec_01 -vm_os_username administrator
-vm_os_password mypassword
```

**示例：使用默认授权文件克隆 VM 并修改现有硬盘**

本示例使用数据中心 VAS/MyCity 上的模板 “BaseW2k3” 创建 VM “testvm01”，并为数据存储 storage1 创建两个硬盘。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc-clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name VAS/MyCity -datastore_name storage1
-compute_resource_name VPMcluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -create_disk 10000,storage1,1000 -modify_disk
2000,storage2
```

## dpmvc-createdatastore 命令—创建数据存储 (Funclet)

dpmvc-createdatastore 命令用于创建数据存储。

此命令具有以下格式：

```
dpmvc-createdatastore
-datastore_name datastorename
-esx_host_name esxhostname
-vc_server vcservername
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
-datastore_type {local|nas|vmfs}
[-vmfs_device_path path]
[-vmfs_block_size_mb blocksize]
[-local_path lpath]
[-nas_user_name nasuser]
[-nas_password naspasswd]
[-nas_remote_hostname nasremhost]
[-nas_remote_path nsarempath]
[-nas_access_mode {read-only|read-write}]
[-sc sc_url]
```

### **-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

### **-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

### **-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

### **-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

### **-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

### **-datastore\_type {vmfs|nas|local}**

指定要创建的数据存储的类型（VMFS、NAS 或 local）。



**-vmfs\_device\_path *path***

(可选) 指定创建数据存储时使用的 VMFS 磁盘的设备路径。您可以使用 “getavailablescsidisks” 命令获取该路径。

**-vmfs\_block\_size\_mb *blocksize***

(可选) 指定创建数据存储时使用的块大小 (MB)。可用的块大小为 1、2、4 和 8。块大小使 VM 磁盘文件分别达到最大值 256 GB、512 GB、1024 GB 和 2048 GB。

**-local\_path *lpath***

(可选) 创建本地数据存储时，该路径指定创建数据存储时使用的 ESX 主机本地文件系统路径。

**-nas\_user\_name *nasuser***

(可选) 为创建 NAS 数据存储指定远程主机用户名。

**-nas\_password *naspasswd***

(可选) 为创建 NAS 数据存储指定远程主机密码。

**-nas\_remote\_hostname *nasremhost***

(可选) 指定托管基于网络的存储的服务器主机名。

**-nas\_remote\_path *nasrempath***

(可选) 指定远程服务器上用于 NAS 数据存储的文件系统路径

**-nas\_access\_mode *nasaccmode***

(可选) 指定用于 NAS 数据存储的基于网络存储的所需访问模式。可用的模式为 “只读” 和 “读写”。

**默认值:** “读写”

## dpmvc-cycle 命令—循环使用 VM (Funclet)

dpmvc-cycle 命令用于打开、关闭、重置或挂起 VM。

此命令具有以下格式:

```
dpmvc-cycle
-powerop {poweron|poweroff|reset|suspend|shutdown guest}
{-datacenter_name datacentername | -vc vcservername}
-vm_name vmname
```

**-powerop {poweron|poweroff|reset|suspend|shutdown guest}**

指定要在 VM 上执行的电源操作。选项包括:

**poweron**

打开 VM 电源。

**poweroff**

关闭 VM 电源。

**reset**

重置 VM。

**挂起**

暂时挂起 VM。

**shutdown guest**

正常关闭来宾操作系统。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

**示例：当数据中心内有多个 vCenter 服务器时关闭 VM**

本示例关闭特定 vCenter 服务器中的 VM。

```
dpmvc-cycle -powerop poweroff -vc my_server1 -vm_name my_vm
```

**示例：在单 vCenter 服务器环境中打开 VM**

本示例关闭属于数据中心“VAS/MyCity”的 VM “testvm”。

```
dpmvc-cycle -vm_name testvm -powerop poweron -datacenter_name VAS/MyCity
```

**dpmvc-datastore 命令—获取数据存储属性 (Funclet)**

dpmvc-datastore 命令用于检索特定 VMware vCenter 数据存储的可用空间和容量设置。

此命令具有以下格式：

```
dpmvc-datastore  
{-datacenter_name datacentername | -vc vcenterservername}  
-datastore_name datastorename  
[-getproperty {all|capacity|freespace}]
```

**-getproperty {all|capacity|freespace}**

（可选）指定要检索哪个属性。选项包括：

**all**

检索数据存储中的容量和可用空间。

**capacity**

检索数据存储中的容量。

**freespace**

检索数据存储中的可用空间。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**示例：获取数据存储的全部属性**

本示例检索数据存储的可用空间和容量。

```
dpmvc-datastore -getproperty all -datacenter_name VAS/MyCity -datastore_name storage1
```

**dpmvc-delete 命令—销毁 VM (Funclet)**

dpmvc-delete 命令用于删除处于关闭状态的 VM。使用此命令清理并释放未用过的资源。

**重要信息！** 在发出该命令之前，请验证是否已备份所有重要数据。该命令会删除 VM 数据存储以及 VM 的数据文件，包括磁盘映像。

此命令具有以下格式：

```
dpmvc-delete  
{-datacenter_name datacentername | -vc vcenterservername}  
-vm_name vmname
```

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

**示例：在单 vCenter 服务器环境中删除 VM**

本示例删除 VM “vm11”。

```
dpmvc-delete -vm_name vm11 -datacenter_name lab444
```

**示例：在多 vCenter 服务器环境中删除 VM**

本示例删除 VM “vm11”。

```
dpmvc-delete -vm_name vm11 -datacenter_name lab444 -vc vc_server_1
```

**dpmvc-distributedswitch 命令—管理虚拟分布式交换机 (Funclet)**

dpmvc distributedswitch 命令用于管理虚拟分布式交换机。

- 将新的虚拟分布式交换机添加到数据中心
- 更新虚拟分布式交换机的属性
- 删除虚拟分布式交换机
- 为虚拟分布式交换机创建新的分布式端口组
- 更新虚拟分布式交换机的端口组属性
- 从虚拟分布式交换机中删除分布式端口组
- 重命名虚拟分布式交换机的分布式端口组

此命令具有以下格式：

```
dpmvc-distributedswitch {-vds_add | -vds_update}  
-vc_server vcservername  
-datacenter_name datacentername  
-vds_folder vdsfolder  
-switch_name switchname  
[-hostnics hostname1:nic1,nic2,...nicn [;hostname2:nic2,...nicn >]*]  
[-uplink_port_names uplink1[,uplink2,...,uplinkn]]  
[-maxports maxports]  
  
dpmvc-distributedswitch -vds_remove  
-vc_server vcservername  
-switch_name switchname  
  
dpmvc-distributedswitch -add_portgroup  
-vc_server vcservername  
-switch_name switchname  
-portgroup_name portgroupname  
[-bindtype earlyBinding | ephemeral | lateBinding]  
[-vlan vlanID]  
[-numports numberofports]
```

```
dpmvc-distributedswitch -update_portgroup  
-vc_server vcservername  
-portgroup_name portgroupname  
[-portgroup_newname portgroupnewname]  
[-bindtype earlyBinding | ephemeral | lateBinding]  
[-vlan vlanID]  
[-numports numberofports]
```

```
dpmvc-distributedswitch -remove_portgroup  
-vc_server vcservername  
-portgroup_name portgroupname
```

```
dpmvc-distributedswitch -rename_portgroup  
-vc_server vcservername  
-portgroup_name portgroupname  
-portgroup_newname portgroupnewname
```

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-datacenter\_name *datacentername***

指定虚拟分布式交换机所属的数据中心。

**-vds\_folder *vdsfolder***

在 CA Virtual Assurance 资源管理器窗格中指定虚拟分布式交换机的文件夹。

**-switch\_name *switchname***

指定操作所针对的交换机名称。

**-hostnics *hostname1:nic1,nic2,...nicn* [*;hostname2:nic2,...nicn >*]**

(可选) 指定与 ESX 主机成员相关联的 NIC 列表。

**-uplink\_port\_names *uplink1[,uplink2,...,uplinkn]***

(可选) 指定要使用的上行链路端口名称的逗号分隔列表。

**-maxports *maxports***

(可选) 指定用于交换机的最大端口数。该数值可限制交换机中所有端口组的端口总数 (numports)。

**-bindtype *earlyBinding* | *ephemeral* | *lateBinding***

(可选) 指定端口组的绑定类型。有效值是:

#### **earlyBinding**

在 VM 绑定至端口组时分配端口。这种类型的绑定可确保始终连接，但永久保留端口。此绑定类型为默认值。

#### **lateBinding**

如果 VM 电源打开且其 NIC 处于连接状态，则将端口分配给该 VM。当 VM 电源关闭或其 NIC 断开连接时，此绑定类型将重新分配端口。LateBinding 可通过 vCenter 进行配置。

#### **ephemeral**

如果 VM 电源打开且其 NIC 处于连接状态，则将端口分配给该 VM。当 VM 电源关闭或其 NIC 断开连接时，此绑定类型将重新分配端口。Ephemeral 绑定可通过 ESX 主机和 vCenter 进行配置。

**-numports *numberofports***

(可选) 指定端口组的端口数。

**-portgroup\_name *portgroupname***

指定端口组名称。

**-portgroup\_newname *portgroupnewname***

指定新的端口组名称。

**-vlan *vlanID***

（可选）指定用于虚拟端口组操作的整数值 (vlan ID)。

**示例：创建新的虚拟分布式交换机**

本示例创建新的虚拟分布式交换机。

```
dpmvc-distributedswitch -vds_add -vc_server vc5master -datacenter_name dc3
-vds_folder vds -switch_name vdsSwitch1
```

**示例：更新现有的虚拟分布式交换机**

本示例更新虚拟分布式交换机。为 MYSERVER1 指定 NIC，并从虚拟分布式交换机中删除 MYSERVER2。如果要从虚拟分布式交换机中删除主机，请指定服务器名称，但不要为其分配 NIC（服务器名称后跟一个冒号，如 MYSERVER2:）。

```
dpmvc-distributedswitch -vds_update -vc_server VAS-VC5 -datacenter_name VC5
-switch_name vdistSwitch -hostnics MYSERVER1:nxmgt2,tmp2;MYSERVER2:
```

该命令用于更新 vdistSwitch:

- 主机 MYSERVER1 正在使用 NIC nxmgt2 和 tmp2
- 主机 MYSERVER2 已从交换机中删除

**dpmvc-entermaintenancemode 命令—设置为维护模式 (Funclet)**

dpmvc-entermaintenancemode 命令用于将 ESX 主机设置为维护模式。

此命令具有以下格式:

```
dpmvc-entermaintenancemode
-esx_host_name esxhostname
-vc_server vcservername
-timeout timeout
[-evacuate_powered_off_vm {no | yes}]
```

**-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-timeout *timeout***

指定进入或退出维护模式的超时时间（秒）。

**-evacuate\_powered\_off\_vm {no | yes}**

（可选）指定是否在 VM 停止后退出。

### 示例：进入维护模式

本示例设置 ESX 主机使其进入维护模式。

```
dpmvc-entermaintenancemode -esx_host_name esx1 -vc_server myvcserver -timeout 60
```

## dpmvc-exitmaintenancemode 命令—退出维护模式 (Funclet)

dpmvc-exitmaintenancemode 命令用于将 ESX 主机设置为退出维护模式。

此命令具有以下格式：

```
dpmvc-exitmaintenancemode  
-esx_host_name esxhostname  
-vc_server vcservername  
-timeout timeout
```

### **-esx\_host\_name esxhostname**

指定 VM 驻留的 VMware ESX 服务器。

### **-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-timeout timeout**

指定进入或退出维护模式的超时时间（秒）。

### 示例：退出维护模式

本示例设置 esx1 ESX 主机，使其退出维护模式。

```
dpmvc-exitmaintenancemode -esx_host_name esx1 -vc_server myvcserver  
-timeout 60
```



## dpmvc-faulttolerance 命令—指定容错操作 (Funclet)

dpmvc-faulttolerance 命令用于打开、关闭、启用、禁用、测试容错或迁移辅助 VM。

此命令具有以下格式：

```
dpmvc-faulttolerance
-turn_on -vm_name <value> [-esx_host_name <value>]|
-turn_off -vm_name <value>|
-enable -vm_name <value>|
-disable -vm_name <value>|
-test_ft -vm_name <value>|
-migrate_secondary -vm_name <value> -esx_host_name <value>
```

### -turn\_on

对指定 VM 打开容错。

（可选）指定辅助 VM 的 ESX 主机名称。

### -turn\_off

对指定 VM 关闭容错。

**注意：**该操作将删除辅助 VM。

### -enable

对指定 VM 启用容错。

### -disable

对指定 VM 禁用容错。

### -test\_ft

在指定 VM 上测试容错。

### -migrate\_secondary

将容错辅助 VM 迁移到其他 ESX 服务器上。-esx\_host\_name 选项是必需选项。

### -vm\_name

定义计算机资源的名称。

### -esx\_host\_name

指定 VM 所在的 ESX 服务器。

### -vc\_server

指定虚拟中心服务器主机名。

## dpmvc-getavailablesids 命令—获取可用 SCSI 磁盘 (Funclet)

dpmvc-getavailablesids 命令用于列出可用的 SCSI 磁盘。

此命令具有以下格式：

```
dpmvc-getavailablesids  
[-datastore_name datastorename]  
-esx_host_name esxhostname  
-vc_server vcservername  
[-ws_user wsuser]  
[-ws_password wspassword]  
[-prompt {yes|no}]
```

### **-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

### **-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

### **-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

### **-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

### **-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

## dpmvc-gethostb 命令—获取主机总线适配器 (Funclet)

dpmvc-gethostb 命令用于检索在指定 ESX 主机上配置的主机总线适配器列表。

此命令具有以下格式：

```
dpmvc-gethostb  
-esx_host_name esxhostname  
-vc_server vcservername  
[-ws_user wsuser]  
[-ws_password wspassword]  
[-prompt {yes|no}]
```

**-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**dpmvc-getresources 命令—获取 VM 资源 (Funclet)**

dpmvc-getresources 命令用于检索 CPU 或内存的份额限制或保留。

此命令具有以下格式:

```
dpmvc-getresources
-vm_name vmname
{-datacenter_name datacentername | -vc vcservername}
[-resource {all|vm_cpulimit|vm_cpureserv|vm_memlimit|vm_memresrv}]
```

**-vm\_name *vmname***

指定 VM。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcservername***

指定 VM 所在的 vCenter 服务器。

**-resource {all /vm\_cpulimit /vm\_cpureserv /vm\_memlimit /vm\_memresrv}**

指定是检索特定资源还是所有资源。选项包括：

**all**

检索用于 VM 的所有 CPU 和内存份额限制和保留。

**vm\_cpulimit**

检索用于 VM 的 CPU 份额数的限制。

**vm\_cpureserv**

检索专用于 VM 的 CPU 份额数。

**vm\_memlimit**

检索用于 VM 的内存份额数的限制。

**vm\_memresrv**

检索专用于 VM 的内存份额数。

**示例：在单 vCenter 服务器环境中获取 VM 的所有资源**

本示例检索 usa-vm2 的所有资源。

```
dpmvc-getresources -resource all -datacenter_name MyCity -vm_name usa-vm2
```

## **dpmvc-getshares 命令—获取 VM 份额 (Funclet)**

dpmvc-getshares 命令用于查看分配给某个 VM 的 CPU 或内存份额的数量。

此命令具有以下格式：

```
dpmvc-getshares  
{-datacenter_name datacentername | -vc vcenterservername}  
-vm_name vmname  
[-share {cpu | memory}]
```

**-share {cpu | memory}**

显示分配给某个 VM 的 CPU 或内存份额的数量。份额用于确定与其他正在运行的 VM 相比，分配给哪些 VM 的可用物理资源的比例较高。如果向某个 VM 分配的份额较多，则会向该 VM 分配较多的物理资源。分配给 VM 的份额越少，提供给该 VM 的物理资源就越少。

**cpu**

指定正在检索 CPU 值。

**memory**

指定正在检索内存值。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

#### 示例：获取 VM 的 CPU 值

本示例检索 usa-vm2 的 CPU 值。

```
dpmvc-getshares -resource cpu -datacenter_name MyCity -vm_name usa-vm2
```

### dpmvc-getsnapshots 命令—列出 VM 的快照 (Funclet)

dpmvc-getsnapshots 命令用于列出 VM 的快照。

此命令具有以下格式：

```
dpmvc-getsnapshots  
{-datacenter_name datacentername | -vc vcenterservername}  
-vm_name vmname  
[-name_only]
```

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

**-name\_only**

（可选）仅提供快照名称。

#### 示例：列出快照

本示例列出某个 VM 的快照。

```
dpmvc-getsnapshots -vc my_server1 -vm_name my_vm
```

## dpmvc-imgjobcheck 命令—获取克隆作业状态 (Funclet)

dpmvc-imgjobcheck 命令用于获取特定 CA Virtual Assurance 作业 ID 或特定 vCenter 任务 ID 的克隆作业状态。为某些 dpmvc 命令提供作业 ID。使用该命令可检查 vCenter 上的状态。

此命令具有以下格式：

```
dpmvc-imgjobcheck  
-status job ID  
[-vc vcenterservername]
```

### **-status *jobID***

指定用于获取作业状态的作业 ID。

### **-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

### 示例：在单 vCenter 服务器环境中，使用作业 ID 获取克隆作业状态

本示例使用 CA Virtual Assurance 作业 ID 获取克隆作业状态。

```
dpmvc-imgjobcheck -status 42
```

## dpmvc-migrate 命令—迁移 VM (Funclet)

dpmvc-migrate 命令用于将 VM 从一台主机服务器迁移到另一台主机服务器。

此命令具有以下格式：

```
dpmvc-migrate  
{-datacenter_name datacentername | -vc vcenterservername}  
-vm_name vm  
-migrate_to_target_host name  
-migrate_to_target_resource_pool_name name  
-migrate_to_compute_resource_name name
```

### **-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

### **-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

### **-vm\_name *vmname***

指定 VM。

### **-migrate\_to\_target\_host *name***

指定正在将 VM 迁移到的目标主机的名称。

### **-migrate\_to\_target\_resource\_pool\_name *name***

指定目标资源池。

**-migrate\_to\_compute\_resource\_name *name***

指定正在将 VM 迁移到的目标群集或 VMware ESX 主机的名称。

**示例：迁移 VM**

本示例将 VM “MyVM1” 从数据中心 VPM/MyCity 迁移到 ServerC-dcavc。

```
dpmvc-migrate -vm_name MyVM1 -datacenter_name VAS/MyCity
-migrate_to_compute_resource_name VASCluster/ClusterA
-migrate_to_target_resource_pool_name Resources/VASTest
-migrate_to_target_host ServerC-dcavc.MyCompany.com
```

**dpmvc-removedatastore 命令—删除数据存储 (Funclet)**

dpmvc-removedatastore 命令用于删除数据存储。

此命令具有以下格式：

```
dpmvc-removedatastore
-datastore_name datastorename
-esx_host_name esxhostname
-vc_server vcservername
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
```

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

## dpmvc-removevmvdisk 命令—删除虚拟磁盘 (Funclet)

dpmvc-removevmvdisk 命令用于从 VM 中删除虚拟磁盘。

此命令具有以下格式：

```
dpmvc-removevmvdisk  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
-disk_device_key diskdevicekey  
-delete_data {yes | no}
```

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-disk\_device\_key *diskdevicekey***

指定磁盘的设备密钥的整数，例如 2000。

**-delete\_data {yes | no}**

指示删除磁盘时是否删除数据。

### 示例：删除虚拟磁盘

本示例从 VM myvm 中删除设备密钥为 2000 的虚拟磁盘。

```
dpmvc-removevmvdisk -vm_name myvm -vc_server myvcserver -disk_device_key 2000  
-delete_data yes
```

## dpmvc-removevmvnic 命令—删除虚拟 NIC (Funclet)

dpmvc-removevmvnic 命令用于从 VM 中删除虚拟 NIC。

此命令具有以下格式：

```
dpmvc removevmvnic  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
-device_key devicekey
```

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。



**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-device\_key *devicekey***

指定网络接口的设备密钥的整数，例如 3000。

**示例：删除 vNIC**

本示例从 VM 中删除 vNIC。

```
dpmvc-removevmnic -vm_name myvm -vc_server myvcserver -device_key 3000
```

**dpmvc-setresources 命令—设置 VM 资源 (Funclet)**

dpmvc-setresources 命令用于调整 CPU 或内存份额限制或保留。

此命令具有以下格式：

```
dpmvc-setresources
-resource {setcpulimit|setcpureserv|setmemlimit|setmemresrv}
-value value
{-datacenter_name datacentername | -vc vcserverservername}
-vm_name vmname
```

**setcpulimit**

定义用于 VM 的 CPU 份额数的限制。

**setcpureserv**

定义专用于 VM 的 CPU 份额数。

**setmemlimit**

定义 VM 内存份额数的限制。

**setmemresrv**

定义专用于 VM 的内存份额数。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcserverservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

**-value value**

指定资源参数的值。

**示例：在单 vCenter 服务器环境中将资源 CPU 限制设置为 1000 个份额**

本示例将虚拟机“MyVM1”的 CPU 份额限制调整为 1000。

```
dpmvc-setresources -resource setcpulimit -value 1000 -vm_name MyVM1  
-datacenter_name VAS/MyCity
```

**dpmvc-setshares 命令—设置 VM 份额 (Funclet)**

dpmvc-setshares 命令用于设置分配给某个 VM 的 CPU 或内存份额的数量。

此命令具有以下格式：

```
dpmvc-setshares  
{-datacenter_name datacentername | -vc vccenterservername}  
-vm_name vmname  
-setshare {cpu_add_prop_value | cpu_subtract_prop_value |  
cpu_overwrite_prop_value | mem_add_prop_value | mem_subtract_prop_value |  
-mem_overwrite_prop_value}  
-value value  
  
-setshare {cpu_add_prop_value | cpu_subtract_prop_value |  
cpu_overwrite_prop_value | mem_add_prop_value |  
mem_subtract_prop_value | mem_overwrite_prop_value}
```

定义要添加、减少或更改的 CPU 或内存份额的数量。份额用于确定与其他正在运行的 VM 相比，分配给哪些 VM 的可用物理资源的比例较高。如果向某个 VM 分配的份额较多，则会向该 VM 分配较多的物理资源。分配给 VM 的份额越少，提供给该 VM 的物理资源就越少。

**cpu\_add\_prop\_value**

定义添加到 VM 的 CPU 的份额数。

**-cpu\_subtract\_prop\_value value**

定义从 VM 减去的 CPU 的份额数。

**-cpu\_overwrite\_prop\_value value**

定义用于 VM 的 CPU 的份额数。

**-mem\_add\_prop\_value value**

定义添加到 VM 的内存的份额数。

**-mem\_subtract\_prop\_value value**

定义从 VM 减去的内存的份额数。

**-mem\_overwrite\_prop\_value value**

定义用于 VM 的 CPU 的份额数。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

**-value *value***

指定资源参数的值。

### 示例：在个 vCenter 服务器环境中增加 VM 的 CPU 份额数

本示例在数据中心“VAS/MyCity”中，将虚拟机“MyVM01”的 CPU 份额数增加 1000。

```
dpmvc-setshares -vm_name MyVM01 -datacenter_name VAS/MyCity -setshare
cpu_add_prop_value -value 1000
```

## dpmvc-snapshot 命令—管理快照 (Funclet)

dpmvc-snapshot 命令用于管理 VM 的快照。使用该命令，可以创建快照、删除一个或所有快照或还原到某个快照。

此命令具有以下格式：

```
dpmvc-snapshot
-create
{-datacenter_name datacentername | -vc vcenterservername}
-vm_name vmname
-vm_snapshot_name vm_snapshotname
[-vm_snapshot_desc description]
[-withmemory {true|false}]
[-poweron]
```

```
dpmvc-snapshot
-remove
{-datacenter_name datacentername / -vc vcenterservername}
-vm_name vmname
[-vm_snapshot_name vm_snapshotname [-withchildren]]
```

```
dpmvc-snapshot
-revert
{-datacenter_name datacentername / -vc vcenterservername}
-vm_name vmname
-vm_snapshot_name vm_snapshotname
```

**-operation {create|remove|revert}**

指定要创建、删除或还原 VM 快照。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-vm\_name *vmname***

指定 VM。

**-vm\_snapshot\_name *vmsnapshotname***

定义 VM 快照的名称。

**-vm\_snapshot\_desc *description***

(可选) 定义 VM 快照的说明。

**-withmemory {true | false}**

(可选) 指定快照包含内存。该参数对于 -remove 或 -revert 无效。

**-withchildren**

(可选) 指定您希望删除快照的所有子项。该参数对于 -create 或 -revert 无效。

**-poweron**

(可选) 指定您希望打开 VM 的电源。该参数对于 -remove 或 -revert 无效。

#### 示例：在单 vCenter 服务器环境中创建 VM 快照

本示例为名为 MyVM01 的 VM 系统创建一个名为 “Hello world” 的快照，并包含说明 “This is my first snapshot”。

```
dpmvc-snapshot -create -vm_snapshot_name "Hello world" -vm_snapshot_desc "First snapshot"
-vm_name MyVM01 -datacenter_name VAS/MyCity
```

#### 示例：在单 vCenter 服务器环境还原到 VM 快照

本示例将 MyVM01 还原到名为 “Hello world” 的快照。

```
dpmvc-snapshot -revert -vm_snapshot_name "Hello world" -vm_name MyVM01
-datacenter_name VAS/MyCity
```

#### 示例：在单 vCenter 服务器环境中删除 VM 快照

本示例从名为 MyVM01 的 VM 系统中删除名为 “Hello world” 的快照。

```
dpmvc-snapshot -remove -vm_snapshot_name "Hello world" -vm_name MyVM01
-datacenter_name VAS/MyCity
```

## dpmvc-templatetovm 命令—将模板转换为 VM (Funclet)

dpmvc-templatetovm 命令用于将虚拟机模板转换回 VM。例如，使用该功能，您可以向转换后的 VM 应用修补程序或软件更新，然后再将其转换回模板。

此命令具有以下格式：

```
dpmvc-templatetovm
-template_name templatename
{-datacenter_name datacentername | -vc vcenterservername}
-compute_resource_name hostname
-resource_pool_name resourcepool
-esx_host_name hostname
```

**-template\_name *templatename***

定义模板的名称。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

**-compute\_resource\_name *name***

指定在其中创建 VM 的群集或 VMware ESX 主机。

**-resource\_pool\_name *resourcepool***

指定您希望从中选择用于克隆的 VM 的资源池的名称。

**-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

### 示例：在单 vCenter 服务器环境中将模板转换为 VM

本示例将名为 convertTest 的模板转换为 VM。

```
dpmvc-templatetovm -datacenter_name VAS/MyCity -template_name convertTest
-compute_resource_name onDemand/myhost.myco.com -esx_host_name myhost.myco.com
-resource_pool_name Resources/QA
```

## dpmvc vapp 命令—管理 vApp

dpmvc vapp 命令支持在 vApp 上执行以下操作：

- 创建新 vApp
- 克隆
- 打开电源
- 关闭电源
- 挂起
- 从 VMware vCenter 中删除 vApp
- 从 VMware vCenter 取消注册
- 将 VM 添加到 vApp 中
- 将资源池添加到 vApp 中
- 将 vApp 添加到 vApp 中
- 更新 vApp 配置

此命令具有以下格式：

```
dpmvc-vapp -create
```

```
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-cpu_allocation isExpandableReservation, limit, reservation, sharesLevel,  
shares]  
[-mem_allocation isExpandableReservation, limit, reservation, sharesLevel,  
shares]
```

```
dpmvc-vapp -clone
```

```
-vc_server vcservername  
-vapp_path vapppath  
[-target_vapp_path targetvapppath]  
[-vapp_name vappname]  
[-target_vapp_host targetvapphost]  
[-target_vapp_datastore targetvappdatastore]
```

```
dpmvc-vapp [-power_on | -power_off | -suspend | -delete | -unregister ]
```

```
-vc_server vcservername  
-vapp_path vapppath
```

```
dpmvc-vapp -add_vms_to_vapp
```

```
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-vms vms]
```

```
dpmvc-vapp -add_rps_to_vapp
```

```
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-rps resourcepool]
```

```
dpmvc-vapp -add_vapps_to_vapp
```

```
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-vapps vapps]  
[-locale iso639value]
```

```
dpmvc-vapp -update_vapp_config
```

```
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-config_settings  
vappName,objectType,startAction,startDelay,startOrder,stopAction,stopDelay,wai  
tForGuest]  
[-cpu_allocation isExpandableReservation, limit, reservation, sharesLevel,  
shares]  
[-mem_allocation isExpandableReservation, limit, reservation, sharesLevel,  
shares]  
[-locale iso639value]
```

#### **-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

#### **-vapp\_path vapppath**

指定 vApp 路径。

#### **-vapp\_name vappname**

（可选）指定 vApp 名称。

**-cpu\_allocation *isExpandableReservation, limit, reservation, sharesLevel, shares***

（可选）指定更新 vApp 配置时的 CPU 分配设置。

**isExpandableReservation**

指定分配是否可扩展。可能的值：true 或 false。

**limit**

指定分配限制。

**reservation**

指定保留分配。

**sharesLevel**

指定份额水平，0 代表低，1 代表正常，2 代表高，3 代表自定义。  
例如 true,-1,1000,1,4000

**shares**

指定分配份额。

**-mem\_allocation *isExpandableReservation, limit, reservation, sharesLevel, shares***

指定更新 vApp 配置时的内存分配设置。

**isExpandableReservation**

指定分配是否可扩展。可能的值：true 或 false。

**limit**

指定分配限制。

**reservation**

指定保留分配。

**sharesLevel**

指定份额水平，0 代表低，1 代表正常，2 代表高，3 代表自定义。  
例如 true,-1,1000,1,4000

**shares**

指定分配份额。

**-vms *vms***

（可选）指定您希望添加到 vApp 的 VM。

**-rps *resourcepool***

（可选）指定您希望添加到 vApp 的资源池的名称。



**-vapps vapps**

(可选) 指定您希望添加到 vApp 的 vApp。

**-config\_settings**

**vappName, objectType, startAction, startDelay, startOrder, stopAction, stopDelay, waitForGuest**

(可选) 指定 vApp 的配置设置。正确的格式为 "

**startAction none /powerOn**

指定启动操作。可用的选项有 none 或 powerOn

**stopAction none /powerOff /guestShutdown /suspend**

指定启动操作。可用的选项有 none、powerOff、guestShutdown 或 suspend。

**-target\_vapp\_path targetvapppath**

指定您希望创建的 vApp 的目标 vApp 路径。

**-target\_vapp\_host targetvapphost**

(可选) 指定新的 vApp 要驻留在其上的主机。

**-target\_vapp\_datastore targetvappdatastore**

(可选) 指定 vApp 要驻留在其上的数据存储的名称。

## dpmvc-virtualswitch 命令—管理虚拟交换机 (Funclet)

dpmvc-virtualswitch 命令用于管理虚拟交换机。

- 创建虚拟交换机
- 更新虚拟交换机的属性
- 删除虚拟交换机
- 为虚拟交换机创建端口组
- 更新虚拟交换机的端口组属性
- 从虚拟交换机中删除端口组
- 重命名虚拟交换机的端口组

此命令具有以下格式：

```
dpmvc-virtualswitch {-vs_add | -vs_update}
-vc_server vcservername
-esx_host_name esxhostname
-switch_name switchname
-nic_names nicname1 [,nicname2, ...]
```

```
dpmvc-virtualswitch -vs_remove  
-vc_server vcservername  
-esx_host_name esxhostname  
-switch_name switchname
```

```
dpmvc-virtualswitch -add_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-switch_name switchname  
-portgroup_name portgroupname  
[-vlan vlanID]
```

```
dpmvc-virtualswitch -update_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-switch_name switchname  
-portgroup_name portgroupname  
[-portgroup_newname portgroupnewname]  
[-vlan vlanID]
```

```
dpmvc-virtualswitch -remove_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-portgroup_name portgroupname  
[-vlan vlanID]
```

```
dpmvc-virtualswitch -rename_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-switch_name switchname  
-portgroup_name portgroupname  
-portgroup_newname portgroupnewname
```

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

**-switch\_name *switchname***

指定操作所针对的交换机名称。

**-nic\_names *nicname1* [*nicname2*, ...]**

指定以逗号分隔的物理 NIC 列表。如果指定多个 NIC，请使用双引号来转义参数，例如："sc1,nfs1,mnic1,mnic2"。

**-portgroup\_name *portgroupname***

指定端口组名称。

**-portgroup\_newname *portgroupnewname***

指定新的端口组名称。

**-vlan *vlanID***

(可选) 指定用于虚拟端口组操作的整数值 (vlan ID)。

### 示例

本示例创建新的虚拟交换机。

```
dpmvc-virtualswitch -vs_add -vc_server vc5master -esx_host_name esx5
-switch_name switch1 -nic_names "sc1,nfs1,mnic1,mnic2"
```

## dpmvc-vmtotemplate 命令—将 VM 转换为模板 (Funclet)

dpmvc-vmtotemplate 命令用于将虚拟机转换为模板。该功能有助于在数据中心的多台服务器之间实现一致性。

此命令具有以下格式：

```
dpmvc-vmtotemplate
-vm_name vmname
{-datacenter_name datacentername | -vc vcenterservername}
```

**-vm\_name *vmname***

指定 VM。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc *vcenterservername***

指定 VM 所在的 vCenter 服务器。

### 示例：在单 vCenter 服务器环境中将 VM 转换为模板

本示例将名为 convertTest 的 VM 转换为模板。

```
dpmvc-vmtotemplate -datacenter_name VAS/MyCity -vm_name convertTest
```

## dpmvc-wait 命令—等待作业的时间 (Funclet)

dpmvc-wait 命令用于指定执行作业之前的等待时间。在作业完成或发生超时之前，该命令不返回状态。

此命令具有以下格式：

```
dpmvc-wait
-job <jobid>
[-timeout timeout]
```

**-job *jobid***

指定作业 ID。

**-timeout *timeout***

指定用于指定作业状态的超时值。

**示例：作业状态**

本示例指定在作业 125635 执行之前等待时间为 20 秒。

```
dpmvc-wait -job 125635 -timeout 20
```

## CA Microsoft Hyper-V AutoShell 命令

可以使用 AutoShell 编写 CA Hyper-V AutoShell 命令的脚本并使其自动化，然后根据命令的结果运行操作。CLI 中也提供了相应的命令。

**注意：***vmname* 选项在 Hyper-V 服务器环境中不是唯一的。为了方便起见，可以通过 *vmname* 指定 VM。但是，如果 *vmname* 不唯一，命令将失败。在这种情况下，必须改用 *vmid* 选项。不能在 AutoShell 命令中使用以下参数对：*vmname*、*vmid* 和 *template*、*templateid*。例如，如果使用 *vmname*，则不能使用 *vmid*。

### dpmhv-addVMNic (cmdlet) 命令-将网络接口控制器添加到 VM 中

dpmhv-addVMNic 命令用于将网络适配器添加到 VM 中，并提供将它连接到虚拟交换机的选项。

此命令具有以下格式：

```
dpmhv-addVMNic  
-host hostname  
-vm vmname|-vmid vmguid  
[-mac_address macaddress]  
[-switch switchname]  
[-legacy]  
[-retval]  
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-vm *vmname***

指定 VM 的名称。

**-vmid *vmguid***

指定 VM 的唯一 ID。

**-mac\_address macaddress**

（可选）定义新网络适配器的 MAC 地址。有效条目：12 位十六进制值（例如，A-F、0-9、a-f），格式如下：0013724C2140 或 00:13:72:4C:21:40。

**注意：**如果没有指定 macaddress，Hyper-V 将自动生成该地址。

**-switch switchname**

指定连接 VM 的网络适配器的虚拟网络交换机名称。如果不指定该选项，VM 的网络适配器将不会连接到虚拟网络。

**默认：**无

**-legacy**

（可选）创建传统网络适配器，而非 Hyper-V 合成适配器。

**注意：**在安装没有 Hyper-V 集成服务的操作系统时使用该选项。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：将网络接口控制器添加到 VM 中**

本示例将 NIC 添加到 VM “TestVM” 中。

```
dpmhv-addVMNic
-host hvserver
-vm TestVM
-switch "Local Area Connection
- Virtual Network"
```

**dpmhv-addVMSCSIController (cmdlet) 命令-将 SCSI 控制器添加到 VM 中**

dpmhv-addVMSCSIController” 命令用于向 VM 中添加最多 4 个 SCSI 控制器。最多可以向 SCSI 控制器中添加 64 个磁盘。

此命令具有以下格式：

```
dpmhv-addVMSCSIController
-host hostname
-vm vmname|-vmid vmguid
[-retval]
[-silent]
```

**-host hostname**

指定 Hyper-V 服务器主机的名称。

**-vm *vmname***

指定 VM 的名称。

**-vmid *vmguid***

指定 VM 的唯一 ID。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：将 SCSI 控制器添加到 VM 中**

本示例将 SCSI 控制器添加到 VM “TestVM” 中。

```
dpmhv-addVMSCSIController -host hvserver -vm TestVM
```

## **dpmhv-changeVMState (cmdlet) 命令-更改 VM 的状态**

dpmhv-changeVMState 命令用于更改 VM 的状态。状态包括：start、suspend、stop、pause、shutdown 和 reboot。

此命令具有以下格式：

```
dpmhv-changeVMState  
-host hostname  
-vm vmname|-vmid vmguid  
-state {start,suspend,stop,pause,shutdown,reboot}  
[-retval]  
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-vm *vmname***

指定 VM 的名称。

**-vmid *vmguid***

指定 VM 的唯一 ID。

**-state {start,suspend,stop,pause,shutdown,reboot}**

指定 VM 的状态。状态选项如下所示：

**start**

启动或恢复 VM。

**挂起**

暂时挂起 VM。

**stop**

停止 VM。

**pause**

暂时停止 VM。

**shutdown**

关闭 VM。

**reboot**

重新启动 VM。

**注意：**

如果指定无效的状态更改，系统将报告异常。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：更改 VM 的状态**

本示例启动 VM “TestVM”。

```
dpmhv-changeVMState -host hvserver -vm TestVM -state start
```

## dpmhv-createTemplateFromVM (Funclet) 命令-从 VM 创建模板

dpmhv-createTemplateFromVM 命令用于使用现有 VM 创建模板。可以使用模板创建多个 VM。

模板将复制 VM 规格和附加到 VM 的虚拟硬盘。如果使用附带快照的 VM 创建模板，将仅导出当前状态（不包括快照）。该命令可为每个 Hyper-V 服务器创建模板目录。该模板目录已进行维护，并可从以下位置进行查询：

询： %ALLUSERSPROFILE%\ca\vpvm\Hyper-V\_Templates\TemplateCatalog.xml

此命令具有以下格式：

```
dpmhv-createTemplateFromVM
-host hostname
-vm vmname|-vmid vmguid
-template templatename
[-async]
[-path pathname]
[-description description]
[-retval]
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-template *templatename***

指定在模板目录中创建的通用模板的名称。

**注意：**模板名称在本地模板目录中必须是唯一的。

### **-async**

（可选）指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

### **-dest *pathname***

（可选）指定要创建的 VM 的路径（模板已存储）。如果没有指定该选项，将在 Hyper-V 服务器的默认位置创建 VM。创建的模板名称存储在以下位置： %ALLUSERSPROFILE%\ca\vpvm\Hyper-V\_Templates。

**注意：**建议不要在 Hyper-V 默认位置中创建 VM。使用模板创建 VM 时，它与未更改的虚拟磁盘映像文件名产生冲突。



**-description *description***

（可选）指定模板说明。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**返回值**

同步执行 `dpmhv-createTemplateFromVM` 命令时，它将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给 `dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。如果在执行期间发生错误，则该命令将报告异常。

**示例：从 VM 创建模板**

本示例从 VM “TestVM” 创建模板 “MyTemplate”。

```
dpmhv-createTemplateFromVM -vm TestVM -template MyTemplate -host hvserver
```

**dpmhv-createVirtDisk (Funclet) 命令-创建虚拟磁盘**

`dpmhv-createVirtDisk` 命令用于创建虚拟磁盘。

此命令具有以下格式：

```
dpmhv-createVirtDisk  
-host hostname  
-path pathname  
-type{fixed,dynamic}  
-sizeGB size  
[-async]  
[-retval]  
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-path *pathname***

指定要创建的虚拟磁盘的路径。

**-type {fixed, dynamic}**

指定要创建的虚拟磁盘的类型。选项包括：

**fixed**

指定虚拟磁盘类型为固定。

**dynamic**

指定虚拟磁盘类型为动态。

**-sizeGB size**

指定虚拟磁盘的大小 (GB)。

**-async**

(可选) 指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：创建虚拟磁盘**

本示例创建大小为 1 GB 的虚拟磁盘。

```
dpmhv-createVirtDisk -host hvserver -path C:\data\Disks\01.vhd -type fixed  
-sizeGB 1
```

## **dpmhv-createVirtFloppy (Cmdlet) 命令-创建虚拟软盘**

**dpmhv-createVirtFloppy** 命令用于创建虚拟软盘映像文件。使用 **dpmhv-setVMVirtFloppy** 命令可将虚拟软盘附加到 VM。

此命令具有以下格式：

```
dpmhv-createVirtFloppy  
-host hostname  
-path pathname[-retval]  
[-silent]
```

**-host hostname**

指定 Hyper-V 服务器主机的名称。

**-path pathname**

指定要创建的虚拟软盘映像文件的路径。如果不指定，将在 Hyper-V 虚拟硬盘的默认目录中创建映像文件。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：创建虚拟软盘**

本示例创建虚拟软盘映像文件“fdd.vfd”。

```
dpmhv-createVirtFloppy -host hvserver -path C:\VMDisks\fdd.vfd
```

**dpmhv-createVM (Funclet) 命令-创建 VM**

dpmhv-createVM 命令用于创建 VM。该命令创建没有系统组件（如网络适配器或硬盘）的 VM。创建 VM 后，可以使用相应的命令创建系统组件。

此命令具有以下格式：

```
dpmhv-createVM
-host hostname
-name name
[-path pathname]
[-template templatename]
[-templateid vmguid]
[-mem mem]
[-cpus cpus]
[-cpuidlimit limit]
[-cpureserve reserve]
[-cpulimit limit]
[-cpuweight weight]
[-startAction {none|auto|always}]
[-startDelay delay]
[-stopAction {save,off,shutdown}]
[-recoveryAction {none,restart,revert}]
[-retval]
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-name *name***

（可选）指定 VM 的名称。

**注意：**VM 的名称必须是唯一的。

**-dest *pathname***

(可选) 指定要创建的 VM 的路径 (模板已存储)。如果没有指定该选项, 将在 Hyper-V 服务器的默认位置创建 VM。创建的模板名称存储在以下位置: %ALLUSERSPROFILE%\ca\vpms\Hyper-V\_Templates。

**注意:** 建议不要在 Hyper-V 默认位置中创建 VM。使用模板创建 VM 时, 它与未更改的虚拟磁盘映像文件名产生冲突。

**-template *templatename***

(可选) 指定现有 VM 的名称。该选项用于复制现有 VM 的配置设置以创建新的 VM。建议不要从现有 VM 创建 VM。两个 VM 共享资源 (如虚拟磁盘), 并且无法同时运行。

**注意:** 模板名称在本地模板目录中必须是唯一的。

**-templateid *vmguid***

(可选) 指定现有 VM 的唯一 ID (GUID)。该选项用于复制现有 VM 的配置设置以创建新的 VM。建议不要从现有 VM 创建 VM。两个 VM 可以共享资源 (如虚拟磁盘), 但不能同时运行。

**默认值:** -1

**注意:** 该选项不表示 VM 目录中的模板 ID, 但表示 Hyper-V 环境中的现有 VM。

**-mem *mem***

(可选) 指定要创建的 VM 的 RAM 内存 (MB)。

**默认值:** -1

**-cpus *cpus***

(可选) 指定要分配给 VM 的 CPU 核心数。

**-cpuidlimit *limit***

(可选) 指定 VM 的 CPU ID 功能的限制。该选项可改进与传统操作系统 (如 Windows NT) 的兼容性。

**默认值:** -1

**-cpureserve *reserve***

(可选) 指定要为 VM 保留的 CPU 的百分比。如果没有指定该选项, Hyper-V 服务器会根据系统整体使用情况分配 CPU 周期。

**默认值:** -1

**-cpulimit *limit***

(可选) 指定 VM 使用的 CPU 资源数的限制。该选项可改进在具有不同 CPU 功能的物理主机之间移动 VM 时的兼容性。

**默认值:** -1

**-cpuweight *weight***

（可选）指定虚拟机的相对权重。该选项控制多个 VM 运行时的资源分配。有效条目：1 到 10000 的整数。

**默认值：** -1

**-startAction {none,auto,always}**

（可选）指定 Hyper-V 主机启动时要在 VM 上执行的操作。选项包括：  
无

不执行任何操作。

**自动**

自动启动 VM。

**注意：** 如果 Hyper-V 主机关闭之前 VM 正在运行，则使用该选项。

**always**

始终启动 VM。

**-startDelay *delay***

（可选）指定 Hyper-V 主机启动并运行后启动 VM 的延迟（秒）。

**默认值：** -1

**-stopAction {save,off,shutdown}**

（可选）指定物理 Hyper-V 关闭时要在 VM 上执行的操作。选项包括：

**save**

挂起 VM。

**关**

关闭 VM 电源。

**shutdown**

关闭系统。

**-recoveryAction {none,restart,revert}**

（可选）指定因意外关闭而重新启动 Hyper-V 主机后要在 VM 上执行的操作。选项包括：

**无**

不执行任何操作。

**restart**

重新启动 VM。

**revert**

还原到 VM 的最后一个快照。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**返回值**

该命令将返回包含新建 VM 的唯一标识符 (GUID) 的字符串。

**示例：使用默认设置创建 VM**

本示例使用默认设置创建 VM “TestVM”。

```
dpmhv-createVM -host hvserver -name TestVM
```

**示例：在用户定义的目录中使用默认设置创建 VM**

本示例在用户定义的目录中使用默认设置创建 VM “TestVM”。

```
dpmhv-createVM -host hvserver -name TestVM -path c:\VMs\TestVM
```

**示例：使用自定义的内存和 CPU 核心值创建 VM**

本示例使用自定义的内存和 CPU 核心值创建 VM “TestVM”。

```
dpmhv-createVM -host hvserver -name TestVM -mem 1024 -cpus 2
```

## dpmhv-createVMFromTemplate (Funclet) 命令-使用模板创建 VM

dpmhv-createVMFromTemplate 命令用于从先前创建的 Microsoft System Center Virtual Machine Manager (SCVMM) 模板创建 VM，并选择性地自定义 Windows 映像的设置。

该模板保留未使用，以便单个模板可用于创建任意数量的虚拟机。该命令仅支持 (SCVMM) 托管模板。有关没有 SCVMM 的本地模板支持的信息，请参阅 dpmhv-createVMFromTemplateEx。

此命令具有以下格式：

```
dpmhv-createVMFromTemplate
-host hostname
-vm vmname
-template templatename
-dest pathname
-scvmmHost scvmmHostname
-adminPass adminPW
[-computerName computerName]
[-ip4addr ip4addr]
[-ip4dhcp ip4dhcpInt]
[-ip4mask ip4mask]
[-ip4gw ip4gw]
[-ip4metric ip4metric]
[-ip4dns ip4dns]
[-productKey key]
[-userName UserName]
[-organization orgname]
[-domain domainname]
[-domainAdmin domad]
[-domainAdminPass dompw]
[-workgroup wgname]
[-adminUser adminUser]
[-adminUserPass adminUserPW]
[-mem mem]
[-cpus cpus]
[-hardwareProf hwprof]
[-guestOSProf osprof]
[-startVM]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-vm *vmname***

指定 VM 的名称。

### **-template *templatename***

指定要用于创建 VM 的源模板的名称。

**注意：**模板名称在本地模板目录中必须是唯一的。

**-dest *pathname***

(可选) 指定要创建的 VM 的路径(模板已存储)。如果没有指定该选项, 将在 Hyper-V 服务器的默认位置创建 VM。创建的模板名称存储在以下位置: %ALLUSERSPROFILE%\ca\vpn\Hyper-V\_Templates。

**注意:** 建议不要在 Hyper-V 默认位置中创建 VM。使用模板创建 VM 时, 它与未更改的虚拟磁盘映像文件名产生冲突。

**-scvmmHost *scvmmHostname***

(可选) 指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器的主机名。使用 SCVMM 集成开通 VM 时, 该参数有效。

**-adminPass *adminPW***

(可选) 该选项用于设置 VM 的默认管理员帐户密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。在异步执行中将忽略该参数。

**注意:** 要成功设置该选项, 请将使用 Sysprep 工具创建的 Windows Server 管理员密码设置为空。

**-computerName *computerName***

(可选) 指定 VM 的计算机名称。对该参数的支持需要使用 Sysprep 工具的映像。该选项对于异步执行命令无效。

**-ip4addr *ip4addr***

(可选) 指定要分配给 VM 的接口的静态 IPv4 地址。要设置专用接口的 IP 地址, 请使用已知的接口名称作为 IP 地址的前缀并以 “#” 分隔。例如,

-ip4addr "Local Area Connection#192.168.1.200"。如果模板映像具有多个网络适配器, 则 IP 地址将分配给第一个接口。该选项对于异步执行命令无效。

**-ip4dhcp *ip4dhcpInt***

(可选) 指定一个选项来打开 VM 的特定接口的 DHCP。您也可为该选项指定接口名称。例如,

-ip4dhcp "Local Area Connection"。如果模板映像具有多个网络适配器, 则为第一个接口打开 DHCP。该选项对于异步执行命令无效。

**默认值:** local

**-ip4mask *ip4mask***

(可选) 指定要为 VM 分配的子网掩码。该选项与 -ip4addr 选项结合使用。如果在

-ip4addr 选项中指定了接口名称, 则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。



**-ip4gw *ip4gw***

（可选）指定该选项来为 VM 设置网关。该选项与 **-ip4addr** 选项结合使用。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**注意：**如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。

**-ip4metric *ip4metric***

（可选）指定要为 VM 设置的接口度量标准。该选项与 **-ip4addr** 选项结合使用。如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认：** 1

**-ip4dns *ip4dns***

（可选）指定要为 VM 设置的 DNS 服务器。该选项与 **-ip4addr** 选项结合使用。如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-adminPass *adminPW***

（可选）该选项用于设置 VM 的默认管理员帐户密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。在异步执行中将忽略该参数。

**注意：**要成功设置该选项，请将使用 Sysprep 工具创建的 Windows Server 2003 管理员密码设置为空。

**-productKey *key***

（可选）指定 VM 的 Windows 产品激活密钥。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-userName *UserName***

（可选）指定在新 VM 上安装的 Windows 映像（使用 Sysprep 工具创建）的用户名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-organization *orgname***

（可选）指定在新 VM 上安装的 Windows 映像（使用 Sysprep 工具创建）的组织名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-domain *domainname***

（可选）指定 VM 的域名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。域和工作组规格互不相容。

**-domainAdmin *domad***

（可选）将用户帐户指定为要在域中创建的域管理员。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-domainAdminPass *dompw***

（可选）指定要在域中创建的域管理员用户帐户的密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-workgroup *wgname***

（可选）指定要为 VM 创建的工作组。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。域和工作组规格互不相容。该选项对于异步执行命令无效。

**-adminUser *adminUser***

（可选）指定要作为默认 Administrators 组的一部分创建的管理员用户名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-adminUserPass *adminUserPW***

（可选）指定要作为默认 Administrators 组的一部分创建的管理员用户帐户的密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-mem *mem***

（可选）指定要创建的 VM 的 RAM 内存 (MB)。

**默认值：** -1

**-cpus *cpus***

（可选）指定要分配给 VM 的 CPU 核心数。

**默认值：** -1

**-scvmmHost *scvmmHostname***

（可选）指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器的主机名。使用 SCVMM 集成开通 VM 时，该参数有效。

**-hardwareProf hwprof**

(可选) 指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器定义的硬件配置文件的名称。该参数覆盖存储在 SCVMM 模板中的硬件配置设置。使用 SCVMM 集成开通 VM 时, 该参数有效。

**-guestOSProf osprof**

(可选) 指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器定义的来宾操作系统配置文件的名称。该参数覆盖存储在 SCVMM 库服务器中的操作系统配置设置。使用 SCVMM 集成开通 VM 时, 该参数有效。

**-startVM**

(可选) 指定该选项可在创建 VM 后自动启动它。默认情况下, 新 VM 处于关闭状态。您必须自定义用于启动 VM 的 IP 配置选项。

**返回值**

同步执行 `dpmhv-createVMFromTemplate` 命令时, 它将返回空字符串。如果异步执行, 会将代表作业 ID 的字符串传递给 `dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。如果在执行期间发生错误, 则该命令将报告异常。

**示例: 从 SCVMM 模板创建 VM**

本示例使您可以使用指定硬件和来宾 OS 配置文件的选项从 SCVMM 模板创建 VM:

```
dpmhv-createVMFromTemplate -host hvserver -vm NewVM -dest c:\VMs\NewBox -template Win2k3Clean  
-scvmmHost SCVMMLibSrv -hardwareProf hw_def -guestOSProf os_def -async -adminPass  
#admin#
```

**另请参阅:**

[dpmhv-createVMFromTemplateEx \(Funclet\) 命令-使用模板创建 VM](#) (p. 204)

## dpmhv-createVMFromTemplateEx (Funclet) 命令-使用模板创建 VM

dpmhv-createVMFromTemplate 命令用于使用现有模板创建 VM。如果原始模板是在 Sysprep 状态中创建，则可自定义使用命令参数创建的 VM（映像）的 (Windows) 设置。创建 VM 后，可以使用相应的命令创建系统组件。可使用一个模板创建多个虚拟机。

**注意：**在同步运行命令时，可以自定义 Windows 设置映像。对于异步执行，在执行完成后，您可以使用 dpmhv-setSysprepProperties 命令自定义映像。

此命令具有以下格式：

```
dpmhv-createVMFromTemplateEx
-host hostname
-vm vmname
-template templatename
-dest pathname
[-computerName computerName]
[-ip4addr ip4addr]
[-ip4dhcp ip4dhcpInt]
[-ip4mask ip4mask]
[-ip4gw ip4gw]
[-ip4metric ip4metric]
[-ip4dns ip4dns]
[-disableAdmin]
[-adminPass adminPW]
[-autoLogon alcnt]
[-duplicatorString dupstr]
[-timeZone timezone]
[-productKey key]
[-userName UserName]
[-organization orgname]
[-domain domainname]
[-domainAdmin domad]
[-domainAdminPass dompw]
[-workgroup wgname]
[-adminUser adminUser]
[-adminUserPass adminUserPW]
[-custom custom]
[-async]
[-mem mem]
[-cpus cpus]
[-cpuidlimit limit]
[-cpufeatlimit limit]
[-cpureserve reserve]
[-cpulimit limit]
[-cpuweight weight]
[-startAction {none,auto,always}]
[-startDelay delay]
[-stopAction {save,off,shutdown}]
[-recoveryAction {none,restart,revert}]
[-scvmmHost scvmmHostname]
[-hardwareProf hwprof]
[-guestOSProf osprof]
[-startVM]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-vm *vmname***

指定 VM 的名称。

**-template *templatename***

指定要用于创建 VM 的源模板的名称。

**注意：**模板名称在本地模板目录中必须是唯一的。

**-dest *pathname***

(可选)指定要创建的 VM 的路径(模板已存储)。如果没有指定该选项,将在 Hyper-V 服务器的默认位置创建 VM。创建的模板名称存储在以下位置: %ALLUSERSPROFILE%\ca\vpvm\Hyper-V\_Templates。

**注意：**建议不要在 Hyper-V 默认位置中创建 VM。使用模板创建 VM 时,它与未更改的虚拟磁盘映像文件名产生冲突。

**-computerName *computerName***

(可选)指定 VM 的计算机名称。对该参数的支持需要使用 Sysprep 工具的映像。该选项对于异步执行命令无效。

**-ip4addr *ip4addr***

(可选)指定要分配给 VM 的接口的静态 IPv4 地址。要设置专用接口的 IP 地址,请使用已知的接口名称作为 IP 地址的前缀并以“#”分隔。例如,

-ip4addr "Local Area Connection#192.168.1.200"。如果模板映像具有多个网络适配器,则 IP 地址将分配给第一个接口。该选项对于异步执行命令无效。

**-ip4dhcp *ip4dhcpInt***

(可选)指定一个选项来打开 VM 的特定接口的 DHCP。您也可为该选项指定接口名称。例如,

-ip4dhcp "Local Area Connection"。如果模板映像具有多个网络适配器,则为第一个接口打开 DHCP。该选项对于异步执行命令无效。

**默认值:** local

**-ip4mask *ip4mask***

(可选)指定要为 VM 分配的子网掩码。该选项与

-ip4addr 选项结合使用。如果在

-ip4addr 选项中指定了接口名称,则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-ip4gw *ip4gw***

(可选)指定该选项来为 VM 设置网关。该选项与 -ip4addr 选项结合使用。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**注意：**如果在

-ip4addr 选项中指定了接口名称,则必须在该选项中使用相同的接口名称。

**-ip4metric *ip4metric***

（可选）指定要为 VM 设置的接口度量标准。该选项与 **-ip4addr** 选项结合使用。如果在 **-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认：1**

**-ip4dns *ip4dns***

（可选）指定要为 VM 设置的 DNS 服务器。该选项与 **-ip4addr** 选项结合使用。如果在 **-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-disableAdmin**

（可选）指定该选项可禁用 VM 的默认管理员帐户。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-adminPass *adminPW***

（可选）该选项用于设置 VM 的默认管理员帐户密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。在异步执行中将忽略该参数。

**注意：**要成功设置该选项，请将使用 Sysprep 工具创建的 Windows Server 管理员密码设置为空。

**-autoLogon *alcnt***

（可选）指定一个选项来设置要创建以使用默认管理员帐户自动登录的帐户数目。帐户会在 Sysprep 进程完成之后创建。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：**

-1

**-duplicatorString *dupstr***

（可选）指定要在 VM 注册表中设置的系统复印机的名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-timeZone *timezone***

（可选）指定使用模板创建的 VM 所用的时区。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:**

-1

**-productKey *key***

（可选）指定 VM 的 Windows 产品激活密钥。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-userName *UserName***

（可选）指定在新 VM 上安装的 Windows 映像（使用 Sysprep 工具创建）的用户名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-organization *orgname***

（可选）指定在新 VM 上安装的 Windows 映像（使用 Sysprep 工具创建）的组织名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-domain *domainname***

（可选）指定 VM 的域名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。域和工作组规格互不相容。

**-domainAdmin *domad***

（可选）将用户帐户指定为要在域中创建的域管理员。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-domainAdminPass *dompw***

（可选）指定要在域中创建的域管理员用户帐户的密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-workgroup *wgname***

（可选）指定要为 VM 创建的工作组。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。域和工作组规格互不相容。该选项对于异步执行命令无效。



**-adminUser *adminUser***

（可选）指定要作为默认 Administrators 组的一部分创建的管理员用户名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-adminUserPass *adminUserPW***

（可选）指定要作为默认 Administrators 组的一部分创建的管理员用户帐户的密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-custom *custom***

（可选）指定要在 Sysprep 进程结束时执行的自定义命令的逗号分隔列表。该参数需要 Sysprep 的 Windows 映像。在异步执行中将忽略该参数。

**-async**

（可选）指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

**-mem *mem***

（可选）指定要创建的 VM 的 RAM 内存 (MB)。

**默认值：** -1

**-cpus *cpus***

（可选）指定要分配给 VM 的 CPU 核心数。

**默认值：** -1

**-cpuidlimit *limit***

（可选）指定 VM 的 CPU ID 功能的限制。该选项可改进与传统操作系统（如 Windows NT）的兼容性。

**默认值：** -1

**-cpufeatlimit *limit***

（可选）指定 VM 使用的 CPU 资源的功能的限制。该选项可改进在具有不同 CPU 功能的物理主机之间移动 VM 时的兼容性。

**默认值：** -1

**-cpureserve *reserve***

（可选）指定要为 VM 保留的 CPU 的百分比。如果没有指定该选项，Hyper-V 服务器会根据系统整体使用情况分配 CPU 周期。

**默认值：** -1

**-cpulimit *limit***

（可选）指定 VM 使用的 CPU 资源数的限制。该选项可改进在具有不同 CPU 功能的物理主机之间移动 VM 时的兼容性。

**默认值：** -1

**-cpuweight *weight***

（可选）指定虚拟机的相对权重。该选项控制多个 VM 运行时的资源分配。有效条目：1 到 10000 的整数。

**默认值：** -1

**-startAction {none,auto,always}**

（可选）指定 Hyper-V 主机启动时要在 VM 上执行的操作。选项包括：

**无**

不执行任何操作。

**自动**

自动启动 VM。

**注意：** 如果 Hyper-V 主机关闭之前 VM 正在运行，则使用该选项。

**always**

始终启动 VM。

**-startDelay *delay***

（可选）指定 Hyper-V 主机启动并运行后启动 VM 的延迟（秒）。

**默认值：** -1

**-stopAction {save,off,shutdown}**

（可选）指定物理 Hyper-V 关闭时要在 VM 上执行的操作。选项包括：

**save**

挂起 VM。

**关**

关闭 VM 电源。

**shutdown**

关闭系统。

**-recoveryAction {none,restart,revert}**

（可选）指定因意外关闭而重新启动 Hyper-V 主机后要在 VM 上执行的操作。选项包括：

**无**

不执行任何操作。

**restart**

重新启动 VM。

**revert**

还原到 VM 的最后一个快照。

**-scvmmHost *scvmmHostname***

（可选）指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器的主机名。使用 SCVMM 集成开通 VM 时，该参数有效。

**-hardwareProf *hwprof***

（可选）指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器定义的硬件配置文件的名称。该参数覆盖存储在 SCVMM 模板中的硬件配置设置。使用 SCVMM 集成开通 VM 时，该参数有效。

**-guestOSProf *osprof***

（可选）指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器定义的来宾操作系统配置文件的名称。该参数覆盖存储在 SCVMM 库服务器中的操作系统配置设置。使用 SCVMM 集成开通 VM 时，该参数有效。

**-startVM**

（可选）指定该选项可在创建 VM 后自动启动它。默认情况下，新 VM 处于关闭状态。您必须自定义用于启动 VM 的 IP 配置选项。

**返回值**

同步执行 `dpmhv-createVMFromTemplateEX` 命令时，它将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给

`dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。如果在执行期间发生错误，则该命令将报告异常。

### 示例：从模板创建 VM

本示例使用模板设置作为完整的静态 IPv4 配置创建 VM “TestVM”：

```
dpmhv-createVMFromTemplateEx -host hvserver -vm NewVM -dest c:\VMs\NewBox  
-template Win2k3SysPrepped  
-ip4addr 192.168.1.25 -ip4mask 255.255.255.0 -ip4gw 192.168.1.1 -ip4dns  
192.168.1.1  
-computerName NewBox
```

另请参阅：

[dpmhv-createVMFromTemplate \(Funclet\) 命令-使用模板创建 VM](#) (p. 199)

## dpmhv-destroyTemplate (Cmdlet) 命令-删除模板

dpmhv-destroyTemplate 命令用于从模板目录中删除现有模板。

此命令具有以下格式：

```
dpmhv-destroyTemplate  
-template templatename  
-host hostname
```

### **-template *templatename***

指定在模板目录中创建的通用模板的名称。

**注意：**模板名称在本地模板目录中必须是唯一的。

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### 示例：删除模板

本示例从模板目录删除模板。

```
dpmhv-destroyTemplate -host hvserver -template MyTemplate
```

## dpmhv-destroyVM (Cmdlet) 命令-删除 VM

dpmhv-destroyVM 命令用于删除现有的 VM 规格、虚拟磁盘、软盘映像以及其相关快照。

此命令具有以下格式：

```
dpmhv-destroyVM  
-host hostname  
-vm vmname [-vmid vmguid]  
[-delvhd]  
[-delvfd]  
[-deliso]  
[-retval]  
[-silent]
```

### **-host hostname**

指定 Hyper-V 服务器主机的名称。

### **-vm vmname**

指定 VM 的名称。

### **-vmid vmguid**

指定 VM 的唯一 ID。

### **-delvhd**

（可选）删除分配给 VM 的虚拟硬盘。

### **-delvfd**

（可选）删除分配给 VM 的虚拟软盘。

### **-deliso**

（可选）删除分配给 VM 的 ISO 映像。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

### 示例：删除 VM

本示例删除附有虚拟硬盘和软盘的 VM。

```
dpmhv-destroyVM -host hvserver -vm TestVM -delvhd -delvfd
```

## dpmhv-expandVirtDisk (Funclet) 命令-扩展虚拟磁盘

dpmhv-expandVirtDisk 命令用于增加虚拟磁盘大小。要使用扩展的虚拟磁盘空间，您必须在虚拟机上运行的来宾操作系统中对虚拟磁盘进行分区。

此命令具有以下格式：

```
dpmhv-expandVirtDisk
-host hostname
-path pathname
-sizeGB size
[-async]
[-retval]
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-path *pathname***

指定要扩展的虚拟磁盘的路径名。

### **-sizeGB *size***

指定虚拟磁盘的大小 (GB)。

### **-async**

(可选) 指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

## 返回值

同步执行 dpmhv-expandVirtDisk 命令时，它将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给 *dpmhv-getJobErrorInfo*、*dpmhv-getJobInfo* 或 *dpmhv-getJobStatus*。在同步模式下，如果在执行期间发生错误，命令将返回错误代码并报告错误。

## 示例：扩展虚拟磁盘

本示例将虚拟磁盘的大小扩展为 4 GB。

```
dpmhv-expandVirtDisk -host hvserver -path C:\VMDisks\hdd.vhd -sizeGB 4
```

## dpmhv-exportVM (Funclet) 命令-导出 VM

dpmhv-exportVM 命令用于导出 VM 规格，包括其相关的虚拟磁盘和软盘。该命令已由 dpmhv-exportVMEx 代替。

此命令具有以下格式：

```
dpmhv-exportVM
-host hostname
-vm vmname [/vmid vmguid]
-dest dest
[-nostate]
[-async]
[-retval]
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-dest *dest***

指定要导出的 VM 的目标路径。

### **-nostate**

(可选)指定仅导出 VM 规格。该选项防止导出的 VM 引用原始的 VM 磁盘映像。

### **-async**

(可选)指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

## 返回值

同步执行 dpmhv-exportVM 命令时，它将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给 *dpmhv-getJobErrorInfo*、*dpmhv-getJobInfo* 或 *dpmhv-getJobStatus*。在同步模式下，如果在执行期间发生错误，命令将返回错误代码并报告错误。

### 示例：导出 VM

本示例导出 VM 规格，包括其相关的虚拟磁盘和软盘。

```
dpmhv-exportVM -host hvserver -vm VM_00000001 -dest c:\DATA\Export\
```

## dpmhv-exportVMEx (Funclet) 命令-导出 VM

dpmhv-exportVMEx 命令用于导出 VM 规格，包括其相关虚拟磁盘、软盘和快照。

此命令具有以下格式：

```
dpmhv-exportVMEx  
-host hostname  
-vm vmname|-vmid vmguid  
-dest dest  
[-noruntimeInfo]  
[-nosnapshots]  
[-nostorage]  
[-nosubdir]  
[-snapshot ss]  
[-snapshotid ssguid]  
[-async]  
[-retval]  
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-dest *dest***

指定要导出的 VM 的目标路径。

### **-noruntimeInfo**

（可选）指定该选项将不导出 VM 的运行时信息。

### **-nosnapshots**

（可选）指定该选项将不导出 VM 的快照。该选项仅导出 VM 的当前状态。

### **-nostorage**

（可选）指定该选项将不导出 VM 的虚拟磁盘。

**注意：**在您导出 VM 的快照时，不要使用该选项。



**-nosubdir**

(可选)指定该选项将不在指定为导出路径的目录下创建包含 VM 显示名称的子目录。

**-snapshot ss**

(可选)指定 VM 快照的显示名称。导出的 VM 处于类似于源 VM 的状态(快照)。导出的 VM 不包括快照。

**注意:** 使用

-ssid 选项, 因为快照显示名称在 Hyper-V 环境中不是唯一的。

**-snapshotid ssguid**

(可选)指定 VM 快照的唯一 ID (GUID)。导出的 VM 处于类似于源 VM 的状态(快照)。导出的 VM 不包括快照。

**-async**

(可选)指定要异步执行命令的选项。默认情况下, 同步执行该命令。

**注意:** 如果执行该选项之前此命令完成执行, 它将返回用于查看操作状态的作业 ID。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**返回值**

同步执行 `dpmhv-exportVMEx` 命令时, 它将返回空字符串。如果异步执行, 会将代表作业 ID 的字符串传递给 `dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。在同步模式下, 如果在执行期间发生错误, 命令将返回错误代码并报告错误。

**示例: 导出没有快照的 VM**

本示例导出没有快照的 VM。

```
dpmhv-exportVMEx -host hvserver -vm TestVM -dest c:\Ex\TestVM1 -nosubdir  
-nosnapshots
```

## dpmhv-getHostSwitches (Funclet) 命令-显示 Hyper-V 主机上的主机开关

dpmhv-getHostSwitches 命令用于显示在 Hyper-V 主机上定义的主机开关（虚拟网络）。

此命令具有以下格式：

```
dpmhv-getHostSwitches  
-host hostname  
[-retval]  
[-silent]  
[-detail]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

### **-detail**

（可选）显示虚拟网络（主机网络）的详细信息。

## 返回值

`dpmhv-getHostSwitches` 命令将返回一组对象，其中每个对象代表指定主机上的一个虚拟开关（虚拟主机）。

以常规模式返回下列属性：

- BytesReceivedPersec
- BytesSentPersec
- 标题
- ElementName
- 名称
- StatusDescriptions

以详细模式返回下列属性：

- BroadcastPacketsReceivedPersec
- BroadcastPacketsSentPersec
- BytesPersec
- BytesReceivedPersec
- BytesSentPersec
- 标题
- CreationClassName
- Dedicated
- 描述
- DirectedPacketsReceivedPersec
- DirectedPacketsSentPersec
- ElementName
- EnabledDefault
- EnabledState
- HealthState
- IdentifyingDescriptions
- InstallDate
- LearnedMacAddresses
- LearnedMacAddressesPersec
- MaxChimneyOffloads
- MaxVMQOffloads
- MulticastPacketsReceivedPersec
- MulticastPacketsSentPersec

- 名称
- Name1
- NameFormat
- NumLearnableAddresses
- OperationalStatus
- OtherDedicatedDescriptions
- OtherEnabledState
- OtherIdentifyingInfo
- PacketsFlooded
- PacketsFloodedPersec
- PacketsPersec
- PacketsReceivedPersec
- PacketsSentPersec
- 路径
- PowerManagementCapabilities
- PrimaryOwnerContact
- PrimaryOwnerName
- PurgedMacAddresses
- PurgedMacAddressesPersec
- RequestedState
- ResetCapability
- 角色
- ScopeOfResidence
- 状态
- StatusDescriptions
- TimeOfLastStateChange

#### 示例：显示主机开关

本示例显示 Hyper-V 主机上的主机开关列表。

```
dpmhv-getHostSwitches -host hvserver
```

## dpmhv-getJobInfo (Funclet) 命令-显示作业信息

dpmhv-getJobInfo 命令用于检索所有与异步作业相关的信息（如完成百分比、错误信息和开始时间）。

此命令具有以下格式：

```
dpmhv-getJobInfo  
-jobref job  
[-silent]  
[-retval]  
[-detail]
```

### **-jobref *job***

指定异步作业的作业 ID。

**注意：**异步作业执行完成时，Hyper-V 仅保留作业信息几分钟（五分钟），超出该时间后信息将无效。

### **-silent**

指定不输出到屏幕。

### **-retval**

返回一个值以进一步处理。

### **-detail**

（可选）显示异步作业的详细信息。

## 返回值

dpmhv-getJobInfo 命令将返回对象，其中每个对象均代表当前的作业状态。

Hyper-V 作业：

以常规模式针对本地 Hyper-V 返回下列属性：

- 标题
- ElementName
- ErrorCode
- ErrorDescription
- JobStatus
- PercentComplete
- ResultVMName
- 状态
- StatusDescriptions

以详细模式针对本地 Hyper-V 返回下列属性：

- Cancellable
- 标题
- DeleteOnCompletion
- 描述
- ElapsedTime
- ElementName
- ErrorCode
- ErrorDescription
- ErrorSummaryDescription
- HealthState
- InstallDate
- InstanceID
- JobRunTimes
- JobState
- JobStatus
- LocalOrUtcTime
- 名称
- 通知
- OII- OperationalStatus

- OtherRecoveryAction
- 所有者
- 路径
- PercentComplete
- 优先级
- RecoveryAction
- ResultVMName
- RunDay
- RunDayOfWeek
- RunMonth
- RunStartInterval
- ScheduledStartTime
- StartTime
- 状态
- StatusDescriptions
- TimeBeforeRemoval
- TimeOfLastStateChange
- TimeSubmitted
- UntilTime

Microsoft System Center Virtual Machine Manager (SCVMM) 作业:

以常规模式针对 SCVMM 返回的属性:

- CurrentStep
- 描述
- ErrorInfo
- IsCompleted
- OId
- ProgressValue
- ResultVMName
- 状态
- StatusString

以详细模式针对 SCVMM 返回的属性:

- AdditionalMessages[]
- AreAuditRecordsAvailable

- AuditRecords[]
- CmdletName
- CurrentStep
- 描述
- EndTime
- ErrorInfo
- ID
- IsCompleted
- IsFullyCached
- IsRestartable
- IsStoppable
- IsVisible
- MarkedForDeletion
- 名称
- OId
- 所有者
- OwnerSID
- PROTipID
- 进度
- ProgressValue
- ResultName
- ResultObjectID
- ResultObjectType
- ResultObjectTypeName
- ResultVMName
- ServerConnection
- 源
- StartTime
- 状态
- StatusString
- Steps[]
- 目标
- TargetObjectID
- TargetObjectType
- WasNotifiedOfCancel



### 示例：显示作业信息

本示例显示导出作业的完成状态。

```
jref = dpmhv-exportVMEx -host hvserver -vm TestVM -dest c:\Ex-async
do
{
    sleep(5000)
    job = dpmhv-getJobInfo -jobref (jref) -silent -retval
    ? "Completion:", job.PercentComplete+"%"
} while(job.ErrorCode==0 && job.PercentComplete<100)
? job.StatusDescriptions
if(job.ErrorCode!=0)
{
    ? job.ErrorDescription
}
```

### 示例：显示 SCVMM 作业信息

本示例显示使用 SCVMM 模板创建的新 VM 的完成状态。

```
jref=dpmhv-createVMFromTemplate -host hvserver -vm NewVM -dest c:\VMs\NewBox
-template Win2k3Clean -scvmmHost SCVMMLibSrv -async
do
{
    sleep(5000)
    job = dpmhv-getJobInfo -jobref (jref) -silent -retval
    ? "Completion:", job.ProgressValue+"%"
} while(!job.IsCompleted)
? job.StatusString
s = job.ErrorInfo;
errorCode = parseInt(s.substr(s.lastIndexOf('(')+1).left(-1));
if(errorCode!=0)
{
    ? job.ErrorInfo
}
```

## dpmhv-getJobStatus (Funclet) 命令-显示作业状态

dpmhv-getJobStatus 命令用于检索异步作业的状态。

此命令具有以下格式：

```
dpmhv-getJobStatus
-jobref job

-jobref job
```

指定异步作业的作业 ID。

**注意：**异步作业执行完成时，Hyper-V 仅保留作业信息几分钟（五分钟），超出该时间后信息将无效。

## 返回值

`dpmhv-getJobStatus` 命令返回指定作业的以下状态信息：

- 已取消
- 已创建
- 失败
- 正在进行
- 已成功
- 未知
- 正在等待

## 示例：显示作业状态

本示例显示导出作业的状态。

```
jref = dpmhv-exportVMEx -host hvserver -vm TestVM -dest c:\Ex -async  
dpmhv-getJobState -jobref (jref)
```

## `dpmhv-getPhysDisks (Funclet)` 命令-显示所有物理磁盘

`dpmhv-getPhysDisks` 命令用于列出要附加到 Hyper-V 服务器上的虚拟机的物理磁盘。在 Windows 磁盘管理器中，物理磁盘应当显示为脱机状态。

此命令具有以下格式：

```
dpmhv-getPhysDisks  
-host hostname  
[-silent]  
[-retval]  
[-detail]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

### **-detail**

（可选）显示附加到虚拟机的物理磁盘的详细信息。

## 返回值

dpmhv-getPhysDisks 命令返回一组代表指定虚拟机的 SCSI 控制器的对象。

以常规模式返回的属性：

- 标题
- DriveNumber
- ElementName
- 名称
- 路径
- StatusDescriptions

以详细模式返回的属性：

- AdditionalAvailability
- 可用性
- Capabilities
- CapabilityDescriptions
- 标题
- CompressionMethod
- CreationClassName
- DefaultBlockSize
- 描述
- DeviceID
- DriveNumber
- ElementName
- EnabledDefault
- EnabledState
- ErrorCleared
- ErrorDescription
- ErrorMethodology
- HealthState
- IdentifyingDescriptions
- InstallDate
- LastCleaned
- LastErrorCode
- LoadTime

- LocationIndicator
- MaxAccessTime
- MaxBlockSize
- MaxMediaSize
- MaxQuiesceTime
- MaxUnitsBeforeCleaning
- MedialsLocked
- MinBlockSize
- MountCount
- 名称
- NeedsCleaning
- NumberOfMediaSupported
- OperationalStatus
- OtherEnabledState
- OtherIdentifyingInfo
- 路径
- PowerManagementCapabilities
- PowerManagementSupported
- PowerOnHours
- RequestedState
- 安全
- 状态
- StatusDescriptions
- StatusInfo
- SystemCreationClassName
- SystemName
- TimeOfLastMount
- TimeOfLastStateChange
- TotalMountTime
- TotalPowerOnHours
- UncompressedDataRate
- UnitsDescription
- UnitsUsed
- UnloadTime

### 示例：显示所有物理磁盘

本示例显示附加到虚拟机的物理磁盘的基本信息。

```
dpmhv-getPhysDisks -host hvserver
```

## dpmhv-getVersion (Funclet) 命令-显示 Web 服务版本

dpmhv-getVersion 命令用于检索 CA Virtual Assurance 的 Web 服务版本。

此命令具有以下格式：

```
dpmhv-getVersion  
[-retval]  
[-silent]
```

### -retval

返回一个值以进一步处理。

### -silent

指定不输出到屏幕。

### 返回值

dpmhv-getVersion 命令返回表示 CA Virtual Assurance 的 web 服务版本的字符串。

### 示例：显示 Web 服务版本

本示例显示 CA Virtual Assurance 的 Web 服务版本。

```
ver = dpmhv-getVersion -retval -silent  
? ver
```

## dpmhv-getVMInfo (Funclet) 命令-显示所有 VM

dpmhv-getVMInfo 命令用于检索虚拟机的信息和设置。

此命令具有以下格式：

```
dpmhv-getVMInfo  
-vm vmname|-vmid vmguid  
-host hostname  
[-retval]  
[-silent]  
[-detail]
```

### -vm *vmname*

指定 VM 的名称。

### -vmid *vmguid*

指定 VM 的唯一 ID。

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

## 返回值

dpmhv-getVMInfo 命令返回具有虚拟机信息的对象。

以常规模式返回下列属性：

- vm
- 标题
- ElementName
- 名称
- OperationalStatus
- StatusDescriptions
- vminfo
- MemoryUsage
- ProcessorLoad
- ProcessorLoadHistory
- vmsettings
- AutoActivate
- AutomaticRecoveryAction
- AutomaticShutdownAction
- AutomaticStartupAction
- AutomaticStartupActionDelay
- 描述
- ElementName
- ExternalDataRoot
- SnapshotDataRoot
- SystemName
- vmcursettings
- BaseBoardSerialNumber
- BIOSGUID
- BIOSSerialNumber
- ChassisAssetTag
- ChassisSerialNumber
- SystemName

以详细模式返回下列属性：

- vm
- AssignedNumaNodeList
- 标题

- CreationClassName
- Dedicated
- 描述
- ElementName
- HealthState
- IdentifyingDescriptions
- InstallDate
- 名称
- NameFormat
- OnTimeInMilliseconds
- OperationalStatus
- OtherDedicatedDescriptions
- OtherEnabledState
- OtherIdentifyingInfo
- PowerManagementCapabilities
- PrimaryOwnerContact
- PrimaryOwnerName
- ProcessID
- RequestedState
- ResetCapability
- 角色
- 状态
- StatusDescriptions
- TimeOfLastConfigurationChange
- TimeOfLastStateChange
- vminfo
- AsynchronousTasks
- CreationTime
- ElementName
- EnabledState
- GuestOperatingSystem
- HealthState
- Heartbeat
- MemoryUsage
- 名称



- 注意:
- NumberOfProcessors
- OperationalStatus
- 路径
- ProcessorLoad
- ProcessorLoadHistory
- 快照
- StatusDescriptions
- ThumbnailImage
- UpTime
- vmsettings
- AdditionalRecoveryInformation
- AllowFullISCSICommandSet
- AutoActivate
- AutomaticRecoveryAction
- AutomaticShutdownAction
- AutomaticStartupAction
- AutomaticStartupActionDelay
- 标题
- CreationTime
- DebugChannelId
- 描述
- ElementName
- ExternalDataRoot
- InstanceID
- OtherVirtualSystemType
- 路径
- ScopeOfResidence
- SettingType
- SnapshotDataRoot
- SystemName
- 版本
- VirtualSystemType
- vmcursettings
- BIOSGUID

- BIOSSerialNumber
- BaseBoardSerialNumber
- ChassisAssetTag
- ChassisSerialNumber
- SystemName

#### 示例：显示所有 VM

本示例显示虚拟机的信息。

```
dpmhv-getVMInfo -host hvserver -vm TestVM
```

### dpmhv-getVMProperties (Funclet) 命令-显示 VM 属性

dpmhv-getVMProperties 命令用于检索 VM 的属性，如 CPU、内存和操作设置。

此命令具有以下格式：

```
dpmhv-getVMProperties  
-vm vmname|-vmid vmguid  
-host hostname  
[-retval]  
[-silent]
```

#### **-vm *vmname***

指定 VM 的名称。

#### **-vmid *vmguid***

指定 VM 的唯一 ID。

#### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

#### **-retval**

返回一个值以进一步处理。

#### **-silent**

指定不输出到屏幕。

## 返回值

dpmhv-getVMProperties 命令返回对象，其中每个对象代表 VM 属性，如下所示：

- memory
- cpu
- cpuSocketCount
- cpuidLimit
- cpuReserve
- cpuLimit
- cpuWeight
- 操作
- startAction
- startDelay
- stopAction
- recoveryAction

### 示例：显示 VM 属性

本示例显示下列 VM 属性：VM CPU 和内存分配。

```
dpmhv-getVMProperties -host hvserver -vm TestVM
```

### 示例：显示 VM 的 CPU 和内存分配

本示例显示 VM “TestVM” 的 CPU 以及内存分配。

```
props=dpmhv-getVMProperties -host hvserver -vm TestVM -retval -silent
? "Memory:", props.memory+"MB"
? "CPU cores:", props.cpu.cpuSocketCount
```

## dpmhv-getVMState (Funclet) 命令-显示 VM 状态

dpmhv-getVMState 命令用于检索 VM 的电源状态。

此命令具有以下格式：

```
dpmhv-getVMState
-vm vmname[-vmid vmguid]
-host hostname
[-retval]
[-silent]
```

**-vm *vmname***

指定 VM 的名称。

**-vmid *vmguid***

指定 VM 的唯一 ID。

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-silent**

指定不输出到屏幕。

**-retval**

返回一个值以进一步处理。

**返回值**

`dpmhv-getVMState` 命令返回表示 VM 电源状态的字符串。返回的值为：

- disabled: VM 已关闭
- enabled: VM 正在运行
- paused: VM 已暂停
- suspended: VM 已挂起

**示例：显示 VM 状态**

本示例检查 VM 是否在运行。

```
s = dpmhv-getVMState -host hvserver -vm TestVM -retval -silent
if(s=="enabled")
? "VM is running"
else
? "VM is not running"
```

**dpmhv-importVM (Funclet) 命令-导入 VM**

`dpmhv-importVM` 命令用于导入使用 `dpmhv-exportVM` 命令导出的 VM。与导出的 VM 类似，将 VM 导入到某个位置时，只能导入一次。

此命令具有以下格式：

```
dpmhv-importVM
-host hostname
-path pathname
[-vm vmname]
[-nonewid]
[-async]
[-retval]
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-path pathname**

指定要导入的 VM 的路径。

**-vm vmname**

指定 VM 的名称。

**-nonewid**

（可选）指定不为要导入的 VM 生成唯一的 GUID。导入的 VM 具有与源 VM（导出的 VM）类似的 GUID。具有相同 GUID 的两个 VM 会发生冲突，建议不要使用。

**-async**

（可选）指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**返回值**

同步执行 `dpmhv-importVM` 命令时，将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给 `dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。在同步模式下，如果在执行期间发生错误，命令将返回错误代码并报告错误。

**示例：导入 VM**

本示例导入使用 `dpmhv-exportVM` 命令导出的 VM。

```
dpmhv-exportVM -host hvserver -vm TestVM -dest c:\Ex\MyVM dpmhv-importVM  
-host hvserver -path c:\Ex\MyVM -vm TestVM1
```

此外，还可发出 `dpmhv-importVM` 命令，从原始导出的 VM 创建多个 VM。

## dpmhv-importVMEx (Funclet) 命令-复制和导入 VM

dpmhv-importVMEx 命令用于导入和复制使用 *dpmhv-exportVMEx* 命令导出的 VM。将 VM 导入到与导出的 VM 不同的位置时，可多次导入 VM。

此命令具有以下格式：

```
dpmhv-importVMEx
-host hostname
-path pathname
[-dest dest]
[-vm vmname]
[-nonewid]
[-async]
[-retval]
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-path *pathname***

指定要导入和复制的 VM 的路径。

### **-dest *dest***

指定要将 VM 导入和复制到的目标路径。

### **-vm *vmname***

指定 VM 的名称。

### **-nonewid**

（可选）指定不为要导入的 VM 生成唯一的 GUID。导入的 VM 具有与源 VM（导出的 VM）类似的 GUID。具有相同 GUID 的两个 VM 会发生冲突，建议不要使用。

### **-async**

（可选）指定要异步执行命令的选项。默认情况下，同步执行该命令。

**注意：**如果执行该选项之前此命令完成执行，它将返回用于查看操作状态的作业 ID。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

## 返回值

同步执行 `dpmhv-importVMEx` 命令时，将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给 `dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。在同步模式下，如果在执行期间发生错误，命令将返回错误代码并报告错误。

## 示例：导入 VM

本示例导入使用 `dpmhv-exportVMEx` 命令导出的 VM。

```
dpmhv-exportVMEx -host hvserver -vm TestVM -dest c:\Ex\MyVM -nosubdir
dpmhv-importVMEx -host hvserver -path c:\Ex\MyVM -dest C:\VMs\TestVM1 -vm TestVM1
```

此外，还可以发出 `dpmhv-importVMEx` 命令，从原始导出的 VM 创建多个 VM。

## dpmhv-setSysprepProperties (Cmdlet) 命令-创建虚拟磁盘

`dpmhv-setSysprepProperties` 命令用于创建虚拟磁盘映像。

此命令具有以下格式：

```
dpmhv-setSysprepProperties
-host hostname
-vm vmname|-vmid vmguid|-jobref job
[-computerName computerName]
[-ip4addr ip4addr]
[-ip4dhcp ip4dhcpInt]
[-ip4mask ip4mask]
[-ip4gw ip4gw]
[-ip4metric ip4metric]
[-ip4dns ip4dns]
[-disableAdmin]
[-adminPass adminPW]
[-autoLogon alcnt]
[-duplicatorString dupstr]
[-timeZone timezone]
[-productKey key]
[-userName UserName]
[-organization orgname>]
[-domain domname]
[-domainAdmin domad]
[-domainAdminPass dompw]
[-workgroup wgname]
[-adminUser adminUser]
[-adminUserPass adminUserPW]
[-custom custom]
```

### -host *hostname*

指定 Hyper-V 服务器主机的名称。

默认值：`$$HVHost`

**-vm *vmname***

指定 VM 的名称。

**-vmid *vmguid***

指定 VM 的唯一 ID。

**-jobref *job***

指定异步作业的作业 ID。

**注意：**异步作业执行完成时，Hyper-V 仅保留作业信息几分钟（五分钟），超出该时间后信息将无效。

**-computerName *computerName***

（可选）指定计算机的名称。对该参数的支持需要使用 Sysprep 工具的映像。该选项对于异步执行命令无效。

**默认值：**空

**-ip4addr *ip4addr***

（可选）指定要分配给 VM 的接口的静态 IPv4 地址。要设置专用接口的 IP 地址，请使用已知的接口名称作为 IP 地址的前缀并以“#”分隔。例如，

**-ip4addr "Local Area Connection#192.168.1.200"**。如果模板映像具有多个网络适配器，则 IP 地址将分配给第一个接口。该选项对于异步执行命令无效。

**默认值：**""

**-ip4dhcp *ip4dhcpInt***

（可选）指定一个选项来打开 VM 的特定接口的 DHCP。您也可为该选项指定接口名称。例如，

**-ip4dhcp "Local Area Connection"**。如果模板映像具有多个网络适配器，则为第一个接口打开 DHCP。该选项对于异步执行命令无效。

**默认值：**local

**-ip4mask *ip4mask***

（可选）指定要为 VM 分配的子网掩码。该选项与 **-ip4addr** 选项结合使用。如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：**""



**-ip4gw *ip4gw***

（可选）指定该选项来为 VM 设置网关。该选项与 **-ip4addr** 选项结合使用。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**注意：**如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。

**默认值：** ""

**-ip4metric *ip4metric***

（可选）指定要为 VM 设置的接口度量标准。该选项与 **-ip4addr** 选项结合使用。如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认：** 1

**-ip4dns *ip4dns***

（可选）指定要为 VM 设置的 DNS 服务器。该选项与 **-ip4addr** 选项结合使用。如果在

**-ip4addr** 选项中指定了接口名称，则必须在该选项中使用相同的接口名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：** ""

**-disableAdmin**

（可选）指定该选项可禁用 VM 的默认管理员帐户。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**-adminPass *adminPW***

（可选）该选项用于设置 VM 的默认管理员帐户密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。在异步执行中将忽略该参数。

**注意：**要成功设置该选项，请将使用 Sysprep 工具创建的 Windows Server 2003 管理员密码设置为空。

**默认值：** 空

**-autoLogon *alcnt***

(可选)指定一个选项来设置要创建以使用默认管理员帐户自动登录的帐户数目。帐户会在 Sysprep 进程完成之后创建。`hyperv_sp_dupstr`。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:**

-1

**-duplicatorString *dupstr***

(可选)指定要在 VM 注册表中设置的系统复印机的名称。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:** 空

**-timeZone *timezone***

(可选)指定使用模板创建的 VM 所用的时区。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:**

-1

**-productKey *key***

(可选)指定 VM 的 Windows 产品激活密钥。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:** 空

**-userName *UserName***

(可选)指定许可 VM 的 Windows 用户名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:** 空

**-organization *orgname***

(可选)指定许可 VM 中的 Windows 副本的组织。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值:** 空

**-domain *domainname***

（可选）指定 VM 的域名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。域和工作组规格互不相容。

**默认值：**空

**-domainAdmin *domad***

（可选）将用户帐户指定为要在域中创建的域管理员。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：**空

**-domainAdminPass *dompw***

（可选）指定要在域中创建的域管理员用户帐户的密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：**空

**-workgroup *wgname***

（可选）指定要为 VM 创建的工作组。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。域和工作组规格互不相容。该选项对于异步执行命令无效。

**默认值：**空

**-adminUser *adminUser***

（可选）指定要作为默认 Administrators 组的一部分创建的管理员用户名。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：**空

**-adminUserPass *adminUserPW***

（可选）指定要作为默认 Administrators 组的一部分创建的管理员用户帐户的密码。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项对于异步执行命令无效。

**默认值：**空

**-custom *custom***

（可选）指定要在 Sysprep 进程结束时执行的自定义命令的逗号分隔列表。该参数需要 Sysprep 的 Windows 映像。在异步执行中将忽略该参数。

**默认值：**空

## 返回值

同步执行 `dpmhv-setSysprepProperties` 命令时，将返回空字符串。如果异步执行，会将代表作业 ID 的字符串传递给 `dpmhv-getJobErrorInfo`、`dpmhv-getJobInfo` 或 `dpmhv-getJobStatus`。如果在执行期间发生错误，将报告异常。

## 示例：创建虚拟磁盘

本示例创建 2 GB 的动态虚拟磁盘。

```
dpmhv-createVirtDisk -host hvserver -path C:\VMDisks\hdd.vhd -type dynamic  
-sizeGB 2
```

## `dpmhv-setVMClustered (Cmdlet)` 命令-在群集中设置 VM

`dpmhv-setVMClustered` 命令用于从共享群集资源添加或删除 VM。在 Hyper-V 服务器环境中，群集中的 VM 可以是 Hyper-V 服务器的共享群集资源或专用资源。如果 VM 是群集共享资源，则无法将其启用以在群集节点之间进行快速或实时迁移。此外，还无法迁移位于专用存储（如本地硬盘上的虚拟磁盘），而不是群集共享存储上的 VM 资源。

此命令具有以下格式：

```
dpmhv-setVMClustered  
-host hostname  
-vm vmname|-vmid vmguid  
-clustered {on,off}[-retval]  
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-clustered {on, off}**

指定群集的共享模式。

开

VM 是共享群集资源。

关

VM 是 Hyper-V 服务器的专用资源。

### **-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：在群集中设置 VM**

本示例将 VM 添加到共享群集资源。

```
dpmhv-setVMClustered -host hvserver -vm TestVM -clustered on
```

**dpmhv-setVMDisk (Funclet) 命令-在驱动器控制器上创建驱动器或磁盘**

dpmhv-setVMDisk 命令用于在驱动器控制器上创建驱动器或磁盘。您可以使用该选项来连接/断开映像。

此命令具有以下格式：

```
dpmhv-setVMDisk
-host hostname
-vm vmname|-vmid vmguid
[-path pathname]
-drive{HDD,DVD}
-type{scsi,ide}[-id id]
[-lun lun]
[-retval]
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-vm *vmname***

指定 VM 的名称。

**-vmid *vmguid***

指定 VM 的唯一 ID。

**-path *pathname***

（可选）指定在驱动器控制器上创建的驱动器或磁盘所在的路径。

**-drive {*HDD, DVD*}**

指定要分配给 VM 的驱动器类型。您可以分配以下驱动器类型：

**HDD**

表示硬盘驱动器

**DVD**

表示压缩磁盘 (CD) 或数字视盘 (DVD) 的驱动器。

**-type {scsi, ide}**

指定要连接到 VM 的虚拟硬盘的控制器类型。Hyper-V 支持下列两种类型：

**SCSI**

将控制器类型指定为 SCSI。

**IDE**

将控制器类型指定为 IDE。

**-id id**

（可选）指定在控制器类型中提到的控制器的 ID。对于控制器类型 IDE，值是 0 或 1。对于 SCSI，值是 0、1、2 或 3，具体取决于添加到 VM 的 SCSI 控制器数目。如果未指定 id（或 -1），该命令将选择具有可用通道的指定类型的第一个控制器。

**默认值：**

-1

**-lun lun**

（可选）以 <type> 和 <id> 的形式指定控制器的逻辑单元。对于 IDE，值是 0 或 1，而对于 SCSI，限制范围是 [0, 63]。如果未指定单元编号（或 -1），该命令将选择指定控制器上的可用通道。

**默认值：**

-1

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例：在 SCSI 控制器上创建虚拟驱动器**

本示例将虚拟硬盘映像文件连接到第一个 SCSI 控制器的第一个通道。

```
dpmhv-setVMDisk -host hvserver -vm TestVM -type scsi -id 0 -lun 0 -drive HDD  
-path c:\VHDs\disk.vhd
```

**示例：将 DVD 连接到 VM**

本示例将物理 DVD-ROM 驱动器连接到 VM “Test VM”。

```
dpmhv-setVMDisk -host hvserver -vm TestVM -type ide -id 0 -lun 1 -drive DVD  
-path D:
```

### 示例：将物理主机磁盘连接到 SCSI 控制器

本示例获取物理主机磁盘，并将其连接到第一个可用的 SCSI 通道。

```
pd=dpmhv-getPhysDisks -host hvserver -retval -silent
```

```
dpmhv-setVMDisk -host hvserver -vm TestVM -type scsi -drive HDD -path (pd[0].Path)
```

## dpmhv-setVMProperties (Cmdlet) 命令-设置 VM 属性

dpmhv-setVMProperties 命令用于设置 VM 的属性。

此命令具有以下格式：

```
dpmhv-setVMProperties
-host hostname
-vm vmname|-vmid vmguid
[-name name]
[-mem mem]
[-cpus cpus]
[-cpuidlimit limit]
[-cpureserve reserve]
[-cpulimit limit]
[-cpuweight weight]
[-startAction {none,auto,always}]
[-startDelay delay]
[-stopAction {save,off,shutdown}]
[-recoveryAction {none,restart,revert}]
[-retval]
[-silent]
```

### -host *hostname*

指定 Hyper-V 服务器主机的名称。

### -vm *vmname*

指定 VM 的名称。

### -vmid *vmguid*

指定 VM 的唯一 ID。

### -name *name*

显示 VM 的名称。

### -mem *mem*

(可选) 指定要创建的 VM 的 RAM 内存 (MB)。

默认值：-1

### -cpus *cpus*

(可选) 指定要分配给 VM 的 CPU 核心数。

-

**-cpuidlimit *limit***

（可选）指定 VM 的 CPU ID 功能的限制。该选项可改进与传统操作系统（如 Windows NT）的兼容性。

**默认值：** -1

**-cpureserve *reserve***

（可选）指定要为 VM 保留的 CPU 的百分比。如果没有指定该选项，Hyper-V 服务器会根据系统整体使用情况分配 CPU 周期。

**默认值：** -1

**-cpulimit *limit***

（可选）指定 VM 使用的 CPU 资源数的限制。该选项可改进在具有不同 CPU 功能的物理主机之间移动 VM 时的兼容性。

**默认值：** -1

**-cpuweight *weight***

（可选）指定虚拟机的相对权重。该选项控制多个 VM 运行时的资源分配。有效条目：1 到 10000 的整数。

**默认值：** -1

**-startAction {none,auto,always}**

（可选）指定 Hyper-V 主机启动时要在 VM 上执行的操作。选项包括：  
无

不执行任何操作。

自动

自动启动 VM。

**注意：** 如果 Hyper-V 主机关闭之前 VM 正在运行，则使用该选项。

always

始终启动 VM。

**-startDelay *delay***

（可选）指定 Hyper-V 主机启动并运行后启动 VM 的延迟（秒）。

**默认值：** -1



**-stopAction {save,off,shutdown}**

(可选) 指定物理 Hyper-V 关闭时要在 VM 上执行的操作。选项包括:

**save**

挂起 VM。

**关**

关闭 VM 电源。

**shutdown**

关闭系统。

**-startDelay *delay***

(可选) 指定 Hyper-V 主机启动并运行后启动 VM 的延迟 (秒)。

默认值: -1

**-recoveryAction {none,restart,revert}**

(可选) 指定因意外关闭而重新启动 Hyper-V 主机后要在 VM 上执行的操作。选项包括:

**无**

不执行任何操作。

**restart**

重新启动 VM。

**revert**

还原到 VM 的最后一个快照。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**示例: 更改 CPU 核心的数目**

本示例将分配给 VM 的 CPU 核心数更改为 2。要成功运行该命令, 请关闭 VM。

```
dpmhv-setVMPProperties -host hvserver -vm TestVM -cpus 2
```

## dpmhv-SetVMVirtFloppy (Cmdlet) 命令-将虚拟软盘映像连接到 VM 软盘驱动器

dpmhv-SetVMVirtFloppy 命令用于将虚拟软盘映像连接到 Hyper-V 主机上指定的 VM 软盘驱动器。

此命令具有以下格式：

```
dpmhv-setVMVirtFloppy  
-host hostname  
-vm vmname|-vmid vmguid  
[-path pathname]  
[-retval]  
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-path *pathname***

(可选) 指定要创建的虚拟软盘映像 (\*.vfd) 文件的路径。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

## 示例：将软盘映像连接到 Hyper-V 主机上的 VM 软盘驱动器

本示例将软盘映像 “floppy.vfd” 连接到 Hyper-V 主机 “TestVM” 上的 VM 软盘驱动器。

```
dpmhv-setVMVirtFloppy  
-host hvserver  
-vm TestVM  
-path c:\img\floppy.vfd
```

## 示例：从 Hyper-V 主机上的 VM 软盘驱动器分离软盘映像

本示例从 Hyper-V 主机上的 VM 软盘驱动器分离软盘映像。

```
dpmhv-setVMVirtFloppy -host hvserver -vm TestVM
```

## dpmhv-ShowClusterSharedVolumes (Funclet) 命令-显示所有群集共享卷

dpmhv-ShowClusterSharedVolumes 命令用于显示 Hyper-V 服务器上的所有群集共享卷。

此命令具有以下格式：

```
dpmhv-showClusterSharedVolumes  
-host hostname  
[-retval]  
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

## 返回值

dpmhv-ShowClusterSharedVolumes 命令从一组对象返回当前的密钥和值对。

### 示例：显示所有群集共享卷

本示例返回主机“hvserver”上共享卷的列表。

```
dpmhv-showClusterSharedVolumes -host hvserver
```

## dpmhv-ShowDirectories (Funclet) 命令-显示所有目录

dpmhv-ShowDirectories 命令用于显示系统中指定目录下的所有子目录。指定的系统可以是 Hyper-V 服务器或启用了远程 WMI 的 Windows 系统。

此命令具有以下格式：

```
dpmhv-showDirectories  
-host hostname  
-dir dir  
[-retval]  
[-silent]  
[-detail]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-dir *dir***

指定目录的名称。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**-detail**

（可选）显示目录列表的详细信息。

## 返回值

`dpmhv-ShowDirectories` 命令返回一组对象, 这些对象表示位于指定父目录下的子目录。

以常规模式返回下列属性:

- 标题
- Drive
- FileName
- 名称
- 路径
- Writeable \$<<prop\_detail
- AccessMask
- Archive
- CSCreationClassName
- CSName
- 标题
- 压缩
- CompressionMethod
- CreationClassName
- CreationDate
- 描述
- Drive
- EightDotThreeFileName
- Encrypted
- EncryptionMethod
- 扩展名
- FSCreationClassName
- FSName
- FileName
- FileSize
- FileType
- Hidden
- InUseCount
- InstallDate
- LastAccessed
- LastModified

- 名称
- 路径
- Readable
- 状态
- 系统
- Writeable

#### 示例：显示所有目录

本示例显示主机 hvserver 的根目录（c: 驱动器）下的子目录。

```
dpmhv-showDirectories  
-host hvserver  
-dir c:\
```

#### 示例：显示 Program Files 下的所有目录

本示例显示 “C:\Program Files” 的子目录。

```
function walkDir(sPath)  
{  
    dirs = dpmhv-showDirectories -host hvserver -dir (sPath) -retval -silent  
    for each (subdir in dirs)  
    {  
        ? subdir.Name  
        walkDir(subdir.Name);  
    }  
}  
walkDir("C:\Program Files");
```

### dpmhv-ShowHosts (Funclet) 命令-显示所有 Hyper-V 服务器

dpmhv-ShowHosts 命令用于显示所有 Hyper-V 服务器。

此命令具有以下格式：

```
dpmhv-showHosts  
[-detail]  
[-retval]  
[-silent]
```

#### -detail

（可选）显示虚拟网络（主机网络）的详细信息。

#### -retval

返回一个值以进一步处理。

#### -silent

指定不输出到屏幕。

## 返回值

dpmhv-ShowHosts 命令返回一组表示 Hyper-V 服务器的对象。以常规模式返回属性 “Name”。

以详细模式返回下列属性：

- AssignedNumaNodeList
- CIMV2\_AdminPasswordStatus
- CIMV2\_AutomaticManagedPagefile
- CIMV2\_AutomaticResetBootOption
- CIMV2\_AutomaticResetCapability
- CIMV2\_BootOptionOnLimit
- CIMV2\_BootOptionOnWatchDog
- CIMV2\_BootROMSupported
- CIMV2\_BootupState
- CIMV2\_Caption
- CIMV2\_ChassisBootupState
- CIMV2\_CreationClassName
- CIMV2\_CurrentTimeZone
- CIMV2\_DNSHostName
- CIMV2\_DaylightInEffect
- CIMV2\_Description
- CIMV2\_Domain
- CIMV2\_DomainRole
- CIMV2\_EnableDaylightSavingsTime
- CIMV2\_FrontPanelResetStatus
- CIMV2\_InfraredSupported
- CIMV2\_InitialLoadInfo
- CIMV2\_InstallDate
- CIMV2\_KeyboardPasswordStatus
- CIMV2\_LastLoadInfo
- CIMV2\_Manufacturer
- CIMV2\_Model
- CIMV2\_Name
- CIMV2\_NameFormat
- CIMV2\_NetworkServerModeEnabled
- CIMV2\_NumberOfLogicalProcessors

- CIMV2\_NumberOfProcessors
- CIMV2\_OEMLogoBitmap
- CIMV2\_OEMStringArray
- CIMV2\_PCSystemType
- CIMV2\_PartOfDomain
- CIMV2\_PauseAfterReset
- CIMV2\_PowerManagementCapabilities
- CIMV2\_PowerManagementSupported
- CIMV2\_PowerOnPasswordStatus
- CIMV2\_PowerState
- CIMV2\_PowerSupplyState
- CIMV2\_PrimaryOwnerContact
- CIMV2\_PrimaryOwnerName
- CIMV2\_ResetCapability
- CIMV2\_ResetCount
- CIMV2\_ResetLimit
- CIMV2\_Roles
- CIMV2\_Status
- CIMV2\_SupportContactDescription
- CIMV2\_SystemStartupDelay
- CIMV2\_SystemStartupOptions
- CIMV2\_SystemStartupSetting
- CIMV2\_SystemType
- CIMV2\_ThermalState
- CIMV2\_TotalPhysicalMemory
- CIMV2\_UserName
- CIMV2\_WakeUpType
- CIMV2\_Workgroup
- 标题
- ClusterName
- CreationClassName
- Dedicated
- 描述
- ElementName
- EnabledDefault



- EnabledState
- HealthCritical
- HealthOk
- HealthState
- HostCPUUtilisationPercent
- HostMemoryAvailableMB
- IdentifyingDescriptions
- InstallDate
- IsHyperVClusterNode
- NET\_ArpAlwaysSourceRoute
- NET\_ArpUseEtherSNAP
- NET\_Caption
- NET\_DHCPEnabled
- NET\_DHCPLeaseExpires
- NET\_DHCPLeaseObtained
- NET\_DHCPServer
- NET\_DNSDomain
- NET\_DNSDomainSuffixSearchOrder
- NET\_DNSEnabledForWINSResolution
- NET\_DNSHostName
- NET\_DNSServerSearchOrder
- NET\_DatabasePath
- NET\_DeadGWDetectEnabled
- NET\_DefaultIPGateway
- NET\_DefaultTOS
- NET\_DefaultTTL
- NET\_Description
- NET\_DomainDNSRegistrationEnabled
- NET\_ForwardBufferMemory
- NET\_FullDNSRegistrationEnabled
- NET\_GatewayCostMetric
- NET\_IGMPLevel
- NET\_IPAddress
- NET\_IPConnectionMetric
- NET\_IPEnabled
- NET\_IPFilterSecurityEnabled

- NET\_IPPortSecurityEnabled
- NET\_IPSecPermitIPProtocols
- NET\_IPSecPermitTCPPorts
- NET\_IPSecPermitUDPPorts
- NET\_IPSubnet
- NET\_IPUseZeroBroadcast
- NET\_IPXAddress
- NET\_IPXEnabled
- NET\_IPXFrameType
- NET\_IPXMediaType
- NET\_IPXNetworkNumber
- NET\_IPXVirtualNetNumber
- NET\_Index
- NET\_InterfaceIndex
- NET\_KeepAliveInterval
- NET\_KeepAliveTime
- NET\_MACAddress
- NET\_MTU
- NET\_NumForwardPackets
- NET\_PMTUBHDetectEnabled
- NET\_PMTUDiscoveryEnabled
- NET\_ServiceName
- NET\_SettingID
- NET\_TcpMaxConnectRetransmissions
- NET\_TcpMaxDataRetransmissions
- NET\_TcpNumConnections
- NET\_TcpUseRFC1122UrgentPointer
- NET\_TcpWindowSize
- NET\_TcpipNetbiosOptions
- NET\_WINSEnableLMHostsLookup
- NET\_WINSHostLookupFile
- NET\_WINSPrimaryServer
- NET\_WINSScopeID
- NET\_WINSSecondaryServer
- 名称

- NameFormat
- OnTimeInMilliseconds
- OperationalStatus
- OtherDedicatedDescriptions
- OtherEnabledState
- OtherIdentifyingInfo
- 路径
- PowerManagementCapabilities
- PrimaryOwnerContact
- PrimaryOwnerName
- ProcessID
- RequestedState
- ResetCapability
- 角色
- 状态
- StatusDescriptions
- TimeOfLastConfigurationChange
- TimeOfLastStateChange
- VSMS\_BiosLockString
- VSMS\_Caption
- VSMS\_DefaultExternalDataRoot
- VSMS\_DefaultVirtualHardDiskPath
- VSMS\_Description
- VSMS\_ElementName
- VSMS\_InstanceID
- VSMS\_MaximumMacAddress
- VSMS\_MinimumMacAddress
- VSMS\_NumaSpanningEnabled
- VSMS\_PrimaryOwnerContact
- VSMS\_PrimaryOwnerName
- VSMS\_ScopeOfResidence

**示例：显示所有 Hyper-V 服务器**

本示例返回 Hyper-V 服务器的列表。

dpmhv-showHosts

**示例：显示所有 Hyper-V 服务器和主机上的 VM**

本示例显示 Hyper-V 服务器列表和每个主机上的 VM。

```
a = dpmhv-showHosts -retval -silent
for each(h in a)
{
    ?
    ? "Host:", h.Name
    dpmhv-showVMs -host (h.Name)
}
```

**dpmhv-ShowLogicalDisks (Funclet) 命令-显示所有逻辑磁盘**

dpmhv-ShowLogicalDisks 命令用于显示连接到 Windows 系统的所有逻辑磁盘。指定的系统可以是 Hyper-V 服务器或启用了远程 WMI 的 Windows 系统。该命令确定 VM 在远程 Hyper-V 服务器的 Windows 文件系统的位置。

此命令具有以下格式：

```
dpmhv-showLogicalDisks
-host hostname
[-retval]
[-silent]
[-detail]
```

**-host hostname**

指定主机名。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**-detail**

（可选）显示逻辑磁盘列表的详细信息。

## 返回值

dpmhv-ShowLogicalDisks 命令返回一组对象，这些对象表示连接到指定主机的逻辑磁盘。

以常规模式返回下列属性：

- 标题
- DeviceID
- FreeSpace
- 名称
- 路径
- 大小

以详细模式返回下列属性：

- 访问权
- 可用性
- BlockSize
- 标题
- 压缩
- ConfigManagerErrorCode
- ConfigManagerUserConfig
- CreationClassName
- 描述
- DeviceID
- DriveType
- ErrorCleared
- ErrorDescription
- ErrorMethodology
- FileSystem
- FreeSpace
- InstallDate
- LastErrorCode
- MaximumComponentLength
- MediaType
- 名称
- NumberOfBlocks
- PNPDeviceID
- 路径

- PowerManagementCapabilities
- PowerManagementSupported
- ProviderName
- Purpose
- QuotasDisabled
- QuotasIncomplete
- QuotasRebuilding
- 大小
- 状态
- StatusInfo
- SupportsDiskQuotas
- SupportsFileBasedCompression
- SystemCreationClassName
- SystemName
- VolumeDirty
- VolumeName
- VolumeSerialNumber

#### 示例：显示所有逻辑磁盘

本示例返回连接到主机的逻辑磁盘列表。

```
dpmhv-showLogicalDisks -host hvserver
```

### dpmhv-ShowSCVMMHardwareProfiles (Funclet) 命令-显示所有 SCVMM 硬件配置文件

dpmhv-ShowSCVMMHardwareProfiles 命令用于显示由 Microsoft System Center Virtual Machine Manager (SCVMM) 定义的硬件配置文件列表，使用 SCVMM 模板创建 VM 时，这些配置文件可用于 VM。

此命令具有以下格式：

```
dpmhv-showSCVMMHardwareProfiles  
-scvmmHost scvmmHostname  
[-retval]  
[-silent]  
[-detail]
```

#### **-scvmmHost *scvmmHostname***

（可选）指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器的主机名。使用 SCVMM 集成开通 VM 时，该参数有效。

#### **-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**-detail**

（可选）显示目录列表的详细信息。

## 返回值

dpmhv-ShowSCVMMHardwareProfiles 命令返回一组对象，这些对象表示在指定的库服务器下定义的硬件配置文件。

以常规模式返回下列属性：

- CPUCount
- 描述
- 内存
- 名称

以详细模式返回下列属性：

- Accessibility
- AddedTime
- BootOrder[]
- 连接
- CPUCount
- CPUMax
- CPUReserve
- CPUType
- ID
- IsFullyCached
- MarkedForDeletion
- 名称
- ServerConnection
- 描述
- DiskIO
- 已启用
- ExpectedCPUUtilization
- HostDrive
- ID
- IsFullyCached
- IsHighlyAvailable
- LimitCPUForMigration
- LimitCPUFunctionality
- MarkedForDeletion



- 内存
- ModifiedTime
- MostRecentTask
- 名称
- NetworkUtilization
- NumLockEnabled
- ObjectType
- 所有者
- RelativeWeight
- ServerConnection
- CEIPOptIn
- CPUPriority
- Channel
- CompanyName
- DatabaseInstanceName
- DatabaseName
- DatabaseServerName
- DiskIOPriority
- EvaluationDaysLeft
- FQDN
- FullyQualifiedDomainName
- IsConnected
- IsEvaluationVersion
- IsWorkgroupEdition
- LibraryRefresherEnabled
- LibraryRefresherFrequency
- MOMReportingEnabled
- MOMReportingServerURL
- MemoryPriority
- MinimumSupportedAgentVersion
- NetworkPriority
- ObjectCache
- OpsMgrReportingEnabled
- OpsMgrReportingServerURL
- OpsMgrServer

- PROAutomationLevel
- PROMonitoringLevel
- PhysicalAddressRangeEnd
- PhysicalAddressRangeStart
- PlacementGoal
- ProductID
- ProductVersion
- 配置文件
- SelfServiceContactEmail
- ServerInterfaceVersion
- UserName
- VMConnectDefaultPort
- VMRCAccessAccount
- VMRCDefaultPort
- ShareSCSIBus
- UndoDisksEnabled
- VirtualCOMPorts[]
- VirtualDVDDrives[]
- VirtualFloppyDrive
- ServerConnection
- VirtualFloppyDisk
- VirtualNetworkAdapters[]
- VirtualSCSIAdapters[]

#### 示例：显示所有 **SCVMM** 硬件配置文件

本示例显示 SCVMM 库服务器 SCVMMLibSrv 上的硬件配置文件。

```
dpmhv-showSCVMMHardwareProfiles  
-scvmmHost SCVMMLibSrv
```

#### 示例：显示多个 **CPU** 的硬件配置文件

本示例显示为多个 CPU 核心定义的所有硬件配置文件。

```
profs = dpmhv-showSCVMMHardwareProfiles -scvmmHost SCVMM -retval -silent  
for each(prof in profs)  
{  
    if(prof.CPUCount > 1)  
        ? prof.Name, "has more than 1 CPU cores"  
}
```

## dpmhv-ShowSCVMMOSProfiles (Funclet) 命令-显示所有来宾 OS 配置文件

dpmhv-ShowSCVMMOSProfiles 命令用于显示所有来宾操作系统 (OS) 配制文件，使用 SCVMM 模板创建 VM 时，这些配置文件可用于 VM。

此命令具有以下格式：

```
dpmhv-showSCVMMOSProfiles  
-scvmmHost scvmmHostname  
[-retval]  
[-silent]  
[-detail]
```

### **-scvmmHost *scvmmHostname***

（可选）指定 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器的主机名。使用 SCVMM 集成开通 VM 时，该参数有效。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

### **-detail**

（可选）显示目录列表的详细信息。

## 返回值

dpmhv-ShowSCVMMOSProfiles 命令返回一组对象, 这些对象表示指定 SCVMM 库服务器上的来宾 OS 配置文件。

以常规模式返回下列属性:

- 管理
- AdminPasswordHasValue
- ComputerName
- 描述
- DomainAdmin
- DomainAdminPasswordHasValue
- JoinDomain
- JoinWorkgroup
- 名称
- OperatingSystem
- OrgName
- ProductKeyHasValue

以详细模式返回下列属性:

- Accessibility
- AddedTime
- 管理
- AdminPasswordHasValue
- ComputerName
- 描述
- DomainAdmin
- DomainAdminPasswordHasValue
- 已启用
- FullName
- GuiRunOnceCommands
- ID
- IsFullyCached
- JoinDomain
- JoinWorkgroup
- MarkedForDeletion
- MergeAnswerFile
- ModifiedTime

- MostRecentTask
- 名称
- ObjectType
- OperatingSystem
- AllowsOrgNameInSysprep
- 体系结构
- 描述
- Edition
- ID
- IsCustomizationAllowed
- IsFullyCached
- MarkedForDeletion
- 名称
- ProductType
- RequiresAdministratorAccountNameInSysprep
- RequiresPIDInSysprep
- RequiresXMLSysprepFormat
- ServerConnection
- 版本
- OrgName
- 所有者
- ProductKeyHasValue
- ServerConnection
- CEIPOptIn
- CPUPriority
- Channel
- CompanyName
- DatabaseInstanceName
- DatabaseName
- DatabaseServerName
- DiskIOPriority
- EvaluationDaysLeft
- FQDN
- FullyQualifiedDomainName
- IsConnected

- IsEvaluationVersion
- IsWorkgroupEdition
- LibraryRefresherEnabled
- LibraryRefresherFrequency
- MOMReportingEnabled
- MOMReportingServerURL
- MemoryPriority
- MinimumSupportedAgentVersion
- 名称
- NetworkPriority
- ObjectCache
- OpsMgrReportingEnabled
- OpsMgrReportingServerURL
- OpsMgrServer
- PROAutomationLevel
- PROMonitoringLevel
- PhysicalAddressRangeEnd
- PhysicalAddressRangeStart
- PlacementGoal
- ProductID
- ProductVersion
- 配置文件
- SelfServiceContactEmail
- ServerInterfaceVersion
- UserName
- VMConnectDefaultPort
- VMRCAccessAccount
- VMRCDefaultPort
- SysprepScript
- TimeZone

**示例：显示所有来宾 OS 配置文件**

本示例显示 SCVMM 库服务器 SCVMMLibSrv 上的来宾 OS 配置文件：

```
dpmhv -showSCVMMOSProfiles  
-scvmmHost SCVMMLibSrv
```

### 示例：显示没有密码的所有来宾 OS 配置文件

本示例显示没有管理员密码的所有来宾 OS 配置文件。

```
profs = dpmhv-showSCVMMOSProfiles -scvmmHost SCVMMLibSrv -retval -silent
for each(prof in profs)
{
    if(prof.AdminPasswordHasValue=="false")
        ? prof.Name,"does not have an Admin password defined"
}
```

## dpmhv-ShowTemplates (Funclet) 命令-显示 Hyper-V 主机上的模板

dpmhv-ShowTemplates 命令用于返回 Hyper-V 主机或 Microsoft System Center Virtual Machine Manager (SCVMM) 库服务器上的模板列表。模板是在模板目录中管理的已导出 VM，用于开通新的 VM。

此命令具有以下格式：

```
dpmhv-showTemplates
[-host hostname]
[-scvmmHost scvmmHost]
[-detail]
[-retval]
[-silent]
```

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-scvmmHost *scvmmHost***

（可选）指定 SCVMM 库服务器的名称。该参数返回使用指定库服务器上的 SCVMM，在外部创建和维护的模板。使用 SCVMM 集成开通 VM 时，该参数有效。

### **-detail**

（可选）显示模板列表的详细信息。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

## 返回值

dpmhv-ShowTemplates 命令返回一组表示模板的对象。

以常规模式返回下列属性：

- ElementName
- 名称

以详细模式返回下列属性：

- DiskspaceRequired
- ElementName
- MemoryAllocation
- 名称
- NetworkInterfaceAddresses
- NetworkInterfaces
- NumberOfProcessors
- TemplateCreationDate
- TemplateDescription
- TemplateFileSystemPath
- TemplateSourceVM

### 示例：获取 VM 模板列表

本示例返回 VM 模板的列表。

```
dpmhv-showTemplates  
-host hvserver
```

### 示例：获取模板和进程的详细信息

本示例返回包含模板和进程详细信息的模板的列表。

```
tt = dpmhv-showTemplates -host hvserver -silent -retval  
for each(t in tt)  
? t.ElementName, t.Description
```



## dpmhv-ShowVMDisks (Funclet) 命令-显示所有虚拟磁盘

dpmhv-ShowVMDisks 命令用于显示 Hyper-V 主机服务器上 VM 中的所有虚拟磁盘。

此命令具有以下格式：

```
dpmhv-showVMDisks
-vm vmname|-vmid vmguid
-host hostname
[-retval]
[-silent]
[-detail]
```

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

### **-detail**

（可选）显示虚拟磁盘列表的详细信息。

## 返回值

dpmhv-ShowVMDisks 命令返回一组对象，这些对象表示连接到 VM 的磁盘。

以常规模式返回下列属性：

- 标题
- 连接
- ElementName
- HostResource
- ResourceSubType

以详细模式返回下列属性：

- Address
- AllocationUnits
- AutomaticAllocation
- AutomaticDeallocation
- 标题
- 连接
- ConsumerVisibility
- 描述
- ElementName
- HostResource
- InstanceID
- Limit
- MappingBehavior
- OtherResourceType
- 父项
- 路径
- PoolID
- Reservation
- ResourceSubType
- ResourceType
- VirtualQuantity
- VirtualSystemIdentifiers
- Weight

### 示例：显示 VM 的所有虚拟磁盘

本示例返回连接到 VM 的虚拟磁盘的列表。

```
dpmhv-showVMDisks
-host hvserver
-vm TestVM
```

### 示例：显示所有磁盘和虚拟硬盘映像文件

本示例返回连接到 VM 的磁盘和虚拟硬盘映像文件的列表。

```
dd = dpmhv-showVMDisks -host hvserver -vm TestVM -detail -retval -silent
for each(d in dd)
{
  if(d.ResourceSubType=="Microsoft Virtual Hard Disk")
  {
    ? d.Connection
  }
}
```

## dpmhv-showVMFloppy (Funclet) 命令-显示软盘驱动器的属性

dpmhv-showVMFloppy 命令用于检索有关连接到 Hyper-V 主机上每个 VM 的所有软盘驱动器的信息。

此命令具有以下格式：

```
dpmhv-showVMFloppy
-vm vmname|-vmid vmguid
-host hostname
[-retval]
[-silent]
[-detail]
```

#### **-vm *vmname***

指定 VM 的名称。

#### **-vmid *vmguid***

指定 VM 的唯一 ID。

#### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

#### **-retval**

返回一个值以进一步处理。

#### **-silent**

指定不输出到屏幕。

#### **-detail**

（可选）显示 VM 的软盘驱动器的详细信息。

## 返回值

dpmhv-showVMFloppy 命令返回表示 VM 软盘驱动器属性详细信息的对象。

以常规模式返回下列属性：

- 标题
- 连接
- ElementName

以详细模式返回下列属性：

- Address
- AllocationUnits
- AutomaticAllocation
- AutomaticDeallocation
- 标题
- 连接
- ConsumerVisibility
- 描述
- ElementName
- HostResource
- InstanceID
- Limit
- MappingBehavior
- OtherResourceType
- 父项
- 路径
- PoolID
- Reservation
- ResourceSubType
- ResourceType
- VirtualQuantity
- VirtualSystemIdentifiers
- Weight

### 示例：显示 VM 的所有软盘驱动器

本示例返回连接到 VM 的软盘驱动器的列表。

```
dpmhv-showVMFloppy -host hvserver -vm TestVM
```

## dpmhv-showVMNICs (Funclet) 命令-显示所有网络接口卡

dpmhv-showVMNICs 命令用于显示 Hyper-V 主机上 VM 的所有网络接口卡 (NIC)。

此命令具有以下格式：

```
dpmhv-showVMNICs
-vm vmname|-vmid vmguid
-host hostname
[-retval]
[-silent]
[-detail]
```

### **-vm *vmname***

指定 VM 的名称。

### **-vmid *vmguid***

指定 VM 的唯一 ID。

### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

### **-retval**

返回一个值以进一步处理。

### **-silent**

指定不输出到屏幕。

### **-detail**

（可选）显示 NIC 列表的详细信息。

## 返回值

dpmhv-showVMNICs 命令返回对象。

以常规模式返回下列属性：

- Address
- ElementName
- StaticMacAddress

以详细模式返回下列属性：

- Address
- AllocationUnits
- AutomaticAllocation
- AutomaticDeallocation
- 标题
- 连接
- ConsumerVisibility
- 描述
- ElementName
- HostResource
- InstanceID
- Limit
- MappingBehavior
- OtherResourceType
- 父项
- 路径
- PoolID
- Reservation
- ResourceSubType
- ResourceType
- StaticMacAddress
- VirtualQuantity
- VirtualSystemIdentifiers
- Weight

### 示例：显示 VM NIC 信息

本示例显示主机“hvserver”上 VM “TestVM”的 NIC 信息。

```

dpmhv-showVMNICs -host hvserver -vm TestVM
//查找 VM 的第一个适配器连接到的虚拟网络
anic = dpmhv-showVMNICs -host hvserver -vm TestVM -detail -silent -retval
lines = anic[0].Connection.split(",");
for each(line in lines)
{
    if(line.startsWith("SystemName="))
    {
        nsw = line.substr(11);
        nsw = nsw.trim('');
        break;
    }
}
// 获取交换机
asw = dpmhv-getHostSwitches -host hvserver -silent -retval
// 查找匹配项
for each(sw in asw)
{
    if(sw.Name == nsw)
    {
        ?? "NIC",anic[0].ElementName
        ? " is connected to switch", sw.ElementName
    }
}

```

### dpmhv-showVMSCSI (Funclet) 命令-显示所有 SCSI 控制器

dpmhv-showVMSCSI 命令用于显示 Hyper-V 主机上 VM 中的所有 SCSI 控制器。您可以将最多四个 SCSI 控制器分配给 Hyper-V 服务器。

此命令具有以下格式：

```

dpmhv-showVMSCSI
-vm vmname|-vmid vmguid
-host hostname
[-retval]
[-silent]
[-detail]

```

#### **-vm *vmname***

指定 VM 的名称。

#### **-vmid *vmguid***

指定 VM 的唯一 ID。

#### **-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**-detail**

（可选）显示 SCSI 控制器列表的详细信息。



## 返回值

dpmhv-showVMSCSI 命令返回一组对象，这些对象表示指定 VM 的 SCSI 控制器。

以常规模式返回下列属性：

- 标题
- ElementName

以详细模式返回下列属性：

- Address
- AllocationUnits
- AutomaticAllocation
- AutomaticDeallocation
- 标题
- 连接
- ConsumerVisibility
- 描述
- ElementName
- HostResource
- InstanceID
- Limit
- MappingBehavior
- OtherResourceType
- 父项
- 路径
- PoolID
- Reservation
- ResourceSubType
- ResourceType
- VirtualQuantity
- VirtualSystemIdentifiers
- Weight

**示例：显示所有 SCSI 控制器**

本示例显示主机“hvserver”上 VM “TestVM”的所有 SCSI 控制器的基本详细信息。

```
dpmhv-showVMSCSI  
-host hvserver  
-vm TestVM
```

**示例：显示所有 SCSI 控制器的详细信息**

本示例显示主机“hvserver”上 VM “TestVM”的 SCSI 控制器的详细信息。

```
dpmhv-showVMSCSI -host hvserver -vm TestVM -detail
```

**dpmhyperv ShowVMs (Funclet) 命令-获取所有 VM**

dpmhv-showVMs 命令用于返回 Hyper-V 主机服务器上 VM 的列表。该命令获取指定 VM 的名称和相应的 GUID。

此命令具有以下格式：

```
dpmhv-showVMs  
-host hostname[-retval]  
[-silent]
```

**-host *hostname***

指定 Hyper-V 服务器主机的名称。

**-retval**

返回一个值以进一步处理。

**-silent**

指定不输出到屏幕。

**返回值**

dpmhv-showVMs 命令返回一组对象，这些对象表示指定主机上的 VM。

该对象包括以下属性：

**guid**

表示 VM 的唯一 ID。

**name**

表示 VM 的显示名称。

**示例：获取所有 VM**

本示例返回主机“hvserver”上的 VM 及其相应 GUID 的列表。

```
dpmhv-showVMs
-host hvserver
```

**示例：显示所有 VM 的 VM 属性**

本示例显示主机“hvserver”上所有 VM 的属性。

```
a = dpmhv-showVMs -host hvserver -retval -silent
for each(vm in a)
{
?
? "VM:", vm.ElementName
dpmhv-getVMProperties -host hvserver -vmid (vm.Name)
}
```

## CA IBM LPAR AutoShell 命令

可以使用 AutoShell 编写 CA IBM LPAR 命令的脚本并使其自动化，然后根据命令的结果运行操作。CLI 中也提供了相应的命令。

### dpmlpar-cycle 命令—循环使用逻辑分区 (Funclet)

dpmlpar-cycle 命令用于打开、关闭、重置或挂起逻辑分区。

此命令具有以下格式：

```
dpmlpar-cycle
-powerop {activate|restart|shutdown}
-hmc name
-managed_system managedsystemname
-partition_name partitionname
[-type {delayed|immediate|os_shutdown|immediate_os_shutdown}]
[-profile_name profilename]
[-activate_bootmode {normal|open_firmware}]
[-activate_keylock {normal|manual}]
[-verbose add_commandinfo]
[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

**-powerop {activate|restart|shutdown}**

指定要在 LPAR 上执行的电源操作。选项包括：

**activate**

打开 LPAR。

**restart**

在必要时关闭 LPAR，然后再将其打开。

**shutdown**

关闭 LPAR。

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {immediate|os\_shutdown|immediate\_os\_shutdown}**

指定使用映像操作类型资源组 (*res\_group*) 或单个资源 (*individual\_res*)。选项包括：

**immediate**

立即关闭分区。如果只更新部分数据，该选项可能导致不适当的结果。

**os\_shutdown**

通过发出用于典型关闭的命令来关闭分区。必须映像分区，才能成功执行该选项。

**immediate\_os\_shutdown**

通过发出操作系统命令以尽快关闭系统来关闭分区。该命令会跳过典型的关闭活动（包括向其他用户发送消息）。必须映像分区，才能成功执行该选项。

**-profile\_name *lparprofile* (仅 HMC)****[-profile\_name *lparname*] (仅 IVM)**

指定激活逻辑分区时要使用的分区配置文件。对于 HMC 为必需。对于 IVM 为可选，如果已指定，该名称必须与分区名称匹配。

**-activate\_bootmode {normal|open\_firmware}**

（可选）指定激活操作的键盘锁定模式。选项包括：

**normal**

以典型方式启动分区。

**open\_firmware**

启动分区并打开“打开固件提示”。

**-activate\_keylock {normal|manual|bypass}**

（可选）指定激活操作的键盘锁定模式。选项包括：

**normal**

以无人照管模式启动分区，激活过程中无需用户交互。

**手工**

以照管模式启动分区，激活期间需要用户交互。

**bypass**

不激活键盘锁定模式。

**-verbose add\_commandinfo**

提供有关如何执行命令的其他信息。

**-sc URL**

（可选）指定服务控制器的 URL。

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**示例：使用默认配置文件激活逻辑分区**

本示例使用默认配置文件激活逻辑分区“testlpar”。

```
dplpar-cycle -powerop activate -hmc uslihmc -managed_system testComputer  
-partition_name testlpar -profile_name testlparprofile
```

### 示例：使用指定的配置文件激活逻辑分区

本示例使用配置文件 `testlparprofile` 激活逻辑分区 “`testlpar`”。

```
dpmlpar-cycle -powerop activate -hmc uslihmc -managed_system testComputer
-partition_name testlpar -profile_name testlparprofile
```

### 示例：激活逻辑分区以打开固件提示

本示例使用配置文件 `testlparprofile` 激活逻辑分区 “`testlpar`” 以打开固件提示。

```
dpmlpar-cycle -powerop activate -hmc usliivm -managed_system testMS
-partition_name testlpar -activate_bootmode open_firmware
```

### 示例：延迟关闭逻辑分区

本示例执行逻辑分区 “`testlpar`” 的延迟关闭。

```
dpmlpar-cycle -hmc testivm -powerop shutdown -managed_system managedsys1
-partition testlpar -type delayed
```

## dpmlpar-delete 命令—删除逻辑分区 (Funclet)

`dpmlpar-delete` 命令用于删除处于停用状态的逻辑分区。

**重要信息！** 在发出该命令之前，请验证是否已备份所有重要数据。该命令删除 LPAR 及其数据文件。在您发出该命令时，必须关闭 LPAR。

此命令具有以下格式：

```
dpmlpar-delete
-hmc name
-managed_system managedsystemname
-partition_name partitionname
[-verbose add_commandinfo]
[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

#### **-hmc name**

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

#### **-managed\_system managedsystemname**

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

#### **-partition\_name partitionname**

指定要删除的逻辑分区的名称。该分区必须存在于受管系统上。

**-verbose *add\_commandinfo***

提供有关如何执行命令的其他信息。

**-sc *URL***

(可选) 指定服务控制器的 URL。

**-ws\_remote\_user *username***

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

(可选) 指定 CA Virtual Assurance 远程密码。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**示例：删除逻辑分区**

本示例删除逻辑分区 lpar01。

```
dpmlpar-delete -hmc hmc02 -managed_system system05 -partition_name lpar01
```

**dpmlpar-getresources 命令—获取 LPAR 资源 (Funclet)**

dpmlpar-getresources 命令用于检索 IBM AIX LPAR 中的内存单元、处理器单元或处理器。

此命令具有以下格式：

```
dpmlpar-getresources
-resource {all|memory|processors|processor_units}
-hmc name
-managed_system managedsystemname
-partition_name partitionname
[-verbose add_commandinfo]
[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

**-resource {all|memory|processors|processor\_units}**

指定是检索特定资源还是所有资源。选项包括：

**all**

检索并显示所有分区资源。

**memory**

仅检索并显示分区的内存资源。

**processors**

仅检索并显示分区的处理器资源。

**processor\_units**

仅检索并显示分区的处理器单元资源。

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定为其列出资源的逻辑分区。该分区必须存在于受管系统上。

**-verbose *add\_commandinfo***

提供有关如何执行命令的其他信息。

**-sc *URL***

(可选) 指定服务控制器的 URL。

**-ws\_remote\_user *username***

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

(可选) 指定 CA Virtual Assurance 远程密码。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**示例：获取 LPAR 的所有资源**

本示例检索 lpar06 的所有资源。

```
dpm_lpar-getresources -resource all -partition_name lpar06  
-hmc hmc_02 -managed_system abcsystem4
```



**示例：获取 LPAR 的内存资源**

本示例检索 lpartest01 的内存资源。

```
dpmlpar-getresources -resource memory -partition_name lpartest01
hmc ivm_02 -managed_system abcsystem2
```

**dpmlpar-image 命令—创建 IBM AIX LPAR (Funclet)**

dpmlpar-image 命令用于使用 HMC 和目标 IBM AIX 受管系统创建 IBM AIX LPAR。可以选择在创建 LPAR 之后部署 IBM AIX 操作系统。

此命令具有以下格式：

```
dpmlpar-image
-hmc name -managed_system managedsystemname
-partition_name partitionname
-profile_name profilename
-min_mem size
-desired_mem size
-max_mem size

-proc_mode {shared|dedicated}
-min_proc_units units
-desired_proc_units units
-max_proc_units units
-min_procs number
-desired_procs number
-max_procs number

-max_virtual_slots number
-share_mode {capped|uncapped}
[-uncap_weight weight]
[-io_slot DRC index, {true|false}]
[-lpar_io_pool_id id {id, ...}]

[-virtual_serial_adapter slot_num, remote_lpar_name, remote_slot_num,
is_required]
[-virtual_scsi_client_adapter slot_num, remote_lpar_name, remote_slot_num,
is_required] |
[-virtual_scsi_server_client_adapter virtual_io_server_name,
backing_device_name, client_slot_num, is_required]
[-virtual_eth_adapter slot_num, is_IEEE, port_vlan_id, (additional_vlan_id,
additional_vlan_id, ...), trunk_priority, is_required]

[-virtual_fc_server_client_adapter virtual_io_server_name,
physical_fc_port_name, client_slot_num, is_required
[virtual_io_server_name, physical_fc_port_name, client_slot_num, is_required; ...
]]
[-verbose add_commandinfo]
```

```
[-create_logical_volume <false|true>]
[-logical_volume_size <value>]
[-volume_group_names <value>]
[-logical_volume_name <value>]
[-use_logical_volume_name_as_prefix {no|yes}]
[-default_vio_server_name vioserver]

[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

用于通过资源组开通 AIX 的其他参数

```
-provision_aix true
-type res_group
-res_group_name resourcegroupname
-machine_res_name machineresourcenam
-target_username targetusername
[-target_password targetpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
-nim_master_host_name nimmasterhostname
[-scalability_server servername]
[-deploy_template templatename]
-auto_deploy {yes|no}
[-wait [timeout]]
```

用于通过单个资源开通 AIX 的其他参数

```
-provision_aix true
-type individual_res
-machine_res_name machineresourcenam
-lpp lppresource
-spot spotresource
-bosinst_data bosdata
-resolv_conf resolveconf
-fb_script fbscript
-post_inst_scripts script1,script2,script3
-target_username targetusername
[-target_password targetpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
-nim_master_host_name nimmasterhostname
[-scalability_server servername]
[-deploy_template templatename]
-auto_deploy {yes|no}
[-wait [timeout]]
```

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要创建的逻辑分区的名称。

**-profile\_name *lparprofile***（仅 HMC）**[-profile\_name *lparname*]**（仅 IVM）

指定激活逻辑分区时要使用的分区配置文件。对于 HMC 为必需。对于 IVM 为可选，如果已指定，该名称必须与分区名称匹配。

**-min\_mem *size***

定义分区的最小内存量。

**-desired\_mem *size***

定义分区所需的内存量。

**-max\_mem *size***

定义分区的最大内存量。

**-proc\_mode {*shared* | *dedicated*}**

指定分区的处理器模式的类型。选项包括：

***shared***

与其他分区共享处理器资源。

***dedicated***

指定分区具有专用处理器资源。

**-min\_proc\_units *units***

定义该分区的最小处理单元数。

限制：.01 增量

注意：该选项仅适用于共享处理器。

**-desired\_proc\_units *units***

定义为该分区分配的处理单元的数量。

限制：.01 增量

注意：该选项仅适用于共享处理器。

**-max\_proc\_units *units***

定义该分区的处理单元的最大数量。

限制：.01 增量

注意：该选项仅适用于共享处理器。

**-min\_procs *number***

定义分区的虚处理器的最小数量。

**注意：**该选项仅适用于共享处理器。

**-desired\_procs *number***

定义为该分区分配的虚处理器的数量。

**注意：**该选项仅适用于共享处理器。

**-max\_procs *number***

定义该分区的虚拟处理器的最大数量。

**注意：**该选项仅适用于共享处理器。

**-shared\_mode {capped|uncapped}**

指定受管系统是否允许逻辑分区使用共享处理器池中未指派给其他分区的空闲处理单元。

**注意：**该选项仅适用于共享处理器。

选项包括：

**capped**

指定分区只能使用指派给它的特定数量的处理单元。

**uncapped**

指定分区可以使用共享处理器池中空闲的处理器单元（当它们可用时）。

**-uncap\_weight *weight***

（可选）定义在选择不受限共享模式时处理优先级的加权平均值。

**限制：**0 - 255

**-max\_virtual\_slots *number***

定义该分区的虚拟适配器的最大数量。

**默认值：**2

**限制：**2 - 65,536

**-io\_slot *DRC-Index*,{true|false}**

（可选）指定物理组件的 I/O 插槽。您可以为不同的 I/O 设备多次指定该值。

**DRC-Index**

指定插槽动态重新配置连接器 (DRC) 索引。

**真**

指定 DRC 索引对于 I/O 插槽是必需的。

**假**

指定 DRC 索引对于 I/O 插槽不是必需的。

**-io\_pool\_id *id* {*id* ...}**

（可选）定义 I/O 适配器组，这些适配器可在没有来自 HMC 的任何活动干预的情况下被任何指定的逻辑分区组接管并使用。分区组可以是逗号分隔列表形式的 I/O 池 ID。

**-virtual\_serial\_adapter *slot\_num*, *remote\_lpar\_name*, *remote\_slot\_num*, *is\_required***

（可选）定义该逻辑分区的虚拟串行适配器。变量 *is\_required* 可以设置为 true 或 false。仅支持客户端串行适配器，但可以多次指定。

**默认值：**在插槽 1 和 2 中创建两个服务器串行适配器。

**注意：**不支持在 IVM 上创建 LPAR。

**-virtual\_eth\_adapter\_client *slot\_num*, *is\_IEEE*, *port\_vlan\_id1* (*vlanid2*,*vlanid3*,...), *trunk\_priority*, *is\_required***

（可选）定义该逻辑分区的虚拟以太网适配器，可以多次指定。可选值有 *vlan\_idn* 和 *trunk\_priority*。变量 *is\_IEEE* 和 *is\_required* 可以设置为 true 或 false。其他 *vlan\_ids* 必须使用逗号分隔，并且包含在圆括号内。

**注意：**在 IVM 上创建 LPAR 时，*slot\_num* 必须大于或等于 4。

**示例：**4,false,1,,,true

**-virtual\_scsi\_client\_adapter *slot\_num*, *remote\_lpar\_name*, *remote\_slot\_num*, *is\_required***

（可选）定义该逻辑分区的虚拟客户端 SCSI 适配器，可以多次指定。变量 *is\_required* 可以设置为 true 或 false。

**-virtual\_scsi\_server\_client\_adapter *virtual\_io\_server\_name*,  
*backing\_device\_name*, *client\_slot\_num*, *is\_required***

（可选）定义虚拟 SCSI 服务器设备和虚拟 SCSI 客户端设备，在创建服务器设备并确定服务器插槽号之后自动链接客户端设备。该参数可以多次指定。变量 *is\_required* 可以设置为 true 或 false。

示例：CUST-VIOSERVER,hdisk5,3,true

注意：如果使用 `dpmlpar image` 命令创建逻辑卷，并将 `-create_logical_volume` 选项设置为“是”，则 `-virtual_scsi_server_client_adapter` 选项不是必需的，应忽略。

**-virtual\_fc\_server\_client\_adapter *virtual\_io\_server\_name*,  
*physical\_fc\_port\_name*, *client\_slot\_num*, *is\_required*  
[;*virtual\_io\_server\_name*, *physical\_fc\_port\_name*, *client\_slot\_num*,  
*is\_required*;...]**

（可选）在 VIO 服务器上定义虚拟光纤通道 (FC) 客户端适配器、FC 服务器适配器，并将 FC 服务器适配器与物理 FC 端口关联。将 `-provision_aix` 选项设置为 true 时，该选项无效。该参数可以多次指定。变量 *is\_required* 可以设置为 true 或 false。

**-create\_logical\_volume {no|yes}**

（可选）指定是否创建逻辑卷。

默认值：no

注意：如果使用 `dpmlpar image` 命令创建逻辑卷，并将 `-create_logical_volume` 选项设置为“是”，则 `-virtual_scsi_server_client_adapter` 选项不是必需的，应忽略。

**-logical\_volume\_size *logical\_volume\_size***

（可选）定义逻辑卷的大小 (MB)。

**-volume\_group\_names *volume\_group\_1*, *volume\_group\_2*,  
*volume\_group\_3***

（可选）以逗号分隔的列表形式指定逻辑卷组名称。

**-logical\_volume\_name *logical\_volume\_name***

（可选）定义逻辑卷的名称。

**-use\_logical\_volume\_name\_as\_prefix {no|yes}**

（可选）指定是否将逻辑卷名称用作前缀。

默认值：no

**-default\_vio\_server\_name *vioserver***

（可选）指定用于创建逻辑卷的 VIO 服务器。

**-itcm\_server *itcm\_servername***

（可选）指定 Unicenter DSM 服务器的名称。

**-sc *URL***

（可选）指定服务控制器的 URL。

**-ws\_remote\_user *username***

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

（可选）指定 CA Virtual Assurance 远程密码。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

#### 用于开通的必需参数

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-auto\_deploy {*yes|no*}**

指定是否自动部署 CA Virtual Assurance 代理。选项包括：

**是**

自动部署 CA Virtual Assurance 代理。

**否**

防止自动部署 CA Virtual Assurance 代理。

**默认值：** no

**-bosinst\_data bos\_install\_data\_resource**

指定包含基础操作系统 (BOS) 安装程序的信息的文件。如果 -install\_type 为 mksysb, 则为可选。如果 -install\_type 为 rte, 则为必需。

**-deploy\_template templatename**

(可选) 指定在 CA Virtual Assurance 中创建的通用模板的名称。

**注意:** 请勿将该模板与由 VMware vCenter 创建并管理的模板相混淆。

**-fb\_script fbscript**

(可选) 定义在 BOS 安装过程完成之后, NIM 客户端初始启动时用于配置设备的文件的名称。

**-lpp lpp\_resource**

指定许可的程序产品文件, 以用于映像请求。如果 -install\_type 为 mksysb, 则为可选。如果 -install\_type 为 rte, 则为必需。

**-machine\_res\_name machineresourcenam**

定义计算机资源的名称。该名称必须在 NIM 主服务器中预定义。

**-nim\_master\_host\_name nimmasterhostname**

定义用于执行映像部署的 NIM 主服务器主机名。

**-post\_inst\_scripts script1,script2,script3**

(可选) 指定安装后运行的逗号分隔脚本列表。

**-provision\_aix {true|false}**

表示创建分区之后, 是否使用 NIM 对其进行映像。

**真**

使用 NIM 对创建的分区进行映像。如果您将其设置为 true, 则使用 NIM 资源组或 NIM 单个资源参数。请参阅 `dpmnim image` 命令 | 使用资源组和 `dpmnim image` 命令部署 IBM AIX 映像 | 使用单个资源部署 IBM AIX 映像。NIM 开通开始时, 返回作业 ID。

**假**

不使用 NIM 对创建的分区进行映像。不返回作业 ID。

**-res\_group\_name resource group name**

定义资源组的名称。

**-resolve\_conf resolveconf**

(可选) 定义一个文件, 该文件包含为本地解析程序例程定义域名协议名称服务器信息的有效 `/etc/resolv.conf` 条目。



**-scalability\_server servername**

(可选) 指定要用于软件分发的 Unicenter DSM 可扩展性服务器。

**-spot spotResource [mkysbResource]**

定义共享产品对象树，以用于映像制作请求。

**-target\_password targetpassword**

(可选) 定义用于将代理部署到的目标主机服务器（要将映像部署到该服务器）的用户密码。如果未指定密码，将从授权文件检索密码。

**注意：**可使用 dpmutil CLI 设置授权文件。

**-target\_username targetusername**

定义用于将代理部署到的目标主机服务器（要将映像部署到该服务器）的用户名。

**-type {res\_group|individual\_res}**

指定使用映像操作类型资源组或单个资源。

**res\_group**

指定使用资源组进行映像。

**individual\_res**

指定使用单个资源进行映像。

**-wait timeout**

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 caimgconf.cfg 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-verbose add\_commandinfo**

提供有关如何执行命令的其他信息。

### 示例：创建 IBM AIX 逻辑分区

本示例创建逻辑分区，但不开通。

```
dpmlpar-image -hmc ivm01 -managed_system testMS -partition_name lptest01
-profile_name Default -max_virtual_slots 10 -min_mem 128 -desired_mem 256
-max_mem 2048 -min_procs 1 -desired_procs 2 -max_procs 5 -proc_mode shared
-share_mode uncapped -uncap_weight 123 -min_proc_units .5 -max_proc_units 3.25
-desired_proc_units 1.75 -virtual_serial_adapter 2,serial_partition,1,true
-virtual_scsi_client_adapter 3,scsi_partition,7,true
-virtual_eth_adapter 4,true,1,(22,35,54),,true -io_slot 21020003,false
```

### 示例：创建 IBM AIX 逻辑分区并使用单个资源对其进行映像

本示例创建逻辑分区并使用单个资源将其开通。

```
dpmlpar-image -hmc hmc01 -managed_system testMS -partition_name IRTEST
-profile_name Default -max_virtual_slots 10 -min_mem 128 -desired_mem 256
-max_mem 2048 -min_procs 1 -desired_procs 1 -max_procs 1 -proc_mode dedicated
-virtual_serial_adapter 2,SerialPartition,1,true
-virtual_scsi_client_adapter 3,SCSIPartition,7,true
-virtual_eth_adapter 4,true,1,(22,35,54),0,true -io_slot 21020003,false
-provision_aix true -type individual_res -lpp 530lpp_res -spot 530spot_res
-bosinst_data 530_bid_ow -resolv_conf master_net_conf -post_inst_scripts
piScript
-machine_res_name MachineName -nim_master_host_name machine.mydomain.com
-auto_deploy no -target_username root -target_password password
```

### 示例：创建 IBM AIX 逻辑分区并使用资源组对其进行映像

本示例创建逻辑分区并使用资源组将其开通。

```
dpmlpar-image -hmc hmc01 -managed_system testMS -partition_name RGTEST
-profile_name Default -max_virtual_slots 10 -min_mem 128 -desired_mem 256
-max_mem 2048 -min_procs 1 -desired_procs 2 -max_procs 5 -proc_mode dedicated
-virtual_serial_adapter 2,SerialPartition,1,true
-virtual_scsi_client_adapter 3,SCSIPartition,7,true
-virtual_eth_adapter 4,true,1,(22,35,54),,true -io_slot 21020003,false
-provision_aix true -type res_group -res_group_name Res_grp -machine_res_name
Machine
-nim_master_host_name machine.mydomain.com -auto_deploy no
-target_username root -target_password password
```

## dpmlpar-imgjobcheck 命令—检索 IBM AIX 映像作业的状态 (Funclet)

dpmlpar-imgjobcheck 命令用于针对特定的 CA Virtual Assurance 作业 ID 检索 IBM AIX 逻辑分区映像作业的状态。

此命令具有以下格式：

```
dpmlpar-imgjobcheck -status jobID [-verbose add_commandinfo]
```

**-status *jobID***

指定用于获取作业状态的 CA Virtual Assurance 作业 ID。

**-verbose *add\_commandinfo***

提供有关如何执行命令的其他信息。

**示例：使用作业 ID 检索 IBM AIX 映像作业的状态**

本示例使用 CA Virtual Assurance 作业 ID 42 获取 IBM AIX 逻辑分区映像作业的作业状态。

```
dpmlpar-imgjobcheck -status 42
```

**dpmlpar-list 命令—列出 HMC 资源 (Funclet)**

dpmlpar-list 命令用于列出来自硬件管理控制台的信息。

此命令具有以下格式：

```
dpmlpar-list
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-display {managed_systems | managed_system_details | partitions | io_components
| backing_devices | scsi_adapters | partition_details | profiles / wwpns}
[-verbose add_commandinfo]
[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

**-display**

```
{managed_systems | managed_system_details | partitions | io_components |
backing_devices | scsi_adapters | partition_details | profiles | wwpns}
```

指定要列出哪些资源。选项包括：

**managed\_systems**

列出由管理硬件控制台控制的受管系统。

**managed\_system\_details**

提供有关受管系统的详细信息。

**partitions**

列出受管系统上的分区。

**io\_components**

列出受管系统的 I/O 组件。

**backing\_devices**

列出可作为后备设备附加到服务器虚拟 SCSI 适配器的设备。

**scsi\_adapters**

列出受管系统的 SCSI 适配器。

**partition\_details**

列出分区的详细信息。

**profiles**

列出分区的配置文件的列表。

**wwpns**

显示分区的全球端口名称。

**-hmc name**

指定要为其列出信息的 HMC/IVM。

**-managed\_system managedsystemname**

(可选) 指定想要列出其信息的逻辑分区驻留的受管系统。

**注意：**请不要与 managed\_systems 选项一起使用。

**-partition\_name partitionname**

(可选) 指定您想要列出其信息的逻辑分区。

**注意：**请不要与 managed\_systems 选项一起使用。

**-verbose add\_commandinfo**

提供有关如何执行命令的其他信息。

**-sc URL**

(可选) 指定服务控制器的 URL。

**-ws\_remote\_user username**

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

(可选) 指定 CA Virtual Assurance 远程密码。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**示例：显示受管系统的列表**

本示例显示硬件管理控制台 “labhmc” 的受管系统。

```
dpmlpar-list -display managed_systems -hmc labhmc
```

**示例：显示 SCSI 适配器**

本示例显示受管系统 “testMS” 的 SCSI 适配器。

```
dpmlpar-list -display scsi_adapters -hmc labhmc
-managed_system testMS
```

**示例：显示分区详细信息**

本示例显示逻辑分区 “testlpar” 的详细信息。

```
dpmlpar-list -display partition_details -hmc uslihmc
-managed_system testMS -partition_name testlpar
```

**dpmlpar-setresources 命令—增加内存资源 (Funclet)**

dpmlpar-setresources 命令用于调整 IBM AIX LPAR 中的内存单元。

此命令具有以下格式：

```
dpmlpar-setresources
-add_memory value
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-type {dynamic|all}
[-verbose add_commandinfo]
[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

**-add\_memory value**

定义添加到分区的内存量。

**-hmc name**

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system managedsystemname**

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name partitionname**

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {dynamic|all}**

指定调整是临时的还是永久的。选项包括：

**dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

**all**

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-verbose add\_commandinfo**

提供有关如何执行命令的其他信息。

**-sc URL**

（可选）指定服务控制器的 URL。

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**示例：向逻辑分区添加内存**

本示例向逻辑分区“lptest01”中添加 128 MB 的内存。

```
dpmlpar-setresources -add_memory 128 -partition_name lptest01  
-hmc uni02 -managed_system usil01system1 -type dynamic
```

**示例：向逻辑分区添加内存并更新配置文件**

本示例向逻辑分区“lptest01”中添加 128 MB 的内存并更新分区配置文件。

```
dpmlpar-setresources -add_memory 128 -partition_name lptest01  
-hmc uni02 -managed_system usil01system1 -type all -profile_name lptest01
```

## dpmlpar-setresources 命令—增加处理器资源 (Funclet)

dpmlpar-setresources 命令用于为 IBM AIX LPAR 增加处理器和处理器单元。

此命令具有以下格式：

```
dpmlpar-setresources
{-add_processors|add_processor_units} value
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-type {dynamic|all}
[-verbose add_commandinfo]
[-sc URL]
[-ws_remote_user username]
[-ws_remote_password password]
[-pre]
[-post]
```

### **-add\_processors value**

定义添加到分区的处理器的数量。

### **-add\_processor\_units value**

定义添加到分区的处理器单元。

限制：.01 增量

### **-hmc name**

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

### **-managed\_system managedsystemname**

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

### **-partition\_name partitionname**

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

### **-type {dynamic|all}**

指定调整是临时的还是永久的。选项包括：

#### **dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

#### **all**

调整当前资源和当前配置文件值。将为分区保存调整的值。

### **-verbose add\_commandinfo**

提供有关如何执行命令的其他信息。

**-sc URL**

（可选）指定服务控制器的 URL。

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**示例：为分区添加处理器并更新配置文件**

本示例向分区“lptest01”中添加处理器并更新分区配置文件。

```
dpmlpar-setresources -add_processors 1 -partition_name lptest01  
-hmc uni02 -managed_system MSystem1 -type all -profile_name lptest01
```

**示例：为分区添加处理器和处理器单元**

本示例为分区“lptest01”暂时添加处理器和处理器单元资源。

```
dpmlpar-setresources -add_processors 1 -add_processor_units 4.5  
partition_name lptest01 -hmc uni02 -managed_system MSystem1 -type dynamic
```

**dpmlpar-setresources 命令—减去内存资源 (Funclet)**

dpmlpar-setresources 命令用于删除 IBM AIX LPAR 中的内存单元。

此命令具有以下格式：

```
dpmlpar-setresources  
-subtract_memory value  
-hmc name  
-managed_system managedsystemname  
-partition_name partitionname  
-type {dynamic|all}  
[-verbose add_commandinfo]  
[-sc URL]  
[-ws_remote_user username]  
[-ws_remote_password password]  
[-pre]  
[-post]
```

**-subtract\_memory value**

定义要从分区删除的内存量。



**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {*dynamic*|*all*}**

指定调整是临时的还是永久的。选项包括：

**dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

**all**

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-verbose *add\_commandinfo***

提供有关如何执行命令的其他信息。

**-sc *URL***

(可选) 指定服务控制器的 URL。

**-ws\_remote\_user *username***

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

(可选) 指定 CA Virtual Assurance 远程密码。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**示例：减少逻辑分区中的内存**

本示例从逻辑分区 “lptest01” 中减少 128 MB 的内存。

```
dpm\par-setresources -subtract_memory 128 -partition_name lptest01  
-hmc con02 -managed_system MSystem1 -type dynamic
```

**示例：减少逻辑分区中的内存并更新配置文件**

本示例从逻辑分区“lptest01”中减少 128 MB 的内存并更新分区配置文件。

```
dpmlpar-setresources -subtract_memory 128 -partition_name lptest01  
-hmc con02 -managed_system MSystem1 -type all
```

**dpmlpar-setresources 命令—减去处理器资源 (Funclet)**

dpmlpar-setresources 命令用于从 IBM AIX LPAR 删除处理器和处理器单元。

此命令具有以下格式：

```
dpmlpar-setresources  
{-subtract_processors|subtract_processor_units} value  
-hmc name  
-managed_system managedsystemname  
-partition_name partitionname  
-type {dynamic|all}  
[-verbose add_commandinfo]  
[-sc URL]  
[-ws_remote_user username]  
[-ws_remote_password password]  
[-pre]  
[-post]
```

**-subtract\_processors value**

定义要从分区删除的处理器数量。

**-subtract\_processor\_units value**

定义要从分区删除的处理器单元。

限制：.01 增量

**-hmc name**

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system managedsystemname**

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name partitionname**

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {dynamic|all}**

指定调整是临时的还是永久的。选项包括：

**dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

**all**

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-verbose add\_commandinfo**

提供有关如何执行命令的其他信息。

**-sc URL**

（可选）指定服务控制器的 URL。

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**示例：减少分区中的处理器并更新配置文件**

本示例从分区“lptest01”中减少一个处理器并更新分区配置文件

```
dpmldpar-setresources -subtract_processors 1 -partition_name lptest01
-hmc hmc02 -managed_system MSystem1 -type all
```

**示例：减少分区中的处理器和处理器单元**

本示例暂时调整分区“lptest01”中的处理器和处理器单元资源。

```
dpmldpar-setresources -subtract_processors 1 -subtract_processor_units 4.5
-partition_name lptest01 -hmc ivm02 -managed_system MSystem1 -type dynamic
```

## CA Oracle Solaris Zones AutoShell 命令

可以使用 AutoShell 编写 CA Oracle Solaris Zones 命令的脚本并使其自动化，然后根据命令的结果运行操作。CLI 中也提供了相应的命令。

## dpmzone-associateproject 命令—将任务与项目关联 (Funclet)

dpmzone-associateproject 命令用于将任务与项目关联起来。项目和任务是用于识别 Solaris 10 操作系统工作负荷的基本实体。项目是任务的集合，任务是进程的集合。每个进程仅属于一个任务，每个任务仅属于一个项目。

此命令具有以下格式：

```
dpmzone-associateproject
-host hostname -name zonename
-proj_name pname -task_id tid
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定提供项目的区域。

### **-proj\_name *pname***

指定项目。

### **-task\_id *tid***

指定任务的 ID 号。

## 示例

将任务与项目关联：

```
dpmzone-associateproject -host SolarisServer2 -name myzone1 -project_name
myproject1
-task_id 1954
```

## dpmzone-clonezone 命令—克隆区域 (Funclet)

dpmzone-clonezone 命令用于基于现有区域创建重复区域。源区域必须暂停才能启动克隆进程。克隆将在 Solaris 10 服务器上运行，且可能需要花费一些时间。

此命令具有以下格式：

```
dpmzone-clonezone
-host hostname
-name zonename
-new_name nzonename
-new_path npath
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要克隆的区域。

**-new\_name *nzonename***

定义新的区域名称。

**-new\_path *npath***

定义从全局区域到当前区域的路径的名称。

### 示例

在主机 SolarisServer2 上克隆区域：

```
dpmzone-clonezone -host SolarisServer2 -name myzone1 -new_name myzone2
-new_path /opt/zones/myzone2
```

## dpmzone-createandinstallzone 命令--创建并安装区域

createandinstallzone 命令用于使用自定义参数在 Solaris 主机上创建并安装区域。

此命令具有以下格式：

```
dpmzone-createandinstallzone
[-sc sc_host]
-host hostname
-name zonename
-path zonepath
-type type
[-archive_path path]
[-autoboot]
[-if_type name]
[-ip ip]
[-pool_name pool]
[-sched_type sched]
[-phy_mem pmem]
[-swap_mem smem]
[-lock_mem lmem]
[-desc desc]
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

定义新区域的名称。

**-path *zonepath***

定义新区域的路径。

**-type *type***

指定新区域的类型。选项包括：

**native**

使用 Solaris 10 操作环境创建非全局区域来运行应用程序。

**whole-root**

创建不继承软件包的整个根区域。

**branded**

创建包含非本地操作环境的非全局区域来运行应用程序。

**默认值：** native

**-archive\_path *path***

（可选）为标记区域指定操作环境安装程序的路径。

**限制：** 该参数是标记区域所需的参数，本地区域不需要该参数。

**-autoboot**

（可选）指定在系统引导时自动引导区域。

**注意：** 如果在服务器上禁用区域服务，则不论该属性的设置为何，均不能自动引导区域。可通过以下命令启用该服务：

```
svcadm enable svc:/system/zones:default
```

**-if\_type *name***

（可选）指定区域使用的网络接口的类型，例如 eri0。

**-ip *ip***

（可选）定义区域的 IP 地址。

**-pool\_name *pool***

（可选）指定与区域关联的池的名称。

**-sched\_type *shed***

（可选）指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。有效值包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**-phy\_mem *pmem***

（可选）定义分配给区域的物理内存。可对该数值应用单位（K、M、G、T），例如，1M 是一个兆字节。

**-swap\_mem *smem***

（可选）定义分配给区域的交换内存。

**-lock\_mem *lmem***

（可选）定义分配给区域的锁定内存。锁定内存无法分页。

**-desc *description***

（可选）定义区域的说明。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

## dpmzone-createpool 命令—创建资源池 (Funclet)

dpmzone-createpool 命令用于在 Solaris 10 主机上创建资源池。

此命令具有以下格式：

```
dpmzone-createpool
-host hostname
-pset_name pset
-cpu_min mincpu
-cpu_max maxcpu
-pool_name pool
-sched_type sched
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-pset\_name *pset***

定义处理器集。每个处理器集（CPU 组）可以包含零个或多个 CPU。

**-cpu\_min *mincpu***

定义处理器集中 CPU 的最小数目。

**-cpu\_max *maxcpu***

定义处理器集中 CPU 的最大数目。

**-pool\_name pool**

定义与区域关联的池的名称。

**-sched\_type shed**

指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。选项包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**示例**

在主机 SolarisServer2 上创建资源池 POOL1。

```
dpmzone-createpool -host SolarisServer2 -pset_name PSET1 -cpu_min 1 -cpu_max 20  
-pool_name POOL1 -sched_type FSS
```

**dpmzone-createproject 命令—创建项目 (Funclet)**

dpmzone-createproject 命令用于在区域中创建项目。

此命令具有以下格式：

```
dpmzone-createproject  
-host hostname -name zonename  
-user_id uid  
[-project_id pid]  
-proj_name pname
```

**-host hostname**

指定 Solaris Zones 服务器。该参数区分大小写。

**-name zonename**

指定区域名称。

**-user\_id uid**

指定区域的用户之一。

示例：root 用户

**-project\_id pid**

（可选）定义项目 ID。您可以指定 ID，也可以让系统自动生成一个 ID。



**-proj\_name pname**

定义新项目的名称。

**示例**

在区域中创建项目：

```
dpmzone-createproject -host SolarisServer2 -name myzone1 -user_id root  
-proj_name myProject
```

**dpmzone-createzone 命令—创建区域 (Funclet)**

dpmzone-createzone 命令用于使用自定义参数在 Solaris 10 主机上创建区域。

此命令具有以下格式：

```
dpmzone-createzone  
-host hostname  
-name zonename  
-path zonepath  
-type type  
[-autoboot]  
[-if_type name]  
[-ip ip]  
[-pool_name pool]  
[-sched_type sched]  
[-phy_mem pmem]  
[-swap_mem smem]  
[-lock_mem lmem]  
[-desc desc]
```

**-host hostname**

指定 Solaris Zones 服务器。该参数区分大小写。

**-name zonename**

定义新区域的名称。

**-path zonepath**

定义从全局区域到当前区域的路径的名称。

**-type *type***

指定新区域的类型。选项包括：

**native**

使用 Solaris 10 操作环境创建非全局区域来运行应用程序。

**whole-root**

创建不继承软件包的整个根区域。

**branded**

创建包含非本地操作环境的非全局区域来运行应用程序。

**默认值：** native

**-autoboot**

指定在系统引导时自动引导区域。

**注意：**如果在服务器上禁用区域服务，则不论该属性的设置为何，均不能自动引导区域。可通过以下命令启用该服务：

```
svcadm enable svc:/system/zones:default
```

**-if\_type *name***

指定区域使用的网络接口，例如，eri0。

**-ip *ip***

（可选）定义区域的 IP 地址。

**-pool\_name *pool***

指定与区域关联的池。

**-sched\_type *shed***

指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。选项包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**-phy\_mem *pmem***

定义分配给区域的物理内存。可对该数值应用单位（K、M、G、T），例如，1M 是一个兆字节。

**-swap\_mem *smem***

定义分配给区域的交换内存。

**-lock\_mem *lmem***

定义分配给区域的锁定内存。锁定内存无法分页。

**-desc *description***

（可选）定义区域的说明。

**示例**

创建名为 WebServer4 的区域：

```
dpmzone-createzone -host MySolarisServer -name WebServer4 -path /Zones/Apache
-type native -autoboot -if_type eri0 -ip 192.168.100.100 -pool_name WebServer
-sched_type fss -phy_mem 1024M -swap_mem 2048M
```

**dpmzone-customcommand 命令—运行自定义命令 (Funclet)**

dpmzone-customcommand 命令用于在 Solaris 服务器上运行外部命令或脚本。

**注意：**有关受限外部命令的信息，请参阅本指南中的“自定义命令”附录。

此命令具有以下格式：

```
dpmzone-customcommand
-host hostname
-cmd command
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-cmd *command***

定义要运行的程序或脚本。

**示例**

在主机 SolarisServer2 上运行命令。

```
dpmzone-customcommand -host SolarisServer2 -cmd
/usr/local/lab/scripts/test_connections.pl
```

## dpmzone-deleteproject 命令—删除项目 (Funclet)

dpmzone-deleteproject 命令用于从区域中删除项目。

此命令具有以下格式：

```
dpmzone-deleteproject  
-host hostname  
-name zonename  
-proj_name pname
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定与项目关联的区域。

### **-proj\_name *pname***

指定要删除的项目。

### 示例

删除项目：

```
dpmzone-deleteproject -host SolarisServer2 -name myzone1 -project_name  
myproject1
```

## dpmzone-deletezone 命令—删除区域 (Funclet)

dpmzone-deletezone 命令用于删除 Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone-deletezone  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要删除的区域。

### 示例

删除主机 SolarisServer2 上的区域。

```
dpmzone-deletezone -host SolarisServer2 -name myzone
```

## dpmzone-installzone 命令—安装区域 (Funclet)

dpmzone-installzone 命令用于在 Solaris 10 服务器上启动安装命令。安装过程可能要花费一些时间。使用 showprocesses 命令检查区域的状态，以确认安装过程正在运行。

此命令具有以下格式：

```
dpmzone-installzone
-host hostname
-name zonename
[-archive_path path]
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要安装的区域。

### **-archive\_path *path***

（可选）为标记区域定义操作环境安装程序的路径。

**限制：**该参数是标记区域所需的参数，本地区域不需要该参数。

## 示例

安装本地区域：

```
dpmzone-installzone
-host SolarisServer
-name MyZone3
```

在 SolarisServer 上使用 Solaris 8 安装标记区域：

```
dpmzone-installzone -host SolarisServer -name MyZone4
-archive_path /opt/zoneos/branded/Solaris8Installer
```

## dpmzone-movezone 命令—移动区域 (Funclet)

dpmzone-movezone 命令用于将区域从一个路径移动到新路径。

此命令具有以下格式：

```
dpmzone-movezone
-host hostname
-name zonename
-new_path path
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要移动到新位置的区域。

**-new\_path *path***

为要移动的区域定义新路径。

**示例**

移动主机 SolarisServer2 上的区域：

```
dpmzone-movezone -host SolarisServer2 -name myzone1  
-new_path /opt/zones/myzone1
```

## **dpmzone-rebootzone 命令—重新启动区域 (Funclet)**

dpmzone-rebootzone 命令用于重新启动 Oracle Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone-rebootzone  
-host hostname  
-name zonename
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要重新启动的区域。

**示例**

重新启动主机 SolarisServer2 上的区域：

```
dpmzone-rebootzone -host SolarisServer2 -name myzone
```

## **dpmzone-renamezone 命令—重命名区域 (Funclet)**

dpmzone-renamezone 命令用于重命名 Oracle Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone-renamezone  
-host hostname  
-name czonename  
-new_name nzonename
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *czonename***

指定当前的区域名称。

**-new\_name *nzonename***

定义新的区域名称。

**示例**

重命名主机 SolarisServer2 上的区域：

```
dpmzone-renamezone -host SolarisServer2 -name myzone -new_name yourzone
```

**dpmzone-setpoolscheduler 命令—设置池排定程序 (Funclet)**

**dpmzone-setpoolscheduler** 命令设置用于资源池的排定程序的类型。资源池可以使用两种类型的排定程序，即公平份额排定程序 (FSS) 和时间份额排定程序 (TS)。

此命令具有以下格式：

```
dpmzone-setpoolscheduler
-host hostname
-pool_name poolname
-sched_type sched
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-pool\_name *poolname***

指定可进行分区的资源的关联组。

**-sched\_type *shed***

指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。选项包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**示例**

将资源池的排定设置为公平份额排定程序：

```
dpmzone-setpoolscheduler
-host SolarisServer
-pool_name test_pool
-sched_type FSS
```

将资源池的排定设置为时间份额排定程序。

```
dpmzone-setpoolscheduler
-host SolarisServer
-pool_name test_pool
-sched_type TS
```

## **dpmzone-setprocessorset 命令—设置处理器集的最大 CPU 数目 (Funclet)**

**dpmzone-setprocessorset** 命令用于设置处理器集的最大 CPU 数目。

此命令具有以下格式：

```
dpmzone-setprocessorset
-host hostname
-processor_set_name psetname
-cpu_max maxcpu
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-processor\_set\_name *psetname***

指定处理器集。每个处理器集（CPU 组）可以包含零个或多个处理器。

### **-cpu\_max *maxcpu***

定义处理器集中 CPU 的最大数目。

### 示例

设置处理器集的最大 CPU 数目：

```
dpmzone-setprocessorset -host SolarisServer -processor_set_name test_pset
-cpu_max 64
```

## **dpmzone-showinterfaces 命令—显示网络接口 (Funclet)**

**dpmzone-showinterfaces** 命令用于列出 Solaris 10 主机上的所有网络接口。

此命令具有以下格式：

```
dpmzone-showinterfaces
-host hostname
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### 示例

列出主机 SolarisServer2 上的所有网络接口：

```
dpmzone-showinterfaces -host SolarisServer2
```



## dpmzone-showpools 命令—显示资源池 (Funclet)

dpmzone-showpools 命令用于列出 Solaris 10 主机上的所有资源池。

此命令具有以下格式：

```
dpmzone-showpools  
-host hostname
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### 示例

列出主机 SolarisServer2 上的所有资源池：

```
dpmzone-showpools -host SolarisServer2
```

## dpmzone-showprojects 命令—列出项目的属性 (Funclet)

dpmzone-showprojects 命令用于列出项目的属性。

此命令具有以下格式：

```
dpmzone-showprojects  
-host hostname  
-name zonename  
-proj_name pname
```

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定区域。

**-proj\_name *pname***

指定项目。

### 示例

列出项目的属性：

```
dpmzone-showprojects -host SolarisServer2 -name myzone1 -project_name myproject1
```

## dpmzone-showresourceset 命令—显示资源集 (Funclet)

dpmzone-showresourceset 命令用于列出资源池中的资源。

此命令具有以下格式：

```
dpmzone-showresourceset
-host hostname
-pool_name poolname
-host hostname
```

指定 Solaris Zones 服务器。该参数区分大小写。

**-pool\_name *poolname***

指定可进行分区的资源的关联组。

### 示例

列出 MyZonesResourcePool 中的资源集：

```
dpmzone-showresource_set -host SolarisServer2 -pool_name MyZonesResourcePool
```

## dpmzone-showtasks 命令—列出项目中的任务 (Funclet)

dpmzone-showtasks 命令用于列出项目中的任务。

此命令具有以下格式：

```
dpmzone-showtasks
-host hostname
-name zonename
-proj_name pname
-host hostname
```

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定区域。

**-proj\_name *pname***

指定项目。

### 示例

列出项目中的任务：

```
dpmzone-showtasks -host SolarisServer2 -name myzone1 -project_name myproject1
```

## dpmzone-showzoneinterfaces 命令—显示区域的网络接口 (Funclet)

dpmzone-showzoneinterfaces 命令用于列出区域中的所有网络接口。

此命令具有以下格式：

```
dpmzone-showzoneinterfaces  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定区域。

### 示例

列出 SolarisServer2 上区域 MyZone3 中的所有网络接口：

```
dpmzone-showzoneinterfaces -host SolarisServer2 -name MyZone3
```

## dpmzone-showzoneprocess 命令—显示区域中的进程 (Funclet)

dpmzone-showzoneprocess 命令用于列出区域上的所有进程。

此命令具有以下格式：

```
dpmzone-showzoneprocess  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定区域。

### 示例

列出 SolarisServer2 上区域 MyZone3 的进程：

```
dpmzone-showzoneprocess -host SolarisServer2 -name MyZone3
```

## dpmzone-showzoneprojects 命令—显示区域中的项目 (Funclet)

dpmzone-showzoneprojects 命令用于列出区域中的所有项目。

此命令具有以下格式：

```
dpmzone-showzoneprojects  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定区域。

### 示例

列出 SolarisServer2 上区域 MyZone3 的所有项目：

```
dpmzone-showzoneprojects -host SolarisServer2 -name MyZone3
```

## dpmzone-showzones 命令—显示区域 (Funclet)

dpmzone-showzones 命令用于列出 Solaris 主机上的所有区域。

此命令具有以下格式：

```
dpmzone-showzones  
-host hostname
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### 示例

列出主机 SolarisServer2 上的所有区域：

```
dpmzone-showzones -host SolarisServer2
```

## dpmzone-startzone 命令—启动区域 (Funclet)

dpmzone-startzone 命令用于启动 Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone-startzone  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要启动的区域。

### 示例

启动主机 SolarisServer2 上的区域：

```
dpmzone-startzone -host SolarisServer2 -name myzone
```

## dpmzone-stopzone 命令—停止区域 (Funclet)

dpmzone-stopzone 命令用于停止 Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone-stopzone  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要停止的区域。

### 示例：停止主机上的区域

停止主机 SolarisServer2 上的区域：

```
dpmzone-stopzone -host SolarisServer2 -name myzone
```

## dpmzone-uninstallzone 命令—卸载区域 (Funclet)

dpmzone-uninstallzone 命令用于卸载 Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone-uninstallzone  
-host hostname  
-name zonename
```

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要卸载的区域。

### 示例

卸载主机 SolarisServer2 上的区域：

```
dpmzone-uninstallzone -host SolarisServer2 -name myzone
```

## 部署和策略配置 AutoShell 命令

本节按字母顺序详细描述了部署和策略配置 AutoShell 命令。您可以从 AutoShell 命令行访问 ISM 函数。您可以从 AutoShell 命令行提示符使用命令 `help ism-*` 来显示函数的完整列表。

## ism-applyPolicyForService 命令—将策略应用于所有系统 (Funclet)

ism-applyPolicyForService 命令用于将策略应用于受管服务中的所有计算机系统。

此命令具有以下格式：

```
ism-applyPolicyForService  
-policyName policy_name  
-policyType policy_type  
-serviceName service_name  
-templateName templateName  
-templateOs templateOs
```

### **-policyName *policy\_name***

指定要应用的策略名称。

**默认值：**空

### **-policyType *policy\_type***

指定策略类型。

**-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

**-templateName *templateName***

指定模板的名称。

**-templateOs *templateOs***

指定策略模板操作系统。

**示例：将策略应用于所有系统**

本示例将策略 SystemEDGE 应用于受管的服务 CA Virtual Assurance。

```
ism-applyPolicyForService -policyName policy5 -policyType SystemEDGE
-serviceName "Enterprise\\Data Center\\CA Virtual Assurance Services"
-templateName template2 -templateOs windows
```

**ism-applyPolicies 命令—将策略应用于一组主机 (Funclet)**

ism-applyPolicies 命令用于将指定的策略应用于一组主机。

此命令具有以下格式：

```
ism-applyPolicies
-policyName policy_name
-policyType policy_type
-hostName host_name
[-timeout time]
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-hostName *host\_name***

指定主机计算机的名称。

**-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：**30 秒。

**注意：**如果指定零，会发生默认系统范围超时。

**示例：将策略应用于一组主机**

本示例将指定的策略应用于一组主机。

```
ism-applyPolicies -policyName policy5 -policyType SystemEDGE -hostName  
"s1"-timeout 30
```

**ism-applyPolicyWait 命令-应用策略并等待 (Funclet)**

ism-applyPolicyWait 命令用于将策略应用于主机并返回应用了该策略的主机名。应用该策略或发生超时之前，此命令不会返回主机名。

此命令具有以下格式：

```
ism-applyPolicyWait  
-policyName policy_name  
-policyType policy_type  
-hostName host_name  
[-timeout time]
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-hostName *host\_name***

指定主机计算机的名称。

**-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：**30 秒。

**示例：应用策略并等待**

将策略 policy5 应用于主机 hvserver。

```
ism-applyPolicy -policyName policy5 -policyType SystemEDGE.SRM.Policies  
-hostname hvserver -timeout 400
```



## ism-applyPolicyTemplates 命令—将策略模板应用于一组主机 (Funclet)

ism-applyPolicyTemplates 命令用于将指定的策略模板应用于一组主机。

此命令具有以下格式：

```
ism-applyPolicyTemplates
-templateName templateName
-templateOs templateOs
-hostName hostname
[-timeout time]
[-replaceExisting replaceExisting]
```

### **-templateName *templateName***

指定模板的名称。

### **-templateOs *templateOs***

指定策略模板操作系统。

### **-hostName *host\_name***

指定主机计算机的名称。

### **-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：** 30 秒。

**注意：** 如果指定零，会发生默认系统范围超时。

### **-replaceExisting *replaceExisting***

（可选）使用新配置替换主机的现有配置。

### 示例：将策略模板应用于一组主机

将策略模板应用于一组主机。

```
ism-applyPolicyTemplates -templateName template2 -templateOs windows -hostName
"s1","s2","s3"-timeout 400 -replaceExisting yes
```

## ism-applyPoliciesAndTemplates 命令—将策略和模板应用于一组主机 (Funclet)

ism-applyPoliciesAndTemplates 命令用于将策略和模板应用于一组主机。

此命令具有以下格式：

```
ism-applyPoliciesAndTemplates
-policyName policyName
-policyType policyType
-hostName hostname
-templateName templateName
-templateOs templateOs
[-timeout time]
[-replaceExisting replaceExisting]
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-hostName *host\_name***

指定主机计算机的名称。

**-templateName *templateName***

指定模板的名称。

**-templateOs *templateOs***

指定策略模板操作系统。

**-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：** 30 秒。

**注意：** 如果指定零，会发生默认系统范围超时。

**-replaceExisting *replaceExisting***

（可选）使用新配置替换主机的现有配置。

### 示例：将策略和模板应用于一组主机

将策略和模板应用于一组主机。

```
ism-applyPoliciesAndTemplates -policyName policy6 -policyType
SystemEDGE.SRM.Policies -hostName "s4","s5","s6" -templateName template2
-templateOs windows -timeout 400 -replaceExisting yes
```

## ism-applyPoliciesForService 命令—将策略应用于所有系统 (Funclet)

ism-applyPoliciesForService 命令用于将策略应用于受管服务中的所有计算机系统。

此命令具有以下格式：

```
ism-applyPoliciesForService
-policyName policy_name
-policyType policy_type
-serviceName service_name
[-timeout time]
```

### **-policyName *policy\_name***

指定要应用的策略名称。

**默认值：**空

### **-policyType *policy\_type***

指定策略类型。

### **-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

### **-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：**30 秒。

**注意：**如果指定零，会发生默认系统范围超时。

## 示例：将策略应用于所有系统

本示例将策略应用于受管服务 CA Virtual Assurance。

```
ism-applyPoliciesForService -policyName policy7 -policyType SystemEDGE
-serviceName "Enterprise\\Data Center\\CA Virtual Assurance Services" -timeout
400
```

## ism-applyPolicyTemplatesForService 命令—将策略模板应用于所有系统 (Funclet)

ism-applyPolicyTemplatesForService 命令用于将策略模板应用于受管服务中的所有计算机系统。

此命令具有以下格式：

```
ism-applyPolicyTemplatesForService
-templateName templateName
-templateOs templateOs
-serviceName serviceName
[-timeout time]
[-replaceExisting replaceExisting]
```

### **-templateName *templateName***

指定模板的名称。

### **-templateOs *templateOs***

指定策略模板操作系统。

### **-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

### **-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：**30 秒。

**注意：**如果指定零，会发生默认系统范围超时。

### **-replaceExisting *replaceExisting***

（可选）使用新配置替换主机的现有配置。

## 示例：将策略模板应用于所有系统

本示例将策略模板 SystemEDGE 应用于受管服务 CA Virtual Assurance。

```
ism-applyPolicyTemplatesForService -templateName template3 -templateOs windows
-serviceName "Enterprise\\Data Center\\CA Virtual Assurance Services" -timeout
200 -replaceExisting yes
```

## ism-applyPoliciesAndTemplatesForService 命令—将策略和模板应用于所有系统 (Funclet)

ism-applyPoliciesAndTemplatesForService 命令用于将策略和模板应用于受管服务中的所有计算机系统。

此命令具有以下格式：

```
ism-applyPoliciesAndTemplatesForService
-policyName policy_name
-policyType policy_type
-serviceName service_name
-templateName templateName
-templateOs templateOs
[-timeout time]
[-replaceExisting replaceExisting]
```

### **-policyName *policy\_name***

指定要应用的策略名称。

**默认值：**空

### **-policyType *policy\_type***

指定策略类型。

### **-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

### **-templateName *templateName***

指定模板的名称。

### **-templateOs *templateOs***

指定策略模板操作系统。

### **-timeout *time***

（可选）为该命令等待主机成功配置指定时间限制（以秒为单位），在此时间限制之后，将发生超时。如果未指定，该命令将不等待主机成功配置。

**默认值：**30 秒。

**注意：**如果指定零，会发生默认系统范围超时。

### **-replaceExisting *replaceExisting***

（可选）使用新配置替换主机的现有配置。

**示例：将策略和模板应用于所有系统**

本示例将策略和模板应用于受管服务 CA Virtual Assurance。

```
ism-applyPoliciesAndTemplatesForService -policyName policy5 -policyType
SystemEDGE -serviceName "Enterprise\\Data Center\\CA Virtual Assurance Services"
-templateName template2 -templateOs windows -timeout 400 -replaceExisting yes
```

**ism-cancelJob 命令-取消作业 (Funclet)**

ism-cancelJob 命令用于取消作业中任何未决的任务。可以指定作业 UUID 或友好 ID。

**注意：**该命令无法取消活动任务。

此命令具有以下格式：

```
ism-cancelJob -jobID job_ID
```

**-jobID *job\_ID***

指定作业 UUID 或友好 ID。

**示例：取消作业**

本示例取消作业 J00004 中所有未决的任务。

```
ism-cancelJob -jobID J00004
```

**ism-copyPolicy 命令-复制策略 (Funclet)**

ism-copyPolicy 命令用于通过从现有策略复制来创建策略。

此命令具有以下格式：

```
ism-copyPolicy -policyName policy_name
-policyType policy_type
-policyVersion [policy_version]
-newName new_policy_name
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-hostName *host\_name***

指定主机计算机的名称。

**-policyVersion *policy\_version***

(可选) 指定要应用的策略的版本。

**默认值:** 空

**-newName *new\_policy\_name***

定义复制的策略的名称。

**示例: 复制策略**

以下示例使用现有的默认策略创建策略 test1ver1。

```
ism-copyPolicy -policyName default -policyType SystemEDGE -policyVersion 1
-newName test1ver1
```

**ism-createJobTemplate 命令-创建作业模板 (Funclet)**

ism-createJobTemplate 命令用于从现有作业创建作业模板。此命令返回创建的作业的模板 ID。

此命令具有以下格式:

```
ism-createJobTemplate -name new_template_name
-jobID existing_job_ID
```

**-name *new\_template\_name***

定义模板的名称。

**-jobID *job\_ID existing\_job\_ID***

指定现有作业 UUID 或友好 ID。

**示例: 创建作业模板**

以下示例从现有作业 ID J00004 创建作业模板 Job\_template\_08。

```
ism-createJobTemplate -name Job_template_08 -jobID J00004
```

**ism-deleteJobTemplate 命令-删除作业模板 (Funclet)**

ism-deleteJobTemplate 命令用于删除现有作业模板。如果指定重复名称, 命令将返回错误。

此命令具有以下格式:

```
ism-deletejobTemplate job_template_name
```

***job\_template\_name***

指定要删除的模板名称。您还可以指定模板 UUID。

### 示例：删除作业模板

本示例删除现有作业模板 `Job_template_08`。

```
ism-deleteJobTemplate Job_template_08
```

### ism-deletePolicy 命令-删除策略 (Funclet)

`ism-deletePolicy` 命令用于删除现有策略。如果不指定策略版本，则将选择最新版本。

此命令具有以下格式：

```
ism-deletePolicy -policyName policy_name  
-policyType policy_type [-policyVersion policy_version]
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-policyVersion *policy\_version***

（可选）指定要应用的策略的版本。

**默认值：**空

### 示例：删除策略

本示例删除现有策略 `policy5`。

```
ism-deletePolicy -policyName policy5 -policyType SystemEDGE.SRM.Policies  
-policyVersion 3.2
```



## ism-deployPackageToHost 命令-将软件包部署到主机 (Funclet)

ism-deployPackageToHost 命令用于将指定的软件包打包程序部署到主机。此命令返回部署作业的 UUID。

此命令具有以下格式：

```
ism-deployPackageToHost
[-jobName job_name]
-package package_name
[-packageVer package_version]
[-wrapper wrapper_name]
-hostName host_name
-user user_name
-password password
-acceptEula
[-encrypted]
[-redeploy]
[-ignoreDepends]
```

### **-jobName *job\_name***

（可选）指定部署作业列表和审核记录中使用的作业名称。

### **-package *package\_name***

指定部署软件包的名称或 ID。

### **-packageVer *package\_version***

（可选）指定部署软件包版本。

### **-wrapper *wrapper\_name***

（可选）指定要使用的软件包打包程序的名称。如果没有指定打包程序，将使用默认打包程序。

### **-hostName *host\_name***

指定主机计算机的名称。

### **-user *user\_name***

指定用于登录的用户名。

### **-password *user\_password***

指定用于登录的用户密码。

### **-acceptEula**

指示接受部署的所有软件包的所有最终用户许可协议 (EULA)。

### **-encrypted**

（可选）指示用户名和密码已加密。

**-redeploy**

(可选) 指示强制重新部署现有软件包。

**-ignoreDepends**

(可选) 指示即使缺少软件包依存关系也要强制部署。

**示例：将软件包部署到主机**

本示例将软件包 SystemEDGE 部署到主机 hvserver 上。

```
ism-deployPackageHost
-package "CA SystemEDGE"
-hostName hvserver
-user admin
-password #test#
-acceptEula
-redeploy
-ignoreDepends
```

**ism-deployTemplate 命令-从模板部署软件包 (Funclet)**

ism-deployTemplate 命令用于将模板中定义的所有软件包部署到所有计算机。此命令返回部署作业的 UUID。

此命令具有以下格式：

```
ism-deployTemplate
[-jobName job_name]
-template template_name
-user user_name
-password password
-acceptEula
[-encrypted]
[-redeploy]
[-ignoreDepends]
```

**-jobName *job\_name***

(可选) 指定部署作业列表和审核记录中使用的作业名称。

**-template *template\_name***

指定用作部署源的作业模板的名称或 UUID。如果多个模板具有相同的名称，则使用模板 UUID。

**-user *user\_name***

指定用于登录的用户名。

**-password *user\_password***

指定用于登录的用户密码。

**-hostName *host\_name***

指定主机计算机的名称。

**-acceptEula**

指示接受部署的所有软件包的所有最终用户许可协议 (EULA)。

**-encrypted**

(可选) 指示用户名和密码已加密。

**-redeploy**

(可选) 指示强制重新部署现有软件包。

**-ignoreDepends**

(可选) 指示即使缺少软件包依存关系也要强制部署。

**示例：从模板部署所有软件包**

本示例将模板 `SysTemplate` 中定义的所有软件包部署到所有计算机。

```
ism-deployTemplate
-template SysTemplate
-user admin
-password #test#
-hostName hvserver
-acceptEula
-redeploy
-ignoreDepends
```

**ism-deployTemplateForService 命令-在服务中部署模板软件包 (Funclet)**

`ism-deployTemplate` 命令用于将作业模板中定义的所有软件包部署到服务中列出的所有计算机。此命令返回部署作业的 UUID。

此命令具有以下格式：

```
ism-deployTemplateForService
[-jobName job_name]
-template template_name
-service service_name
-user user_name
-password password
-acceptEula
[-encrypted]
[-redeploy]
[-ignoreDepends]
```

**-jobName *job\_name***

(可选) 指定部署作业列表和审核记录中使用的作业名称。

**-template *template\_name***

指定用作部署源的作业模板的名称或 UUID。如果多个模板具有相同的名称，则使用模板 UUID。

**-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

**-user *user\_name***

指定用于登录的用户名。

**-password *user\_password***

指定用于登录的用户密码。

**-acceptEula**

指示接受部署的所有软件包的所有最终用户许可协议 (EULA)。

**-encrypted**

（可选）指示用户名和密码已加密。

**-redeploy**

（可选）指示强制重新部署现有软件包。

**-ignoreDepends**

（可选）指示即使缺少软件包依存关系也要强制部署。

**示例：在服务中部署模板软件包**

本示例在服务 Enterprise\Data Center\MyNewService 中部署模板 SysTemplate 中定义的所有软件包。

```
ism-deployTemplateForService -template SysTemplate  
-service Enterprise\Data Center\MyNewService  
-user admin -password #test# -hostName hvserver  
-acceptEula -redeploy -ignoreDepends
```

## ism-deployTemplateToHost 命令-在主机上部署模板软件包 (Funclet)

ism-deployTemplate 命令用于将作业模板中定义的所有软件包部署到指定主机。此命令返回部署作业的 UUID。

此命令具有以下格式：

```
ism-deployTemplate
[-jobName job_name]
-template template_name
-hostname host_name
-user user_name
-password password
-acceptEula
-encrypted
[-redeploy]
[-ignoreDepends]
```

### **-jobName *job\_name***

（可选）指定部署作业列表和审核记录中使用的作业名称。

### **-template *template\_name***

指定用作部署源的作业模板的名称或 UUID。如果多个模板具有相同的名称，则使用模板 UUID。

### **-hostName *host\_name***

指定主机计算机的名称。

### **-user *user\_name***

指定用于登录的用户名。

### **-password *user\_password***

指定用于登录的用户密码。

### **-hostName *host\_name***

指定主机计算机的名称。

### **-acceptEula**

指示接受部署的所有软件包的所有最终用户许可协议 (EULA)。

### **-encrypted**

（可选）指示用户名和密码已加密。

### **-redeploy**

（可选）指示强制重新部署现有软件包。

### **-ignoreDepends**

（可选）指示即使缺少软件包依存关系也要强制部署。

### 示例：在主机上部署模板软件包

本示例将模板 SysTemplate 中定义的所有软件包部署到主机 hvserver。

```
ism-deployTemplate -template SysTemplate
-hostname hvserver
-user admin
-password #test#
-hostName hvserver
-acceptEula
-redeploy
-ignoreDepends
```

### ism-encryptString 命令-加密字符串 (Funclet)

ism-encryptString 命令用于加密给定文本并将加密文本返回到控制台。

此命令具有以下格式：

```
ism-ism-encryptString encrypted_text
```

***encrypted\_text***

指定要加密的文本。

### 示例：加密字符串

本示例加密文本字符串 passwordText。

```
ism-deleteJobTemplate passwordText
```

### ism-listComputerSystems 命令-显示所有计算机系统 (Funclet)

ism-listComputerSystems 命令用于列出所有计算机系统（包括主机名、平台、受管服务和已部署软件包）。

此命令具有以下格式：

```
ism-listComputerSystems [hostFilter]
```

***hostFilter***

（可选）指定部分主机名。使用诸如星号 \* 等通配符。

**默认值：**空

### 示例：显示所有计算机系统

以下示例显示主机名中第一个字符为 h 的所有计算机系统。

```
ism-listComputerSystems h*
```

## ism-listDeployedSystems 命令-列出已部署软件包的所有主机 (Funclet)

ism-listDeployedSystems 命令用于列出在其中部署了特定软件包的所有主机名。

**注意：**此视图是审核跟踪视图，因此如果主机稍后被删除或重新映像，它们将仍然显示在该列表中。

此命令具有以下格式：

```
ism-listDeployedSystems -package package_name  
-platform platform_name [-version package_version]
```

**-package *package\_name***

指定部署软件包的名称或 ID。

**-platform *platform\_name***

指定软件包的操作系统。

**-version *package\_version***

（可选）指定软件包的版本。

**注意：**仅当有多个软件包版本时才指定版本。

### 示例：列出已部署软件包的所有主机

本示例列出在其中部署了软件包 7e033e5d-test-404d-82in-a2f7bbe960a4 的所有主机名。

```
ism-listDeployedSystems -package 7e033e5d-test-404d-82in-a2f7bbe960a4 -platform  
Windows
```

## ism-listJobs 命令-列出所有作业 (Funclet)

ism-listJob 命令用于列出所有部署作业。可以使用通配符按照作业状态和部分操作环境名称进行筛选。

此命令具有以下格式：

```
ism-listJobs [-status job_status] [-platformFilter platform_filter]
```

**-status *job\_status***

（可选）表示部署作业的状态。可以按照状态名称（活动、未决、完成或失败）的逗号分隔列表指定筛选。

**-platformFilter *platform\_filter***

（可选）表示使用通配符的完整或部分操作环境名称。

**示例：列出状态为“活动”的所有作业**

本示例列出状态为“活动”的所有作业。

```
ism-listJobs -status Active
```

**ism-listJobStatus 命令-出作业状态 (Funclet)**

ism-listJobStatus 命令用于显示部署作业的状态。

此命令具有以下格式：

```
ism-listJobStatus -jobID job_ID
```

**-jobID *job\_ID***

指定作业 UUID 或友好 ID。

**示例：列出作业状态**

本示例显示作业 aa-12348-test0-9883 的状态

```
ism-listJobStatus -jobId aa-12348-test0-9883
```

**ism-listJobTemplates 命令-列出所有作业模板 (Funclet)**

ism-listJobTemplates 命令用于列出所有定义的部署作业模板的 UUID 和名称。

此命令具有以下格式：

```
ism-listJobTemplates [-detail]
```

**-detail**

显示每个作业模板的软件包和目标详细信息。

**示例：列出作业状态**

本示例显示部署作业模板。

```
ism-listJobTemplates
```

**ism-listPackages 命令-列出所有软件包 (Funclet)**

ism-listPackages 命令用于列出所有打包程序和软件包。

**注意：**不显示使用现有打包程序引用的软件包。

此命令具有以下格式：

```
ism-listJob [-platformFilter platform_filter]
```



**-platformFilter *platform\_filter***

（可选）表示完整或部分操作环境名称。使用诸如星号 \* 等通配符。

**示例：列出所有软件包**

本示例列出操作环境名称以字母 **w** 开头的所有打包程序和软件包。

```
ism-listPackages -platformFilter w*。
```

**ism-listPolicies 命令-列出所有策略 (Funclet)**

ism-listPolicies 命令用于列出所有策略。可以按照策略类型筛选策略。

此命令具有以下格式：

```
ism-listPolicies [-policyTypeFilter policytype_filter] [-allVersions]
```

**-policyTypeFilter *policytype\_filter***

（可选）表示策略类型筛选。

**-allVersions**

（可选）表示列出所有策略的所有版本。

**示例：列出所有策略**

本示例列出所有策略的所有版本。

```
ism-listPolicies -allVersions
```

**ism-listPolicyStatus 命令-列出策略状态 (Funclet)**

ism-listPolicyStatus 命令用于列出所有主机的策略交付状态。

此命令具有以下格式：

```
ism-listPolicyStatus -policyName policy_name  
-policyType policy_type [-policyVersion policy_version]
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-policyVersion *policy\_version***

（可选）指定要应用的策略的版本。

**默认值：** 空

**示例：列出策略状态**

本示例列出策略 `policy5` 的策略交付状态

```
ism-listPolicyStatus -policyName policy5  
-policyType SystemEDGE.SRM.Policies -policyVersion 3.2
```

**ism-listServiceSystems 命令-列出服务中的所有计算机系统 (Funclet)**

`ism-listServiceSystems` 命令用于列出服务中与筛选字符串匹配的所有计算机系统。该列表包括主机名、平台、受管服务和已部署软件包。

此命令具有以下格式：

```
ism-listServiceSystems service_filter  
service_filter
```

指定完整或部分服务名称。使用诸如星号 (\*) 等通配符。

**示例：列出服务中的所有计算机系统**

本示例列出以字符 `b` 开头的所有服务中的所有计算机系统。

```
ism-listServiceSystems b*
```

**ism-listSysEdgeConfig 命令-列出 SystemEDGE 配置 (Funclet)**

`listSysEdgeConfig` 命令用于显示在给定主机名和策略类型的管理器服务器上已缓存的 SystemEDGE 配置。

此命令具有以下格式：

```
ism-listSysEdgeConfig -hostName host_name -policyType policy_type
```

**-hostName *host\_name***

指定主机计算机的名称。

**-policyType *policy\_type***

指定策略类型。

**示例：列出 SystemEDGE 配置**

本示例列出 systemEDGE 配置。

```
ism-listSysEdgeConfig -hostname hvserver -policyType SystemEDGE.SRM.Policies
```

## ism-listSystemPoliciesForService 命令-列出服务中的所有策略 (Funclet)

ism-listSystemPoliciesForService 命令用于列出部署到服务中每个主机的所有策略。

此命令具有以下格式：

```
ism-listSystemPoliciesForService -serviceName service_name
```

**-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

### 示例：列出服务中的所有策略

本示例列出服务 Enterprise\DataCenter\MyNewService 中的所有策略。

```
ism-listServiceSystems Enterprise\DataCenter\MyNewService
```

## ism-listTemplateEulas 命令-显示模板中的 EULA (Funclet)

ism-listTemplateEulas 命令用于显示模板中所有最终用户许可协议 (EULA) 的完整文本。如果存在多个模板名称，请使用模板 UUID。

此命令具有以下格式：

```
ism-listTemplateEulas -template template_name
```

**-template *template\_name***

指定模板名称。您还可以指定模板 UUID。

### 示例：显示模板中的所有 EULA

本示例显示模板 test\_template 中的所有 EULA。

```
ism-listTemplateEulas -template test_template
```

## ism-renamePolicy 命令-重命名策略 (Funclet)

ism-renamePolicy 命令用于重命名策略的最新版本。

**注意：**不能重命名以前的策略版本，请改为创建副本。

此命令具有以下格式：

```
ism-renamePolicy -policyName policy_name
```

```
-policyType policy_type
```

```
-newName new_policy_name
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

**-newName *new\_policy\_name***

定义新策略名称。

#### 示例：重命名策略

本示例将策略 `policy5` 重命名为新名称 `NewPolicy5`。

```
ism-applyPolicy -policyName policy5  
-policyType SystemEDGE.SRM.Policies -newName NewPolicy5
```

### ism-setMaintenanceMode 命令-设置主机的维护模式 (Funclet)

`ism-setMaintenanceMode` 命令用于设置主机的维护模式。

此命令具有以下格式：

```
ism-setMaintenanceMode -hostname host_name  
[-on {true|false}]
```

**-hostName *host\_name***

指定主机计算机的名称。

**-on {true|false}**

（可选）定义主机是否设置为维护模式。指定 `false` 可使系统退出受管模式。

**默认值：** `true`

#### 示例：设置主机维护模式

本示例将主机 `hvserver` 设置为维护模式。

```
ism-setMaintenanceMode -hostname hvserver
```

### ism-listSystemPolicies 命令-列出所有策略 (Funclet)

`ism-listSystemPolicies` 命令用于列出部署到每个主机的所有策略和策略模板。

此命令具有以下格式：

```
ism-listSystemPolicies -hostName hostName
```

**-hostName *hostName***

指定主机名。

**示例：列出服务中的所有策略**

本示例列出部署到给定主机 `test2` 的所有策略和策略模板以及交付状态。上次应用策略和策略模板的时间以及例外（如果有）。

```
ism-listSystemPolicies -hostName test2
```

输出

```
Policy                Status              OS Type Ver   Applied          Exc
-----
-- test2 -----
-- Policies-----
default SystemEDGE.SR Delivery requested Windows 0 2011-08-02 18:35:36 No
default SystemEDGE    Configured           Windows 0 2011-08-08 12:54:59 No
-- Templates -----
test3 SystemEDGE       Configured           Windows 0 2011-08-08 12:54:59 No
```

**ism-setMaintenanceModeForService 命令-设置服务中的维护模式 (Funclet)**

`ism-setMaintenanceModeForService` 命令用于设置服务中所有计算机系统的维护模式。

此命令具有以下格式：

```
ism-setMaintenanceModeForService
-servicename service_name
[-on {true|false}]
```

**-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

**-on {true|false}**

定义服务中的所有计算机系统是否设置为维护模式。指定 `false` 可使这些系统退出受管模式。

**默认值：** `true`

**示例：设置服务中的维护模式**

本示例将服务 Enterprise\DataCenter\MyNewService 中的所有主机设置为维护模式。

```
ism-setMaintenanceModeForService -servicename  
Enterprise\DataCenter\MyNewService -on true
```

**ism-setSysEdgeLogLevel 命令-设置 SystemEDGE 记录级别 (Funclet)**

ism-setSysEdgeLogLevel 命令用于设置主机的 SystemEDGE 记录级别。

此命令具有以下格式：

```
ism-setSysEdgeLogLevel -hostname host_name -level level
```

**-hostname *host\_name***

指定主机计算机的名称。

**-level *level***

指示要将消息记录到给定级别的代理。级别包括：致命、严重、警告、信息、调试、调试 1、调试 2 和调试 3。值越高，记录的信息越多。致命是最低级别，调试 3 是最高级别。

**示例：设置 SystemEDGE 记录级别**

本示例将 SystemEdge 记录级别设置为致命。

```
ism-setSysEdgeLogLevel -hostname hvserver -level fatal
```

**ism-setSysEdgeLogLevelForService 命令-设置服务中的 SystemEDGE 记录级别 (Funclet)**

ism-setSysEdgeLogLevelForService 命令用于设置服务中所有主机的 SystemEDGE 记录级别。

此命令具有以下格式：

```
ism-setSysEdgeLogLevelForService -servicename service_name -level level
```

**-serviceName *service\_name***

指定受管服务的名称。

**注意：**在服务名称中，使用双反斜杠作为分隔符。使用单反斜杠作为分隔符会被解释为转义序列。例如，将服务名称指定为“企业\\数据中心\\服务名称”，而不是“企业\数据中心\服务名称”。

**-level *level***

指示要将消息记录到给定级别的代理。级别包括：致命、严重、警告、信息、调试、调试 1、调试 2 和调试 3。值越高，记录的信息越多。致命是最低级别，调试 3 是最高级别。

**示例：设置服务中的 SystemEDGE 记录级别**

本示例设置服务 Enterprise\DataCenter\MyNewService 中的 SystemEDGE 记录级别。

```
ism-setSysEdgeLogLevel -servicename Enterprise\DataCenter\MyNewService -level info
```

**ism-updateAllExceptionSystems 命令-应用最新策略 (Funclet)**

ism-updateAllExceptionSystems 命令用于查找具有策略例外的所有主机并应用最新策略。此命令在成功时返回 OK，或者显示错误消息。

此命令具有以下格式：

```
ism-updateAllExceptionSystems
```

**示例：应用最新策略**

本示例查找具有策略例外的所有主机并应用最新策略。

```
ism-updateAllExceptionSystems
```

**ism-updateAllOldPolicies 命令-更新所有旧策略 (Funclet)**

ism-updateAllOldPolicies 命令用于查找具有过期策略配置的所有主机并应用最新策略。

此命令具有以下格式：

```
ism-updateAllOldPolicies
```

**示例：更新所有旧策略**

本示例查找具有旧策略的所有主机并应用最新策略。

```
ism-updateAllOldPolicies
```

**ism-updateAllPolicies 命令-更新所有策略 (Funclet)**

ism-updateAllPolicies 命令用于将最新策略应用于所有计算机。

此命令具有以下格式：

```
ism-updateAllPolicies
```

**示例：更新所有策略**

本示例将最新策略应用于所有计算机。

```
ism-updateAllPolicies
```

## ism-updateAllUpToDateSystems 命令-更新所有最新系统 (Funclet)

ism-updateAllUpToDateSystems 命令用于将最新策略重新应用于所有最新主机。

此命令具有以下格式：

```
ism-updateAllUpToDateSystems
```

### 示例：更新所有最新系统

本示例将最新策略重新应用于所有最新计算机。

```
ism-updateAllUpToDateSystems
```

## ism-updateExceptionSystems 命令-应用最新策略 (Funclet)

ism-updateExceptionSystems 命令用于查找具有策略例外的所有主机并应用最新配置。

此命令具有以下格式：

```
ism-updateExceptionSystems -policyName policy_name  
-policyType policy_type
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

### 示例：更新至最新策略

本示例更新默认策略的所有策略例外并应用最新配置。

```
ism-updateExceptionSystems -policyName default -policyType SystemEDGE
```

## ism-updateOldSystems 命令-更新旧策略 (Funclet)

ism-updateOldSystems 命令用于查找具有指定策略过期配置的所有主机并应用最新策略配置。

此命令具有以下格式：

```
ism-updateOldSystems -policyName policy_name  
-policyType policy_type
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。



### 示例：更新旧策略

本示例更新具有默认策略的过期配置的所有主机并应用最新配置。

```
ism-updateOldSystems -policyName default -policyType SystemEDGE
```

## ism-updatePolicies 命令-更新策略的配置 (Funclet)

ism-updateSystems 命令用于将最新配置应用于运行指定策略的所有主机。

此命令具有以下格式：

```
ism-updateSystems -policyName policy_name  
-policyType policy_type
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

### 示例：更新策略的配置

本示例将最新配置应用于运行默认策略的所有主机。

```
ism-updateSystems -policyName default -policyType SystemEDGE
```

## ism-updateUpToDateSystems 命令-重新应用策略 (Funclet)

ism-updateUpToDateSystems 命令用于将指定策略的最新配置重新应用于被视为该策略的最新主机的所有主机。

此命令具有以下格式：

```
ism-updateUpToDateSystems -policyName policy_name  
-policyType policy_type
```

**-policyName *policy\_name***

指定要应用的策略名称。

**-policyType *policy\_type***

指定策略类型。

### 示例：重新应用策略

本示例将最新配置重新应用于运行默认策略的所有主机。

```
ism-updateUpToDateSystems -policyName default -policyType SystemEDGE
```

## ism-copyPolicyTemplate 命令—复制策略模板 (Funclet)

ism-copyPolicyTemplate 命令用于通过从现有策略模板复制来创建策略模板。

此命令具有以下格式：

```
ism-copyPolicyTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs  
-newName newName
```

### **-templateName *templateName***

指定模板的名称。

### **-templateType *templateType***

指定策略模板类型。

### **-templateOs *templateOs***

指定策略模板操作系统。

### **-newName *newName***

定义复制的策略模板的新名称。

### 示例：复制策略模板

本示例创建选定策略模板 test 1 的新副本。使用新名称 test 2 创建副本。

```
ism-copyPolicyTemplate -templateName test1 -templateType SystemEDGE -templateOs  
windows -newName test2
```

## ism-deletePolicyTemplate 命令—删除策略模板 (Funclet)

ism-deletePolicyTemplate 命令用于删除现有策略模板。

此命令具有以下格式：

```
ism-copyPolicyTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs
```

### **-templateName *templateName***

指定模板的名称。

### **-templateType *templateType***

指定策略模板类型。

### **-templateOs *templateOs***

指定策略模板操作系统。

### 示例：删除策略模板

本示例删除现有策略模板。

```
ism-deletePolicyTemplate -templateName test3 -templateType SystemEDGE  
-templateOs windows
```

## ism-listPolicyTemplateStatus 命令—列出策略模板状态 (Funclet)

ism-listPolicyTemplateStatus 命令用于列出使用指定策略模板的所有计算机系统的策略模板交付状态。

此命令具有以下格式：

```
ism-copyPolicyTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs
```

### **-templateName *templateName***

指定模板的名称。

### **-templateType *templateType***

指定策略模板类型。

### **-templateOs *templateOs***

指定策略模板操作系统。

### 示例：列出策略模板状态

本示例列出所有计算机系统的策略模板交付状态。

```
ism-listPolicyTemplateStatus -templateName Test1 -templateType SystemEDGE  
-templateOs windows
```

## ism-removePolicyTemplate 命令—删除策略模板 (Funclet)

ism-removePolicyTemplate 命令用于从与给定主机关联的策略列表中删除策略模板。

此命令具有以下格式：

```
ism-removePolicyTemplate -templateName templateName  
-templateType templateType  
-hostName hostName
```

### **-templateName *templateName***

指定模板的名称。

### **-templateType *templateType***

指定策略模板类型。

**-hostName *hostName***

指定主机计算机的名称。

**示例：删除策略模板**

本示例从与指定主机名关联的策略列表中删除策略模板。

```
ism-removePolicyTemplate -templateName templateName -templateType templateType  
- hostName hostName
```

**ism-renamePolicyTemplate 命令—重命名策略模板 (Funclet)**

ism-renamePolicyTemplate 命令用于重命名策略模板。

此命令具有以下格式：

```
ism-renamePolicyTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs  
-newName newName
```

**-templateName *templateName***

指定模板的名称。

**-templateType *templateType***

指定策略模板类型。

**-templateOs *templateOs***

指定策略模板操作系统。

**-newName *newName***

指定重命名的策略模板的新名称。

**示例：重命名策略模板**

本示例重命名选定的策略模板。

```
ism-renamePolicyTemplate -templateName test2 -templateType SystemEDGE  
-templateOs windows -newName test3
```

**ism-updateExceptionSystemsTemplate 命令—应用最新策略模板 (Funclet)**

ism-updateExceptionSystemsTemplate 命令用于为策略模板查找具有策略例外的所有主机并应用最新配置。

此命令具有以下格式：

```
ism-updateExceptionSystemsTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs
```

**-templateName *templateName***

指定模板的名称。

**-templateType *templateType***

指定策略模板类型。

**-templateOs *templateOs***

指定策略模板操作系统。

**示例：更新至最新策略模板**

本示例更新策略模板的所有策略例外并应用最新配置。

```
ism-updateExceptionSystemsTemplate -templateName tpl1 -templateType SystemEDGE  
-templateOs windows
```

**ism-updateOldSystemsTemplate 命令—更新所有旧系统模板 (Funclet)**

ism-updateOldSystemsTemplate 命令用于为所有系统模板查找具有过期配置的所有主机并应用最新配置。

此命令具有以下格式：

```
ism-updateOldSystemsTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs
```

**-templateName *templateName***

指定模板的名称。

**-templateType *templateType***

指定策略模板类型。

**-templateOs *templateOs***

指定策略模板操作系统。

**示例：更新所有旧系统模板**

本示例为所有系统模板查找具有旧配置的所有主机并应用最新策略。

```
ism-updateOldSystemsTemplate -templateName tpl1 -templateType SystemEDGE  
-templateOs windows
```

## ism-updateSystemsTemplate 命令—应用系统模板 (Funclet)

ism-updateSystemsTemplate 命令用于将指定系统模板的最新配置应用于所有主机。

此命令具有以下格式：

```
ism-updateSystemsTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs
```

**-templateName *templateName***

指定模板的名称。

**-templateType *templateType***

指定策略模板类型。

**-templateOs *templateOs***

指定策略模板操作系统。

### 示例：应用系统模板

本示例将最新配置应用于运行系统模板的所有主机。

```
ism-updateSystemsTemplate -templateName tpl1 -templateType SystemEDGE  
-templateOs windows
```

## ism-updateUpToDateSystemsTemplate 命令—重新应用系统模板 (Funclet)

ism-updateUpToDateSystemsTemplate 命令用于将指定系统模板的最新配置重新应用于被视为该系统模板的最新主机的所有主机。

此命令具有以下格式：

```
ism-updateUpToDateSystemsTemplate -templateName templateName  
-templateType templateType  
-templateOs templateOs
```

**-templateName *templateName***

指定模板的名称。

**-templateType *templateType***

指定策略模板类型。

**-templateOs *templateOs***

指定策略模板操作系统。

### 示例：重新应用系统模板

本示例将最新配置重新应用于运行系统模板的所有主机。

```
ism-updateUpToDateSystemsTemplate -templateName tpl1 -templateType SystemEDGE  
-templateOs windows
```

### ism-listPolicyTemplate 命令—列出所有策略模板 (Funclet)

ism-listPolicyTemplate 命令用于列出所有计算机系统的所有策略模板。

此命令具有以下格式：

```
ism-listPolicyTemplate -policyTypeFilter policyTypeFilter
```

**-policyTypeFilter policyTypeFilter**

指定筛选类型以排序和显示所有策略模板。

**默认值：** 空

### 示例：列出所有策略模板

本示例列出所有计算机系统的所有策略模板。

```
ism-listPolicyTemplate -policyTypeFilter policyTypeFilter
```





## 第 3 章： 命令行脚本

---

CA Virtual Assurance 提供您可以从命令提示符窗口运行的平台命令。这些命令与 VMware vCenter 服务器、Solaris Zones、Hyper-V、MSCS 等的 AutoShell 命令相对应。

此部分包含以下主题：

- [命令行指令 \(CLI\)](#) (p. 361)
- [CA VMware vCenter Server CLI 命令](#) (p. 361)
- [CA Hyper-V CLI 命令](#) (p. 442)
- [CA IBM LPAR CLI 命令](#) (p. 511)
- [CA Microsoft Cluster Server CLI 命令](#) (p. 547)
- [CA Solaris Zones CLI 命令](#) (p. 565)
- [CA Cisco UCS 命令](#) (p. 592)
- [管理命令行实用工具](#) (p. 630)
- [通用发现命令](#) (p. 642)
- [收集引擎命令](#) (p. 647)
- [事件命令](#) (p. 665)
- [常规外壳命令](#) (p. 670)
- [帮助中心命令](#) (p. 680)
- [映像命令](#) (p. 688)
- [策略命令](#) (p. 695)
- [策略配置](#) (p. 709)
- [资源管理器命令](#) (p. 711)
- [远程监控命令](#) (p. 757)
- [日志文件](#) (p. 765)

### 命令行指令 (CLI)

命令行指令 (CLI) 可用于为任务编写脚本并使其自动执行。对于可以在 CA Virtual Assurance 用户界面中执行的任何操作，都可以使用这些命令编写脚本。CLI 命令会生成自己的日志文件。

**重要信息：** 请确保具有足够的权限从命令提示符运行 CLI 命令。

### CA VMware vCenter Server CLI 命令

您可以使用 CLI 为 CA VMware vCenter Server 命令编写脚本并使其自动执行，然后根据命令结果运行操作。也可以在 AutoShell 中使用对应的命令。

## dpmvc addesxhost 命令—添加 ESX 主机

dpmvc addesxhost 命令将 ESX 主机添加到 vCenter 服务器中。

此命令具有以下格式：

```
dpmvc addesxhost
-esx_host_name esxhostname
-vc_server vcservername
[-sc sc_url]
[-wait timeout]
[-pre]
[-post]
-esx_host_user esxhostuser
-esx_host_password esxhostpassword
[-management_ip managementip]
[-host_connect_port hostconnectport]
[-folder_name foldername]
[-datacenter_name datacentername]
[-cluster_name clustername]
[-vim_account_user vimaccountuser]
[-vim_account_password vimaccountpassword]
[-locale iso639value]
```

### **-esx\_host\_name esxhostname**

指定托管 VM 的 ESX 服务器的名称。

### **-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

**-wait timeout**

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-esx\_host\_user esxhostuser**

（可选）指定用于访问 ESX 主机的用户名。

**-esx\_host\_password esxhostpassword**

指定用于访问 ESX 主机的密码。

**-management\_ip managementip**

（可选）指定管理 IP。

**-host\_connect\_port hostconnectport**

（可选）指定 ESX 主机用于与 vCenter 服务器通信的端口。

**-folder\_name foldername**

（可选）指定可以添加 ESX 主机的文件夹的名称。

**-datacenter\_name datacentername**

（可选）指定 VM 所在的数据中心。

**-cluster\_name clustername**

（可选）指定可以添加 ESX 主机的群集的名称。

**-vim\_account\_user vimaccountuser**

（可选）指定 VIM 帐户的用户名。

**-vim\_account\_password vimaccountpassword**

（可选）指定用于访问 VIM 帐户的密码。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：添加 ESX 主机**

本示例将 ESX 主机 “esx1” 添加到 vCenter 服务器 “myvcenter” 中。

```
dpmvc addesx -esx_host_name esx1 -vc_server myvcenter -esx_host_user admin  
-esx_host_password topsecret
```

## **dpmvc addvmvdisk 命令—添加虚拟磁盘**

dpmvc addvmvdisk 命令将虚拟磁盘添加到 VM 中。

此命令具有以下格式：

```
dpmvc addvmvdisk  
-datastore_name datastorename  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
[-sc sc_url]  
[-wait timeout]  
[-pre]  
[-post]  
-capacity_in_mb disk_capacity  
[-controller_key controllerkey]  
[-disk_mode  
{append|independent_nonpersistent|independent_persistent|nonpersistent|persistent|undoable}]  
-thin_provisioning {no|yes}  
[-unit_number unitnumber]  
[-locale iso639value]
```

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

"""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-capacity\_in\_mb *disk\_capacity***

指定虚拟磁盘的容量 (MB)。

**-controller\_key *controllerkey***

(可选) 指定磁盘的控制器密钥。

**-disk\_mode {append | independent\_nonpersistent | independent\_persistent | nonpersistent | persistent | undoable}**

(可选) 指定磁盘的模式。

**-thin\_provisioning {no | yes}**

(可选) 指示磁盘是否必须精简开通。

**-unit *disk\_unitnumber***

（可选）指定磁盘的单元编号。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：添加虚拟磁盘**

本示例将虚拟磁盘添加到 VM “myvm” 中。

```
dpmvc addvmvmdisk -datastore_name disk1 -vm_name myvm -vc_server myvcenterserver  
-capacity_in_mb 20000 -thin_provisioning no
```

## dpmvc addvmvnic 命令—添加虚拟 NIC

dpmvc addvmvnic 命令将虚拟 NIC 添加到 VM 中。

此命令具有以下格式：

```
dpmvc addvmvnic  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
[-sc sc_url]  
[-wait timeout]  
[-pre]  
[-post]  
-devicetype {e1000 | vmxnet}  
-network_name networkname  
[-mac_address macaddress]  
-wake_on_lan_enabled {no | yes}  
[-locale iso639value]
```

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

"""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-devicetype {*e1000* | *vmxnet*}**

表示网络设备的类型。

**-network\_name *networkname***

指定 NIC 必须连接到的虚拟网络。您可以根据以下命名约定区分标准交换机和分布式虚拟交换机的名称:

- 对于标准交换机, 其名称为网络名称。
- 对于分布式虚拟交换机, 其名称为 `dvPort` 组名称后跟用括号括起来的分布式虚拟交换机名称: `dvPortGroupName (dvSwitchName)`

**-mac\_address *macaddress***

(可选) 指定网络设备的 MAC (介质访问控制) 地址。

**-wake\_on\_lan\_enabled {no | yes}**

表示是否在启用 LAN 时启动 VM。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

#### 示例：添加 vNIC

本示例将虚拟 NIC 添加到 VM 中。

```
dpmvc addvmvnic -vm_name myvm -vc_server myvcenterserver -devicetype vmxnet  
-network_name net1 -wake_on_lan_enabled no
```



## dpmvc clone 命令—克隆 VM

dpmvc clone 命令用于克隆 VM。

此命令具有以下格式：

```
dpmvc clone
[-sc sc_url]
-datacenter_name datacentername
-compute_resource_name name
-esx_host_name esxhostname
-datastore_name datastorename
-vm_name vmname
-resource_pool_name resourcepool
-spec_name specificationname
[-auto_deploy value]
[-deploy_template templatename]
[-network_connection
"nic=sequence_number,network_name=name,ip_address=ip_address,def_gateway=defa
ult_gateway,alt_gateway=alt_gateway,subnet_mask=subnet_mask[,win_dns_server=w
ins_dns_server,win_alt_dns=wins_alt_dns,wins_primary=wins_primary,wins_second
ary=wins_secondary"]]]
[-global_dns_search_suffix suffix1[,suffix2,suffix3,...]]
[-linux_domain_name domain_name]
[-linux_dns_servers primary_dns=value,secondary_dns=value,tertiary_dns=value]
[-create_disk MB,datastorename,controller]
[-modify_disk key, datastorename]
[-set_memory number]
[-set_cpu number]
-vm_os_username username
[-vm_os_password password]
[-vc_server vcservername]
[-vc_user vcuser]
[-vc_password vcpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID] {-template_name templatename |
-vm_name_cloned_from -vmname}
[-scalability_server scalabilityservername]
[-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-compute\_resource\_name *name***

指定在其中创建 VM 的群集或 VMware ESX 主机。

**-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-vm\_name *vmname***

指定 VM。

**-resource\_pool\_name *resourcepool***

指定您希望从中选择用于克隆的 VM 的资源池的名称。

**-spec\_name *specificationname***

(可选) 指定要用于所克隆虚拟机的规格的名称。

**-auto\_deploy {*yes*|*no*}**

指定是否自动部署 CA Virtual Assurance 代理。选项包括:

**是**

自动部署 CA Virtual Assurance 代理。

**否**

防止自动部署 CA Virtual Assurance 代理。

**默认值:** `no`

**-deploy\_template *templatename***

（可选）指定在 CA Virtual Assurance 中创建的通用模板的名称。

**注意：**请勿将该模板与由 VMware vCenter 创建并管理的模板相混淆。

**-network\_connection**

**"nic=sequence\_number,network\_name=name,ip\_address=ip\_address,def\_gateway=default\_gateway,alt\_gateway=alt\_gateway,subnet\_mask=subnet\_mask,win\_dns\_server=wins\_dns\_server,win\_alt\_dns=wins\_alt\_dns,wins\_primary=wins\_primary,wins\_secondary=wins\_secondary"**

**Windows:**

（可选）为 VM 创建网络连接。该参数可以多次指定。序列号是以 (1) 开头的整数值，用于标识网络连接。如果参数包含任何空格，则必须用括号将整组参数括起来。通过输入没有值的名称（例如，

“nic=123,alt\_gateway=”）可以重置除 nic、network\_name、ip\_address、def\_gateway 和 subnet\_mask 以外的所有值。

**-network\_connection**

**"nic=sequence\_number,network\_name=name,ip\_address=ip\_address,def\_gateway=default\_gateway,alt\_gateway=alt\_gateway,subnet\_mask=subnet\_mask"**

**Linux:**

（可选）创建一个或多个网络连接。该参数可以多次指定。序列号是以 (1) 开头的整数值，用于标识网络连接。如果参数包含任何空格，则必须用括号将整组参数括起来。通过输入没有值的名称（例如，

“nic=123,alt\_gateway=”）只能重置 alt\_gateway。

**-global\_dns\_search\_suffix *suffix1{,suffix2,suffix3,...}*****Linux:**

指定一个或多个 DNS 搜索后缀。

**Windows:**

（可选）指定一个或多个 DNS 搜索后缀。

**-linux\_domain\_name *domain\_name*****Linux:**

（可选）指定 Linux VM 的域名。

**-linux\_dns\_servers*****primary\_dns=value{,secondary\_dns=value,tertiary\_dns=value}*****Linux:**

(可选) 指定 Linux VM 的 DNS 服务器的 IP 地址。可以省略一个或多个值 (例如, `primary_dns=123.123.123.13,secondary_dns=,tertiary_dns=123.123.123.15`), 以便取消设置该值。

**-create\_disk *MB,datastore\_name,controller***

(可选) 创建一个或多个附加的硬盘驱动器。该参数可以多次指定。

**MB**

定义硬盘驱动器的大小 (MB)。

**Datastore\_name**

指定该硬盘驱动器的数据存储的名称。当前, 每个 VM 仅支持一个数据存储。

**控制器**

指定该硬盘驱动器的控制器密钥。

**-modify\_disk {*key,datastore\_name*}**

(可选) 指定硬盘驱动器驻留的数据存储。可以多次指定。

**key**

指定现有磁盘。

**datastore\_name**

指定该磁盘移动到的数据存储的名称。

**-set\_memory *number***

(可选) 定义用于克隆的 VM 的内存量 (MB)。该值会覆盖在模板中指定的值。

**-set\_cpu *number***

(可选) 定义克隆的 VM 的 CPU 的数量。该值会覆盖在模板中指定的值。

**-vm\_os\_username *user name***

指定所克隆 VM 的用户。自动部署映像时, 还可以使用该用户名进行身份验证。

**Windows:** 必须在自定义规格中定义用户名。

**Linux:** 必须与模板中定义的用户名相同。

**-vm\_os\_password *password***

（可选）指定所克隆 VM 的用户密码。自动部署映像时，还可以使用该密码进行身份验证。

**Windows:** 必须与自定义规格中定义的密码相同。

**Linux:** 必须与模板中定义的密码相同。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户（-vc\_user 选项）和组件（-auth\_comp 选项），则将其检索加密密码。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 dpmutil set auth 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-template\_name *templatename***

指定要用于所克隆虚拟机的模板的名称。该参数取代了 -vm\_name\_cloned\_from，不能两者一起指定。

**-vm\_name\_cloned\_from *vmname***

指定将用作克隆的虚拟机的模板的虚拟机的名称。该参数用于替换 -template\_name，您无法同时指定两者。

**-resource\_pool\_name *resourcepool***

指定您希望从中选择用于克隆的 VM 的资源池的名称。

**-scalability\_server *servername***

（可选）指定要用于软件分发的 Unicenter DSM 可扩展性服务器。

**-spec\_name *specificationname***

（可选）指定要用于所克隆虚拟机的规格的名称。

**-compute\_resource\_name *name***

指定在其中创建 VM 的群集或 VMware ESX 主机。

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

"""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {*yes*|*no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-encrypted\_password {*yes*|*no*}**

(可选) 指定您是否希望加密 VC 用户密码。

默认值: Yes

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

### 示例：使用全局凭据克隆 VM

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -vc_server vc_server_1
```

### 示例：使用授权文件和组件克隆 VM

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -vc_server vc_server_1 -vc_user VCAdmin1
-auth_file c:\localauth.dat -auth_comp Imaging
```

### 示例：使用全局凭据克隆 VM 并创建两个 CPU

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”，并为该 VM 创建两个 CPU。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -set_cpu 2 -vc_server vc_server_1
```

### 示例：使用默认授权文件克隆 VM 并将内存设置为 4096 MB

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”，并将内存设置为 4096。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -set_memory 4096 -vm_server vm_server_1
-vm_user VCAdmin1
```

**示例：使用默认授权文件克隆 VM 并创建两个硬盘**

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”，并为数据存储 storage1 创建两个硬盘。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -create_disk 10000,storage1,1000
-create_disk 10000,storage1,2001 -vm_server_1 -vm_user VCAAdmin1
```

**示例：克隆 VM 并创建网络连接**

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”，创建网络连接并设置初始值。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -network_connection "nic=1,network_name=VM
Network,ip_address=123.321.5.22,def_gatwat=123.321.1.108,subnet_mask=255.255
.255.0,win_dns_server=123.321.42.1,win_alt_dns=123.321.3.101,wins_primary=123
.321.3.3"
```

**示例：使用默认授权文件从已关闭电源的 VM 克隆 VM**

此示例在数据中心 DCA/MyCity 上使用 VM “testvm02”（而不是模板）创建新的 VM “testvm01”。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -datacenter_name "DCA/MyCity" -vm_name testvm01 -datastore_name
storage1
-esx_host_name ussd-dpmvc.ca.com -resource_pool_name Resources/DPMTest
-compute_resource_name DPMCluster/ussdCluster -vm_name_cloned_from testvm02
-auto_deploy yes -spec_name w2k3_spec_01 -vm_os_username administrator
-vm_os_password mypassword -vc_server vm_server_1.myco.com -vc_user VCAAdmin1
```

**示例：使用默认授权文件克隆 VM 并修改现有硬盘**

此示例在数据中心 DCA/MyCity 上使用模板 “BaseW2k3” 创建新的 VM “testvm01”，并为数据存储 storage1 创建两个硬盘。在克隆操作完成时，将自动部署 CA Virtual Assurance 代理。

```
dpmvc clone -vm_name testvm01 -template_name Templates/BaseW2k3 -auto_deploy yes
-spec_name w2k3_spec_01 -datacenter_name DCA/MyCity -datastore_name storage1
-compute_resource_name DCACluster/ClusterServerA -vm_os_username administrator
-vm_os_password testpassword -create_disk 10000,storage1,1000 -modify_disk
2000,storage2
-vm_server_1 -vm_user VCAAdmin1
```



## dpmvc createdatastore 命令—创建数据存储

dpmvc createdatastore 命令用于创建新的数据存储。

此命令具有以下格式：

```
dpmvc createdatastore
-datastore_name datastorename
-esx_host_name esxhostname
-vc_server vcservername
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
-datastore_type {local|nas|vmfs}
[-vmfs_device_path path]
[-vmfs_block_size_mb blocksize]
[-local_path lpath]
[-nas_user_name nasuser]
[-nas_password naspasswd]
[-nas_remote_hostname nasremhost]
[-nas_remote_path nsarempath]
[-nas_access_mode {read-only|read-write}]
[-sc sc_url]
[-locale iso639value]
```

### **-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

### **-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

### **-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

### **-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

### **-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值：Yes

### **-datastore\_type {vmfs|nas|local}**

指定要创建的数据存储的类型：VMFS、NAS 或本地。

**-vmfs\_device\_path *path***

（可选）指定在创建新的数据存储时要使用的 VMFS 磁盘的设备路径。您可以使用 “getavailablescsidisks” 命令获取该路径。

**-vmfs\_block\_size\_mb *blocksize***

（可选）指定在创建新的数据存储时要使用的块大小（以 MB 为单位）。可用的块大小为 1、2、4 和 8。对应的 VM 磁盘文件最大大小分别为 256GB、512GB、1024GB 以及 2048GB。

**-local\_path *lpath***

（可选）在创建本地数据存储时，该路径指定用于创建数据存储的 ESX 主机本地文件系统路径。

**-nas\_user\_name *nasuser***

（可选）为创建 NAS 数据存储指定远程主机用户名。

**-nas\_password *naspasswd***

（可选）为创建 NAS 数据存储指定远程主机密码。

**-nas\_remote\_hostname *nasremhost***

（可选）指定托管基于网络的存储的服务器主机名。

**-nas\_remote\_path *nasrempath***

（可选）指定要用于 NAS 数据存储的远程服务器上的文件系统路径

**-nas\_access\_mode *nasaccmode***

（可选）指定对基于网络的存储的期望访问权限类型，该存储将用于 NAS 数据存储。可用的模式为“只读”和“读写”。如果不指定，则默认模式为“读写”。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

***port***

定义服务控制器服务器的侦听端口。

示例：80

## dpmvc cycle 命令—循环 VM

dpmvc cycle 命令打开电源、关闭电源、重置或挂起 VM。

此命令具有以下格式：

```
dpmvc cycle
[-sc sc_url]
-powerop {poweron|poweroff|reset|suspend|shutdown guest}
-datacenter_name datacentername
-vm_name vmname
[-vc_server vcserver]
[-vc_user vcuser]
[-vc_password vcpasssword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
[-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspasssword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-powerop {poweron|poweroff|reset|suspend|shutdown guest}**

指定要在 VM 上执行的电源操作。选项包括：

#### **poweron**

打开 VM 电源。

#### **poweroff**

关闭 VM 电源。

#### **reset**

重置 VM。

**挂起**

暂时挂起 VM。

**shutdown guest**

正常关闭来宾操作系统。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户（-vc\_user 选项）和组件（-auth\_comp 选项），则将其检索加密密码。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-wait *timeout***

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-encrypted\_password {yes|no}**

（可选）指定您是否希望加密 VC 用户密码。

默认值: Yes

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：当数据中心内有多个 vCenter 服务器时关闭 VM**

本示例从特定 vCenter 服务器关闭 VM。

```
dpmvc cycle -powerop poweroff -vc_server my_server1 -datacenter_name my_dc -
vm_name my_vm
```

**示例：在单 vCenter 服务器环境中，使用全局凭据关闭 VM**

本示例关闭属于数据中心 “DCA/MyCitySan Diego” 的 VM “testvm”。

```
dpmvc cycle -vm_name testvm -powerop poweroff -datacenter_name DCA/MyCity
```

**示例：使用默认授权文件关闭 VM**

本示例关闭属于数据中心 “DCA/MyCitySan Diego” 的 VM “testvm”。

```
dpmvc cycle -vm_name testvm -powerop poweroff -datacenter_name DCA/MyCity
-vc_server vc_server_1
```

**示例：使用指定的授权文件和组件打开 VM**

本示例打开属于数据中心 “DCA/MyCitySan Diego” 的 VM “testvm”。

```
dpmvc cycle -vm_name testvm -powerop poweron -datacenter_name DPM/San Diego
-vc_server vc_server_1 -vc_user VCAdmin1 -auth_file c:\localauth.dat
-auth_component Imaging
```

**示例：使用默认授权文件和指定的组件打开 VM**

本示例打开属于数据中心 “DCA/MyCitySan Diego” 的 VM “testvm”。

```
dpmvc cycle -vm_name testvm -powerop poweron -datacenter_name DPM/San Diego
-vc_server vc_server_1 -vc_user VCAdmin1 -auth_component Imaging
```

## dpmvc datastore 命令—获取数据存储属性

dpmvc datastore 命令检索特定 VMware vCenter 数据存储的可用空间和容量设置。

此命令具有以下格式：

```
dpmvc datastore
[-sc sc_url]
-getproperty {all|capacity|freespace}
-datacenter_name datacentername
-datastore_name datastorename
[-vc_server vcserver]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-getproperty {all|capacity|freespace}**

（可选）指定要检索哪个属性。选项包括：

**all**

检索数据存储中的容量和可用空间。

**capacity**

检索数据存储中的容量。

**freespace**

检索数据存储中的可用空间。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：使用全局凭据获取数据存储的全部属性**

本示例检索数据存储的可用空间和容量。

```
dpmvc datastore -getproperty all datacenter_name DCA/MyCity  
-datastore_name storage1 -vc_server vc_server_1
```

**示例：获取数据存储的全部属性**

本示例检索数据存储的可用空间和容量。

```
dpmvc datastore -getproperty all datacenter_name DCA/MyCity  
-datastore_name storage1 -vc_server vc_server_1
```

**示例：获取数据存储的全部属性**

本示例获取数据存储的可用空间和容量。

```
dpmvc datastore -getproperty all datacenter_name DCA/MyCity  
-datastore_name storage1 -vc_server vc_server_1
```

## dpmvc delete 命令—销毁 VM

dpmvc delete 命令删除电源已关闭的 VM。使用此命令清理并释放未用过的资源。

**重要信息！** 在发出该命令之前，请验证是否已备份所有重要数据。该命令会删除 VM 数据存储以及 VM 的数据文件，包括磁盘映像。

此命令具有以下格式：

```
dpmvc delete  
[-sc sc_url]  
-datacenter_name datacentername  
-vm_name vmname  
[-vc_server vcservername]  
[-vc_user vcuser]  
[-vc_password vcpassword]  
[-auth_file authorizationfilename]  
[-auth_comp componentID] [-wait [timeout]]  
[-pre]  
[-post]  
[-ws_user wsuser]  
[-ws_password wspassword]  
[-prompt {yes|no}]  
[-encrypted_password {yes|no}]  
[-locale iso639value]
```



**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (`-auth_file`) 指定用户 (`-vc_user` 选项) 和组件 (`-auth_comp` 选项)，则将其检索加密密码。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-wait timeout**

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user wsuser**

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password wspassword**

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-encrypted\_password {yes|no}**

(可选) 指定您是否希望加密 VC 用户密码。

默认值: Yes

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

**示例：在单 vCenter 服务器环境中，使用全局凭据销毁 VM**

本示例销毁 VM “vm11”。

```
dpmvc delete -vm_name vm11 -datacenter_name "lab 444"
```

**示例：在多 vCenter 服务器环境中，使用全局凭据销毁 VM**

本示例销毁 VM “vm11”。

```
dpmvc delete -vm_name vm11 -datacenter_name "lab 444" -vc_server vc_server_1
```

**dpmvc distributedswitch 命令—管理虚拟分布式交换机**

dpmvc distributedswitch 命令用于管理虚拟分布式交换机。

- 将新的虚拟分布式交换机添加到数据中心
- 更新虚拟分布式交换机的属性
- 删除虚拟分布式交换机
- 为虚拟分布式交换机创建新的分布式端口组
- 更新虚拟分布式交换机的端口组属性
- 从虚拟分布式交换机中删除分布式端口组
- 重命名虚拟分布式交换机的分布式端口组

此命令具有以下格式：

```
dpmvc distributedswitch {-vds_add | -vds_update}
-vc_server vcservername
[-datacenter_name datacentername]
-vds_folder vdsfolder
-switch_name switchname
[-hostnics hostname1:nic1,nic2,...nicn [;hostname1:nic2,...nicn ]]
[-uplink_port_names uplink1[,uplink2,...,uplinkn]]
[-maxports maxports]
[-sc sc_url]
[-wait timeout]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc distributedswitch -vds_remove
-vc_server vcservername
-switch_name switchname
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc distributedswitch -add_portgroup
-vc_server vcservername
-switch_name switchname
-portgroup_name portgroupname
[-bindtype earlyBinding | ephemeral | lateBinding]
[-vlan vlanID]
[-numports numberofports]
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc distributedswitch -update_portgroup
-vc_server vcservername
-portgroup_name portgroupname
[-portgroup_newname portgroupnewname]
[-bindtype earlyBinding | ephemeral | lateBinding]
[-vlan vlanID]
[-numports numberofports]
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc distributedswitch -remove_portgroup
-vc_server vcservername
-portgroup_name portgroupname
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc distributedswitch -rename_portgroup
-vc_server vcservername
-portgroup_name portgroupname
-portgroup_newname portgroupnewname
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vds\_folder *vdsfolder***

在 CA Virtual Assurance 资源管理器窗格中指定虚拟分布式交换机文件夹。

**-switch\_name *switchname***

指定操作所针对的交换机名称。

**-hostnics *hostname1:nic1,nic2,...nicn [;hostname1:nic2,...nicn ]***

(可选) 指定与 ESX 主机成员相关联的 NIC 列表。

**-uplink\_port\_names *uplink1[,uplink2,...,uplinkn]***

(可选) 指定要使用的上行链路端口名称的逗号分隔列表。

**-maxports *maxports***

(可选) 指定端口的最大数目。

**-bindtype *earlyBinding | ephemeral | lateBinding***

(可选) 指定端口组的绑定类型。有效值是：

#### **earlyBinding**

在 VM 绑定至端口组时分配端口。这种类型的绑定可确保始终连接，但永久保留端口。此绑定类型为默认值。

#### **lateBinding**

如果 VM 电源打开且其 NIC 处于连接状态，则将端口分配给该 VM。当 VM 电源关闭或其 NIC 断开连接时，此绑定类型将重新分配端口。LateBinding 可通过 vCenter 进行配置。

#### **ephemeral**

如果 VM 电源打开且其 NIC 处于连接状态，则将端口分配给该 VM。当 VM 电源关闭或其 NIC 断开连接时，此绑定类型将重新分配端口。Ephemeral 绑定可通过 ESX 主机和 vCenter 进行配置。

**-numports *numberofports***

(可选) 指定端口组的端口数。

**-portgroup\_name *portgroupname***

指定端口组名称。

**-portgroup\_newname *portgroupnewname***

指定新的端口组名称。

**-vlan *vlanID***

(可选) 指定用于虚拟端口组操作的整数值 (vlan ID)。

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

"""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：创建新的虚拟分布式交换机**

此示例将创建新的虚拟分布式交换机。

```
dpmvc distributedswitch -vds_add -vc_server vc5master -datacenter_name dc3
-vds_folder vds -switch_name vds1
```

**示例：更新现有的虚拟分布式交换机**

此示例将更新虚拟分布式交换机。它为 MYSERVER1 指定 NIC 并从虚拟分布式交换机中删除 MYSERVER2。如果要从虚拟分布式交换机中删除主机，请指定服务器名称，但不要为其分配 NIC（服务器名称后跟随一个冒号，如 MYSERVER2:）。

```
dpmvc distributedswitch -vds_update -vc_server VAS-VC5 -datacenter_name VC5
-switch_name vdistSwitch -hostnics MYSERVER1:nxmgt2,tmp2;MYSERVER2:
-ws_user admin -ws_password ca_admin -post
```

该命令用于更新 vdistSwitch:

- 主机 MYSERVER1 正在使用 NIC nxmgt2 和 tmp2
- 主机 MYSERVER2 已从交换机中删除

**dpmvc entermaintenancemode 命令-设置为维护模式**

dpmvc entermaintenancemode 命令用于将 ESX 主机设置为维护模式

此命令具有以下格式:

```
dpmvc entermaintenancemode
-esx_host_name esxhostname
-vc_server vcservername
[-sc sc_url]
[-wait timeout]
[-pre]
[-post]
-timeout timeout
[-evacuate_powered_off_vm {no | yes}]
[-locale iso639value]
```

**-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-timeout *timeout***

指定进入或退出维护模式的超时时间（秒）。

**-evacuate\_powered\_off\_vm {no | yes}**

（可选）指定是否在 VM 停止后退出。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。



### 示例：进入维护模式

本示例设置 ESX 主机使其进入维护模式。

```
dpmvc entermaintenancemode -esx_host_name esx1 -vc_server myvcserver -timeout 60
```

## dpmvc exitmaintenancemode 命令-退出维护模式

dpmvc exitmaintenancemode 命令用于设置 ESX 主机，使其退出维护模式。

此命令具有以下格式：

```
dpmvc exitmaintenancemode  
-esx_host_name esxhostname  
-vc_server vcservername  
[-sc sc_url]  
[-wait timeout]  
[-pre]  
[-post]  
-timeout timeout  
[-locale iso639value]
```

### **-esx\_host\_name esxhostname**

指定 VM 驻留的 VMware ESX 服务器。

### **-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

### **-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

**-wait timeout**

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-timeout timeout**

指定进入或退出维护模式的超时时间（秒）。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：退出维护模式**

本示例设置 `esx1` ESX 主机，使其退出维护模式。

```
dpmvc exitmaintenancemode -esx_host_name esx1 -vc_server myvcserver  
-timeout 60
```

## dpmvc faulttolerance 命令—指定容错操作

**dpmvc faulttolerance** 命令用于打开、关闭、启用、禁用、测试容错，或迁移辅助 VM。

此命令具有以下格式：

```
dpmvc faulttolerance
{-turn_on|-turn_off|-enable|-disable|-test_ft|-migrate_secondary }
[-datacenter_name datacentername]
-vm_name vmname
[-esx_host_name esxhostname]
-vc_server vcservername
[-sc sc_url]
[-wait timeout]
[-pre]
[-post]
[-locale iso639value]
```

### **-turn\_on**

对指定 VM 打开容错。

（可选）指定辅助 VM 的 ESX 主机名称。

### **-turn\_off**

对指定 VM 关闭容错。

**注意：** 该操作将删除辅助 VM。

### **-enable**

对指定 VM 启用容错。

### **-disable**

对指定 VM 禁用容错。

### **-test\_ft**

在指定 VM 上测试容错。

### **-migrate\_secondary**

将容错辅助 VM 迁移到其他 ESX 服务器上。-esx\_host\_name 选项是必需选项。

### **-vm\_name *vmname***

指定 VM。

### **-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

### **-esx\_host\_name *esxhostname***

（可选）指定 VM 所在 VMware ESX 服务器。

**-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait timeout**

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

## dpmvc getavailablescsidisks 命令—获取可用的 SCSI 磁盘

dpmvc getavailablescsidisks 命令用于列出可用的 SCSI 磁盘。

此命令具有以下格式：

```
dpmvc getavailablescsidisks  
[-datastore_name datastorename]  
-esx_host_name esxhostname  
-vc_server vcservername  
[-ws_user wsuser]  
[-ws_password wspassword]  
[-prompt {yes|no}]  
[-sc sc_url]  
[-locale iso639value]
```

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “`native`”。

## dpmvc gethosthba 命令—获取主机总线适配器

`dpmvc gethosthba` 命令用于检索在指定 ESX 主机上配置的主机总线适配器列表。

此命令具有以下格式:

```
dpmvc gethosthba
-esx_host_name esxhostname
-vc_server vcservername
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-sc sc_url]
[-locale iso639value]
```

**-esx\_host\_name *esxhostname***

指定 VM 驻留的 VMware ESX 服务器。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: 80

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

## dpmvc getresources 命令-获取 VM 资源

`dpmvc getresources` 命令用于检索 CPU 或内存份额限制或保留。

此命令具有以下格式：

```
dpmvc getresources
[-sc sc_url]
-vm_name vmname
-datacenter_name datacentername
-resource {all|cpulimit|cpureserv|memlimit|memresrv}
[-vc_server vcserver]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-vm\_name *vmname***

指定 VM。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-resource {*all* /*vm\_cpulimit* /*vm\_cpureserv* /*vm\_memlimit* /*vm\_memresrv*}**

指定是检索特定资源还是所有资源。选项包括:

***all***

检索用于 VM 的所有 CPU 和内存份额限制和保留。

***vm\_cpulimit***

检索用于 VM 的 CPU 份额数的限制。

***vm\_cpureserv***

检索专用于 VM 的 CPU 份额数。

***vm\_memlimit***

检索用于 VM 的内存份额数的限制。

***vm\_memresrv***

检索专用于 VM 的内存份额数。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。



**-ws\_user wsuser**

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password wspassword**

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 获取 VM 的所有资源**

本示例检索 usa-vm2 的所有资源。

```
dpmvc getresources -resource all -datacenter_name MyCity -vm_name usa-vm2
```

**示例: 获取 VM 的 CPU 限制**

本示例检索 VM uss-vm3 的单个资源 CPU 限制。

```
dpmvc getresources -resource cpulimit -datacenter_name MyCity  
-vm_name uss-vm3 -vc_server vc_server_1
```

**dpmvc getshares 命令-获取 VM 份额**

dpmvc getshares 命令用于查看分配给某个 VM 的 CPU 或内存份额的数量。

此命令具有以下格式:

```
dpmvc getshares  
[-sc sc_url] {-cpu|-memory}  
-datacenter_name datacentername  
-vm_name vmname  
[-vc_server vcserver]  
[-pre]  
[-post]  
[-ws_user wsuser]  
[-ws_password wspassword]  
[-prompt {yes|no}]  
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-cpu | -memory**

显示分配给某个 VM 的 CPU 或内存份额的数量。份额用于确定与其他正在运行的 VM 相比, 分配给哪些 VM 的可用物理资源的比例较高。如果向某个 VM 分配的份额较多, 则会向该 VM 分配较多的物理资源。如果向某个 VM 分配的份额较少, 则会向该 VM 分配较少的物理资源。

**-cpu**

指定正在检索 CPU 值。不要与 -memory 一起使用。

**-memory**

指定正在检索内存值。不要与 -cpu 一起使用。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：获取 VM 的 CPU 值**

本示例检索 usa-vm2 的 CPU 值。

```
dpmvc getshares -resource all -datacenter_name MyCity -vm_name usa-vm2
-vc_server vc_server_1
```

**示例：在单 vCenter 服务器环境中获取某个 VM 的 CPU 值**

本示例检索 usa-vm2 的 CPU 值。

```
dpmvc getshares -resource all -datacenter_name MyCity -vm_name usa-vm2
```

**示例：在多 vCenter 服务器环境中，使用全局凭据获取某个 VM 的内存值**

本示例显示获取 VM uss-vm3 的单个资源 CPU 限制。

```
dpmvc getshares -resource cpulimit -datacenter_name MyCity -vm_name uss-vm3
-vc_server vc_server_1
```

**dpmvc getsnapshots 命令-获取 VM 快照**

dpmvc getsnapshots 命令用于列出 VM 的快照。

此命令具有以下格式：

```
dpmvc getsnapshots
[-sc sc_url]
-datacenter_name datacentername
-vm_name vmname
[-vc_server vcserver]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {*yes*|*no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: `Yes`

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “`native`”。

**示例：列出快照**

本示例列出某个 VM 的快照。

```
dpmvc getsnapshots -vc_server my_server1 -datacenter_name my_dc
                  -vm_name my_vm
```

**dpmvc imgjobcheck 命令-获取克隆作业状态**

dpmvc imgjobcheck 命令用于获取特定 CA Virtual Assurance 作业 ID 或特定 VC 任务 ID 的克隆作业状态。为某些 dpmvc 命令提供作业 ID。使用该命令验证 VC 上的状态。

此命令具有以下格式：

```
dpmvc imgjobcheck
[-sc sc_url]
-status {job ID/vc task ID}
[-vc_server vcserver]
[-vc_user vcuser]
[-vc_password vcpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://*hostname:port*/...

***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

***port***

定义服务控制器服务器的侦听端口。

示例：80

**-status *jobID***

指定用于获取作业状态的作业 ID。

**-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user vcuser**

(可选) 指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password vcpassword**

(可选) 指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户 (-vc\_user 选项) 和组件 (-auth\_comp 选项)，则将其检索加密密码。

**-auth\_file authorizationfilename**

(可选) 指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 dpmutil set auth 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp componentID**

(可选) 指定可用于对主机和用户进行分组的组件 ID。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user wsuser**

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password wspassword**

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-encrypted\_password {yes|no}**

(可选) 指定您是否希望加密 VC 用户密码。

**默认值:** Yes

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例：在单 vCenter 服务器环境中，使用作业 ID 获取克隆作业状态**

本示例使用 CA Virtual Assurance 作业 ID 获取克隆作业状态。

```
dpmvc imgjobcheck -status 42
```

**示例：在单 vCenter 服务器环境中，使用 VC 任务 ID 获取克隆作业状态**

本示例使用 VC 任务 ID 获取克隆作业状态。

```
dpmvc imgjobcheck -status task -21099
```

## dpmvc migrate 命令-迁移 VM

dpmvc migrate 命令用于将 VM 从一台主机服务器迁移到另一台主机服务器。

此命令具有以下格式：

```
dpmvc migrate
[-sc sc_url]
-datacenter_name datacentername
-vm_name vm -migrate_to_target_host name
-migrate_to_target_resource_pool_name name
-migrate_to_compute_resource_name name
[-vc_server vcserver]
[-vc_user vcuser]
[-vc_password vcpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID] [-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-migrate\_to\_target\_host *name***

指定正在将 VM 迁移到的目标主机的名称。

**-migrate\_to\_target\_resource\_pool\_name *name***

指定目标资源池。

**-migrate\_to\_compute\_resource\_name *name***

指定正在将 VM 迁移到的目标群集或 VMware ESX 主机的名称。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户（-vc\_user 选项）和组件（-auth\_comp 选项），则将其检索加密密码。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。



**-wait timeout**

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user wsuser**

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password wspassword**

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值：Yes

**-encrypted\_password {yes|no}**

（可选）指定您是否希望加密 VC 用户密码。

默认值：Yes

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：使用默认授权文件迁移 VM**

本示例将 VM “MyVM1” 从数据中心 DCA/MyCity 迁移到 ServerC-dcavc。

```
dpmvc migrate -vm_name MyVM1 -datacenter_name DCA/MyCity
-migrate_to_compute_resource_name DCACluster/ClusterA
-migrate_to_target_resource_pool_name Resources/DCTest
-migrate_to_target_host ServerC-dcavc.MyCompany.com
-vm_server VM_Server1 -vc_user VCAAdmin1
```

**dpmvc removedatastore 命令—删除数据存储**

dpmvc removedatastore 命令用于删除数据存储。

此命令具有以下格式：

```
dpmvc removedatastore
-datastore_name datastorename
-esx_host_name esxhostname
-vc_server vcservername
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-sc sc_url]
[-locale iso639value]
```

**-datastore\_name *datastorename***

指定新 VM 位于的数据存储的名称。

**-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-ws\_user *wsuser***

（可选）指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

## dpmvc removevmvdisk 命令-删除虚拟磁盘

`dpmvc removevmvdisk` 命令用于从 VM 中删除虚拟磁盘。

此命令具有以下格式:

```
dpmvc removevmvdisk
[-datacenter_name datacentername]
-vm_name vmname
-vc_server vcservername
[-sc sc_url]
[-wait timeout]
[-pre]
[-post]
-disk_device_key diskdevicekey
-delete_data {yes | no}
[-locale iso639value]
```

**-datacenter\_name *datacentername***

(可选) 指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-disk\_device\_key *diskdevicekey***

指定磁盘的设备密钥的整数，例如 2000。

**-delete\_data {yes | no}**

指示删除磁盘时是否删除数据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：删除虚拟磁盘**

本示例从 VM myvm 中删除设备密钥为 2000 的虚拟磁盘。

```
dpmvc removevmdisk -vm_name myvm -vc_server myvcserver -disk_device_key 2000
-delete_data yes
```

**dpmvc removevmvnic 命令-删除虚拟 NIC**

dpmvc removevmvnic 命令用于从 VM 中删除虚拟 NIC。

此命令具有以下格式：

```
dpmvc removevmvnic
[-datacenter_name datacentername]
-vm_name vmname
-vc_server vcservername
[-sc sc_url]
[-wait timeout]
[-pre]
[-post]
-device_key devicekey
[-locale iso639value]
```

**-datacenter\_name *datacentername***

（可选）指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-wait timeout**

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-device\_key devicekey**

指定网络接口的设备密钥的整数, 例如 3000。

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 删除 vNIC**

本示例从 VM 中删除 vNIC。

```
dpmvc removevmnic -vm_name myvm -vc_server myvcserver -device_key 3000
```

## dpmvc setcpu 命令-设置 VM 的 CPU 数目

dpmvc setcpu 命令用于设置 VM 的 CPU 数目。

此命令具有以下格式:

```
dpmvc setcpu  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
-set_cpu number  
[-sc_sc_url]  
[-wait timeout]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-set\_cpu *number***

(可选) 定义克隆的 VM 的 CPU 的数量。该值会覆盖在模板中指定的值。

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

"""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：设置 VM 的 CPU 数目**

本示例将 VM myvm 的 CPU 数目设置为 4。

```
dpmvc setcpu -vm_name myvm -vc_server myvcserver -set_cpu 4
```

## **dpmvc setmemory 命令-设置 VM 的内存大小**

dpmvc setmemory 命令用于设置 VM 的内存大小。

此命令具有以下格式：

```
dpmvc setmemory  
[-datacenter_name datacentername]  
-vm_name vmname  
-vc_server vcservername  
-set_memory number  
[-sc sc_url]  
[-wait timeout]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-set\_memory *number***

（可选）定义用于克隆的 VM 的内存量 (MB)。该值会覆盖在模板中指定的值。



**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-wait *timeout***

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

"""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：设置 VM 的内存大小**

本示例将 VM `myvm` 的内存大小设置为 4 GB。

```
dpmvc setmemory -vm_name myvm -vc_server myvcserver -set_memory 4000
```

## dpmvc setresources 命令-设置 VM 资源

dpmvc setresources 命令用于调整 CPU 或内存份额限制或保留。

此命令具有以下格式：

```
dpmvc setresources
[-sc sc_url] {-setcpulimit|-setcpureserve|-setmemlimit|-setmemresrv} value
-datacenter_name datacentername
-vm_name vmname
[-vc_server vcservername]
[-vc_user vcuser]
[-vc_password vcpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
[-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **setcpulimit**

定义用于 VM 的 CPU 份额数的限制。

### **setcpureserv**

定义专用于 VM 的 CPU 份额数。

### **setmemlimit**

定义 VM 内存份额数的限制。

### **setmemresrv**

定义专用于 VM 的内存份额数。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户（-vc\_user 选项）和组件（-auth\_comp 选项），则将其检索加密密码。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-wait *timeout***

（可选）表示是否显示作业的状态并直到操作完成、失败或达到超时时长（如果您指定了超时值）才返回。如果不指定该选项，CLI 不等待完成即返回。如果无响应，可选的超时值将定义超时时长（分钟）。如果输入不带超时值的等待选项，CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外，也可使用下列超时值：

"""

无超时值。

0

无限等待。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user wsuser**

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password wspassword**

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-encrypted\_password {yes|no}**

(可选) 指定您是否希望加密 VC 用户密码。

**默认值:** Yes

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 在单 vCenter 服务器环境中, 使用全局凭据将资源 CPU 限制设置为 1000 个份额**

本示例将虚拟机 “MyVM1” 的 CPU 份额限制调整为 1000。

```
dpmvc setresources -setcpulimit 1000 -vm_name MyVM1  
-datacenter_name DCA/MyCity
```

**示例: 使用特定组件 ID 将资源 CPU 限制设置为 1000**

本示例将虚拟机 “MyVM1” 的 CPU 份额限制调整为 1000。

```
dpmvc setresources -setcpulimit 1000 -vm_name MyVM1 -datacenter_name DCA/MyCity  
-vc_server vc_server_1 -vc_user VCAdmin1 -auth_comp Imaging
```

## dpmvc setshares 命令-设置 VM 份额

dpmvc setshares 命令用于设置分配给某个 VM 的 CPU 或内存份额的数量。

此命令具有以下格式：

```
dpmvc setshares
[-sc sc_url] {-cpu_add_prop_value value|-cpu subtract_prop_value
value|-cpu_overwrite_prop_value value|-mem_add_prop_value
value|-mem_subtract_prop_value value|-mem_overwrite_prop_value value}
-datacenter_name datacentername
-vm_name vmname
[-vc_server vcservername]
[-vc_user vcuser]
[-vc_password vcpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
[-wait [timeout]] [-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

**-cpu\_add\_prop\_value value | -cpu subtract prop value value | -cpu\_overwrite\_prop\_value value | -mem\_add\_prop\_value value | -mem\_subtract\_prop\_value value | -mem\_overwrite\_prop\_value value**

指定要添加、减少或更改的 CPU 或内存份额的数量。份额用于确定与其他正在运行的 VM 相比，分配给哪些 VM 的可用物理资源的比例较高。如果向某个 VM 分配的份额较多，则会向该 VM 分配较多的物理资源。如果向某个 VM 分配较少的份额，则会向该 VM 分配较少的物理资源。

**-cpu\_add\_prop\_value value**

指定要添加到 VM 的 CPU 份额的数量。

**-cpu\_subtract\_prop\_value value**

指定要从 VM 中减少的 CPU 份额的数量。

**-cpu\_overwrite\_prop\_value value**

指定要用于 VM 的 CPU 份额的数量。

**-mem\_add\_prop\_value value**

指定要添加到 VM 的内存份额的数量。

**-mem\_subtract\_prop\_value value**

指定要从 VM 中减少的内存份额的数量。

**-mem\_overwrite\_prop\_value value**

指定要用于 VM 的 CPU 份额的数量。

**-datacenter\_name datacentername**

指定 VM 所在的数据中心。

**-vm\_name vmname**

指定 VM。

**-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user vcuser**

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password vcpasssword**

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户（-vc\_user 选项）和组件（-auth\_comp 选项），则将其检索加密密码。

**-auth\_file *authorizationfilename***

(可选) 指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 `dpmutil set auth` 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

(可选) 指定可用于对主机和用户进行分组的组件 ID。

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {*yes|no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-encrypted\_password {*yes|no*}**

(可选) 指定您是否希望加密 VC 用户密码。

默认值: Yes

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：在单 vCenter 服务器环境中，使用全局凭据增加 VM 的 CPU 份额**

本示例在数据中心 “DCA/MyCity” 中，将虚拟机 “MyVM01” 的 CPU 份额数增加 1000。

```
dpmvc setshares -vm_name MyVM01 -datacenter_name DCA/MyCity  
-cpu_add_prop_value 1000
```

**示例：在多 vCenter 服务器环境中，使用全局凭据增加 VM 的 CPU 份额**

本示例在数据中心 “DCA/MyCity” 中，将虚拟机 “MyVM01” 的 CPU 份额数增加 1000。

```
dpmvc setshares -vm_name MyVM01 -datacenter_name DCA/MyCity  
-cpu_add_prop_value 1000 -vc_server vc_server_1
```

**示例：使用指定的授权文件和组件 ID 覆盖 VM 的 CPU 份额**

本示例在数据中心 “DCA/MyCity” 中，将虚拟机 “MyVM01” 的 CPU 份额数设置为 1200。

```
dpmvc setshares -vm_name MyVM01 -datacenter_name DCA/MyCity  
-cpu_overwrite_prop_value 1200 -vc_server vc_server_1  
-vc_user VCAdmin1 -auth_file c:\localauth.dat -auth_comp Imaging
```



## dpmvc snapshot 命令-管理快照

dpmvc snapshot 命令用于管理 VM 的快照。使用该命令，可以创建快照、删除一个或所有快照或还原到某个快照。

此命令具有以下格式：

```
dpmvc snapshot
[-sc sc_url]
-operation create
-datacenter_name datacentername -vm_name vmname
-vm_snapshot_name vm_snapshotname
[-vm_snapshot_desc description]
[-withmemory {yes|no}]
[-quiesce {yes|no}][-poweron]
[-vc_server vcservername]
[-vc_user vcusername]
[-vc_password vcpassword]
[-auth_file authorizationfile]
[-auth_comp componentID] [-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

```
dpmvc snapshot
[-sc sc_url]
-operation remove
-datacenter_name datacentername -vm_name vmname
[-vm_snapshot_name vm_snapshotname]
[-vm_snapshot_id snapid]
[-withchildren {yes|no}]
[-vc_server vcservername]
[-vc_user vcusername]
[-vc_password vcpassword]
[-auth_file authorizationfile]
[-auth_comp componentID] [-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

```
dpmvc snapshot
[-sc sc_url]
-operation revert
-datacenter_name datacentername -vm_name vmname
-vm_snapshot_name vmsnapshotname
[-vm_snapshot_id snapid]
[-vc_server vcservername]
[-vc_user vcusername]
[-vc_password vcpassword]
[-auth_file authorizationfile]
[-auth_comp componentID] [-wait [timeout]]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-operation {create | remove | revert}**

指定要创建、删除或还原 VM 快照。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vm\_name *vmname***

指定 VM。

**-vm\_snapshot\_name *vmsnapshotname***

定义 VM 快照的名称。

**-vm\_snapshot\_desc *description***

（可选）定义 VM 快照的说明。

**-withmemory {true | false}**

（可选）指定快照包含内存。该参数对于 -remove 或 -revert 无效。

**-quiesce {yes|no}**

（可选）如果设置为 **yes** 并且在拍摄快照时虚拟机已打开电源，VMware 工具将静止虚拟机中的文件系统。这样便能确保磁盘快照代表来宾文件系统的一致状态。如果虚拟机关闭或 VMware 工具不可用，该命令将忽略静止标志。

**-vm\_snapshot\_id *snapid***

（可选）定义快照标识符。

**-withchildren**

（可选）指定您希望删除快照的所有子项。该参数对于 **-create** 或 **-revert** 无效。

**-poweron**

（可选）指定您希望打开 VM 的电源。该参数对 **-remove** 或 **-revert** 无效。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

（可选）指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

（可选）指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (**-auth\_file**) 指定用户 (**-vc\_user** 选项) 和组件 (**-auth\_comp** 选项)，则将其检索加密密码。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 **dpmutil set auth** 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {*yes*|*no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-encrypted\_password {*yes*|*no*}**

(可选) 指定您是否希望加密 VC 用户密码。

默认值: Yes

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “`native`”。

**示例：在单 vCenter 服务器环境中，使用全局凭据创建 VM 快照**

本示例为名为 MyVM01 的 VM 系统创建一个名为 “Hello world” 的快照，并包含说明 “This is my first snapshot”。

```
dpmvc snapshot -create -vm_snapshot_name "Hello world" -vm_snapshot_desc "First snapshot"
-vm_name MyVM01 -datacenter_name VAS/MyCity
```

**示例：在单 vCenter 服务器环境中，使用全局凭据还原到某个 VM 快照**

本示例将 MyVM01 还原到名为 “Hello world” 的快照。

```
dpmvc snapshot -revert -vm_snapshot_name "Hello world" -vm_name MyVM01
-datacenter_name VAS/MyCity
```

**示例：在单 vCenter 服务器环境中，使用全局凭据删除 VM 快照**

本示例从名为 MyVM01 的 VM 系统中删除名为 “Hello world” 的快照。

```
dpmvc snapshot -remove -vm_snapshot_name "Hello world" -vm_snapshot_desc "First snapshot"
-vm_name MyVM01 -datacenter_name VAS/MyCity
```

**示例：在多 vCenter 服务器环境中，使用全局凭据为 VM 创建快照**

本示例在名为 PayrollVM 的 VM 系统上创建名为 PayrollSnapshot 的快照。

```
dpmvc snapshot -create -vm_snapshot_name PayrollSnapshot -vm_snapshot_desc "First snapshot"
-vm_name MyVM01 -datacenter_name VAS/MyCity -vc_server vc_server_1
```

**示例：使用指定的授权文件和组件 ID 删除 VM 快照**

本示例在数据中心 “DCA/MyCity” 中，将虚拟机 “MyVM01” 的 CPU 份额数设置为 1200。

```
dpmvc snapshot -remove -vm_snapshot_name "Snapshot 1" -vm_name MyVM01
-datacenter_name VAS/MyCity -vc_server vc_server_1 -vc_user VCAdmin1
-auth_file c:\localauth.dat -auth_comp Imaging
```

## dpmvc templatetovm 命令-将模板转换成 VM

dpmvc templatetovm 命令用于将虚拟机模板转换回 VM。例如，使用该功能，您可以向转换后的 VM 应用修补程序或软件更新，然后再将其转换回模板。

此命令具有以下格式：

```
dpmvc templatetovm
[-sc sc_url]
-template_name templatename
-datacenter_name datacentername
-compute_resource_name hostname
-resource_pool_name resourcepool
-esx_host_name hostname
[-vc_server vmserver]
[-vc_user vcuser]
[-vc_password vcpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
[-pre]
[-post]
[-ws_user wsuser]
[-ws_password wspassword]
[-prompt {yes|no}]
[-encrypted_password {yes|no}]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-template\_name *templatename***

定义模板的名称。

### **-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

### **-compute\_resource\_name *name***

指定在其中创建 VM 的群集或 VMware ESX 主机。

**-resource\_pool\_name resourcepool**

指定您希望从中选择用于克隆的 VM 的资源池的名称。

**-esx\_host\_name esxhostname**

指定 VM 驻留的 VMware ESX 服务器。

**-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user vcuser**

(可选) 指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password vcpassword**

(可选) 指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户 (-vc\_user 选项) 和组件 (-auth\_comp 选项)，则将其检索加密密码。

**-auth\_file authorizationfilename**

(可选) 指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 dpmutil set auth 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp componentID**

(可选) 指定可用于对主机和用户进行分组的组件 ID。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user wsuser**

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password wspassword**

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-encrypted\_password {yes|no}**

（可选）指定您是否希望加密 VC 用户密码。

**默认值:** Yes

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：在单 vCenter 服务器环境中，使用全局凭据将模板转换为 VM**

本示例将名为 convertTest 的模板转换为 VM。

```
dpmvc templatetovm -datacenter_name VAS/MyCity -template_name convertTest  
-compute_resource_name onDemand/myhost.myco.com  
-esx_host_name myhost.myco.com -resource_pool_name Resources/QA
```

**示例：在多 vCenter 服务器环境中，使用全局凭据将模板转换为 VM**

本示例将名为 payrollSystem1 的模板转换为 VM。

```
dpmvc templatetovm -datacenter_name VAS/MyCity -template_name payrollSystem1  
-compute_resource_name onDemand/myhost.myco.com -esx_host_name myhost.myco.com  
-resource_pool_name Resources/QA -vc_server vc_server_1
```

**示例：使用指定的授权文件和组件 ID 将模板转换为 VM**

本示例将名为 Win2K3 的模板转换为 VM。

```
dpmvc templatetovm -datacenter_name VAS/MyCity -template_name Win2K3  
-compute_resource_name onDemand/myhost.myco.com  
-esx_host_name myhost.myco.com -resource_pool_name Resources/QA  
-vc_server vc_server_1 -vc_user VCAdmin1 -auth_file c:\localauth.dat  
-auth_comp Imaging
```



## dpmvc vapp 命令—管理 vApp

dpmvc vapp 命令支持在 vApp 上执行以下操作：

- 创建新 vApp
- 克隆
- 打开电源
- 关闭电源
- 挂起
- 从 VMware vCenter 中删除 vApp
- 从 VMware vCenter 取消注册
- 将 VM 添加到 vApp 中
- 将资源池添加到 vApp 中
- 将 vApp 添加到 vApp 中
- 更新 vApp 配置

此命令具有以下格式：

```
dpmvc vapp -create
-vc_server vcservername
-vapp_path vapppath
[-vapp_name vappname]
[-cpu_allocation isExpandableReservation, limit, reservation, sharesLevel,
shares]
[-mem_allocation isExpandableReservation, limit, reservation, sharesLevel,
shares]
[-locale iso639value]
```

```
dpmvc vapp -clone
-vc_server vcservername
-vapp_path vapppath
[-target_vapp_path targetvapppath]
[-vapp_name vappname]
[-target_vapp_host targetvapphost]
```

```
dpmvc vapp [-power_on | -power_off | -suspend | -delete | -unregister ]
-vc_server vcservername
-vapp_path vapppath
```

```
dpmvc vapp -add_vms_to_vapp
-vc_server vcservername
-vapp_path vapppath
[-vapp_name vappname]
[-vms vms]
```

```
dpmvc vapp -add_rps_to_vapp  
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-rps resourcepool]
```

```
dpmvc vapp -add_vapps_to_vapp  
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-vapps vapps]  
[-locale iso639value]
```

```
dpmvc vapp -update_vapp_config  
-vc_server vcservername  
-vapp_path vapppath  
[-vapp_name vappname]  
[-config_settings  
vappName,objectType,startAction,startDelay,startOrder,stopAction,stopDelay,wai  
tForGuest]  
[-cpu_allocation isExpandableReservation, limit, reservation, sharesLevel,  
shares]  
[-mem_allocation isExpandableReservation, limit, reservation, sharesLevel,  
shares]  
[-locale iso639value]
```

**-vc\_server vcservername**

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vapp\_path vapppath**

指定 vApp 路径。

**-vapp\_name vappname**

（可选）指定 vApp 名称。

**-cpu\_allocation *isExpandableReservation, limit, reservation, sharesLevel, shares***

（可选）指定更新 vApp 配置时的 CPU 分配设置。

**isExpandableReservation**

指定分配是否可扩展。可能的值：true 或 false。

**limit**

指定分配限制。

**reservation**

指定保留分配。

**sharesLevel**

指定份额水平，0 代表低，1 代表正常，2 代表高，3 代表自定义。  
例如 true,-1,1000,1,4000

**shares**

指定分配份额。

**-mem\_allocation *isExpandableReservation, limit, reservation, sharesLevel, shares***

指定更新 vApp 配置时的内存分配设置。

**isExpandableReservation**

指定分配是否可扩展。可能的值：true 或 false。

**limit**

指定分配限制。

**reservation**

指定保留分配。

**sharesLevel**

指定份额水平，0 代表低，1 代表正常，2 代表高，3 代表自定义。  
例如 true,-1,1000,1,4000

**shares**

指定分配份额。

**-vms *vms***

（可选）指定您希望添加到 vApp 的 VM。

**-rps *resourcepool***

（可选）指定您希望添加到 vApp 的资源池的名称。

**-vapps vapps**

（可选）指定您希望添加到 vApp 的 vApp。

**-config\_settings**

**vappName,objectType,startAction,startDelay,startOrder,stopAction,stopDelay,waitForGuest**

（可选）指定 vApp 的配置设置。正确的格式为 "

**startAction none /powerOn**

指定启动操作。可用的选项有 none 或 powerOn

**stopAction none /powerOff/guestShutdown /suspend**

指定启动操作。可用的选项有 none、powerOff、guestShutdown 或 suspend。

**-target\_vapp\_path targetvapppath**

指定您希望创建的 vApp 的目标 vApp 路径。

**-target\_vapp\_host targetvapphost**

（可选）指定新的 vApp 要驻留在其上的主机。

**-target\_vapp\_datastore targetvappdatastore**

（可选）指定 vApp 要驻留在其上的数据存储的名称。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

## dpmvc virtualswitch 命令—管理虚拟交换机

dpmvc virtualswitch 命令用于管理虚拟交换机。

- 创建新的虚拟交换机
- 更新虚拟交换机的属性
- 删除虚拟交换机
- 为虚拟交换机创建新的端口组
- 更新虚拟交换机的端口组属性
- 从虚拟交换机中删除端口组
- 重命名虚拟交换机的端口组

**注意：**此操作不会异步运行，并且会立刻返回结果。但是，PMM 将该操作视为任务操作。因此响应中包含任务 ID，但该 ID 始终是空字符串（""）。

此命令具有以下格式：

```
dpmvc virtualswitch {-vs_add | -vs_update}
-vc_server vcservername
-esx_host_name esxhostname
-switch_name switchname
-nic_names nicname1[,nicname2, ...]
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc virtualswitch -vs_remove
-vc_server vcservername
-esx_host_name esxhostname
-switch_name switchname
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc virtualswitch -add_portgroup
-vc_server vcservername
-esx_host_name esxhostname
-switch_name switchname
-portgroup_name portgroupname
[-vlan vlanID]
[-sc sc_url]
[-wait [timeout]]
[-pre]
[-post]
[-locale iso639value]
```

```
dpmvc virtualswitch -update_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-switch_name switchname  
-portgroup_name portgroupname  
[-portgroup_newname portgroupnewname]  
[-vlan vlanID]  
[-sc sc_url]  
[-wait [timeout]]  
[-pre]  
[-post]  
[-locale iso639value]
```

```
dpmvc virtualswitch -remove_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-portgroup_name portgroupname  
[-vlan vlanID]  
[-sc sc_url]  
[-wait [timeout]]  
[-pre]  
[-post]  
[-locale iso639value]
```

```
dpmvc virtualswitch -rename_portgroup  
-vc_server vcservername  
-esx_host_name esxhostname  
-switch_name switchname  
-portgroup_name portgroupname  
-portgroup_newname portgroupnewname  
[-sc sc_url]  
[-wait [timeout]]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-esx\_host\_name *esxhostname***

指定托管 VM 的 ESX 服务器的名称。

**-switch\_name *switchname***

指定操作所针对的交换机名称。

**-nic\_names *nicname1* [*nicname2*, ...]**

指定以逗号分隔的物理 NIC 列表。如果指定多个 NIC，请使用双引号来转义参数，例如："sc1,nfs1,mnic1,mnic2"。

**-portgroup\_name *portgroupname***

指定端口组名称。

**-portgroup\_newname portgroupnewname**

指定新的端口组名称。

**-vlan vlanID**

(可选) 指定用于虚拟端口组操作的整数值 (vlan ID)。

**-sc sc\_url**

(可选) 为服务控制器主机定义 URL。

示例: https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例: localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例: 80

**-wait timeout**

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 caimgconf.cfg 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

此示例将创建新的虚拟交换机。

```
dpmvc virtualswitch -vs_add -vc_server vc5master -esx_host_name esx5  
-switch_name switch1 -nic_names "sc1,nfs1,mnic1,mnic2"
```

## **dpmvc vmtotemplate 命令-将 VM 转换成模板**

dpmvc vmtotemplate 命令用于将虚拟机转换成模板。该功能有助于在数据中心的多台服务器之间实现一致性。

此命令具有以下格式：

```
dpmvc vmtotemplate  
[-sc sc_url]  
-vm_name vmname  
-datacenter_name datacentername  
[-vc_server vmserver]  
[-vc_user vcuser]  
[-vc_password vcpassword]  
[-auth_file authorizationfilename]  
[-auth_comp componentID]  
[-pre]  
[-post]  
[-ws_user wsuser]  
[-ws_password wspassword]  
[-prompt {yes|no}]  
[-encrypted_password {yes|no}]  
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80



**-vm\_name *vmname***

指定 VM。

**-datacenter\_name *datacentername***

指定 VM 所在的数据中心。

**-vc\_server *vcservername***

指定在具有多个 vCenter 服务器时要访问的 vCenter 服务器。对于单 vCenter 服务器环境是可选的。

**-vc\_user *vcuser***

(可选) 指定 vCenter 服务器用户 ID。这是可选的，具体取决于是否已启用全局凭据。

**-vc\_password *vcpassword***

(可选) 指定 vCenter 服务器用户 ID 的密码。省略该选项时，如果从默认授权文件或特定授权文件 (-auth\_file) 指定用户 (-vc\_user 选项) 和组件 (-auth\_comp 选项)，则将其检索加密密码。

**-auth\_file *authorizationfilename***

(可选) 指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 dpmutil set auth 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

(可选) 指定可用于对主机和用户进行分组的组件 ID。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到 VC 平台管理模块 (PMM) 的 Web 服务用户名。

**-ws\_password *wspassword***

(可选) 指定连接到 VC 性能管理模块 (PMM) 的 Web 服务用户的密码。

**-prompt {*yes|no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-encrypted\_password {yes|no}**

（可选）指定您是否希望加密 VC 用户密码。

**默认值:** Yes

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：在单 vCenter 服务器环境中，使用全局凭据将 VM 转换为模板**

本示例将名为 convertTest 的 VM 转换为模板。

```
dpmvc vmtotemplate -datacenter_name VAS/MyCity -template_name convertTest  
-compute_resource_name onDemand/myhost.myco.com -esx_host_name myhost.myco.com  
-resource_pool_name Resources/QA
```

**示例：在多 vCenter 服务器环境中，使用全局凭据将 VM 转换为模板**

本示例将名为 payrollSystem1 的模板转换为 VM。

```
dpmvc vmtotemplate -datacenter_name VAS/MyCity -template_name payrollSystem1  
-compute_resource_name onDemand/myhost.myco.com -esx_host_name myhost.myco.com  
-resource_pool_name Resources/QA -vc_server vc_server_1
```

**示例：使用指定的授权文件和组件 ID 将 VM 转换为模板**

本示例将名为 Win2K3 的模板转换为 VM。

```
dpmvc vmtotemplate -datacenter_name VAS/MyCity -template_name Win2K3  
-compute_resource_name onDemand/myhost.myco.com -esx_host_name myhost.myco.com  
-resource_pool_name Resources/QA -vc_server vc_server_1 -vc_user VCAdmin1  
-auth_file c:\localauth.dat -auth_comp Imaging
```

## CA Hyper-V CLI 命令

使用 CLI 可以编写 CA Hyper-V 命令的脚本并使其自动化，然后根据命令的结果运行操作。也可以在 AutoShell 中使用对应的命令。

## dpmhyperv AddVMNic 命令-将网络适配器控制器添加到 VM 中

dpmhyperv AddVMNic 命令用于将网络适配器添加到 VM 中。

此命令具有以下格式：

```
dpmhyperv AddVMNic  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vm VM_name|-vmguid vm_ID  
[-mac mac_value]  
[-switch switch_name]  
[-legacy]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-mac *mac\_value***

（可选）指定网络适配器的媒体访问控制 (MAC) 地址。

**-switch *switch\_name***

（可选）指定连接到网络适配器的虚拟网络交换机名称。如果未指定该参数，则网络适配器将不会有任何连接。

**-legacy**

（可选）指定是否创建传统网络适配器，而不是 Hyper-V 合成适配器。仅当安装没有可用的 Hyper-V 集成服务的操作系统时，才必须使用该选项。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将网络适配器控制器添加到 VM 中**

本示例将网络适配器控制器添加到 VM “TestVM” 中，并将其连接到交换机 “Local Area Connection - Virtual Network”。

```
dpmhyperv
-addVMNic
-ws_user dcaadmin
-ws_password #test#
-hypervHost hvserver
-vm TestVM
-switch "Local Area Connection - Virtual Network"
```

## **dpmhyperv AddVMSCSIController 命令-为 VM 创建 SCSI 控制器**

dpmhyperv AddVMSCSIController 命令用于为 Hyper-V 主机上的 VM 创建 SCSI 控制器。

此命令具有以下格式：

```
dpmhyperv AddVMSCSIController
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password] -hypervHost hostname
-vm vm_name|-vguid vm_ID
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 为 Hyper-V 主机上的 VM 创建新的 SCSI 控制器**

本示例为 Hyper-V 主机 “hserver” 上的 GUID 为 “38F29B6A-8CE2-42D2-8269-BFED14644376” 的 VM 创建新的 SCSI 控制器。

```
dpmhyperv AddVMSCSIController
-ws_user dcaadmin
-ws_password #test#
-hypervHost hserver
-vmguid 38F29B6A-8CE2-42D2-8269-BFED14644376
```

**dpmhyperv AgentGet 命令-返回代理值**

dpmhyperv AgentGet 命令用于返回代理值。

此命令具有以下格式:

```
dpmhyperv AgentGet
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-modelPath modelpath_value attributeName attribute_name
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

(可选) 指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host *hostname***

指定 Hyper-V 服务器的主机名。

**-jobRef *job\_ref\_identifier***

指定异步作业的引用标识符。

**-modelPath *modelpath\_value***

指定要从其获取值的对象的模型路径。

**-attributeName *attribute\_name***

指定要获取的属性的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：返回代理值**

本示例返回代理值。

```
dpmhyperv AgentGet
-ws_user dcaadmin
-ws_password #test#
-hypervHost hserver -modelPath
"https://localhost/aip/AOM/root/cimv2:CA_ComputerSystem.CreationClassName=\"C
A_ComputerSystem\",Name=\"8a2a30a0-a184-11df-84a8-00155d7c7405\"\" -attribute
Name DPM.HYPERV.SERVER.CRITICALVMS.MONITOR
```

## dpmhyperv AgentSet 命令-设置代理值

dpmhyperv AgentSet 命令用于设置代理值。

此命令具有以下格式：

```
dpmhyperv AgentSet  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-modelPath model_path attributeName attribute_name  
-snmpValue snmp_value  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host *hostname***

指定 Hyper-V 服务器的主机名。

**-jobRef *job\_ref\_identifier***

指定异步作业的引用标识符。

**-modelPath *modelpath\_value***

指定要从其获取值的对象的模型路径。

**-attributeName *attribute\_name***

指定要获取的属性的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：设置代理值

本示例设置代理值。

```
dpmhyperv AgentSet
-ws_user admin
-ws_password #test#
-hypervHost hserver
-modelPath
"https://localhost/aip/AOM/root/cimv2:CA_ComputerSystem.CreationClassName=\"C
A_ComputerSystem\",Name=\"8a2a30a0-a184-11df-84a8-00155d7c7405\"" -attribute
Name DPM.HYPERV.SERVER.CRITICALVMS.MONITOR -snmpValue 1
```

## dpmhyperv ChangeVMState 命令-更改 VM 的状态

dpmhyperv ChangeVMState 命令用于更改 Hyper-V 主机上的 VM 的状态。

此命令具有以下格式：

```
dpmhyperv ChangeVMState
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password] -hypervHost hostname
-vm vm_name|-vguid vm_ID
-state {Start|Suspend|Stop|Pause|Shutdown|Reboot}
-[pre]
[-post]
[-locale iso639value]
```

#### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

#### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

#### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

#### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

#### **-vm *vm\_name***

指定 VM 名称。

#### **-vmguid *vm\_ID***

指定 VM 的唯一 GUID。



**-state {Start|Suspend|Stop|Pause|Shutdown|Reboot}**

指定 VM 的状态。选项包括：

**启动**

打开 VM。

**挂起**

暂时挂起 VM。

**停止**

停止 VM。

**暂停**

暂时停止 VM。

**Shutdown**

关闭 VM。

**重新启动**

执行 VM 的硬重置。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：更改 VM 的状态**

本示例将 VM “vmtest” 的状态设置为启用。

```
dpmhyperv ChangeVMState -ws_user admin -ws_password #test# -hypervHost hvserver  
-vm vmtest -state start
```

## dpmhyperv CreateTemplateFromVM 命令-从 VM 创建模板

dpmhyperv CreateTemplateFromVM 命令用于从现有的 VM 创建模板。

此命令具有以下格式：

```
dpmhyperv CreateTemplateFromVM  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vmname vm_name|-vmguid vm_ID  
-template template_name  
[-description template_description]  
[-path template_location]  
[-async]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-template *template\_name***

定义新模板的名称。

**-description *template\_description***

（可选）指定模板说明。

**-path *template\_location***

（可选）定义要在其中创建模板的目录的完整路径。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：从 VM 创建模板**

本示例从 VM “TestVM” 创建模板 “MyTemplate”。

```
dpmhyperv -CreateTemplateFromVM -ws_user admin -ws_password #test#  
-hypervHost hvserver -vm TestVM -template MyTemplate
```

**dpmhyperv CreateVirtDisk 命令-创建虚拟磁盘映像**

dpmhyperv CreateVirtDisk 命令用于在 Hyper-V 主机上创建虚拟磁盘映像。

此命令具有以下格式：

```
dpmhyperv CreateVirtDisk  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-path location  
-diskType {dynamic|fixed}  
-size drive_size  
[-async]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-path *location***

定义虚拟磁盘映像的完整路径。

**-diskType {fixed | dynamic}**

指定虚拟磁盘映像的类型。选项包括：

**fixed**

指示磁盘的大小已固定，以后无法增加。

**dynamic**

指示磁盘的大小是动态的，以后可根据需要进行增加。

**-size *drive\_size***

指定驱动器的大小 (GB)。

**注意：**稍后可以更改磁盘的大小。

**-async**

(可选) 以异步模式运行命令。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如：法语为 fr\_FR)。要使用命令提示符的区域设置，请指定 “native”。

**示例：创建虚拟磁盘映像**

本示例创建大小为 5 GB 的虚拟磁盘 “01.vhd”。

```
dpmhyperv CreateVirtDisk -hypervHost hvserver -path C:\data\Disks\01.vhd -type
dynamic -size 5 -async
```

## dpmhyperv CreateVirtFloppy 命令-创建虚拟软盘映像

dpmhyperv CreateVirtFloppy 命令用于在 Hyper-V 主机上创建虚拟软盘映像。

此命令具有以下格式：

```
dpmhyperv CreateVirtFloppy  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-path location  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-path *location***

定义虚拟软盘映像的完整路径。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：创建虚拟软盘映像

本示例在 Hyper-V 主机“hvserver”上创建虚拟软盘映像。

```
dpmhyperv CreateVirtFloppy -ws_user dcaadmin -ws_password #test# -hypervHost  
hvserver -path C:\data\Floppy\1.vfd
```

## dpmhyperv CreateVM 命令-创建 VM

dpmhyperv CreateVM 命令用于在 Hyper-V 主机上创建 VM。

此命令具有以下格式：

```
dpmhyperv CreateVM
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
[-templatevm vm_template]
[-templatevmguid template_ID]
[-path location]
[-memory memory_size]
[-cpuSocketCount total_CPU_socket_count]
[-cpuidLimit]
[-cpufeatlimit]
[-cpuReserve cpu_reserve]
[-cpuLimit max_CPU_resources]
[-cpuWeight cpu_weight]
[-startAction {none|auto|always}]
[-startDelay start_delay_secs]
[-stopAction {save|off|shutdown}]
[-recoveryAction {none|restart|revert}]
[-pre]
[-post]
[-locale iso639value]
```

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

### **-templatevm *vm\_template***

（可选）指定您要从其复制配置的 VM 的名称。此参数用于标识 Hyper-V 环境中现有的 VM，而不是 VM 目录中的模板。

**注意：**我们建议您不要通过现有 VM 创建 VM，因为共享的资源可能会导致冲突。

**-templatevmguid *template\_ID***

(可选) 指定您要从其复制配置的 VM 的 GUID。此参数用于标识 Hyper-V 环境中现有的 VM，而不是 VM 目录中的模板。

**注意：**我们建议您不要通过现有 VM 创建 VM，因为共享的资源可能会导致冲突。

**-path *location***

(可选) 定义新的 VM 的完整路径。

**-memory *memory\_size***

(可选) 定义 VM 的内存大小 (MB)。

**-cpuSocketCount *total\_CPU\_socket\_count***

(可选) 定义 VM 中 CPU 内核的数量。VM 的 CPU 内核数量不能超过 Hyper-V 主机中可用的 CPU 内核总数。

**-cpuidLimit**

(可选) 指示处理器是否必须限制最大 CPUID 值。

**-cpufeatlimit**

(可选) 指示 VM 是否必须限制公开给操作系统的 CPU 功能。

**-cpuReserve *cpu\_reserve***

(可选) 指定保留供 VM 使用的 CPU 资源的量。

**-cpuLimit *max\_CPU\_resources***

(可选) 指定 VM 可以占用的 CPU 资源的最大量。

**-cpuWeight *cpu\_weight***

(可选) 在 1 至 10000 的范围内指定虚拟机的相对权重。

**默认：** 100

**-startAction {*none*|*auto*|*always*}**

(可选) 指定在 VM 启动后要执行的操作。选项包括：

**无**

在 VM 上不执行任何操作。

**自动**

如果 VM 在 Hyper-V 主机关闭之前正在运行，则自动启动该 VM。

**always**

在每次 Hyper-V 启动时启动 VM。

**-startDelay *start\_delay\_secs***

(可选) 指定在 Hyper-V 主机启动后延迟启动 VM 的时间（秒）。

**-stopAction {save|off|shutdown}**

（可选）指定在 Hyper-V 主机停止之前要在 VM 上执行的操作。选项包括：

**save**

挂起 VM。

**关**

禁用 VM。

**shutdown**

关闭 VM。要使用该选项，必须安装 Hyper-V 工具。

**-recoveryAction {none|restart|revert}**

（可选）指定 VM 工作进程异常终止并恢复之后要在 VM 上执行的操作。选项包括：

**无**

不执行任何操作。

**restart**

重新启动 VM。

**revert**

恢复为上次的快照。要使用该选项，VM 应至少有一个快照。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：创建 VM**

本示例在“VM”文件夹中创建名为“TestVM”的 VM。

```
dpmhyperv CreateVM -ws_user dcaadmin -ws_password #test# -hypervHost hvserver  
-name TestVM -path c:\VM\TestVM
```



## dpmhyperv CreateVMFromTemplate 命令-通过 SCVMM 服务器模板创建 VM

dpmhyperv CreateVMFromTemplate 命令用于通过 SCVMM 服务器模板创建 VM。

此命令具有以下格式：

```
dpmhyperv CreateTemplateFromVM
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
[-prompt {yes|no}] -hypervHost hostname
-vmname vm_name
-template template_name
[-dest destination]
[-computerName computer_Name]
[-ip4addr ip4addr]
[-ip4dhcp ip4dhcp]
[-ip4dhcpInt ip4dhcpInt]
[-ip4mask ip4mask]
[-ip4gw ip4gw]
[-ip4metric ip4metric]
[-ip4dns ip4dns]
[-adminPass admin_password]
[-productKey product_key]
[-userName user_name]
[-organization org_name]
[-domain domain_name]
[-domainAdmin domain_admin_login]
[-domainAdminPass domain_admin_password]
[-adminUser admin_username]
[-startVM start_vm]
{-memory memory_size}
[cpuSocketCount total_CPU_socket]
[-scvmmHost scvmm_host]
[-scvmmHardwareProf scvmm_hardware_profile]
[-scvmmGuestOSProf scvmm_guest_os_profile]
[-auto_deploy autodeploy]
[-async]
[-pre]
[-post]
[-locale iso639value]
```

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-prompt {yes|no}**

（可选）指定是否提示您输入用户名和密码。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-template *template\_name***

定义新模板的名称。

**-dest *destination\_path***

（可选）指定创建模板的目标路径。

**-computerName *computer\_name***

（可选）指定 VM 的计算机名称。

**-ip4addr *ip4addr***

（可选）指定要分配给 VM 接口的静态 IPv4 地址。

**-ip4dhcp *ip4dhcp***

（可选）指定是否使用 DHCP。

默认值: local

**-ip4dhcpInt *ip4dhcpInt***

（可选）指定 DHCP 地址。

**-ip4mask *network\_mask***

（可选）指定要分配给 VM 的子网掩码。该选项与 **-ip4addr** 选项结合使用。

**-ip4gw *gateway\_address***

（可选）指定该选项可为 VM 设置网关。该选项与 **-ip4addr** 选项结合使用。

**-ip4metric *ip\_metric***

（可选）指定要为 VM 设置的接口度量标准。该选项与 **-ip4addr** 选项结合使用。

**-ip4dns *dnsserver\_ip***

（可选）指定 VM 的 DNS 服务器。该选项与 **-ip4addr** 选项结合使用。

**-adminPass *admin\_password***

（可选）指定 VM 的默认管理员密码。

**-productKey *product\_key***

（可选）指定 VM 的 Windows 产品激活密钥。

**-userName *user\_name***

（可选）指定 VM 中 Windows 的用户名。

**-organization *org\_name***

（可选）指定 VM 中 Windows OS 的组织名称。

**-domain *domain\_name***

（可选）指定 VM 的域名。

**-domainAdmin *domain\_admin\_login***

（可选）指定域管理员登录名。

**-domainAdminPass *domain\_admin\_password***

（可选）指定域管理员帐户的密码。该选项对于异步模式是无效的。

**-adminUser *admin\_username***

（可选）指定是默认 Administrators 组成员的用户名。

**-startVM *start\_vm***

（可选）指示是否启动 VM。

**-memory *memory\_size***

（可选）定义内存大小（字节）。

**-cpuSocketCount *total\_CPU\_socket***

（可选）指定 VM 中处理器插槽的数目。

**-scvmmHost *scvmm\_hostname***

（可选）指定 SCVMM 服务器主机的名称。

**-scvmmHardwareProf *scvmm\_hardware\_profile***

（可选）指定 SCVMM 定义的硬件配置文件名称。

**-scvmmGuestOSProf *scvmm\_guest\_os\_profile***

（可选）指定 SCVMM 定义的来宾操作系统配置文件名称。

**-autodeploy *autodeploy***

（可选）指示自动部署。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**注意：**如果没有使用 `-async` 模式，则会忽略下列参数：`memory`、`cpuSocketCount`、`cpuidLimit`、`cpuFeatureLimit`、`cpuReserve`、`cpuLimit`、`cpuWeight`、`startAction`、`startDelay`、`stopAction` 和 `recoveryAction`。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：使用单机服务器模板创建 VM**

本示例使用单机模板 “Win2kSysPrepped” 创建 VM “NewVM”。

```
dpmhyperv createVMFromTemplate -ws_user dcaadmin -ws_password #test# -hypervHost  
hvserver -vm NewVM -dest c:\VMs\NewBox -template Win2k3SysPrepped -ip4addr  
127.0.0.1 -ip4mask 255.255.255.0 -ip4gw 127.0.0.1 -ip4dns 127.0.0.1 -computerName  
NewBox
```

## dpmhyperv CreateVMFromTemplate 命令-利用模板创建 VM

dpmhyperv CreateVMFromTemplateEx 命令用于通过单机 Hyper-V 服务器模板创建 VM。

此命令具有以下格式：

```
dpmhyperv CreateTemplateFromVMEx
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
[-prompt {yes|no}]
-hypervHost hostname
-vmname vm_name
-template template_name
[-dest destination]
[-computerName computer_Name]
[-ip4addr ip4addr]
[-ip4dhcp ip4dhcp]
[-ip4dhcpInt ip4dhcpInt]
[-ip4mask ip4mask]
[-ip4gw ip4gw]
[-ip4metric ip4metric]
[-ip4dns ip4dns]
[-disableAdmin]
[-adminPass admin_password]
[-autoLogon autologon]
[-duplicatorString duplicate_String]
[-timeZone timezone]
[-productKey product_key]
[-userName user_name]
[-organization org_name]
[-domain domain_name]
[-domainAdmin domain_admin_login]
[-domainAdminPass domain_admin_password]
[-workgroup workgroup_name]
[-adminUser admin_username]
[-adminUserPass admin_password] [adminGroup admin_group_name]
[-startVM start_vm]
[-custom custom]
[-memory memory_size] [cpuSocketCount total_CPU_socket]
[-cpuidLimit cpu_id_limit]
[-cpufeatlimit cpu_feat_limit]
[-cpuReserve cpu_reserve]
[-cpuLimit max_CPU_usage]
[-cpuWeight cpu_weight]
[-startAction {none|auto|always}]
[-startDelay start_delay_secs]
[-stopAction {save|off|shutdown}]
[-recoveryAction {none,restart,revert}]
[-scvmmHost scvmm_host]
[-scvmmHardwareProf scvmm_hardware_profile]
[-scvmmGuestOSProf scvmm_guest_os_profile]
[-async]
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

(可选) 指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-prompt {yes|no}**

（可选）指定是否提示您输入用户名和密码。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-template *template\_name***

定义新模板的名称。

**-dest *destination\_path***

（可选）指定创建模板的目标路径。

**-computerName *computer\_name***

（可选）指定计算机的名称。

**-ip4addr *ip4addr***

（可选）指定要分配给 VM 接口的静态 IPv4 地址。

**-ip4dhcp *ip4dhcp***

（可选）指定是否使用 DHCP。

默认值：local

**-ip4dhcpInt *ip4dhcpInt***

（可选）指定 DHCP 地址。

**-ip4mask *network\_mask***

（可选）指定要分配给 VM 的子网掩码。该选项与 **-ip4addr** 选项结合使用。

**-ip4gw *gateway\_address***

（可选）指定该选项可为 VM 设置网关。该选项与 **-ip4addr** 选项结合使用。

**-ip4metric *ip\_metric***

（可选）指定要为 VM 设置的接口度量标准。该选项与 **-ip4addr** 选项结合使用。

**-ip4dns *dnserver\_ip***

（可选）指定 VM 的 DNS 服务器。该选项与 **-ip4ddr** 选项结合使用。

**-disableAdmin**

（可选）指定该选项可禁用 VM 的默认管理员帐户。

**-adminPass *admin\_password***

（可选）指定 VM 的默认管理员密码。

**-autoLogon *autologon***

（可选）指定该选项可设置在 Sysprep 进程完成后使用默认管理员帐户自动登录的帐户的数目。

**-duplicatorString *duplicate\_string***

（可选）指定要在 VM 注册表中设置的系统复印机的名称。

**-timeZone *timezone***

（可选）指定使用模板创建的 VM 所用的时区。

**-productKey *product\_key***

（可选）指定 VM 的 Windows 产品激活密钥。

**-userName *user\_name***

（可选）指定 VM 中 Windows 的用户名。

**-organization *org\_name***

（可选）指定 VM 中 Windows 的组织名称。

**-domain *domain\_name***

（可选）指定 VM 的域名。

**-domainAdmin *domain\_admin\_login***

（可选）指定域管理员登录名。

**-domainAdminPass *domain\_admin\_password***

（可选）指定域管理员帐户的密码。该选项对于异步模式是无效的。

**-workgroup *workgroup\_name***

（可选）指定 VM 的工作组。该选项对于异步模式是无效的。

**-adminUser *admin\_username***

（可选）指定是默认 Administrators 组成员的用户名。

**-adminUserPass *admin\_password***

（可选）指定默认 Administrators 组的密码。



**-adminGroup *admin\_group\_name***

（可选）指定管理员的组名。

**-startVM *start\_vm***

（可选）指示是否启动 VM。

**-custom *custom***

（可选）指定在 Sysprep 进程结束时执行的自定义命令的逗号分隔列表。

**-memory *memory\_size***

（可选）定义内存大小（字节）。

**-cpuSocketCount *total\_CPU\_socket***

（可选）指定 VM 中处理器插槽的数目。

**-cpuidLimit *cpu\_id\_limit***

（可选）指示处理器是否必须限制最大 CPUID 值。

**cpufeatlimit *cpu\_feat\_limit***

（可选）指示 VM 是否必须限制公开给操作系统的 CPU 功能。

**-cpuReserve *cpu\_reserve***

（可选）指定保留供 VM 使用的 CPU 资源的量。

**-cpuLimit *max\_CPU\_usage***

（可选）指定 VM 可以占用的 CPU 资源的最大量。

**-cpuWeight *cpu\_weight***

（可选）在 1 至 10000 的范围内指定虚拟机的相对权重。

**-startAction {none|auto|always}**

（可选）指定在 Hyper-V 主机启动之后要在 VM 上执行的操作。选项包括：

无

不执行任何操作。

自动

如果 VM 在 Hyper-V 主机关闭之前正在运行，则自动启动该 VM。

always

在每次 Hyper-V 启动时启动 VM。

**-startDelay *start\_delay\_secs***

（可选）指定在 Hyper-V 主机启动后延迟启动 VM 的时间（秒）。

**-stopAction {save|off|shutdown}**

（可选）指定在 Hyper-V 主机关闭之前要在 VM 上执行的操作。选项包括：

**save**

挂起 VM。

**关**

禁用 VM。

**shutdown**

关闭 VM。

**-recoveryAction {none,restart,revert}**

（可选）指定在 Hyper-V 主机意外关闭又重新启动之后要在 VM 上执行的操作。选项包括：

**无**

不执行任何操作。

**restart**

重新启动 VM。

**revert**

恢复为上次的快照。

**-scvmmHost *scvmm\_hostname***

（可选）指定 SCVMM 服务器主机的名称。

**-scvmmHardwareProf *scvmm\_hardware\_profile***

（可选）指定 SCVMM 定义的硬件配置文件的名称。

**-scvmmGuestOSProf *scvmm\_guest\_os\_profile***

（可选）指定 SCVMM 定义的来宾操作系统配置文件的名称。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**注意：**如果没有使用 **-async** 模式，则会忽略下列参数：**memory**、**cpuSocketCount**、**cpuidLimit**、**cpufeatlimit**、**cpuReserve**、**cpuLimit**、**cpuWeight**、**startAction**、**startDelay**、**stopAction** 和 **recoveryAction**。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：通过单机 Hyper-V 服务器模板创建 VM**

本示例使用模板 “Win2kSysPrepped” 创建 VM “NewVM”。

```
dpmhyperv -createVMFromTemplateEx -ws_user dcaadmin -ws_password #test#
-hypervHost hvserver -vm NewVM -dest c:\VMs\NewBox -template Win2k3SysPrepped
-ip4addr 127.0.0.1 -ip4mask 255.255.255.0 -ip4gw 127.0.0.1 -ip4dns 127.0.0.1
-computerName NewBox
```

**dpmhyperv DeleteVM 命令-删除 VM**

dpmhyperv DeleteVM 命令用于删除 Hyper-V 主机上的 VM。该命令还会删除分配给 VM 的虚拟磁盘、虚拟软盘驱动器和 ISO 映像。

此命令具有以下格式：

```
dpmhyperv DeleteVM
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password] -hypervHost hostname
-vm vm_name|-vguid vm_ID
[-delvhd]
[-delvfd]
[-deliso]
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-delvhd**

(可选) 指示是否删除虚拟硬盘驱动器。

**-delvfd**

(可选) 指示是否删除与 VM 关联的虚拟软盘映像。

**-deliso**

(可选) 指示是否删除与 VM 关联的虚拟 CD/DVD 映像。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 从 Hyper-V 主机删除 VM**

本示例从 Hyper-V 主机 “hserver” 删除 GUID 为

“346597794004894538F29B6A-8CE2-42” 的 VM。该命令还会删除分配给 VM 的虚拟磁盘映像、虚拟软盘驱动器和 CD/DVD ISO 映像。

```
dpmhyperv DeleteVM -ws_user admin -ws_password #test# -hypervHost hserver -vmguid 38F29B6A-8CE2-42D2-8269-BFED14644376 -delvhd -delvfd -deliso
```

## dpmhyperv ExpandVirtDisk 命令-扩展虚拟磁盘的大小

dpmhyperv ExpandVirtDisk 命令用于扩展 Hyper-V 主机上的虚拟磁盘映像的大小。

此命令具有以下格式：

```
dpmhyperv ExpandVirtDisk  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-path location  
-size disk_size  
[-async]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-path *location***

定义虚拟磁盘的位置。

**-size *drive\_size***

指定驱动器的大小 (GB)。

**注意：**稍后可以更改磁盘的大小。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：扩展虚拟磁盘的大小**

本示例在异步模式下将 Hyper-V 主机 “hserver” 上的虚拟磁盘映像的大小扩展到 2 GB。该命令将返回所创建任务的 Hyper-V 作业 ID。

```
dpmhyperv ExpandVirtDisk -ws_user admin -ws_password #test# -hypervHost hserver  
-path C:\data\Disks\1.vhd -size 2 -async
```

## **dpmhyperv ExportVM 命令-导出 VM**

dpmhyperv ExportVM 命令用于将包含 VM 信息的文件存储到 Hyper-V 主机上的目录。该命令已过时，改为使用 dpmhv ExportVMEx。

此命令具有以下格式：

```
dpmhyperv ExportVM  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vm vm_name|-vmguid vm_ID  
-dest dest_location  
[-nostate]  
[-async]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-dest *dest\_location***

(可选) 定义导入文件的目标位置。

**-nostate**

(可选) 指定仅导出 VM 规格。该选项可防止导出的 VM 试图引用原始 VM 磁盘映像。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 导出 VM**

本示例存储包含 Hyper-V 主机 “hserver” 上的 VM “TestVM” 的信息的文件。

```
dpmhv ExportVM -hypervHost hserver -vm TestVM -dest c:\DATA\Export
```

**dpmhyperv ExportVMEx 命令--导出 VM**

dpmhyperv ExportVM 命令用于导出虚拟机。

此命令具有以下格式:

```
dpmhyperv ExportVM
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-vm vm_name|-vmguid vm_ID
-dest dest_folder
[-noruntimeInfo]
[-nosnapshots]
[-nostorage]
[-nosubdir]
[-snapshot snapshot_name][[-snapshotid snapshot_ID]
[-nostate]
[-async]
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-dest *dest\_location***

（可选）定义导入文件的目标位置。

**-noruntimeInfo**

（可选）指定不导出运行时相关信息。

**-nosnapshots**

（可选）指定不导出快照。

**-nostorage**

（可选）指定不导出虚拟磁盘映像文件。

**-nosubdir**

（可选）指定不在指定为导出路径的目录中创建包含 VM 显示名称的子目录。

**-snapshot *snapshot\_name***

（可选）指定要导出的快照的名称。

**-snapshot *snapshot\_ID***

（可选）指定要导出的快照的 GUID。

**-nostate**

（可选）指定仅导出 VM 规格。该选项可防止导出的 VM 试图引用原始 VM 磁盘映像。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。



**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：导出 VM**

本示例在没有快照的情况下导出 VM “TestVM”，并创建导出的 VM “textvmExp”。

```
dpmhyperv ExportVMEx -ws_user admin -ws_password #test# -hypervHost hserver -vm
TestVM -dest c:\Ex\TestVMExp -nosubdir -nosnapshots
```

**dpmhyperv GetHostSwitches 命令-获取主机的交换机**

dpmhyperv GetHostSwitches 命令用于返回 Hyper-V 主机的交换机。您可以指定筛选条件以获得特定的交换机。

此命令具有以下格式：

```
dpmhyperv GetHostSwitches
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname [filter filter_condition]
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host hostname**

指定 Hyper-V 服务器的主机名。

**-jobRef job\_ref\_identifier**

指定异步作业的引用标识符。

**-filter condition**

（可选）指定筛选条件，如“\*”。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：获取主机的交换机**

本示例返回主机 “hvserver” 的交换机。

```
dpmhyperv GetHostSwitches -ws_user admin -ws_password #test# -hypervHost hvserver
```

**示例：使用筛选获得主机的交换机**

本示例返回交换机名称中含有 “n” 的所有交换机。

```
dpmhyperv GetHostSwitches -ws_user admin -ws_password #test# -hypervHost hvserver  
-filter *n*
```

## dpmhyperv GetJobErrorInfo 命令-获取作业的错误信息

dpmhyperv GetJobErrorInfo 命令用于返回在 Hyper-V 主机上运行的异步作业的错误详细信息。

此命令具有以下格式：

```
dpmhyperv GetJobErrorInfo  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-jobRef job_ref_identifier  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host *hostname***

指定 Hyper-V 服务器的主机名。

**-jobRef *job\_ref\_identifier***

指定异步作业的引用标识符。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 获取作业的错误信息**

本示例返回作业

“\\hserver\root\virtualization:Msvm\_ConcreteJob.InstanceID="86C9700348-4117-ABA8-445456"” 的错误信息。

```
dpmhyperv GetJobErrorInfo -ws_user admin -ws_password admin -hypervHost hvserver
-jobRef
\\hserver\root\virtualization:Msvm_ConcreteJob.InstanceID="86C9700348-4117-ABA8-445456"
```

**dpmhyperv GetJobInfo 命令-获取作业的信息**

dpmhyperv GetJobInfo 命令用于返回 Hyper-V 主机上的异步作业的详细信息。详细信息包括完成的百分比、错误信息和开始时间。

此命令具有以下格式:

```
dpmhyperv GetJobInfo
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-jobRef job_ref_identifier
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

(可选) 指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host hostname**

指定 Hyper-V 服务器的主机名。

**-jobRef job\_ref\_identifier**

指定异步作业的引用标识符。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：获取作业的信息**

本示例返回 Hyper-V 作业

“\\hserver\root\virtualization:Msvm\_ConcreteJob.InstanceID="86TSTC4-ABA8-C82734036"”的相关信息。

```
dpmhyperv GetJobInfo -ws_user admin -ws_password #test# -hypervHost hvserver  
-jobRef  
\\hserver\root\virtualization:Msvm_ConcreteJob.InstanceID="86TSTC4-ABA8-C8273  
4036"
```

## dpmhyperv GetJobState 命令-获取作业的状态

dpmhyperv GetJobState 命令用于返回在 Hyper-V 主机上运行的异步作业的状态。

此命令具有以下格式：

```
dpmhyperv GetJobState  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-jobRef job_ref-identifier  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host *hostname***

指定 Hyper-V 服务器的主机名。

**-jobRef *job\_ref\_identifier***

指定异步作业的引用标识符。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：获取作业的状态**

本示例返回作业

“\\hvserver\root\virtualization:Msvm\_ConcreteJob.InstanceID="86CD04TEST82795D6F036"” 的状态。

```
dpmhyperv GetJobState -ws_user admin -ws_password #test# -hypervHost hserver  
-jobRef  
\\hserver\root\virtualization:Msvm_ConcreteJob.InstanceID="86CD04TEST82795D6F  
036"
```

## dpmhyperv GetServerInfo 命令-获取所有受管服务器

dpmhyperv GetServerInfo 命令用于返回在 CA Virtual Assurance 系统中注册的所有 Hyper-V 主机及其属性。

此命令具有以下格式：

```
dpmhyperv GetServerInfo  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

本示例返回在 CA Virtual Assurance 系统中注册的每个 Hyper-V 主机的属性。

```
dpmhyperv GetServerInfo -ws_user admin -ws_password #test#
```

## dpmhyperv GetVersion 命令-获取 PMM 版本

dpmhyperv GetVersion 命令用于返回平台管理模块 (PMM) 的版本。

此命令具有以下格式：

```
dpmhyperv GetVersion  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：获取 PMM 版本

本示例显示当前 Hyper-V PMM 的版本。

```
dpmhyperv GetVersion -ws_user admin -ws_password #test#
```

## dpmhyperv GetVMProperties 命令-获取 VM 属性

dpmhyperv GetVMProperties 命令用于查看 VM 属性。

此命令具有以下格式：

```
dpmhyperv GetVMProperties  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vm vmname|-vmguid vm_ID  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。



### 示例：获取 VM 属性

本示例显示主机“hserver”上的 VM “vm-01”的属性。

```
dpmhyperv GetVMProperties -ws_user admin -ws_password #test# -hypervHost hserver  
-vm vm01
```

## dpmhyperv GetVMState 命令-获取 VM 状态

dpmhyperv GetVMState 命令用于返回 VM 的当前状态。以下列表说明四个状态：

- disabled: VM 已关闭
- enabled: VM 已启动并正在运行
- paused: VM 暂时停止
- suspended: VM 已停止

此命令具有以下格式：

```
GetVMState  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vm vmname|-vmguid vm_ID  
[-pre]  
[-post]  
[-locale iso639value]
```

#### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

#### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

#### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

#### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

#### **-vm *vm\_name***

指定 VM 名称。

#### **-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

#### **-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：使用 VM 名称获取 VM 状态**

本示例返回主机 “hserver” 上的 VM “vm-01” 的状态。

```
dpmhyperv GetVMState -ws_user admin -ws_password #test# -hypervHost hserver -vm test-vm01
```

**示例：使用 VM GUID 获取 VM 状态**

本示例返回主机 “hserver” 上 GUID 为 “38FTESTED14644376” 的虚拟机的状态。

```
dpmhyperv GetVMState -ws_user admin -ws_password #test# -hypervHost hserver -vmguid 38FTESTED14644376
```

## dpmhyperv ImportVM 命令-导入 VM

dpmhyperv ImportVM 命令用于将 VM 导入到 Hyper-V 主机。

此命令具有以下格式：

```
dpmhyperv ImportVM  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password] -hypervHost hostname  
-path location  
[-name vm_name]  
[-noNewID]  
[-async]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-path *location***

定义导入的 VM 的位置。

**-vm *vm\_name***

指定 VM 名称。

**-noNewID**

（可选）指示不生成新的 VM GUID。在这种情况下，导入的 VM 和作为导入源的 VM 具有相同的 GUID。建议不要使用该选项，以避免与其他 VM 产生潜在的冲突。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将 VM 导入到 Hyper-V 主机**

本示例将 VM “test-vm01-imported” 导入到 Hyper-V 主机 “hserver”。该命令不生成新的 ID。

```
dpmhyperv ImportVM -ws_user admin -ws_password #test# -hypervHost hserver -path C:\data\Export\test-vm01 -name test-vm01-imported -noNewID.
```

## dpmhyperv ImportVMEx 命令-复制并导入 VM

dpmhyperv ImportVMEx 命令用于将已导出的 VM 复制并导入到 Hyper-V 主机。

此命令具有以下格式：

```
dpmhyperv ImportVMEx  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password] -hypervHost hostname  
-path location [dest dest_location]  
[-name vm_name]  
[-noNewID]  
[-async]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-path *location***

定义服务器的位置。

**-dest *dest\_location***

（可选）定义导入文件的目标位置。

**-vm *vm\_name***

指定 VM 名称。

**-noNewID**

（可选）指示不生成新的 VM GUID。在这种情况下，导入的 VM 和作为导入源的 VM 具有相同的 GUID。建议不要使用该选项，以避免与其他 VM 产生潜在的冲突。

**-async**

（可选）以异步模式运行命令。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将 VM 复制并导入到 Hyper-V 主机**

本示例将已导出的 VM 导入到 Hyper-V 主机，并且不为其生成新的 ID。

```
dpmhyperv ImportVMEx -ws_user admin -ws_password #test# -hypervHost hserver -
path C:\data\Export\test -name test-vm01-imported -noNewID
```

**dpmhyperv SetClustered 命令-在群集中添加或删除 VM**

dpmhyperv SetClustered 命令用于在 Hyper-V 群集中添加或删除 VM。

此命令具有以下格式：

```
dpmhyperv SetClustered
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-vm VM_name|-vmguid vm_ID
-clustered {on|off}
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host hostname**

指定 Hyper-V 服务器的主机名。

**-vm vm\_name**

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-clustered {on|off}**

指示向群集中添加 VM 还是从群集中删除 VM。

开

指示向群集中添加 VM。

关

指示从群集中删除 VM。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将 VM 附加到 Hyper-V 群集**

本示例将 GUID 为 “38F29B6A-TEST-BFED14644376” 的 VM 附加到 Hyper-V 群集。

```
dpmhyperv SetClustered -ws_user admin -ws_password #test# -hypervHost hserver  
-vmguid 38F29B6A-TEST-BFED14644376 -clustered on
```

**示例：从 Hyper-V 群集删除 VM**

本示例从 Hyper-V 群集删除 VM “test-vm01”。

```
dpmhyperv SetClustered -ws_user admin -ws_password #test# -hypervHost hserver -vm  
test-vm01 -clustered off
```

## dpmhyperv SetSysPrepProperties 命令-设置 Sysprep 属性

dpmhyperv SetSysPrepProperties 命令用于在 Hyper-V 服务器上设置 Sysprep 属性。

此命令具有以下格式：

```
dpmhyperv SetSysPrepProperties
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-vm VM_name|-vmguid vm_ID
[-jobRef job_reference]
[-computerName computer_name]
[-ip4addr ip4_address]
[-ip4dhcp]
[-ip4dhcpint ip4dhcpint]
[-ip4mask network_mask]
[-ip4gw gateway_address]
[-ip4metric ip_metric]
[-ip4dns dnsserver_ip]
[-disableAdmin]
[-adminPass administrator_password]
[-autoLogon autologon]
[-duplicatorString duplicator_string]
[-timezone timezone]
[-productKey product_key]
[-userName user_name]
[-organization org_name]
[-domain domain_name]
[-domainAdmin domain_admin_login]
[-domainAdminPass domain_admin_password]
[-workgroup workgroup_name]
[-adminUser admin_username]
[-adminUserPass admin_password]
[-adminGroup admin_groupname]
[-custom custom]
[-pre]
[-post]
[-locale iso639value]
```

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-jobRef *job\_reference***

指定作业标识符。

**-computerName *computer\_name***

(可选) 指定计算机的名称。

**-ip4addr *ip4\_address***

(可选) 指定要分配给 VM 接口的静态 IPv4 地址。

**-ip4dhcp *ip4dhcpInt***

(可选) 指定 DHCP 地址。

**-ip4mask *network\_mask***

(可选) 指定要分配给 VM 的子网掩码。该选项与 **-ip4addr** 选项结合使用。

**-ip4gw *gateway\_address***

(可选) 指定该选项可为 VM 设置网关。该选项与 **-ip4addr** 选项结合使用。

**-ip4metric *ip\_metric***

(可选) 指定要为 VM 设置的接口度量标准。该选项与 **-ip4addr** 选项结合使用。

**-ip4dns *dnserver\_ip***

(可选) 指定 VM 的 DNS 服务器。该选项与 **-ip4addr** 选项结合使用。

**-disableAdmin**

(可选) 指定该选项可禁用 VM 的默认管理员帐户。对该参数的支持需要一个使用 Sysprep 工具创建的 Windows 映像。该选项在异步模式下是无效的。

**-adminPass *administrator\_password***

(可选) 指定 VM 的默认管理员密码。



**-autoLogon *autologon***

（可选）指定该选项可设置在 Sysprep 进程完成后使用默认管理员帐户自动登录的帐户的数目。

**-duplicatorString *duplicate\_string***

（可选）指定要在 VM 注册表中设置的系统复印机的名称。

**-timeZone *timezone***

（可选）指定使用模板创建的 VM 所用的时区。

默认值：

-1

**-productKey *product\_key***

（可选）指定 VM 的 Windows 产品激活密钥。

**-userName *User\_name***

（可选）指定 VM 中 Windows 的用户名。

**-organization *org\_name***

（可选）指定 VM 中 Windows 的用户名。该选项在异步模式下是无效的。

**-domain *domain\_name***

（可选）指定 VM 的域名。

**-domainAdmin *domain\_admin\_login***

（可选）指定域管理员登录名。

**-domainAdminPass *domain\_admin\_password***

（可选）指定域管理员帐户的密码。该选项对于异步模式是无效的。

**-workgroup *workgroup\_name***

（可选）指定 VM 的工作组。该选项对于异步模式是无效的。

**-adminUser *admin\_username***

（可选）指定是默认 Administrators 组成员的用户名。

**-adminUserPass *admin\_password***

（可选）指定默认 Administrators 组的密码。

**-custom *custom***

（可选）指定在 Sysprep 进程结束时执行的自定义命令的逗号分隔列表。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：在 Hyper-V 服务器上设置 Sysprep 属性**

本示例在主机“hserver”上设置 Sysprep 属性。

```
dpmhyperv SetSysPrepProperties -hypervHost hserver -vm VM_05 -ip4addr 130.119.0.40 -ip4mask 255.255.0.0 -ip4gw 130.119.0.100
```

## **dpmhyperv SetVMDisk 命令-将驱动器或磁盘分配给驱动器控制器**

SetVMDisk 命令用于将虚拟驱动器或磁盘分配给驱动器控制器。

此命令具有以下格式：

```
dpmhyperv SetVMDisk  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password] -hypervHost hostname  
-vm vm_name|-vmguid vm_ID  
-drive {HDD|DVD} -driveType {scsi|ide}  
[-path location]  
-id drive_ID_number  
[-lun logical_unit_num]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost hostname**

指定 Hyper-V 主机的名称。

**-vm vm\_name**

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-drive {HDD|DVD}**

指定将映像分配给的驱动器的类型。选项包括：

**HDD**

指示驱动器类型是硬盘驱动器。

**DVD**

指示驱动器类型是 CD/DVD 驱动器。

**-driveType {scsi|ide}**

指定将映像分配给的驱动器控制器的类型。选项包括：

**scsi**

指示驱动器控制器是 SCSI。

**ide**

指示驱动器控制器是 IDE。对于 CD/DVD 映像，控制器类型必须是 IDE。

**-path *location***

（可选）定义虚拟磁盘或驱动器的位置。

**-id *drive\_ID\_number***

指定驱动器的 ID 号。对于 IDE，请指定 0；对于 SCSI 驱动器，请指定 0、1、或 2。

**-lun *logical\_unit\_num***

（可选）指定 SCSI 驱动器的逻辑单元号 (LUN)。对于 IDE 控制器，LUN 必须是 0 或 1；对于 SCSI 控制器，LUN 必须是介于 0 到 63 之间的数字。如果没有指定数字，则会自动分配可用通道。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：在驱动器控制器上分配驱动器

本示例将虚拟硬盘驱动器映像文件“disk.vhd”分配给第一个 SCSI 控制器上的第一条通道：

```
dpmhv-setVMDisk -hypervHost hvserver -vm TestVM -type scsi -id 0 -lun 0 -drive HDD -path c:\VHDs\disk.vhd
```

## dpmhyperv SetVMProperties 命令-设置 VM 的属性

dpmhyperv SetVMProperties 命令用于更改 VM 的属性。

此命令具有以下格式：

```
dpmhyperv SetVMProperties  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vm vm_name|-vguid vm_ID  
[-memory memory_size]  
[-cpuSocketCount total_CPU_socket_count]  
[-cpuidLimit CPU_ID_Limit]  
[-cpufeatlimit CPU_Feature_Limit]  
[-cpuReserve cpu_reserver]  
[-cpuReserve cpu_reserve]  
[-cpuLimit max_CPU_usage]  
[-cpuWeight cpu_weight]  
[-startAction {none|auto|always}]  
[-startDelay start_delay_secs]  
[-stopAction {save|off|shutdown}]  
[-recoveryAction {none|restart|revert}]  
[-pre]  
[-post]  
[-locale iso639value]
```

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

### **-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-memory *memory\_size***

(可选) 定义 VM 的内存大小 (MB)。

**-cpuSocketCount *total\_CPU\_socket\_count***

(可选) 定义 VM 中 CPU 内核的数量。CPU 内核不能超过 Hyper-V 主机中可用的 CPU 内核的总数。

**-cpuidLimit *CPU\_ID\_Limit***

(可选) 指示为了与传统操作系统 (例如 Windows NT) 更好地兼容, VM 是否必须减小 CPU 标识符。

**cpufeatlimit *CPU\_Feature\_Limit***

(可选) 指示 VM 是否必须限制公开给操作系统的 CPU 功能。

**-cpuReserve *cpu\_reserve***

(可选) 指定保留供 VM 使用的 CPU 资源的量。

**-cpuLimit *max\_CPU\_usage***

(可选) 指定 VM 可以占用的 CPU 资源的最大量。

**-cpuWeight *cpu\_weight***

(可选) 在 1 至 10000 的范围内指定虚拟机的相对权重。

默认: 100

**-startAction {none|auto|always}**

(可选) 指定在执行启动操作之前等待的时间间隔。选项包括:

无

不执行任何操作。

自动

如果 VM 在 Hyper-V 主机关闭之前正在运行, 则自动启动该 VM。

always

在每次 Hyper-V 启动时启动 VM。

**-startDelay *start\_delay\_secs***

(可选) 指定在 Hyper-V 主机启动后延迟启动 VM 的时间 (秒)。

**-stopAction {save|off|shutdown}**

（可选）指定在 Hyper-V 主机关闭之前要在 VM 上执行的操作。选项包括：

**save**

挂起 VM。

**关**

禁用 VM。

**shutdown**

关闭 VM。

**-recoveryAction {none|restart|revert}**

（可选）指定在 Hyper-V 主机意外关闭又重新启动之后要在 VM 上执行的操作。选项包括：

**无**

不执行任何操作。

**restart**

重新启动 VM。

**revert**

恢复为上次的快照。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：设置 VM 的属性**

本示例将已分配 CPU 内核的数目更改为 2。

```
dpmhyperv -setVMProperties -ws_user dcaadmin -ws_password #test# -hypervHost  
hserver -vm TestVM -cpus 2
```

## dpmhyperv SetVMVirtFloppy 命令-将虚拟软盘映像分配给 VM 软盘驱动器

dpmhyperv SetVMVirtFloppy 命令用于将虚拟软盘映像分配给 Hyper-V 主机上的 VM 软盘驱动器。

此命令具有以下格式：

```
dpmhyperv SetVMVirtFloppy
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password] -hypervHost hostname
-vm vm_name|-vguid vm_ID
[-path location]
[-pre]
[-post]
[-locale iso639value]
```

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

### **-vm *vm\_name***

指定 VM 名称。

### **-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

### **-path *location***

定义虚拟软盘映像的位置。

### **-pre**

（可选）操作完成之前生成事件。

### **-post**

（可选）操作完成之后生成事件。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：将虚拟软盘映像分配给 Hyper-V 主机上的 VM 软盘驱动器**

本示例将虚拟软盘映像“1.vfd”连接到 Hyper-V 主机“hserver”上的 VM 软盘驱动器。

```
dpmhyperv SetVMVirtFloppy -ws_user admin -ws_password test -hypervHost hserver  
-vm test-vm01 -path C:\data\Floppy\1.vfd
```

**dpmhyperv ShowClusterSharedVolumes 命令-显示所有群集共享卷**

dpmhyperv ShowClusterSharedVolumes 命令用于列出 Hyper-V 服务器上所有的群集共享卷。

此命令具有以下格式：

```
dpmhyperv ShowClusterSharedVolumes  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。



### 示例：显示 Hyper-V 主机上所有的群集共享卷

本示例列出主机“hserver”上的所有共享卷。

```
dpmhyperv ShowClusterSharedVolumes -ws_user admin -ws_password #test#  
-hypervHost hserver
```

## dpmhyperv ShowDirectories 命令-显示 Hyper-V 服务器上的目录属性

dpmhyperv ShowDirectories 命令用于列出 Hyper-V 主机上的目录的所有属性。

此命令具有以下格式：

```
dpmhyperv ShowDirectories  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-directory directory_path  
[-pre]  
[-post]  
[-locale iso639value]
```

#### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

#### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

#### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

#### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

#### **-directory *location***

定义 Hyper-V 主机上的目录的位置。

#### **-pre**

（可选）操作完成之前生成事件。

#### **-post**

（可选）操作完成之后生成事件。

#### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：显示 Hyper-V 主机上的目录的属性**

本示例列出主机“hserver”上的目录“c:\test”的所有属性。

```
dpmhyperv ShowDirectories -ws_user admin -ws_password #test# -hypervHost hserver  
-directory c:\test
```

**dpmhyperv ShowLogicalDisks 命令-显示所有逻辑磁盘**

dpmhyperv ShowLogicalDisks 命令用于列出 Hyper-V 主机上的所有逻辑磁盘及其属性。

此命令具有以下格式：

```
dpmhyperv ShowLogicalDisks  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：显示 Hyper-V 主机上的所有逻辑磁盘

本示例列出 Hyper-V 主机 “hserver” 上的所有逻辑磁盘。

```
dpmhyperv ShowLogicalDisks -ws_user admin -ws_password #test# -hypervHost hserver
```

## dpmhyperv ShowPhysDisks 命令-显示主机的物理磁盘

dpmhyperv ShowPhysDisks 命令用于显示 Hyper-V 主机上的所有物理磁盘。

此命令具有以下格式：

```
dpmhyperv ShowPhysDisks  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
[-prompt]  
-hypervHost hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

#### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

#### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

#### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

#### **-hypervHost *hostname***

指定 Hyper-V 主机的名称。

#### **-prompt {yes|no}**

（可选）指定是否提示您输入用户 ID 和密码。

默认值：Yes

#### **-pre**

（可选）操作完成之前生成事件。

#### **-post**

（可选）操作完成之后生成事件。

#### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：显示主机的物理磁盘**

本示例显示 Hyper-V 主机 “hserver” 上的所有物理磁盘。

```
dpmhyperv ShowPhysDisks -ws_user admin -ws_password #test# -hypervHost hserver
```

## **dpmhyperv ShowSCVMMHardwareProfiles 命令-显示所有 SCVMM 硬件配置文件**

dpmhyperv ShowSCVMMHardwareProfiles 命令用于列出 SCVMM 主机上的所有硬件配置文件。

此命令具有以下格式：

```
dpmhyperv ShowSCVMMHardwareProfiles  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password] -scvmmHost scvmm_hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-scvmmHost *scvmm\_hostname***

（可选）指定 SCVMM 服务器主机的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：获取 SCVMM 主机上的硬件配置文件

本示例显示 SCVMM 主机 “ms01” 上的所有硬件配置文件。

```
dpmhyperv ShowSCVMMHardwareProfiles -ws_user admin -ws_password #test#  
-scvmmHost ms01
```

## dpmhyperv ShowSCVMMGuestOSProfiles 命令-显示来宾操作系统配置文件

dpmhyperv ShowSCVMMGuestOSProfiles 命令用于列出 SCVMM 主机上来宾操作系统的所有配置文件。

此命令具有以下格式：

```
dpmhyperv ShowSCVMMGuestOSProfiles  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-scvmmHost scvmm_hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

#### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

#### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

#### **-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

#### **-scvmmHost *scvmm\_hostname***

（可选）指定 SCVMM 服务器主机的名称。

#### **-pre**

（可选）操作完成之前生成事件。

#### **-post**

（可选）操作完成之后生成事件。

#### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：显示来宾操作系统配置文件**

本示例列出 SCVMM 主机 “ms01” 上的所有来宾操作系统。

```
dpmhyperv ShowSCVMMGuestOSProfiles -ws_user admin -ws_password #test# -scvmmHost ms01
```

**dpmhyperv ShowTemplates 命令-显示所有 VM 模板**

dpmhyperv ShowTemplates 命令用于列出 Hyper-V 主机上的所有 VM 模板。

此命令具有以下格式：

```
dpmhyperv ShowTemplates  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
[-filter condition]  
[-scvmmHost scvmm_hostname]  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-filter *condition***

（可选）指定筛选条件，如 “\*”。

**-scvmmHost *scvmm\_hostname***

（可选）指定 SCVMM 服务器主机的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：显示所有 VM 模板**

本示例列出主机 “hserver” 上的所有 VM 模板。

```
dpmhyperv ShowTemplates -ws_user admin -ws_password #test# -hypervHost hserver
```

**dpmhyperv ShowVirtFloppy 命令-显示虚拟软盘驱动器列表**

dpmhyperv ShowVirtFloppy 命令用于列出分配给 VM 的所有虚拟软盘。

此命令具有以下格式：

```
dpmhyperv GetVMState
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-vm vmname|-vmguid vm_ID
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-vm *vm\_name***

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：显示所有软盘**

本示例列出分配给 GUID 为“38900009000000yy”的 VM 的所有虚拟软盘。

```
dpmhyperv ShowVirtFloppy -ws_user admin -ws_password #test# -hypervHost hserver  
-vmguid 38900009000000yy
```

## dpmhyperv ShowVMDisks 命令-显示所有虚拟磁盘

dpmhyperv ShowVMDisks 命令用于列出 Hyper-V 主机上的 VM 中的所有虚拟磁盘。

此命令具有以下格式：

```
dpmhyperv ShowVMDisks  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-vm vmname|-vmguid vm_ID  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost hostname**

指定 Hyper-V 主机的名称。

**-vm vm\_name**

指定 VM 名称。

**-vmguid vm\_ID**

指定 VM 的唯一 GUID。



**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：显示 Hyper-V 主机上的所有虚拟磁盘**

本示例列出主机 “hserver” 上的 VM “vm01” 中的所有虚拟磁盘及其属性。

```
dpmhyperv ShowVMDisks -ws_user admin -ws_password #test# -hypervHost hserver -vm
test-vm01
```

**dpmhyperv ShowVMNics 命令-显示所有网络接口卡**

dpmhyperv ShowVMNics 命令用于列出 VM 中的所有网络接口卡。

此命令具有以下格式：

```
dpmhyperv ShowVMNics
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-vm vmname|-vmguid vm_ID
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost hostname**

指定 Hyper-V 主机的名称。

**-vm vm\_name**

指定 VM 名称。

**-vmguid *vm\_ID***

指定 VM 的唯一 GUID。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 显示所有网络接口卡**

本示例列出主机 “hserver” 上的 VM “vm01” 中的所有网络接口卡及其属性。

```
dpmhyperv ShowVMDisks -ws_user admin -ws_password #test# -hypervHost hserver -vm test-vm01
```

## dpmhyperv ShowVMs 命令-显示所有 VM

dpmhyperv ShowVMs 命令用于列出 Hyper-V 主机上的所有 VM。该命令将获取 VM 名称及其对应的 GUID。

此命令具有以下格式:

```
dpmhyperv ShowVMs  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

(可选) 指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

(可选) 指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

(可选) 指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：显示所有 VM**

本示例列出主机 “hserver” 上的所有 VM 模板及其对应的 GUID。

```
dpmhyperv ShowVMs -ws_user admin -ws_password #test# -hypervHost hserver
```

**dpmhyperv ShowVMSCSIControllers 命令-显示所有 SCSI 控制器**

dpmhyperv ShowVMSCSIControllers 命令用于列出 VM 中的所有 SCSI 控制器。

此命令具有以下格式：

```
dpmhyperv ShowVMSCSIControllers
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
-vm vmname|-vmguid vm_ID
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**-hypervHost hostname**

指定 Hyper-V 主机的名称。

**-vm vm\_name**

指定 VM 名称。

**-vmguid vm\_ID**

指定 VM 的唯一 GUID。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：显示 SCVMM 主机上的所有 SCSI 控制器**

本示例列出主机 “hserver” 上的 VM “vm01” 中的所有 SCSI 控制器及其属性。

```
dpmhyperv ShowVMSCSIControllers -ws_user admin -ws_password #test# -hypervHost  
hserver -vm vm01
```

## **dpmhyperv ValidateAgentSNMPAccess 命令-验证对代理的 SNMP 访问**

dpmhyperv ValidateAgentSNMPAccess 命令用于验证是否可以使用 SNMP 团体字符串访问代理。

此命令具有以下格式：

```
dpmhyperv ValidateAgentSNMPAccess  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
-hypervHost hostname  
-snmpAccessName snmp_name -checkWriteAccess write_access  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-host *hostname***

指定 Hyper-V 服务器的主机名。

**-snmpAccessName *snmp\_name***

指定 SNMP 团体字符串。

**-checkWriteAccess *write\_access***

指定是否选择写入权限。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：验证对代理的 SNMP 访问**

本示例验证对主机 “hserver” 的 SNMP 访问。

```
dpmhyperv ValidateAgentSNMPAccess -hypervHost hserver -snmpAccessName public
-checkWriteAccess -ws_user admin -ws_password #test#
```

**dpmhyperv ValidateServerAccess 命令-检查服务器访问权限**

dpmhyperv ValidateServerAccess 命令用于检查是否可以访问主机服务器。

此命令具有以下格式：

```
dpmhyperv ValidateServerAccess
[-ws_user username]
[-ws_password password]
[-ws_encrypted_password]
-hypervHost hostname
[-user username]
[-pass password]
[-pre]
[-post]
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-user *username***

（可选）指定用户的名称。

**-pass *password***

（可选）指定用户的密码。

**-pre**

（可选）操作完成之前生成事件。

**-post**

（可选）操作完成之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：验证服务器访问权限**

本示例验证是否可以访问 Hyper-V 主机 “hserver”。

```
dpmhyperv ValidateServerAccess -ws_user admin -ws_password #test# -hypervHost hserver
```

## dpmhyperv DeleteTemplate 命令-删除模板

dpmhyperv DeleteTemplate 命令用于删除 Hyper-V 主机上的模板。

此命令具有以下格式：

```
dpmhyperv Deletetemplate  
[-ws_user username]  
[-ws_password password]  
[-ws_encrypted_password]  
[-prompt]  
-hypervHost hostname  
-template template_name  
[-pre]  
[-post]  
[-locale iso639value]
```

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**-ws\_encrypted\_password**

（可选）指示 CA Virtual Assurance 密码是否加密。

**-hypervHost *hostname***

指定 Hyper-V 主机的名称。

**-prompt {*yes*|*no*}**

(可选) 指定是否提示您输入用户 ID 和密码。

默认值: Yes

**-template *template\_name***

指定要删除的模板名称。

**-pre**

(可选) 操作完成之前生成事件。

**-post**

(可选) 操作完成之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 删除模板**

本示例删除 Hyper-V 主机 “hserver” 上的模板 “template\_5”。

```
dpmhyperv ShowPhysDisks -ws_user admin -ws_password #test# -hypervHost hserver  
-template template_5
```

## CA IBM LPAR CLI 命令

您可以使用 CLI 为 CA IBM LPAR 命令编写脚本并使其自动化, 然后基于命令结果运行操作。也可以在 AutoShell 中使用对应的命令。

## dpmlpar cycle 命令-循环逻辑分区

dpmlpar cycle 命令用于打开、关闭、重置或挂起逻辑分区。

此命令具有以下格式：

```
dpmlpar cycle
[-sc sc_url]
-powerop {activate|restart|shutdown}
-hmc name
-managed_system managementsystemname
-partition_name partitionname
[-type {delayed|immediate|os_shutdown|immediate_os_shutdown}]
[-profile_name profilename]
[-activate_bootmode {normal|open_firmware}]
[-activate_keylock {normal|manual}]
[-pre] [-post]
[-ws_user username]
[-ws_password password]
[-prompt <no|yes>]
[-ws_remote_user username]
[-ws_remote_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-powerop {activate|restart|shutdown}**

指定要在 LPAR 上执行的电源操作。选项包括：

#### **activate**

打开 LPAR。

#### **restart**

在必要时关闭 LPAR，然后再将其打开。

#### **shutdown**

关闭 LPAR。



**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {*immediate* | *os\_shutdown* | *immediate\_os\_shutdown*}**

指定使用映像操作类型资源组 (*res\_group*) 或单个资源 (*individual\_res*)。选项包括：

***immediate***

立即关闭分区。如果只更新部分数据，该选项可能导致不适当的结果。

***os\_shutdown***

通过发出用于典型关闭的命令来关闭分区。必须映像分区，才能成功执行该选项。

***immediate\_os\_shutdown***

通过发出操作系统命令以尽快关闭系统来关闭分区。该命令会跳过典型的关闭活动（包括向其他用户发送消息）。必须映像分区，才能成功执行该选项。

**-profile\_name *lparprofile* (仅 HMC)*****[-profile\_name lparname]* (仅 IVM)**

指定激活逻辑分区时要使用的分区配置文件。对于 HMC 为必需。对于 IVM 为可选，如果已指定，该名称必须与分区名称匹配。

**-activate\_bootmode {*normal* | *open\_firmware*}**

(可选) 指定激活操作的键盘锁定模式。选项包括：

***normal***

以典型方式启动分区。

***open\_firmware***

启动分区并打开“打开固件提示”。

**-activate\_keylock {normal|manual|bypass}**

（可选）指定激活操作的键盘锁定模式。选项包括：

**normal**

以无人照管模式启动分区，激活过程中无需用户交互。

**手工**

以照管模式启动分区，激活期间需要用户交互。

**bypass**

不激活键盘锁定模式。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

---

**-locale *iso639value*****-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

**-ws\_remote\_user *username***

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

（可选）指定 CA Virtual Assurance 远程密码。

---

**-locale *iso639value*****示例：使用默认配置文件激活逻辑分区**

本示例使用默认配置文件激活逻辑分区 “testlpar”。

```
dpmlpar cycle -powerop activate -hmc uslihmc  
-managed_system testComputer -partition_name testlpar
```

### 示例：使用指定的配置文件激活逻辑分区

本示例使用配置文件 testlparprofile 激活逻辑分区 “testlpar”。

```
dpmlpar cycle -powerop activate -hmc usliivm
-managed_system testComputer -partition_name testlpar
-profile_name testlparprofile
```

### 示例：激活逻辑分区以打开固件提示

本示例使用配置文件 testlparprofile 激活逻辑分区 “testlpar” 以打开固件提示。

```
dpmlpar cycle -powerop activate -hmc uslihmc
-managed_system testMS -partition_name testlpar
-activate_bootmode open_firmware
```

### 示例：延迟关闭逻辑分区

本示例执行逻辑分区 “testlpar” 的延迟关闭。

```
dpmlpar cycle -hmc testivm -powerop shutdown
-managed_system managedsys1 -partition testlpar -type delayed
```

## dpmlpar delete 命令-删除逻辑分区

dpmlpar delete 命令用于删除已停用的逻辑分区。

**重要信息！** 在发出该命令之前，请验证是否已备份所有重要数据。该命令删除 LPAR 及其数据文件。在您发出该命令时，必须关闭 LPAR。

此命令具有以下格式：

```
dpmlpar delete
[-sc sc_url]
-hmc name
-managed_system managedsystemname
-partition_name partitionname
[-pre] [-post]
[-ws_user username]
[-ws_password password]
[-prompt <false|true>]
[-ws_remote_user username]
[-ws_remote_password password]
[-delete_logical_value <false|true>]
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要删除的逻辑分区的名称。该分区必须存在于受管系统上。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value*****-prompt {*yes*|*no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: `Yes`

**-ws\_remote\_user *username***

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

(可选) 指定 CA Virtual Assurance 远程密码。

**-locale *iso639value***

### 示例：删除逻辑分区

本示例删除逻辑分区 “lpar01”。

```
dpmlpar delete -hmc hmc02 -managed_system system05
-partition_name lpar01
```

## dpmlpar getresources 命令-获取 LPAR 资源

dpmlpar getresources 命令用于检索 IBM AIX LPAR 的内存单元、处理器单元或处理器。

此命令具有以下格式：

```
dpmlpar getresources
[-sc sc_url]
-resource {all|memory|processors|processor_units}
-hmc name
-managed_system managedsystemname
-partition_name partitionname
[-pre] [-post]
[-ws_user username]
[-ws_password password]
[-prompt <false|true>]
[-ws_remote_user username]
[-ws_remote_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-resource {all|memory|processors|processor\_units}**

指定是检索特定资源还是所有资源。选项包括：

### **all**

检索并显示所有分区资源。

### **memory**

仅检索并显示分区的内存资源。

**processors**

仅检索并显示分区的处理器资源。

**processor\_units**

仅检索并显示分区的处理器单元资源。

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定为其列出资源的逻辑分区。该分区必须存在于受管系统上。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

---

**-locale *iso639value*****-prompt {*yes*|*no*}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-ws\_remote\_user *username***

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

(可选) 指定 CA Virtual Assurance 远程密码。

---

**-locale *iso639value*****示例: 获取 LPAR 的所有资源**

本示例检索 lpar06 的所有资源。

```
dpm_lpar getresources -resource all -partition_name lpar06  
-hmc ivm_02 -managed_system abcsystem4
```

**示例：获取 LPAR 的内存资源**

本示例检索 lpartest01 的内存资源。

```
dpmlpar getresources -resource memory -partition_name lpartest01
-hmc hmc_02 -managed_system abcsystem2
```

**dpmlpar image 命令-创建 IBM AIX LPAR**

dpmlpar image 命令用于创建使用 HMC/IVM 的 IBM AIX LPAR 和目标 IBM AIX 受管系统。（可选）可以在创建 LPAR 之后部署 IBM AIX 操作系统。

此命令具有以下格式：

```
dpmlpar image [-sc sc_url]
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-profile_name lparprofile (仅 HMC)
[-profile_name lparname] (仅 IVM)

-min_mem size
-desired_mem size
-max_mem size

-proc_mode {shared|dedicated}
-min_proc_units units
-desired_proc_units units
-max_proc_units units

-min_procs number
-desired_procs number
-max_procs number

-max_virtual_slots number
-share_mode {capped|uncapped}
[-uncap_weight weight]
[-io_slot DRC index, {true|false}]
[-lpar_io_pool_id id {id, ...}]

[-virtual_serial_adapter slot_num, remote_lpar_name, remote_slot_num,
is_required]
[-virtual_scsi_client_adapter slot_num, remote_lpar_name, remote_slot_num,
is_required] |
[-virtual_scsi_server_client_adapter virtual_io_server_name,
backing_device_name, client_slot_num, is_required]
[-virtual_eth_adapter slot_num, is_IEEE, port_vlan_id, (additional_vlan_id,
additional_vlan_id, ...), trunk_priority, is_required]
[-virtual_fc_server_client_adapter virtual_io_server_name,
physical_fc_port_name, client_slot_num, is_required]
[virtual_io_server_name, physical_fc_port_name, client_slot_num, is_required; ...
]]
```

```
[-pre] [-post]
[-ws_user username]
[-ws_password password]

[-encrypted_password {no|yes}]
[-prompt <no|yes>]

[-ws_remote_user username]
[-ws_remote_password password]

[-create_logical_volume <false|true>]
[-logical_volume_size <value>]
[-volume_group_names <value>]
[-logical_volume_name <value>]
[-use_logical_volume_name_as_prefix {no|yes}]
[-default_vio_server_name vioserver]
[-itcm_server itcmservername]
[-locale iso639value]
```

用于使用 NIM MKSYSB 实用工具为 AIX 制作的参数

```
[-install_type {rte|mksysb}]

-mksysb mksysb_resource

-lpp lpp_resource

-bosinst_data bos_install_data_resource

-image_data image_data_resource
```

使用资源组开通 AIX 的参数

```
-provision_aix true
-type res_group
-res_group_name resourcegroupname
-machine_res_name machineresourcenname
-target_username targetusername
[-target_password targetpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
-nim_master_host_name nimmasterhostname
[-scalability_server servername]
[-deploy_template templatename]
-auto_deploy {yes|no}
[-wait [timeout]]
```



使用单个资源开通 AIX 的参数

```
-provision_aix true
-type individual_res
-machine_res_name machineresourcename
-lpp lppresource
-spot spotresource
-bosinst_data bosdata
-resolv_conf resolveconf
-fb_script fbscript
-post_inst_scripts script1,script2,script3
-target_username targetusername
[-target_password targetpassword]
[-auth_file authorizationfilename]
[-auth_comp componentID]
-nim_master_host_name nimmasterhostname
[-scalability_server servername]
[-deploy_template templatename]
-auto_deploy {yes|no}
[-wait [timeout]]
```

#### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

#### **-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

#### **-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

#### **-partition\_name *partitionname***

指定要创建的逻辑分区的名称。

#### **-profile\_name *lparprofile***（仅 HMC）

#### **[-profile\_name *lparname*]**（仅 IVM）

指定激活逻辑分区时要使用的分区配置文件。对于 HMC 为必需。对于 IVM 为可选，如果已指定，该名称必须与分区名称匹配。

**-min\_mem size**

定义分区的最小内存量。

**-desired\_mem size**

定义分区所需的内存量。

**-max\_mem size**

定义分区的最大内存量。

**-proc\_mode {shared|dedicated}**

指定分区的处理器模式的类型。选项包括：

**shared**

与其他分区共享处理器资源。

**dedicated**

指定分区具有专用处理器资源。

**-min\_proc\_units units**

定义该分区的最小处理单元数。

限制：.01 增量

注意：该选项仅适用于共享处理器。

**-desired\_proc\_units units**

定义为该分区分配的处理单元的数量。

限制：.01 增量

注意：该选项仅适用于共享处理器。

**-max\_proc\_units units**

定义该分区处理单元的最大数量。

限制：.01 增量

注意：该选项仅适用于共享处理器。

**-min\_procs number**

定义分区的虚处理器的最小数量。

注意：该选项仅适用于共享处理器。

**-desired\_procs number**

定义为该分区分配的虚处理器的数量。

注意：该选项仅适用于共享处理器。

**-max\_procs *number***

定义该分区的虚拟处理器的最大数量。

**注意：**该选项仅适用于共享处理器。

**-shared\_mode {capped|uncapped}**

指定受管系统是否允许逻辑分区使用共享处理器池中未指派给其他分区的空闲处理单元。

**注意：**该选项仅适用于共享处理器。

选项包括：

**capped**

指定分区只能使用指派给它的特定数量的处理单元。

**uncapped**

指定分区可以使用共享处理器池中空闲的处理器单元（当它们可用时）。

**-uncap\_weight *weight***

（可选）定义在选择不受限共享模式时处理优先级的加权平均值。

**限制：**0 - 255

**-max\_virtual\_slots *number***

定义该分区的虚拟适配器的最大数量。

**默认值：**2

**限制：**2 - 65,536

**-io\_slot *DRC-Index*,{true|false}**

（可选）指定物理组件的 I/O 插槽。您可以为不同的 I/O 设备多次指定该值。

**DRC-Index**

指定插槽动态重新配置连接器 (DRC) 索引。

**真**

指定 DRC 索引对于 I/O 插槽是必需的。

**假**

指定 DRC 索引对于 I/O 插槽不是必需的。

**-io\_pool\_id *id* {*id* ...}**

（可选）定义 I/O 适配器组，这些适配器可在没有来自 HMC 的任何活动干预的情况下被任何指定的逻辑分区组接管并使用。分区组可以是逗号分隔列表形式的 I/O 池 ID。

**-virtual\_serial\_adapter slot\_num, remote\_lpar\_name, remote\_slot\_num, is\_required**

(可选) 定义该逻辑分区的虚拟串行适配器。变量 *is\_required* 可以设置为 *true* 或 *false*。仅支持客户端串行适配器，但可以多次指定。

**默认值：**在插槽 1 和 2 中创建两个服务器串行适配器。

**注意：**不支持在 IVM 上创建 LPAR。

**-virtual\_eth\_adapter\_client slot\_num, is\_IEEE, port\_vlan\_id1 (vlanid2,vlanid3,...), trunk\_priority, is\_required**

(可选) 定义该逻辑分区的虚拟以太网适配器，可以多次指定。可选值有 *vlan\_idn* 和 *trunk\_priority*。变量 *is\_IEEE* 和 *is\_required* 可以设置为 *true* 或 *false*。其他 *vlan\_ids* 必须使用逗号分隔，并且包含在圆括号内。

**注意：**在 IVM 上创建 LPAR 时，*slot\_num* 必须大于或等于 4。

**示例：**4,false,1,,,true

**-virtual\_fc\_server\_client\_adapter virtual\_io\_server\_name, physical\_fc\_port\_name, client\_slot\_num, is\_required [;virtual\_io\_server\_name, physical\_fc\_port\_name, client\_slot\_num, is\_required;...]**

(可选) 在 VIO 服务器上定义虚拟光纤通道 (FC) 客户端适配器、FC 服务器适配器，并将 FC 服务器适配器与物理 FC 端口关联。将

-provision\_aix 选项设置为 *true* 时，该选项无效。该参数可以多次指定。变量 *is\_required* 可以设置为 *true* 或 *false*。

**-virtual\_scsi\_client\_adapter slot\_num, remote\_lpar\_name, remote\_slot\_num, is\_required**

(可选) 定义该逻辑分区的虚拟客户端 SCSI 适配器，可以多次指定。变量 *is\_required* 可以设置为 *true* 或 *false*。

**-virtual\_scsi\_server\_client\_adapter virtual\_io\_server\_name, backing\_device\_name, client\_slot\_num, is\_required**

(可选) 定义虚拟 SCSI 服务器设备和虚拟 SCSI 客户端设备，在创建服务器设备并确定服务器插槽号之后自动链接客户端设备。该参数可以多次指定。变量 *is\_required* 可以设置为 *true* 或 *false*。

**示例：**CUST-VIOSERVER,hdisk5,3,true

**注意：**如果使用 *dpmlpar image* 命令创建逻辑卷，并将

-create\_logical\_volume 选项设置为“是”，则

-virtual\_scsi\_server\_client\_adapter 选项不是必需的，应忽略。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**-encrypted\_password {*no* | *yes*}**

（可选）指定用户密码是否加密。

**默认值：** No

**-prompt {*yes* | *no*}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

**-ws\_remote\_user *username***

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

（可选）指定 CA Virtual Assurance 远程密码。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**-create\_logical\_volume {*no* | *yes*}**

（可选）指定是否创建逻辑卷。

**默认值：** no

**注意：** 如果使用 `dpmlpar image` 命令创建逻辑卷，并将 `-create_logical_volume` 选项设置为 “是”，则 `-virtual_scsi_server_client_adapter` 选项不是必需的，应忽略。

**-logical\_volume\_size *logical\_volume\_size***

（可选）定义逻辑卷的大小 (MB)。

**-volume\_group\_names *volume\_group\_1*, *volume\_group\_2*, *volume\_group\_3***

（可选）以逗号分隔的列表形式指定逻辑卷组名称。

**-logical\_volume\_name *logical\_volume\_name***

(可选) 定义逻辑卷的名称。

**-use\_logical\_volume\_name\_as\_prefix {no|yes}**

(可选) 指定是否将逻辑卷名称用作前缀。

默认值: no

**-default\_vio\_server\_name *vioserver***

(可选) 指定用于创建逻辑卷的 VIO 服务器。

**-itcm\_server *itcm\_servername***

(可选) 指定 Unicenter DSM 服务器的名称。

#### 用于使用 NIM mksysb 制作映像的参数

**-install\_type {rte|mksysb}**

(可选) NIM 安装类型。接受的值为 *rte* 或 *mksysb*。如果未指定 *-install\_type*，则 *rte* 为默认值。

**-mksysb *mksysbResource***

*mksysb* 资源。仅在 *-install\_type* 为 *mksysb* 时有效。

**-lpp *lpp\_resource***

指定许可的程序产品文件，以用于映像请求。如果 *-install\_type* 为 *mksysb*，则为可选。如果 *-install\_type* 为 *rte*，则为必需。

**-bosinst\_data *bos\_install\_data\_resource***

指定包含基础操作系统 (BOS) 安装程序的信息的文件。如果 *-install\_type* 为 *mksysb*，则为可选。如果 *-install\_type* 为 *rte*，则为必需。

**-image\_data *imageDataResource***

(可选) 指定映像数据资源文件，该文件描述如何在根卷上配置物理磁盘、卷组、逻辑卷、文件系统和分页空间。

#### 用于开通的参数

**-auth\_file *authorizationfilename***

(可选) 指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 *dpmutil set auth* 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

(可选) 指定可用于对主机和用户进行分组的组件 ID。

**-auto\_deploy {yes|no}**

指定是否自动部署 CA Virtual Assurance 代理。选项包括：

**是**

自动部署 CA Virtual Assurance 代理。

**否**

防止自动部署 CA Virtual Assurance 代理。

**默认值：** no

**-bosinst\_data *bos\_install\_data\_resource***

指定包含基础操作系统 (BOS) 安装程序的信息的文件。如果

-install\_type 为 mksysb，则为可选。如果 -install\_type 为 rte，则为必需。

**-deploy\_template *templatename***

(可选) 指定在 CA Virtual Assurance 中创建的通用模板的名称。

**注意：** 请勿将该模板与由 VMware vCenter 创建并管理的模板相混淆。

**-fb\_script *fbscript***

(可选) 定义在 BOS 安装过程完成之后，NIM 客户端初始启动时用于配置设备的文件的名称。

**-lpp *lpp\_resource***

指定许可的程序产品文件，以用于映像请求。如果 -install\_type 为 mksysb，则为可选。如果 -install\_type 为 rte，则为必需。

**-machine\_res\_name *machineresourcenname***

定义计算机资源的名称。该名称必须在 NIM 主服务器中预定义。

**-nim\_master\_host\_name *nimmasterhostname***

定义用于执行映像部署的 NIM 主服务器主机名。

**-post\_inst\_scripts *script1,script2,script3***

(可选) 指定安装后运行的逗号分隔脚本列表。

**-provision\_aix {true|false}**

表示创建分区之后，是否使用 NIM 对其进行映像。

**真**

使用 NIM 对创建的分区进行映像。如果您将其设置为 **true**，则使用 NIM 资源组或 NIM 单个资源参数。请参阅 **dpmnim image** 命令 | 使用资源组和 **dpmnim image** 命令部署 IBM AIX 映像 | 使用单个资源部署 IBM AIX 映像。NIM 开通开始时，返回作业 ID。

**假**

不使用 NIM 对创建的分区进行映像。不返回作业 ID。

**-res\_group\_name resource group name**

定义资源组的名称。

**-resolve\_conf resolveconf**

（可选）定义一个文件，该文件包含为本地解析程序例程定义域名协议名称服务器信息的有效 */etc/resolv.conf* 条目。

**-scalability\_server servername**

（可选）指定要用于软件分发的 Unicenter DSM 可扩展性服务器。

**-spot spotResource [mkysbResource]**

定义共享产品对象树，以用于映像制作请求。

**-target\_password targetpassword**

（可选）定义用于将代理部署到的目标主机服务器（要将映像部署到该服务器）的用户密码。如果未指定密码，将从授权文件检索密码。

**注意：**可使用 **dpmutil** CLI 设置授权文件。

**-target\_username targetusername**

定义用于将代理部署到的目标主机服务器（要将映像部署到该服务器）的用户名。

**-type {res\_group|individual\_res}**

指定使用映像操作类型资源组或单个资源。

**res\_group**

指定使用资源组进行映像。

**individual\_res**

指定使用单个资源进行映像。



**-wait timeout**

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

""

无超时值。

0

无限等待。

**示例: 创建 IBM AIX 逻辑分区**

本示例创建逻辑分区, 但不开通。

```
dpmlpar image -hmc ivm01 -managed_system testMS
-partition_name lpartest01 -profile_name lpartest01
-max_virtual_slots 10 -min_mem 128 -desired_mem 256
-max_mem 2048 -min_procs 1 -desired_procs 2 -max_procs 5
-proc_mode shared -share_mode uncapped -uncap_weight 123
-min_proc_units .5 -max_proc_units 3.25 -desired_proc_units 1.75
-virtual_serial_adapter 2,serial_partition,1,true
-virtual_scsi_client_adapter 3,scsi_partition,7,true
-virtual_eth_adapter 4,true,1,(22,35,54),,true
-io_slot 21020003,false
```

**示例: 创建 IBM AIX 逻辑分区并使用单个资源对其进行映像**

本示例创建逻辑分区并使用单个资源将其开通。

```
dpmlpar image -hmc hmc01 -managed_system testMS
-partition_name IRTEST -profile_name Default
-max_virtual_slots 10 -min_mem 128 -desired_mem 256
-max_mem 2048 -min_procs 1 -desired_procs 1
-max_procs 1 -proc_mode dedicated
-virtual_serial_adapter 2,SerialPartition,1,true
-virtual_scsi_client_adapter 3,SCSIPartition,7,true
-virtual_eth_adapter 4,true,1,(22,35,54),0,true
-io_slot 21020003,false -provision_aix true
-type individual_res -lpp 530lpp_res -spot 530spot_res
-bosinst_data 530_bid_ow -resolv_conf master_net_conf
-post_inst_scripts piScript -machine_res_name MachineName
-nim_master_host_name machine.mydomain.com -auto_deploy no
-target_username root -target_password password
```

**示例：创建 IBM AIX 逻辑分区并使用资源组对其进行映像**

本示例创建逻辑分区并使用资源组将其开通。

```
dpmlpar image -hmc hmc02 -managed_system testMS
-partition_name RGTEST -profile_name Default
-max_virtual_slots 10 -min_mem 128 -desired_mem 256
-max_mem 2048 -min_procs 1 -desired_procs 2 -max_procs 5
-proc_mode dedicated -virtual_serial_adapter 2,SerialPartition,1,true
-virtual_scsi_client_adapter 3,SCSIPartition,7,true
-virtual_eth_adapter 4,true,1,(22,35,54),,true
-io_slot 21020003,false -provision_aix true -type res_group
-res_group_name Res_grp -machine_res_name Machine
-nim_master_host_name machine.mydomain.com -auto_deploy no
-target_username root -target_password password
```

**dpmlpar imgjobcheck 命令-检索 IBM AIX 映像作业的状态**

dpmlpar imgjobcheck 命令用于检索特定 CA Virtual Assurance 作业 ID 的 IBM AIX 逻辑分区映像作业的状态。

此命令具有以下格式：

```
dpmlpar imgjobcheck
[-sc sc_url]
-status jobID
[-pre] [-post]
[-ws_user username]
[-ws_password password]
[-prompt <no|yes>]
[-ws_remote_user username]
[-ws_remote_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-status *jobID***

指定用于获取作业状态的 CA Virtual Assurance 作业 ID。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value*****-prompt {*yes*|*no*}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-ws\_remote\_user *username***

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

（可选）指定 CA Virtual Assurance 远程密码。

**-locale *iso639value*****示例：使用作业 ID 检索 IBM AIX 映像作业的状态**

本示例使用 CA Virtual Assurance 作业 ID 42 获取 IBM AIX 逻辑分区映像作业的作业状态。

```
dpm_lpar imgjobcheck -status 42
```

## dpmlpar list 命令-列出 HMC 资源

dpmlpar list 命令用于列出来自硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM) 的信息。

此命令具有以下格式：

```
dpmlpar list [-sc sc_url]  
-hmc name  
[-managed_system managedsystemname]  
[-partition_name partitionname]  
-display {managed_systems | managed_system_details | partitions | io_components  
| backing_devices | scsi_adapters | partition_details | profiles / wwpns}  
[-pre] [-post]  
[-ws_user username]  
[-ws_password password]  
[-prompt <no|yes>]  
[-ws_remote_user username]  
[-ws_remote_password password]  
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-display**

{*managed\_systems* | *managed\_system\_details* | *partitions* | *io\_components* |  
*backing\_devices* | *scsi\_adapters* | *partition\_details* | *profiles* | *wwpns*}

指定要列出哪些资源。选项包括：

### **managed\_systems**

列出由管理硬件控制台控制的受管系统。

### **managed\_system\_details**

提供有关受管系统的详细信息。

### **partitions**

列出受管系统上的分区。

**io\_components**

列出受管系统的 I/O 组件。

**backing\_devices**

列出可作为后备设备附加到服务器虚拟 SCSI 适配器的设备。

**scsi\_adapters**

列出受管系统的 SCSI 适配器。

**partition\_details**

列出分区的详细信息。

**profiles**

列出分区的配置文件的列表。

**wwpns**

显示分区的全球端口名称。

**-hmc name**

指定要为其列出信息的 HMC/IVM。

**-managed\_system managedsystemname**

(可选) 指定想要列出其信息的逻辑分区驻留的受管系统。

**注意:** 请不要与 managed\_systems 选项一起使用。

**-partition\_name partitionname**

(可选) 指定您想要列出其信息的逻辑分区。

**注意:** 请不要与 managed\_systems 选项一起使用。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-prompt {yes|no}**

(可选) 指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-ws\_remote\_user username**

(可选) 指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

(可选) 指定 CA Virtual Assurance 远程密码。

**-locale *iso639value*****示例：显示受管系统的列表**

本示例显示硬件管理控制台“labhmc”的受管系统。

```
dpmlpar list -display managed_systems -hmc labhmc
```

**示例：显示 SCSI 适配器**

本示例显示受管系统“testMS”的 SCSI 适配器。

```
dpmlpar list -display scsi_adapters -hmc uslihmc  
-managed_system testMS
```

**示例：显示分区详细信息**

本示例显示逻辑分区“testlpar”的详细信息。

```
dpmlpar list -display partition_details -hmc uslihmc  
-managed_system testMS -partition_name testlpar
```

## dpmlpar setresources 命令-添加内存资源

dpmlpar setresources 命令用于调整 IBM AIX LPAR 的内存单元。

此命令具有以下格式：

```
dpmlpar setresources  
[-sc sc_url]  
-add_memory value  
-hmc name  
-managed_system managedsystemname  
-partition_name partitionname  
-type {dynamic|all}  
[-pre] [-post]  
[-ws_user username]  
[-ws_password password]  
[-prompt <no|yes>]  
[-ws_remote_user username]  
[-ws_remote_password password]  
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-add\_memory *value***

定义添加到分区的内存量。

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {*dynamic*|*all*}**

指定调整是临时的还是永久的。选项包括:

***dynamic***

仅调整当前资源, 在您关闭分区时, 调整的值将丢失。

***all***

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

**-locale iso639value**

（可选）指定 **ISO 639\_3166** 组合以覆盖默认的英语输出（例如：法语为 **fr\_FR**）。要使用命令提示符的区域设置，请指定 **“native”**。示例：向逻辑分区添加内存

本示例向逻辑分区 “lptest01” 中添加 128 MB 的内存。

```
dpmlpar setresources -add_memory 128 -partition_name lptest01  
-hmc uni02 -managed_system usi01system1 -type dynamic
```

**示例：向逻辑分区添加内存并更新配置文件**

本示例向逻辑分区 “lptest01” 中添加 128 MB 的内存并更新分区配置文件。

```
dpmlpar setresources -add_memory 128 -partition_name lptest01  
-hmc uni02 -managed_system usi01system1 -type all  
-profile_name lptest01
```

## dpmlpar setresources 命令-添加处理器资源

dpmlpar setresources 命令用于为 IBM AIX LPAR 添加处理器和处理器单元。

此命令具有以下格式：

```
dpmlpar setresources  
[-sc sc_url]  
{-add_processors|add_processor_units} value  
-hmc name  
-managed_system managedsystemname  
-partition_name partitionname  
-type {dynamic|all}  
[-pre] [-post]  
[-ws_user username]  
[-ws_password password]  
[-prompt <no|yes>]  
[-ws_remote_user username]  
[-ws_remote_password password]  
[-locale iso639value]
```



**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: 80

**-add\_processors *value***

定义添加到分区的处理器的数量。

**-add\_processor\_units *value***

定义添加到分区的处理器单元。

限制: .01 增量

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {*dynamic*|*all*}**

指定调整是临时的还是永久的。选项包括:

***dynamic***

仅调整当前资源, 在您关闭分区时, 调整的值将丢失。

***all***

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

---

**-locale iso639value**

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值:** Yes

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

---

**-locale iso639value**

#### 示例：为分区添加处理器并更新配置文件

本示例向分区“lptest01”中添加处理器并更新分区配置文件。

```
dpm_lpar setresources -add_processors 1 -partition_name lptest01  
-hmc uni02 -managed_system MSystem1 -type all -profile_name lptest01
```

#### 示例：为分区添加处理器和处理器单元

本示例为分区“lptest01”暂时添加处理器和处理器单元资源。

```
dpm_lpar setresources -add_processors 1 -add_processor_units 4.5  
-partition_name lptest01 -hmc uni02 -managed_system MSystem1  
-type dynamic
```

## dpmlpar setresources 命令-减少内存资源

dpmlpar setresources 命令用于删除 IBM AIX LPAR 的内存单元。

此命令具有以下格式：

```
dpmlpar setresources
[-sc sc_url]
-subtract_memory value
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-type {dynamic|all}
[-pre] [-post]
[-ws_user username]
[-ws_password password]
[-prompt <no|yes>]
[-ws_remote_user username]
[-ws_remote_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-subtract\_memory *value***

定义要从分区删除的内存量。

### **-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

### **-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

### **-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {dynamic|all}**

指定调整是临时的还是永久的。选项包括：

**dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

**all**

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

---

**-locale iso639value****-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值：Yes

**-ws\_remote\_user username**

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password password**

（可选）指定 CA Virtual Assurance 远程密码。

---

**-locale iso639value****示例：减少逻辑分区中的内存**

本示例从逻辑分区“lptest01”中减少 128 MB 的内存。

```
dpm_lpar setresources -subtract_memory 128 -partition_name lptest01  
-hmc con02 -managed_system MSystem1 -type dynamic
```

**示例：减少逻辑分区中的内存并更新配置文件**

本示例从逻辑分区“lptest01”中减少 128 MB 的内存并更新分区配置文件。

```
dpm_lpar setresources -subtract_memory 128 -partition_name lptest01  
-hmc con02 -managed_system MSystem1 -type all
```

## dpmlpar setresources 命令-减少处理器资源

dpmlpar setresources 命令用于从 IBM AIX LPAR 中删除处理器和处理器单元。

此命令具有以下格式：

```
dpmlpar setresources
[-sc sc_url]
{-subtract_processors|subtract_processor_units} value
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-type {dynamic|all}
[-pre] [-post]
[-ws_user username]
[-ws_password password]
[-prompt <no|yes>]
[-ws_remote_user username]
[-ws_remote_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-subtract\_processors *value***

定义要从分区删除的处理器数量。

### **-subtract\_processor\_units *value***

定义要从分区删除的处理器单元。

限制：.01 增量

### **-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

### **-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {*dynamic*|*all*}**

指定调整是临时的还是永久的。选项包括：

**dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

**all**

调整当前资源和当前配置文件值。将为分区保存调整的值。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

---

**-locale *iso639value*****-prompt {*yes*|*no*}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值：Yes

**-ws\_remote\_user *username***

（可选）指定 CA Virtual Assurance 远程用户名。

**-ws\_remote\_password *password***

（可选）指定 CA Virtual Assurance 远程密码。

---

**-locale *iso639value*****示例：减少分区中的处理器并更新配置文件**

本示例从分区“lptest01”中减少一个处理器并更新分区配置文件

```
dpm_lpar setresources -subtract_processors 1 -partition_name lptest01  
-hmc hmc02 -managed_system MSystem1 -type all
```

**示例：减少分区中的处理器和处理器单元**

本示例暂时调整分区“lptest01”中的处理器和处理器单元资源。

```
dpmlpar setresources -subtract_processors 1 -subtract_processor_units 4.5
-partition_name lptest01 -hmc hmc02 -managed_system MSystem1 -type dynamic
```

**dpmlpar setresources 命令—更新内存资源**

dpmlpar setresources 命令用于调整 IBM AIX LPAR 的内存单元。

此命令具有以下格式：

```
dpmlpar setresources
[-sc sc_url]
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-type {dynamic|all|profile}
-min_mem size
-desired_mem size
-max_mem size
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {dynamic|all/profile}**

指定调整是临时的还是永久的。选项包括：

**dynamic**

仅调整当前资源，在您关闭分区时，调整的值将丢失。

**all**

调整当前资源和配置文件值。将为分区保存调整的值。

**profile**

调整配置文件值。

**-min\_mem size**

定义分区的最小内存量。

**-desired\_mem size**

定义分区所需的内存量。

**-max\_mem size**

定义分区的最大内存量。

## dpmlpar setresources 命令—更新处理器资源

dpmlpar setresources 命令用于更新 IBM AIX LPAR 的处理器。

此命令具有以下格式：

```
dpmlpar setresources
[-sc sc_url]
-hmc name
-managed_system managedsystemname
-partition_name partitionname
-type {dynamic|all/profile}
[-proc_mode {shared |dedicated}]
[-min_proc_units units]
-desired_proc_units units
[-max_proc_units units]
[-min_procs number]
-desired_procs number
[-max_procs number]
[-share_mode {capped |uncapped} [-uncap_weight weight]]
```



**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-hmc *name***

指定控制逻辑分区所在的受管系统的硬件管理控制台 (HMC) 或集成虚拟化管理器 (IVM)。

**-managed\_system *managedsystemname***

指定逻辑分区所驻留的受管系统。受管系统必须存在于 HMC/IVM 上。

**-partition\_name *partitionname***

指定要将资源应用至的逻辑分区。该分区必须存在于受管系统上。

**-type {*dynamic*|*all*|*profile*}**

指定调整是临时的还是永久的。选项包括:

***dynamic***

仅调整当前资源, 在您关闭分区时, 调整的值将丢失。

***all***

调整当前资源和配置文件值。将为分区保存调整的值。

***profile***

调整配置文件值。

**-proc\_mode {*shared*|*dedicated*}**

(可选) 指定分区的处理器模式的类型。选项包括:

***shared***

与其他分区共享处理器资源。

***dedicated***

指定分区具有专用处理器资源。

**-min\_proc\_units *units***

(可选) 定义该分区的最小处理单元数。

限制: .01 增量

注意: 该选项仅适用于共享处理器。

**-desired\_proc\_units *units***

定义为该分区分配的处理单元的数量。

限制: .01 增量

注意: 该选项仅适用于共享处理器。

**-max\_proc\_units *units***

(可选) 定义该分区的最大处理单元数。

限制: .01 增量

注意: 该选项仅适用于共享处理器。

**-min\_procs *number***

(可选) 定义该分区的最小虚拟处理器数。

注意: 该选项仅适用于共享处理器。

**-desired\_procs *number***

定义为该分区分配的虚处理器的数量。

注意: 该选项仅适用于共享处理器。

**-max\_procs *number***

(可选) 定义该分区的最大虚拟处理器数。

注意: 该选项仅适用于共享处理器。

**-shared\_mode {*capped* | *uncapped*}**

指定受管系统是否允许逻辑分区使用共享处理器池中未指派给其他分区的空闲处理单元。

注意: 该选项仅适用于共享处理器。

选项包括:

***capped***

指定分区只能使用指派给它的特定数量的处理单元。

***uncapped***

指定分区可以使用共享处理器池中空闲的处理器单元 (当它们可用时)。

**-uncap\_weight weight**

（可选）定义在选择不受限共享模式时处理优先级的加权平均值。

限制：0 - 255

## CA Microsoft Cluster Server CLI 命令

可以使用 CLI 编写 CA MSCS 命令的脚本并使其自动化，然后根据命令的结果运行操作。也可以在 AutoShell 中使用对应的命令。

### dpmmscs getclusterinfo 命令—获取群集的信息

getclusterinfo 命令用于返回群集的常规信息。

此命令具有以下格式：

```
dpmmscs getclusterinfo
[-sc <scUrl>]
-cluster <cluster>
[-locale iso639value]
[-ws_user username]
[-ws_password password]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-cluster**

指定群集名称。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：获取群集的信息**

```
dpmmcs.exe getclusterinfo -cluster MYCLUSTER -ws_user admin -ws_password admin
```

## dpmmcs getgroupresourcesinfo 命令—获取群集组的资源信息

getgroupresourcesinfo 命令用于返回群集组的资源信息

此命令具有以下格式：

```
dpmmcs getgroupresourcesinfo  
[-sc <scUrl>]  
-cluster <cluster>  
-resgroup <resource group>  
[-locale iso639value]  
[-ws_user username]  
[-ws_password password]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-resgroup**

指定资源组。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：获取群集组的资源信息**

```
dpmmscs.exe getgroupresourcesinfo -cluster MYCLUSTER -resgroup "Group 0" -ws_user
admin -ws_password admin
```

**dpmmscs getnetworkinterfacesinfo 命令—获取网络接口信息**

getnetworkinterfaces info 命令用于返回群集网络接口信息。

此命令具有以下格式：

```
dpmmscs getnetworkinterfacesinfo
[-sc <scUrl>]
-cluster <cluster>
-net <network name>
[-locale iso639value]
[-ws_user username]
[-ws_password password]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-net**

指定网络名称。

**-cluster**

指定群集名称。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：获取群集网络接口信息。**

```
dpmmscs.exe getnetworkinterfacesinfo -cluster MYCLUSTER -net "Local Area  
Connection 2" -ws_user admin -ws_password admin
```

## dpmmcs getnetworkinfo 命令—获取网络信息

getnetworkinfo 命令用于返回群集网络信息。

此命令具有以下格式：

```
dpmmcs getnetworkinfo  
[-sc <scUrl>]  
-cluster <cluster>  
[-locale iso639value]  
[-ws_user username]  
[-ws_password password]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

示例：获取群集网络信息。

```
dpmmcs.exe getnetworksinfo -cluster MYCLUSTER -ws_user admin -ws_password admin
```

## dpmmcs getnodesinfo 命令—获取群集节点信息

getnodesinfo 命令用于返回群集节点信息。

此命令具有以下格式：

```
dpmmcs getnodesinfo
[-sc <scUrl>]
-cluster <cluster>
[-locale iso639value]
[-ws_user username]
[-ws_password password]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

示例：获取群集节点信息。

```
dpmmcs.exe getnodesinfo -cluster MYCLUSTER -ws_user admin -ws_password admin
```

## dpmmcs getresourcegroupsinfo 命令—获取群集资源组信息

getresourcegroupsinfo 命令用于返回群集资源组信息。

此命令具有以下格式：

```
dpmmcs getresourcegroupsinfo  
[-sc <scUrl>]  
-cluster <cluster>  
[-locale iso639value]  
[-ws_user username]  
[-ws_password password]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

示例：获取网络群集资源组信息。

```
dpmmcs.exe getresourcegroupsinfo -cluster MYCLUSTER -ws_user admin -ws_password  
admin
```



## dpmmcs getresourcesdependenciesinfo 命令—获取资源的依存关系信息

getresourcesdependenciesinfo 命令用于返回群集资源的依存关系信息。

此命令具有以下格式：

```
dpmmcs getresourcesdependenciesinfo  
[-sc <scUrl>]  
-cluster <cluster>  
-res <resource name>  
-restype <resource type>  
[-locale iso639value]  
[-ws_user username]  
[-ws_password password]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-res**

指定资源名称。

### **-restype**

指定资源类型。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**示例：获取群集资源的依存关系信息**

```
dpmmcs.exe getresourcedependenciesinfo -cluster MYCLUSTER -res "SQL Server Fulltext" -restype "Generic Service" -ws_user admin -ws_password admin
```

**dpmmcs getresourcetypesinfo 命令—获取群集资源类型信息**

getresourcetypes info 命令用于返回群集资源类型信息。

此命令具有以下格式：

```
dpmmcs getresourcetypesinfo  
[-sc <scUrl>]  
-cluster <cluster>  
[-locale iso639value]  
[-ws_user username]  
[-ws_password password]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**示例：获取群集资源类型信息。**

```
dpmmcs.exe getresourcetypesinfo -cluster MYCLUSTER -ws_user admin -ws_password admin
```

## dpmmcs getservicesinfo 命令—获取群集服务信息

getservicesinfo 命令用于返回群集服务信息。

此命令具有以下格式：

```
dpmmcs getservicesinfo  
[-sc <scUrl>]  
-cluster <cluster>  
[-locale iso639value]  
[-ws_user username]  
[-ws_password password]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

示例：获取群集服务信息。

```
dpmmcs.exe getservicesinfo -cluster MYCLUSTER -ws_user admin -ws_password admin
```

## dpmmscs movegroup 命令—将资源组移动到其他节点

movegroup 命令用于将资源组移动到另一个节点。

此命令具有以下格式：

```
dpmmscs movegroup  
[ -sc <scUrl> ]  
-cluster <cluster>  
-resgroup <resource group>  
-node <node name>  
[ -locale <locale> ]  
[ -ws_user <username> ]  
[ -ws_password <password> ]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-resgroup**

指定资源组。

### **-node**

指定节点名称。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

### 示例：将资源组移动到其他节点

```
dpmmcs.exe movegroup -cluster MYCLUSTER -resgroup "Group 0" -node ATS-281-W2k3CL  
-ws_user admin -ws_password admin
```

## dpmmcs pauseservice 命令—暂停群集服务

您可以通过 `pauseservice` 命令暂停节点上的群集服务。

此命令具有以下格式：

```
dpmmcs pauseservice  
[ -sc <scUrl> ]  
-cluster <cluster>  
-node <node name>  
[ -locale <locale> ]  
[ -ws_user <username> ]  
[ -ws_password <password> ]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-node**

指定节点名称。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

示例：暂停节点上的群集服务：

```
dpmmcs.exe pauseservice -cluster MYCLUSTER -node ATS-281-W2k3CL -ws_user admin  
-ws_password admin
```

## dpmmcs bringgrouponline 命令—使资源组联机

您可以通过 **bringgrouponline** 命令使资源组联机。

此命令具有以下格式：

```
dpmmcs bringgrouponline  
[ -sc <scUrl> ]  
-cluster <cluster>  
-resgroup <resource group>  
[ -locale <locale> ]  
[ -ws_user <username> ]  
[ -ws_password <password> ]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-cluster**

指定群集名称。

### **-resgroup**

指定资源组。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### **-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

### **-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**示例：使资源组联机**

```
dpmmcs.exe bringgrouponline -cluster MYCLUSTER -resgroup "Group 0" -ws_user
admin -ws_password admin
```

**dpmmcs bringresourceonline 命令—使资源联机**

您可以通过 **bringresourceonline** 命令使资源联机。

此命令具有以下格式：

```
dpmmcs bringresourceonline
```

```
[ -sc <scUrl> ]
-cluster <cluster>
-res <resource name>
[ -locale <locale> ]
[ -ws_user <username> ]
[ -ws_password <password> ]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-res**

指定资源名称。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

**示例：使资源联机**

```
dpmmcs.exe bringresourceonline -cluster MYCLUSTER -res "SQL Server Fulltext"
-ws_user admin -ws_password admin
```

**dpmmcs resumeservice 命令—恢复群集服务**

您可以通过 **resumeservice** 命令恢复节点上的群集服务。

此命令具有以下格式：

```
dpmmcs resumeservice
```

```
[ -sc <scUrl> ]
-cluster <cluster>
-node <node name>
[ -locale <locale> ]
[ -ws_user <username> ]
[ -ws_password <password> ]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-node**

指定节点名称。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。



**-ws\_password *password***

（可选）指定 CA Virtual Assurance 用户密码。

#### 示例：恢复节点上的群集服务

```
dpmmscs.exe resumeservice -cluster MYCLUSTER -node ATS-281-W2k3CL -ws_user admin
-ws_password admin
```

## dpmmscs startservice 命令—启动群集服务

您可以通过 **startservice** 命令启动节点上的群集服务。

此命令具有以下格式：

```
dpmmscs startservice
[ -sc <scUrl> ]
-cluster <cluster>
-node <node name>
[ -locale <locale> ]
[ -ws_user <username> ]
[ -ws_password <password> ]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-cluster**

指定群集名称。

**-node**

指定节点名称。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user *username***

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：启动节点上的群集服务**

```
dpmmscs.exe startservice -cluster MYCLUSTER -node ATS-281-W2k3CL -ws_user admin  
-ws_password admin
```

## dpmmscs stopservice 命令—停止群集服务

您可以通过 **stopservice** 命令停止节点上的群集服务。

此命令具有以下格式：

```
dpmmscs stopservice  
  
[ -sc <scUrl> ]  
-cluster <cluster>  
-node <node name>  
[ -locale <locale> ]  
[ -ws_user <username> ]  
[ -ws_password <password> ]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-node**

指定节点名称。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：停止节点上的群集服务**

```
dpmmcs.exe stopservice -cluster MYCLUSTER -node ATS-281-W2k3CL -ws_user admin
-ws_password admin
```

## dpmmcs takegroupoffline 命令—使资源组脱机

您可以通过 **takegroupoffline** 命令使资源组脱机。

此命令具有以下格式：

```
dpmmcs takegroupoffline
[ -sc <scUrl> ]
-cluster <cluster>
-resgroup <resource group>
[ -locale <locale> ]
[ -ws_user <username> ]
[ -ws_password <password> ]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-resgroup**

指定资源组。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：使资源组脱机**

```
dpmmscs.exe takegroupoffline -cluster MYCLUSTER -resgroup "Group 0" -ws_user  
admin -ws_password admin
```

## **dpmmscs takeresourceoffline 命令—使资源脱机**

您可以通过 **Takeresourceoffline** 命令使资源脱机。

此命令具有以下格式：

```
dpmmscs takeresourceoffline  
[ -sc <scUrl> ]  
-cluster <cluster>  
-res <resource name>  
[ -locale <locale> ]  
[ -ws_user <username> ]  
[ -ws_password <password> ]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-cluster**

指定群集名称。

**-res**

指定资源名称。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**-ws\_user username**

（可选）指定 CA Virtual Assurance 用户名。

**-ws\_password password**

（可选）指定 CA Virtual Assurance 用户密码。

**示例：使资源脱机**

```
dpmmscs.exe takeresourceoffline -cluster MYCLUSTER -res "SQL Server Fulltext"
-ws_user admin -ws_password admin
```

## CA Solaris Zones CLI 命令

可以使用 CLI 编写 CA Solaris Zones 命令的脚本并使其自动化，然后根据命令的结果运行操作。也可以在 AutoShell 中使用对应的命令。

### dpmzone associateproject 命令—将任务与项目关联

**associateproject** 命令用于将任务与项目关联起来。任务是多个进程的集合，这些进程表示一段时间内的一系列工作。每个任务与一个项目关联。

此命令具有以下格式：

```
dpmzone associateproject
[-sc sc_host]
-host hostname
-name zonename
-proj_name pname
-task_id tid
[-pre]
[-post]
[-locale iso639value]
```

**-sc sc\_host**

（可选）指定服务控制器的主机名。

**-host hostname**

指定 Solaris Zones 服务器。该参数区分大小写。

**-name zonename**

指定提供项目的区域。

**-proj\_name pname**

指定项目。

**-task\_id tid**

指定任务的 ID 号。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

将任务与项目关联：

```
dpmzone associateproject -host SolarisServer2 -name myzone1 -project_name  
myproject1  
-task_id 1954
```

## dpmzone clonezone 命令—克隆区域

**clonezone** 命令用于基于现有区域创建重复区域。源区域必须暂停才能启动克隆进程。克隆将在 Solaris 10 服务器上运行，且可能需要花费一些时间。

此命令具有以下格式：

```
dpmzone clonezone [-sc sc_host]  
-host hostname  
-name zonename  
-new_name nzonename  
-new_path npath  
[-pre]  
[-post]  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要克隆的区域。

**-new\_name *nzonename***

定义新的区域名称。

**-new\_path *npath***

定义新区域的路径。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 **fr\_FR**）。要使用命令提示符的区域设置，请指定 “**native**”。

**示例**

在 SolarisServer2 主机上克隆区域。

```
dpmzone clonezone -host SolarisServer2 -name myzone1 -new_name myzone2
-new_path /opt/zones/myzone2
```

**dpmzone createandinstallzone 命令--创建和安装区域**

**createandinstallzone** 命令用于使用自定义参数在 Solaris 主机上创建并安装区域。

此命令具有以下格式：

```
dpmzone createandinstallzone
[-sc sc_host]
-host hostname
-name zonename
-path zonepath
-type type
[-archive_path path]
[-autoboot]
[-if_type name]
[-ip ip]
[-pool_name pool]
[-sched_type sched]
[-phy_mem pmem]
[-swap_mem smem]
[-lock_mem lmem]
[-desc desc]
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

定义新区域的名称。

**-path *zonepath***

定义新区域的路径。

**-type *type***

指定新区域的类型。选项包括：

**native**

使用 Solaris 10 操作环境创建非全局区域来运行应用程序。

**whole-root**

创建不继承软件包的整个根区域。

**branded**

创建包含非本地操作环境的非全局区域来运行应用程序。

**默认值：** native

**-archive\_path *path***

（可选）为标记区域指定操作环境安装程序的路径。

**限制：** 该参数是标记区域所需的参数，本地区域不需要该参数。

**-autoboot**

（可选）指定在系统引导时自动引导区域。

**注意：** 如果在服务器上禁用区域服务，则不论该属性的设置为何，均不能自动引导区域。可通过以下命令启用该服务：

```
svcadm enable svc:/system/zones:default
```

**-if\_type *name***

（可选）指定区域使用的网络接口的类型，例如 `eri0`。

**-ip *ip***

（可选）定义区域的 IP 地址。

**-pool\_name *pool***

（可选）指定与区域关联的池的名称。



**-sched\_type *shed***

（可选）指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。有效值包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**-phy\_mem *pmem***

（可选）定义分配给区域的物理内存。可对该数值应用单位（K、M、G、T），例如，1M 是一个兆字节。

**-swap\_mem *smem***

（可选）定义分配给区域的交换内存。

**-lock\_mem *lmem***

（可选）定义分配给区域的锁定内存。锁定内存无法分页。

**-desc *description***

（可选）定义区域的说明。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmzone createpool 命令—创建资源池

`createpool` 命令用于在 Solaris 10 主机上创建资源池。

此命令具有以下格式：

```
dpmzone createpool  
[-sc sc_host]  
-host hostname  
-pset_name pset  
-cpu_min mincpu  
-cpu_max maxcpu  
-pool_name pool  
-sched_type sched  
[-pre]  
[-post]  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-pset\_name *pset***

定义处理器集的名称。每个处理器集（CPU 组）可以包含零个或多个 CPU。

**-cpu\_min *mincpu***

定义处理器集中 CPU 的最小数目。

**-cpu\_max *maxcpu***

定义处理器集中 CPU 的最大数目。

**-pool\_name *pool***

指定与区域关联的池的名称。

**-sched\_type *sched***

指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。选项包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

在 SolarisServer2 主机上创建资源池 POOL1:

```
dpmzone createpool -host SolarisServer2 -pset_name PSET1 -cpu_min 1 -cpu_max 20
-pool_name POOL1 -sched_type FSS
```

**dpmzone createproject 命令—创建项目**

createproject 命令用于在区域中创建项目。

此命令具有以下格式:

```
dpmzone createproject
[-sc sc_host]
-host hostname
-name zonename
-user_id uid
[-project_id pid]
-proj_name pname
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定区域。

**-user\_id *uid***

指定用户。

示例: root 用户

**-project\_id *pid***

(可选)定义项目 ID。您可以指定 ID,也可以让系统自动生成一个 ID。

**-proj\_name *pname***

定义新项目的名称。

**-pre**

(可选)指定是否在执行操作之前生成事件。

**-post**

(可选)指定是否在执行操作之后生成事件。

**-locale *iso639value***

(可选)指定 ISO 639\_3166 组合以覆盖默认的英语输出(例如:法语为 fr\_FR)。要使用命令提示符的区域设置,请指定“native”。

**示例**

在区域中创建项目:

```
dpmzone createproject -host SolarisServer2 -name myzone1 -user_id root
                        -proj_name myProject
```

## dpmzone createzone 命令—创建区域

createzone 命令用于使用自定义参数在 Solaris 主机上创建区域。

此命令具有以下格式:

```
dpmzone createzone
[-sc sc_host]
-host hostname
-name zonename
-path zonepath
-type type
[-autoboot]
[-if_type name]
[-ip ip]
[-pool_name pool]
[-sched_type sched]
[-phy_mem pmem]
[-swap_mem smem]
[-lock_mem lmem]
[-desc desc]
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

(可选) 指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

定义新区域的名称。

**-path *zonepath***

定义新区域的路径。

**-type *type***

指定新区域的类型。选项包括：

**native**

使用 Solaris 10 操作环境创建非全局区域来运行应用程序。

**whole-root**

创建不继承软件包的整个根区域。

**branded**

创建包含非本地操作环境的非全局区域来运行应用程序。

**默认值:** native

**-autoboot**

(可选) 指定在系统引导时自动引导区域。

**注意:** 如果在服务器上禁用区域服务，则不论该属性的设置为何，均不能自动引导区域。可通过以下命令启用该服务：

```
svcadm enable svc:/system/zones:default
```

**-if\_type *name***

(可选) 指定区域使用的网络接口的类型，例如 eri0。

**-ip *ip***

(可选) 定义区域的 IP 地址。

**-pool\_name *pool***

(可选) 指定与区域关联的池的名称。

**-sched\_type *shed***

（可选）指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。有效值包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**-phy\_mem *pmem***

（可选）定义分配给区域的物理内存。可对该数值应用单位（K、M、G、T），例如，1M 是一个兆字节。

**-swap\_mem *smem***

（可选）定义分配给区域的交换内存。

**-lock\_mem *lmem***

（可选）定义分配给区域的锁定内存。锁定内存无法分页。

**-desc *description***

（可选）定义区域的说明。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

创建名为 WebServer4 的区域：

```
dpmzone createzone -host MySolarisServer -name WebServer4 -path /Zones/Apache
-type native
-autoboot -if_type eri0 -ip 192.168.100.100 -pool_name WebServer -sched_type fss
-phy_mem 1024M -swap_mem 2048M
```

## dpmzone customcommand 命令—运行自定义命令

`customcommand` 命令用于在 Solaris 服务器上运行外部命令或脚本。

**注意：**有关受限外部命令的信息，请参阅本指南中的“自定义命令”附录。

此命令具有以下格式：

```
dpmzone customcommand
[-sc sc_host]
-host hostname
-cmd command
[-locale iso639value]
```

### **-sc *sc\_host***

（可选）指定服务控制器的主机名。

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-cmd *command***

指定要运行的程序或脚本的名称。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例

在 SolarisServer2 主机上运行命令：

```
dpmzone customcommand -host SolarisServer2
-cmd /usr/local/private/scripts/test_connections.pl
```

## dpmzone deleteproject 命令—删除项目

`deleteproject` 命令用于从区域中删除项目。

此命令具有以下格式：

```
dpmzone deleteproject
[-sc sc_host]
-host hostname
-name zonename
-proj_name pname
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

(可选) 指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定与项目关联的区域。

**-proj\_name *pname***

指定要删除的项目。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例**

删除项目:

```
dpmzone deleteproject -host SolarisServer2 -name myzone1 -project_name myproject1
```

## **dpmzone deletezone 命令—删除区域**

**deletezone** 命令用于删除 Solaris 服务器上的区域。

此命令具有以下格式:

```
dpmzone deletezone  
[-sc sc_host]  
-host hostname  
-name zonename  
[-pre]  
[-post]  
[-locale iso639value]
```



**-sc *sc\_host***

(可选) 指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要删除的区域。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

**示例**

删除主机 SolarisServer2 上的区域:

```
dpmzone deletezone -host SolarisServer2 -name myzone
```

## dpmzone installzone 命令 — 安装区域

installzone 命令用于在 Solaris 服务器上启动安装命令。安装过程可能要花费一些时间。使用 showprocesses 命令验证区域的状态, 以确认安装过程是否正在运行。

此命令具有以下格式:

```
dpmzone installzone
[-sc sc_host]
-host hostname
-name zonename
[-archive_path path]
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

(可选) 指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name zonename**

指定要安装的区域。

**-archive\_path path**

（可选）为标记区域指定操作环境安装程序的路径。

**限制：**该参数是标记区域所需的参数，本地区域不需要该参数。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：安装本地区域**

本示例在 SolarisServer 上安装 MyZone3。

```
dpmzone installzone  
-host SolarisServer  
-name MyZone3
```

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**示例**

在 SolarisServer 上使用 Solaris 8 安装 MyZone4。

```
dpmzone installzone -host SolarisServer -name MyZone4  
-archive_path /opt/zoneos/branded/Solaris8Installer
```

## **dpmzone movezone 命令—移动区域**

movezone 命令用于将区域从一个路径移动到新路径。

此命令具有以下格式：

```
dpmzone movezone  
[-sc sc_host]  
-host hostname  
-name zonename  
-new_path path  
[-pre]  
[-post]  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要移动到新位置的区域。

**-new\_path *path***

定义区域的新路径。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

移动主机 SolarisServer2 上的区域：

```
dpmzone movezone -host SolarisServer2 -name myzone1
-new_path /opt/zones/myzone1
```

**dpmzone rebootzone 命令—重新启动区域**

rebootzone 命令用于重新启动 Solaris 服务器上的区域。

此命令具有以下格式：

```
dpmzone rebootzone
[-sc sc_host]
-host hostname
-name zonename
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要重新启动的区域。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

重新启动主机 SolarisServer2 上的区域：

```
dpmzone rebootzone -host SolarisServer2 -name myzone
```

## **dpmzone renamezone 命令—重命名区域**

renamezone 命令用于重命名 Solaris 10 服务器上的区域。

此命令具有以下格式：

```
dpmzone renamezone  
[-sc sc_host]  
-host hostname  
-name czonename  
-new_name nzonename  
[-pre]  
[-post]  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-old\_name *czonename***

指定当前的区域名称。

**-new\_name *nzonename***

定义新的区域名称。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 fr\_FR)。要使用命令提示符的区域设置, 请指定 “native”。

### 示例

重命名主机 SolarisServer2 上的区域:

```
dpmzone renamezone -host SolarisServer2 -name myzone -new_name yourzone
```

## dpmzone setpoolscheduler 命令—设置池排定程序

setpoolscheduler 命令设置用于资源池的排定程序的类型。资源池可以使用两种类型的排定程序, 即公平份额排定程序 (FSS) 和时间份额排定程序 (TS)。

此命令具有以下格式:

```
dpmzone setpoolscheduler
[-sc sc_host]
-host hostname
-pool_name poolname
-sched_type sched
[-pre]
[-post]
[-locale iso639value]
```

**-sc *sc\_host***

(可选) 指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-pool\_name *poolname***

指定可进行分区的资源的关联组。

**-sched\_type *shed***

指定根据份额分配 CPU 时间而使用的排定程序的类型。份额是分配给项目的系统 CPU 资源的一部分。选项包括：

**ts**

指定时间份额排定程序，用于公平地将 CPU 资源分配给每个进程，使 CPU 资源不集中于某个特定进程。ts 是 Solaris 操作环境的默认排定程序。

**fss**

指定公平份额排定程序，用于根据份额分配 CPU 时间。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将池排定程序设置为 FSS**

本示例使用公平份额排定程序设置资源池的排定。

```
dpmzone setpoolscheduler -host SolarisServer -pool_name test_pool  
-sched_type FSS
```

**示例：将池排定程序设置为 TS**

本示例使用时间份额排定程序设置资源池的排定。

```
dpmzone setpoolscheduler -host SolarisServer -pool_name test_pool  
-sched_type TS
```

## dpmzone setprocessorset 命令—设置处理器集的最大 CPU 数目

setprocessorset 命令用于设置处理器集的最大 CPU 数目。

此命令具有以下格式：

```
dpmzone setprocessorset
[-sc sc_host]
-host hostname
-processor_set_name psetname
-cpu_max maxcpu
[-pre]
[-post]
[-locale iso639value]
```

### **-sc *sc\_host***

（可选）指定服务控制器的主机名。

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-processor\_set\_name *psetname***

指定处理器集。每个处理器集（CPU 组）可以包含零个或多个处理器。

### **-cpu\_max *maxcpu***

定义处理器集中 CPU 的最大数目。

### **-pre**

（可选）指定是否在执行操作之前生成事件。

### **-post**

（可选）指定是否在执行操作之后生成事件。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

## 示例

设置处理器集的最大 CPU 数目：

```
dpmzone setprocessorset -host SolarisServer -processor_set_name test_pset
-cpu_max 64
```

## dpmzone showinterfaces 命令—显示网络接口

showinterfaces 命令用于列出 Solaris 10 主机上的所有网络接口。

此命令具有以下格式：

```
dpmzone showinterfaces  
[-sc sc_host]  
-host hostname  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### 示例

列出主机 SolarisServer2 上的所有网络接口：

```
dpmzone showinterfaces -host SolarisServer2
```

## dpmzone showpools 命令—显示资源池

showpools 命令用于列出 Solaris 主机上的所有资源池。

此命令具有以下格式：

```
dpmzone showpools  
[-sc sc_host]  
-host hostname  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。



### 示例

列出主机 SolarisServer2 上的所有资源池：

```
dpmzone showpools -host SolarisServer2
```

## dpmzone showprojects 命令—列出项目的属性

showprojects 命令用于列出项目的属性。

此命令具有以下格式：

```
dpmzone showprojects  
[-sc sc_host]  
-host hostname  
-name zonename  
-proj_name pname  
[-locale iso639value]
```

### **-sc *sc\_host***

（可选）指定服务控制器的主机名。

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定区域。

### **-proj\_name *pname***

指定与区域关联的项目。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### 示例

列出项目的属性：

```
dpmzone showprojects -host SolarisServer2 -name myzone1 -project_name myproject1
```

## dpmzone showresourceset 命令—显示资源集

`showresourceset` 命令用于列出资源池中的资源。

此命令具有以下格式：

```
dpmzone showresourceset  
[-sc sc_host]  
-host hostname  
-pool_name poolname  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-pool\_name *poolname***

指定可进行分区的资源的关联组。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

### 示例

列出 `MyZonesResourcePool` 中的资源集：

```
dpmzone showresourceset -host SolarisServer2 -pool_name MyZonesResourcePool
```

## dpmzone showtasks 命令—列出项目中的任务

`showtasks` 命令用于列出项目中的任务。

此命令具有以下格式：

```
dpmzone showtasks  
[-sc sc_host]  
-host hostname  
-name zonename  
-proj_name pname  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定区域。

**-proj\_name *pname***

指定与区域关联的项目。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

列出项目中的任务：

```
dpmzone showtasks -host SolarisServer2 -name myzone1 -project_name myproject1
```

**dpmzone showzoneinterfaces 命令—显示区域的网络接口**

showzoneinterfaces 命令用于列出区域的所有网络接口。

此命令具有以下格式：

```
dpmzone showzoneinterfaces
[-sc sc_host]
-host hostname
-name zonename
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定区域。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### 示例

列出 SolarisServer2 上区域 MyZone3 的所有网络接口：

```
dpmzone showzoneinterfaces -host SolarisServer2 -name MyZone3
```

## dpmzone showzoneprocess 命令—显示区域中的进程

showzoneprocess 命令用于列出区域中的所有进程。

此命令具有以下格式：

```
dpmzone showzoneprocess  
[-sc sc_host]  
-host hostname  
-name zonename  
[-locale iso639value]
```

### **-sc *sc\_host***

（可选）指定服务控制器的主机名。

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定区域。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例

列出 SolarisServer2 上区域 MyZone3 的进程：

```
dpmzone showzoneprocess -host SolarisServer2 -name MyZone3
```

## dpmzone showzoneprojects 命令—显示区域中的项目

showzoneprojects 命令用于列出区域中的所有项目。

此命令具有以下格式：

```
dpmzone showzoneprojects  
[-sc sc_host]  
-host hostname  
-name zonename  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定区域。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

列出 SolarisServer2 上区域 MyZone3 的所有项目：

```
dpmzone showzoneprojects -host SolarisServer2 -name MyZone3
```

**dpmzone showzones 命令—显示区域**

showzones 命令用于列出 Solaris 主机上的所有区域。

此命令具有以下格式：

```
dpmzone showzones
[-sc sc_host]
-host hostname
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例**

列出主机 SolarisServer2 上的所有区域：

```
dpmzone showzones -host SolarisServer2
```

## dpmzone startzone 命令—启动区域

startzone 命令用于启动 Solaris 10 服务器上的区域。

此命令具有以下格式：

```
dpmzone startzone  
[-sc sc_host]  
-host hostname  
-name zonename  
[-pre]  
[-post]  
[-locale iso639value]
```

### **-sc *sc\_host***

（可选）指定服务控制器的主机名。

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要启动的区域。

### **-pre**

（可选）指定是否在执行操作之前生成事件。

### **-post**

（可选）指定是否在执行操作之后生成事件。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## 示例

启动主机 SolarisServer2 上的区域：

```
dpmzone startzone -host SolarisServer2 -name myzone
```

## dpmzone stopzone 命令—停止区域

stopzone 命令用于停止 Solaris 10 服务器上的区域。

此命令具有以下格式：

```
dpmzone stopzone  
[-sc sc_host]  
-host hostname  
-name zonename  
[-pre]  
[-post]  
[-locale iso639value]
```

### **-sc *sc\_host***

（可选）指定服务控制器的主机名。

### **-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

### **-name *zonename***

指定要停止的区域。

### **-pre**

（可选）指定是否在执行操作之前生成事件。

### **-post**

（可选）指定是否在执行操作之后生成事件。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## 示例

停止主机 SolarisServer2 上的区域：

```
dpmzone stopzone -host SolarisServer2 -name myzone
```

## dpmzone uninstallzone 命令—卸载区域

uninstallzone 命令用于卸载 Solaris 10 服务器上的区域。

此命令具有以下格式：

```
dpmzone uninstallzone  
[-sc sc_host]  
-host hostname  
-name zonename  
[-pre]  
[-post]  
[-locale iso639value]
```

**-sc *sc\_host***

（可选）指定服务控制器的主机名。

**-host *hostname***

指定 Solaris Zones 服务器。该参数区分大小写。

**-name *zonename***

指定要卸载的区域。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### 示例

卸载主机 SolarisServer2 上的区域：

```
dpmzone uninstallzone -host SolarisServer2 -name myzone
```

## CA Cisco UCS 命令

您可以使用 dpmucs CLI 命令来检索 Cisco Unified Computing System (UCS) 设备信息、统计信息、服务配置文件、池，以及执行刀片服务器和机箱操作。



## 详细信息

[dpmucs associateserviceprofile](#)—将服务配置文件与刀片服务器相关联 (p. 593)  
[dpmucs cloneserviceprofile](#)—克隆服务配置文件 (p. 595)  
[dpmucs createmacpool](#) 命令—创建 MAC 地址池 (p. 596)  
[dpmucs createorg](#) 命令—创建组织 (p. 597)  
[dpmucs createuuidpool](#) 命令—创建 UUID 池 (p. 603)  
[dpmucs createserverpool](#) 命令—创建服务器池 (p. 599)  
[dpmucs createserviceprofile](#) 命令—创建服务配置文件 (p. 600)  
[dpmucs createtemplatefromserviceprofile](#) 命令—从服务配置文件创建模板 (p. 602)  
[dpmucs createwwpool](#) 命令—创建 WWNN 池 (p. 604)  
[dpmucs deletemacpool](#) 命令—删除 MAC 地址池 (p. 606)  
[dpmucs deleteorg](#) 命令—删除组织 (p. 607)  
[dpmucs deleteserverpool](#) 命令—删除服务器池 (p. 608)  
[dpmucs deleteserviceprofile](#)—删除服务配置文件 (p. 609)  
[dpmucs deleteuuidpool](#) 命令—删除 UUID 池 (p. 610)  
[dpmucs deletewwpool](#) 命令—删除 WWNN 池 (p. 612)  
[dpmucs disassociateserviceprofile](#) 命令—解除关联服务配置文件 (p. 615)  
[dpmucs failoverserviceprofile](#) 命令—将服务配置文件与用于故障转移的刀片服务器相关联 (p. 616)  
[dpmucs getblades](#) 命令—获取刀片服务器信息 (p. 617)  
[dpmucs getchassis](#) 命令—获取机箱信息 (p. 619)  
[dpmucs getserviceprofilestatus](#) 命令—获取服务配置文件状态 (p. 621)  
[dpmucs getpolicy](#) 命令—从 UCS 管理器获取策略 (p. 620)  
[dpmucs getswitches](#) 命令—获取交换机信息 (p. 622)  
[dpmucs gettopsystem](#) 命令—获取系统信息 (p. 623)  
[dpmucs listserviceprofiles](#) 命令—列出服务配置文件 (p. 624)  
[dpmucs powercycle](#) 命令—重新启动刀片服务器 (p. 625)  
[dpmucs renamepool](#) 命令—重命名池 (p. 627)  
[dpmucs serviceprofileupdate](#) 命令—执行服务配置文件操作 (p. 628)

## dpmucs associateserviceprofile—将服务配置文件与刀片服务器相关联

使用该命令可将服务配置文件与特定的刀片服务器相关联。该命令可在关联之前显示刀片服务器的状态。要验证关联，请使用“`getserviceprofilestatus`”命令。

此命令具有以下格式：

```

dpmucs associateserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_dn serviceprofiledn
-blade_dn bladename
[-ws_user username]
[-ws_password password]
[-locale iso639value]
  
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

**-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式: `org-root/serviceprofile`。

**-blade\_dn *bladenam***

刀片服务器的完全限定且唯一的名称。如果未提供刀片服务器，则将选择一个非关联且可用的刀片服务器。有效格式: `systemid/chassisid/bladeid`。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检査的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例: 将配置文件与刀片服务器相关联**

该示例将配置文件与指定的刀片服务器相关联。

```
dpmucs associateserviceprofile -ucs_manager EngManager200
-service_profile_dn org-root/ls-dev-boot-from-san-esx-2
-blade_dn sys/chassis-1/blade-1
```

## dpmucs cloneserviceprofile—克隆服务配置文件

使用该命令可克隆服务配置文件。

此命令具有以下格式：

```
dpmucs cloneserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_dn serviceprofiledn
-cloneName clonename
[-org_path org_path]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式：*org-root/serviceprofile*。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 *caaipsecurity* 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 *fr\_FR*）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：克隆服务配置文件

该示例将克隆服务配置文件。

```
dpmucs cloneserviceprofile -ucs_manager EngManager200
-service_profile_dn org-root/ls-dev-boot-from-san-esx-2
-cloneName org-root/ls-dev-boot-from-san-esx-2
```

## dpmucs createmacpool 命令—创建 MAC 地址池

使用该命令可创建一个或多个槽范围的 MAC 地址池。可在任何组织级别创建池。

此命令具有以下格式：

```
dpmucs createmacpool
[-sc sc_url]
-ucs_manager ucsmanagername
-pool_name poolname
-description pooldescription
-org_path organizationpath
-ranges fromrange|torange fromrange|torange...
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-pool\_name *poolname***

指定池名称。有效条目：1-16 个字母数字字符。

### **-description *pooldescription***

定义正在创建的池。

**-org\_path organizationpath**

定义组织的完全限定路径名。有效格式：

*Org-root/org-suborg1/org-suborg2/org-suborg3...*

**-ranges fromrange | torange fromrange | torange...**

定义池的插槽范围。每个池类型的有效条目：格式为

XX:XX:XX:XX:XX:XX 的 MAC 地址，格式为 XXXX-XXXXXXXXXXXX 的

UUID，格式为 XX:XX:XX:XX:XX:XX:XX:XX 的 WWNN/WWPN。使用 | 分

隔整个范围条目，使用单个空格分隔插槽范围。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：创建 MAC 地址池**

该示例将创建带有多个插槽的 MAC 地址池。

```
dpmucs createmacpool
-ucs_manager ucsmanager
-pool_name TEST_MAC_POOL
-description TEST_POOL -org_path ""
-ranges 00:25:B5:00:00:C0|00:25:B5:00:00:C3 00:25:B5:00:00:C4|00:25:B5:00:00:C9
00:25:B5:00:00:CA|00:25:B5:00:00:CF
-ws_user ba
-ws_password ba
```

**dpmucs createorg 命令—创建组织**

使用该命令可创建组织。

此命令具有以下格式：

```
dpmucs createorg [-sc sc_url]
-ucs_manager ucsmanagername
-parentOrg_dn parentorganization
-org_name organizationname
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

**-parentOrg\_dn *parentorganization***

定义当前组件的父组织的完全限定名称。有效格式:  
*parentorganization/org*。

**-org\_name *organizationname***

定义组织的完全限定名称。有效格式: *org-root/org-name*

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例: 创建组织**

该示例将创建父级和下级组织。

```
dpmucs createorg
-ucs_manager EngManager200
-parentOrg_dn org-root
-org_name TEST_ORG
```

## dpmucs createserverpool 命令—创建服务器池

使用该命令可创建服务器池。

此命令具有以下格式：

```
dpmucs createwpool  
[-sc sc_url]  
-ucs_manager ucsmanagername  
-pool_name poolname  
-description pooldescription  
-org_path organizationpath  
-chassisbladepairs pairs  
[-ws_user username]  
[-ws_password password]  
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-pool\_name *poolname***

指定池名称。有效条目：1-16 个字母数字字符。

### **-description *pooldescription***

定义正在创建的池。

### **-org\_path *organizationpath***

定义组织的完全限定路径名。有效格式：

*Org-root/org-suborg1/org-suborg2/org-suborg3...*

### **-chassisbladepairs *pairs***

定义服务器池的机箱-刀片服务器对。有效条目：机箱名称，后跟刀片服务器名称，以竖线（|）分隔；使用单个空格分隔名称对。示例：  
chassis1|blade1 chassis2|blade2。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

#### 示例：创建服务器池

该示例将创建带有多个机箱刀片服务器对的服务器池。

```
createserverpool
-ucs_manager ucsmanager
-pool_name TEST_SERVER_POOL
-description TEST POOL
-org_path ""
-chassisbladepairs Eng1|Blade2 Test1|Blade1 QA1|Blade3
```

## dpmucs createserviceprofile 命令—创建服务配置文件

使用该命令可基于默认硬件创建一个或多个简单的服务配置文件，还可选择创建默认 vNIC 或 vHBA。还可以基于模板创建服务配置文件。

此命令具有以下格式：

```
dpmucs createserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_name serviceprofilename
[-org_path org_path -description description]
-policyType profiletype
[-templatename templatename]
-number number
-createvnic createvnic
-createvhba createvhba]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```



**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

**-service\_profile\_name *serviceprofilename***

指定新服务配置文件的名称。

**-templatename *templatename***

指定模板名称。

**-number *number***

指定创建的对象数目。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：创建服务配置文件**

该示例将为指定的 UCS 管理器创建服务配置文件。

```
dpmucs.exe createserviceprofile  
-ucs_manager Engmanager  
-service_profile_name sptest_1
```

## dpmucs createtemplatefromserviceprofile 命令—从服务配置文件创建模板

使用该命令可从现有的服务配置文件创建模板。

此命令具有以下格式：

```
dpmucs createtemplatefromserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_dn serviceprofiledn
-templatename templatename
[-templatetype templatetype -org_path org_path]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 **caaipsecurity** 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：从服务配置文件创建模板

该示例可为指定的 UCS 管理器从服务配置文件创建模板。

```
dpmucs.exe createtemplatefromserviceprofile
-ucs_manager Engmanager
-service_profile_dn sptest_1
-templatename template_sp_1
```

## dpmucs createuuidpool 命令—创建 UUID 池

使用该命令可为刀片服务器创建一个或多个槽范围的全球唯一标识符 (UUID) 池。

此命令具有以下格式：

```
dpmucs createuuidpool
[-sc sc_url]
-ucs_manager ucsmanagername
-pool_name poolname
-description pooldescription
-org_path organizationpath
-ranges fromrange|torange fromrange|torange...
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager ucsmanagername**

定义用于检索信息或操作的 UCS 管理器的名称。

### **-pool\_name poolname**

指定池名称。有效条目：1-16 个字母数字字符。

### **-description pooldescription**

定义正在创建的池。

### **-org\_path organizationpath**

定义组织的完全限定路径名。有效格式：

Org-root/org-suborg1/org-suborg2/org-suborg3...

**-ranges *fromrange* | *torange fromrange* | *torange...***

定义池的插槽范围。每个池类型的有效条目：格式为 XX:XX:XX:XX:XX:XX 的 MAC 地址，格式为 XXXX-XXXXXXXXXXXX 的 UUID，格式为 XX:XX:XX:XX:XX:XX:XX:XX 的 WWNN/WWPN。使用 | 分隔整个范围条目，使用单个空格分隔插槽范围。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全验证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

### 示例：创建 UUID 池

该示例将创建具有多个插槽的 UUID 池。

```
dpmucs createuuidpool
-ucs_manager EngManager200
-pool_name TEST_POOL_UUID
-description TEST_POOL
-org_path ""
-ranges 5500-000000000030|5500-000000000032 5500-000000000035|5500-000000000040
5500-000000000042|5500-000000000045
-ws_user cba
-ws_password cba
```

## dpmucs createwwpool 命令—创建 WWNN 池

使用该命令可为刀片服务器创建全球节点名称 (WWNN) 端口或节点池。

此命令具有以下格式：

```
dpmucs createwwpool
[-sc sc_url]
-ucs_manager ucsmanagername
-pool_name poolname
-description pooldescription
-org_path organizationpath
-ranges fromrange|torange fromrange|torange...-wwpool_type type
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

**-pool\_name *poolname***

指定池名称。有效条目: 1-16 个字母数字字符。

**-description *pooldescription***

定义正在创建的池。

**-org\_path *organizationpath***

定义组织的完全限定路径名。有效格式:

`Org-root/org-suborg1/org-suborg2/org-suborg3...`

**-ranges *fromrange* | *torange fromrange* | *torange...***

定义池的插槽范围。每个池类型的有效条目: 格式为

`XX:XX:XX:XX:XX:XX` 的 MAC 地址, 格式为 `XXXX-XXXXXXXXXXXX` 的

UUID, 格式为 `XX:XX:XX:XX:XX:XX:XX:XX` 的 WWNN/WWPN。使用 | 分

隔整个范围条目, 使用单个空格分隔插槽范围。

**-wwpool\_type *type***

将池类型定义为节点或端口。有效条目: `node` 或 `pool`。默认值 = `node`。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如: 法语为 `fr_FR`）。要使用命令提示符的区域设置, 请指定 “`native`”。

### 示例：创建 WWNN 池

该示例将创建 WWNN 节点池。

```
dpmucs createwwpool
-ucs_manager EmgManager200
-pool_name TEST_POOL_WW
-description TEST_POOL
-org_path "" -ranges 20:00:00:25:B6:01:00:F0|20:00:00:25:B6:01:00:F3
20:00:00:25:B6:01:00:F5|20:00:00:25:B6:01:00:F7 -wwpool_type node -ws_user cba
-ws_password cba
```

## dpmucs deletemacpool 命令—删除 MAC 地址池

使用该命令可删除 MAC 地址池。

此命令具有以下格式：

```
dpmucs deletemacpool
[-sc sc_url]
-ucs_manager ucsmanagename
-pool_name poolname
-org_path organizationpath
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-pool\_name *poolname***

指定池名称。有效条目：1-16 个字母数字字符。

### **-org\_path *organizationpath***

定义组织的完全限定路径名。有效格式：

*Org-root/org-suborg1/org-suborg2/org-suborg3...*

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

#### 示例：删除 MAC 地址池

该示例将删除 MAC 地址池。

```
dpmucs deletemacpool
-ucs_manager ucsmanager
-pool_name TEST_POOL
-org_path ""
-ws_user cba -ws_password cba
```

## dpmucs deleteorg 命令—删除组织

使用该命令可删除组织。

此命令具有以下格式：

```
dpmucs createorg
[-sc sc_url]
-ucs_manager ucsmanagername
-org_name organizationname
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例：`80`

**-ucs\_manager ucsmanagername**

定义用于检索信息或操作的 UCS 管理器的名称。

**-org\_name organizationname**

定义组织的完全限定名称。有效格式: *org-root/org-name*

**-ws\_user username -ws\_password password**

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 *caaipsecurity* 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 *fr\_FR*)。要使用命令提示符的区域设置, 请指定 “*native*”。

**示例: 删除组织**

该示例将删除组织。

```
dpmucs deleteorg
-ucs_manager EngManager200
-org_dn org-root/org-TEST_ORG
```

## dpmucs deleteserverpool 命令—删除服务器池

使用该命令可删除服务器池。

此命令具有以下格式:

```
dpmucs deleteserverpool
[-sc sc_url]
-ucs_manager ucsmanagername
-org_path orgpath
-pool_name poolname
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc sc\_url**

(可选) 为服务控制器主机定义 URL。

**示例:** *https://hostname:port/...*

**hostname**

定义安装了服务控制器的完全限定主机。

**示例:** *localhost.ca.com*

**port**

定义服务控制器服务器的侦听端口。

**示例:** *80*



**-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

**-org\_path *organizationpath***

定义组织的完全限定路径名。有效格式：

*Org-root/org-suborg1/org-suborg2/org-suborg3...*

**-pool\_name *poolname***

指定池名称。有效条目：1-16 个字母数字字符。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 *caaipsecurity* 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 *fr\_FR*）。要使用命令提示符的区域设置，请指定 “*native*”。

**示例：删除服务器池**

该示例将删除服务器池。

```
dpmucs.exe deleteserverpool
-ucs_manager ucsmanager
-org_path1/Eng/test
-pool_name TEST_POOL
```

**dpmucs deleteserviceprofile — 删除服务配置文件**

使用该命令可删除服务配置文件。

此命令具有以下格式：

```
dpmucs deleteserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_dn serviceprofiledn
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

**-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式: `org-root/serviceprofile`。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

## dpmucs deleteuuidpool 命令—删除 UUID 池

使用该命令可删除 UUID 池。

此命令具有以下格式：

```
dpmucs deleteuuidpool  
[-sc sc_url]  
-ucs_manager ucsmanagername  
-pool_name poolname  
-org_path organizationpath  
[-ws_user username]  
[-ws_password password]  
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

**-pool\_name *poolname***

指定池名称。有效条目: 1-16 个字母数字字符。

**-org\_path *organizationpath***

定义组织的完全限定路径名。有效格式:

`Org-root/org-suborg1/org-suborg2/org-suborg3...`

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例: 删除 UUID 池**

该示例将删除 UUID 池。

```
dpmucs deleteuuidpool
-ucs_manager EngManager200
-pool_name "TEST_POOL"
-org_path ""
-ws_user cba
-ws_password cba
```

## dpmucs deletewwpool 命令—删除 WWNN 池

使用该命令可删除 WWNN 池。

此命令具有以下格式：

```
dpmucs deletewwpool  
[-sc sc_url]  
-ucs_manager ucsmanagername  
-pool_name poolname  
-org_path organizationpath  
[-ws_user username]  
[-ws_password password]  
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-pool\_name *poolname***

指定池名称。有效条目：1-16 个字母数字字符。

### **-org\_path *organizationpath***

定义组织的完全限定路径名。有效格式：

*Org-root/org-suborg1/org-suborg2/org-suborg3...*

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：删除 WWNN 池**

该示例将删除 WWNN 池。

```
dpmucs deletewwpool
-ucs_manager EngManager200
-pool_name "TEST_POOL"
-org_path ""
-ws_user cba
-ws_password cba
```

**dpmucs deployimage 命令—部署 Cisco UCS 映像**

使用该命令可将映像部署到使用 RSI 或 ITCM 的 Cisco UCS 刀片服务器。

此命令具有以下格式：

```
dpmucs deployimage
[-sc sc_url] -deploytype RSI|ITCM
仅 ITCM: -passwordEncrypted yes|no -osPassword password
-inputparam ucsManagerHost|ucsManager
-inputparam bladeDn|bladeDN
-inputparam macAddress|macaddress
-inputparam osType|os_type
-inputparam profileDn|profileDN
-inputparam profileTemplateDn|profileTemplateDn
-inputparam selectFirstBlade|{false|true}
-inputparam serviceProfileName|serviceprofilename
-inputparam ... deploytype_options
[-ws_user username][-ws_password password]
[-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-inputparam argument|value**

指定映像部署的输入参数。RSI 参数值集必须用引号括起。

**-bootServerName | bootserver**

指定用于 ITCM 部署的引导服务器。

**-ws\_user username -ws\_password password**

(可选) 指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：使用 RSI 部署映像**

该示例将使用 RSI 将映像部署到 Cisco UCS 刀片服务器。

```
dpmucs.exe deployimage -deploytype RSI
-inputparam ucsManagerHost|lodcisco1-mc1.ca.com
-inputparam ipAddress|blade-2
-inputparam macAddress|00:26:51:09:5D:E4
-inputparam osType|Microsoft_Windows-2008_R2_*-x86_64
-inputparam bladeDn|sys/chassis-1/blade-2
-inputparam serviceProfileName|blade2_netboot1
-inputparam profileDn|org-root/ls-blade2_netboot1
-inputparam selectFirstBlade|false
-inputparam profileTemplateName|org-root/ls-template1
-inputparam RSIServer|ucs03-racemi
-inputparam RSIIImage|ATS_WIN2K8_IMG
-inputparam EnableScale|no
```

**示例：使用 ITCM 部署映像**

该示例将使用 ITCM 将映像部署到 Cisco UCS 刀片服务器。

```
dpmucs.exe deployimage -deploytype ITCM
-passwordEncrypted no -osPassword secret123
-inputparam ucsManagerHost|lodcisco1-mc1.ca.com
-inputparam computerName|hostname1
-inputparam ipAddress|blade-2
-inputparam macAddress|00:26:51:09:5D:E4
-inputparam osType|Any
-inputparam osImage|Windows2003_Image -bootServerName|ITCM_BootServer1
-inputparam osUser|Administrator
-inputparam osPassword|password
-inputparam bladeDn|sys/chassis-1/blade-2
-inputparam serviceProfileName|blade2_netboot1
-inputparam profileDn|org-root/ls-blade2_netboot1
-inputparam selectFirstBlade|false
-inputparam templateName|Default_Software_List
-inputparam profileTemplateName|org-root/ls-template1
```

## dpmucs diassociateserviceprofile 命令—解除关联服务配置文件

使用该命令可解除服务配置文件与刀片服务器的关联。

此命令具有以下格式：

```
dpmucs diassociateserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_dn serviceprofiledn
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式：*org-root/serviceprofile*。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 *caaipsecurity* 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 *fr\_FR*）。要使用命令提示符的区域设置，请指定“*native*”。

**示例：解除关联服务配置文件**

该示例将解除服务配置文件的关联。

```
dpmucs diassociateserviceprofile
-ucs_manager EngManager200
-service_profile_dn org-root/ls-dev-boot-from-san-esx-10
```

## **dpmucs failoverserviceprofile 命令—将服务配置文件与用于故障转移的刀片服务器相关联**

使用该命令可将现有的服务配置文件关联转移到其它配置文件以便进行刀片服务器故障转移。

此命令具有以下格式：

```
dpmucs failoverserviceprofile
[-sc sc_url]
-ucs_manager ucsmanagername
-service_profile_dn serviceprofiledn
-blade_dn bladename
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

**-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式：**org-root/serviceprofile**。指定的配置文件必须是相关联的服务配置文件。



**-blade\_dn bladename**

刀片服务器的完全限定且唯一的名称。如果未提供刀片服务器，则选择一个非关联且可用的刀片服务器。有效格式：

*systemid/chassisid/bladeid*。如果未提供刀片服务器（""），将会随机选择可用的刀片服务器。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：创建故障转移服务配置文件**

该示例将随机可用的刀片服务器与刀片服务器相关联以便进行故障转移。

```
dpmucs.exe failoverserviceprofile
-ucs_manager ucsmanager
-service_profile_dn org-root/ls-dev-boot-from-san-esx-2 -blade_dn "
```

**dpmucs getblades 命令—获取刀片服务器信息**

使用该命令可获取 UCS 管理器的所有或特定刀片服务器的信息。

此命令具有以下格式：

```
dpmucs getblades
[-sc sc_url]
-ucs_manager ucsmanagername
[-dn distinguishedname]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

**-dn *distinguishedname***

定义 Cisco UCS 机箱、刀片服务器或交换机的完全限定且唯一的名称。  
有效条目: 表示所有机箱、刀片服务器或交换机的设备名称或空字符串 ("")。如果未指定, 将列出设备详细信息。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全认证的凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 获取刀片服务器信息**

该示例获取 UCS 管理器 EngManager200 中刀片服务器 -dn “sys/chassis-1/blade-1” 的信息。

```
dpmucs getblades
-ucs_manager EngManager200
-dn sys/chassis-1/blade-1
-ws_user cba
-ws_password cba
```

## dpmucs getchassis 命令—获取机箱信息

使用该命令获取 UCS 管理器中所有机箱或特定机箱的信息。

此命令具有以下格式：

```
dpmucs getchassis  
[-sc sc_url]  
-ucs_manager ucsmanagename  
[-dn distinguishedname]  
[-ws_user username]  
[-ws_password password]  
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-dn *distinguishedname***

定义 Cisco UCS 机箱、刀片服务器或交换机的完全限定且唯一的名称。  
有效条目：表示所有机箱、刀片服务器或交换机的设备名称或空字符串 ("")。如果未指定，将列出设备详细信息。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：获取机箱信息

该示例获取 sys/chassis-1 机箱的信息。

```
dpmucs getchassis
-ucs_manager EngManager200
-dn sys/chassis-1
-ws_user cba
-ws_password cba
```

## dpmucs getpolicy 命令—从 UCS 管理器获取策略

使用该命令从 UCS 管理器获取策略。

此命令具有以下格式：

```
dpmucs getpolicy
[-sc sc_url]
-ucs_manager ucsmanagername
[-policyType policytype|All]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：获取策略

该示例获取指定 UCS 管理器的所有策略。

```
dpmucs.exe getpolicy
-ucs_manager Engmanager
-policytype All
```

## dpmucs getserviceprofilestatus 命令—获取服务配置文件状态

使用该命令获取服务配置文件状态。

此命令具有以下格式：

```
dpmucs getserviceprofilestatus
[-sc sc_url]
-ucs_manager ucsmanagename
-service_profile_dn serviceprofiledn
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式：*org-root/serviceprofile*。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 *caaipsecurity* 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：获取服务配置文件状态**

该示例获取服务配置文件状态。

```
dpmucs getserviceprofilestatus
-ucs_manager ucsmanager
-service_profile_dn org-root/ls-dev-boot-from-san-esx-11
```

## dpmucs getswitches 命令—获取交换机信息

使用该命令获取 UCS 管理器中所有交换机或特定交换机的信息。

此命令具有以下格式：

```
dpmucs getswitches
[-sc sc_url]
-ucs_manager ucsmanagername
[-dn distinguishedname]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-ucs\_manager *ucsmanagername***

定义用于检索信息或操作的 UCS 管理器的名称。

**-dn *distinguishedname***

定义 Cisco UCS 机箱、刀片服务器或交换机的完全限定且唯一的名称。  
有效条目：表示所有机箱、刀片服务器或交换机的设备名称或空字符串 ("")。如果未指定，将列出设备详细信息。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：获取交换机信息**

该示例获取 UCS 管理器 EngManager200 上交换机 `syst/switch-A` 的信息。

```
dpmucs getswitches
-ucs_manager EngManager200
-dn syst/switch-A
-ws_user cba
-ws_password cba
```

**dpmucs gettopsystem 命令—获取系统信息**

使用该命令获取 UCS 管理器中最高系统的信息。

此命令具有以下格式：

```
dpmucs gettopsystem
[-sc sc_url]
-ucs_manager ucsmanagename
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

**示例：**`80`

**-ucs\_manager ucsmanagename**

定义用于检索信息或操作的 UCS 管理器的名称。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：获取系统信息**

该示例获取 UCS 管理器 EngManager200 中最高系统的信息。

```
dpmucs gettopsystem
- sc https://HOST/dpm/ucsws
- ucs_manager EngManager200
- ws_user cba
- ws_password cba
```

## **dpmucs listserviceprofiles 命令—列出服务配置文件**

使用该命令列出指定 UCS 管理器的服务配置文件。

此命令具有以下格式：

```
dpmucs listserviceprofiles
[-sc sc_url]
-ucs_manager ucsmanagername
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

**示例：**`80`

**-ucs\_manager ucsmanagername**

定义用于检索信息或操作的 UCS 管理器的名称。



**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

#### 示例：列出服务配置文件

该示例列出 UCS 管理器 EngManager200 的服务配置文件。

```
dpmucs listserviceprofiles
-ucs_manager EngManager200
-ws_user cba
-ws_password cba
```

## dpmucs powercycle 命令—重新启动刀片服务器

使用该命令重新启动刀片服务器。

此命令具有以下格式：

```
dpmucs powercycle
[-sc sc_url]
-ucs_manager ucsmanagername
[-dn distinguishedname]
-operation operation
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-ucs\_manager ucsmanagername**

定义用于检索信息或操作的 UCS 管理器的名称。

**-dn distinguishedname**

定义 Cisco UCS 机箱、刀片服务器或交换机的完全限定且唯一的名称。  
有效条目：表示所有机箱、刀片服务器或交换机的设备名称或空字符串 ("")。如果未指定，将列出设备详细信息。

**-operation operations**

定义刀片服务器重新启动选项。有效条目是：

- 2 = 重新启动。立即关闭刀片服务器的电源，然后再启动刀片服务器。
- 3 = 正常关闭。允许保存数据，然后关闭刀片服务器的电源再启动刀片服务器。
- 4 = 立即重置。允许您拔出刀片服务器的电源，然后在不保存数据的情况下重新插入该电源。
- 5 = 重置等待。在拔出电源之前，允许保存数据，然后重新插入该电源。
- 6 = 软关闭。通知所有运行的应用程序关闭，以便保存数据。
- 7 = 关闭。在不保存数据的情况下关闭刀片服务器。
- 8 = 启动。启动刀片服务器。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：重新启动刀片服务器**

该示例立即重新启动刀片服务器 `sys/chassis-1/blade-3`。

```
dpmucs powercycle
-ucs_manager EngManager200
-dn "sys/chassis-1/blade-3"
-operation 2
-ws_user cba
-ws_password cba
```

## dpmucs renamepool 命令—重命名池

使用该命令对池类型进行重命名。

此命令具有以下格式：

```
dpmucs renamepool
[-sc sc_url]
-ucs_manager ucsmanagename
-pool_type pooltype
-pool_name poolname
-new_pool newpoolname
-org_path organizationpath
-description description
-ranges fromrange|torange fromrange|torange...
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

### **-pool\_type *pooltype***

指定池类型。有效的池类型（区分大小写）：MacPool、UuidPool、ComputePool、WWNodeNamePool、WWPortNamePool。

### **-pool\_name *poolname***

指定池名称。有效条目：1-16 个字母数字字符。

### **-new\_pool\_name *newpoolname***

定义新的池名称。有效条目：1-16 个字符，不包括空格以及  
^!@#%&\*()+={} \ | ; " < > , / ? ` ~

**-org\_path organizationpath**

定义组织的完全限定路径名。有效格式：

*Org-root/org-suborg1/org-suborg2/org-suborg3...*

**-ranges fromrange | torange fromrange | torange...**

定义池的插槽范围。每个池类型的有效条目：格式为

XX:XX:XX:XX:XX:XX 的 MAC 地址，格式为 XXXX-XXXXXXXXXXXX 的

UUID，格式为 XX:XX:XX:XX:XX:XX:XX:XX 的 WWNN/WWPN。使用 | 分隔整个范围条目，使用单个空格分隔插槽范围。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：重命名池**

该示例重命名池。

```
dpmucs.exe renamepool
-ucs_manager ucsmanager
-pool_type UuidPool
-pool_name TEST_POOL
-new_pool_name NEW_TEST_POOL
-org_path ""
-description Windows pool
```

**dpmucs serviceprofileupdate 命令—执行服务配置文件操作**

使用该命令执行下列服务配置文件操作之一：关联、解除关联及故障转移。

此命令具有以下格式：

```
dpmucs serviceprofileupdate
[-sc sc_url]
-ucs_manager ucsmanagername
-update_operation operation
-service_profile_dn serviceprofiledn
[-blade_dn bladedn]
[-ws_user username]
[-ws_password password]
[-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ucs\_manager *ucsmanagename***

定义用于检索信息或操作的 UCS 管理器的名称。

**-update\_operation *operations***

为服务配置文件定义要执行的操作。有效条目：

**associate** = 将服务配置文件与指定的刀片服务器关联。

**disassociate** = 解除服务配置文件与指定刀片服务器的关联。

**failover** = 从一个服务配置文件故障转移到另一个服务配置文件。

**-service\_profile\_dn *serviceprofiledn***

刀片服务器服务配置文件的完全限定且唯一的名称。必须在开通 OS 之前创建刀片服务器服务配置文件。服务配置文件包含启动顺序、网络设置和存储（本地或远程）。有效格式: `org-root/serviceprofile`。

**-blade\_dn *bladename***

刀片服务器的完全限定且唯一的名称。如果未提供刀片服务器，则将选择一个非关联且可用的刀片服务器。有效格式：

`systemid/chassisid/bladeid`。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：执行服务配置文件操作

该示例为特定的刀片服务器关联一个服务配置文件。

```
dpmucs.exe serviceprofileupdate
-ucs_manager Engmanager
-update_operation associate
-service_profile_dn org-root/ls-dev-boot-from-san-esx-1 -blade_dn
sys/chassis-1/blade-1
```

## 管理命令行实用工具

### dpmutil 语法

**dpmutil** 是一个命令行实用工具，您可以使用该工具更改 CA Virtual Assurance 组件的配置。要使用 **dpmutil**，管理数据库信息在注册表中必须可用，并且必须安装 Microsoft SQL 客户端。要访问 **dpmutil**，您必须拥有具有管理员权限的用户名。**dpmutil** 命令行实用工具语法接受 *set*、*get* 或 *delete* 请求作为第一个参数，然后让您设置或获取有关组件的信息。如果您输入的参数无效或不输入参数，会出现一条错误消息，其中显示语法信息和可配置的组件列表。**dpmutil** 语法包括以下内容：

```
dpmutil {-set|-get|-delete} -category|component [-userusername]
[-passwordpassword] [-help] [-validateonly][--locale iso639value]
```

#### **-set**

定义或更新组件的配置设置。

#### **-get**

显示组件的配置设置。

#### **-delete**

删除组件的配置设置。

#### **-category/component**

定义想要从中获取或更改数据的类别或组件。

#### **-userusername**

（可选）指定 EEM 用户。如果没有该参数，**dpmutil** 会提示您输入有效的 EEM 凭据，然后您才能进行配置更新和删除或者显示现有配置信息。

**-password *password***

（可选）指定 EEM 密码。如果没有该参数，dpmutil 会提示您输入有效的 EEM 凭据，然后您才能进行配置更新和删除或者显示现有配置信息。

**注意：**用户和密码参数与它们的值之间没有空格。

**-help**

显示 dpmutil 帮助。

**-validateonly**

（可选）验证使用 **set** 和有效命令时输入的信息，如下所示：

```
dpmutil -set command -validateonly
```

该命令不创建或更新信息。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**注意：**重新启动 Apache HTTP 服务器以使更改生效。

## dpmutil -amq 命令—配置 ActiveMQ 消息代理

dpmutil **set|get|delete** amq 命令用于为 CA Virtual Assurance 配置 ActiveMQ 消息代理。一些 CA Virtual Assurance 组件使用消息代理在彼此之间传递数据。该命令用于设置消息代理 URL。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -amq [-locale iso639value]
```

**-set**

定义 ActiveMQ 消息代理的 URL。

**示例：***tcp://servername:port*

***servername***

标识安装了 ActiveMQ 消息代理的服务器的名称。

***port***

指定 ActiveMQ 消息代理侦听端口。

**默认：**61616

**-get**

显示为 ActiveMQ 消息代理配置的 URL。

**-delete**

删除为 ActiveMQ 消息代理配置的 URL。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmutil ecs 命令-配置 Windows 排定程序

dpmutil set|get ecs 命令用于配置排定程序服务器。

此命令具有以下格式：

```
dpmutil {-set|-get} -ecs [-locale iso639value]
```

**-set**

定义排定程序服务器的配置设置。

**-get**

显示排定程序服务器的配置设置。

**-ecs**

定义要配置的排定程序服务器。系统会提示您确定 ECS 是否为活动的排定程序（键入 Y 表示确定，或者按 Enter 跳过配置）。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmutil -hd 命令—配置帮助台

dpmutil set|get hd 命令用于为 CA Virtual Assurance 配置帮助台信息。您可以使用该命令在 CA Virtual Assurance 的现有安装上配置帮助台系统。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -hd [locale iso639value]
```

**-set**

定义帮助台系统的配置设置。

**-get**

显示帮助台系统的配置设置。



**-delete**

将帮助台服务器从 CA Virtual Assurance 中删除，但是它们仍保留在管理数据库和性能数据库中作为历史记录。系统将提示您输入每个参数。

**-hd**

定义连接到帮助台服务器所需的信息。系统将提示您输入帮助台服务器名称、管理员用户名和密码以及端口。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmutil -hmc 命令--配置 HMC 或 IVM 服务器

使用 dpmutil hmc 命令添加、修改、列出或删除配置的 HMC 或 IVM 服务器。您可以使用该命令在 CA Virtual Assurance 的现有安装上配置 HMC 或 IVM 服务器。需要提供 CA Virtual Assurance 管理员用户名、密码和端口才能设置或获取信息。

**注意：**重新启动 Apache HTTP 服务器以使更改生效。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -hmc [-locale iso639value]
```

**-set**

添加或更改 HMC 或 IVM 服务器的配置设置，包括服务器名称、用户名和密码。

**-get**

显示 HMC 或 IVM 服务器的配置设置，包括服务器名称、用户名。

**-delete**

删除 HMC 或 IVM 服务器，但为了保存历史记录，这些服务器仍保留在管理数据库和性能数据库中。系统将提示您输入每个参数。

**-hmc**

定义 HMC 或 IVM 服务器。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmutil -scvmmserver 命令--配置 Hyper-V 服务器

`dpmutil set|get -hvserver` 命令用于添加、修改、列出或删除配置的 Hyper-V 服务器，以进行虚拟机 (VM) 开通。您可以使用该命令在现有的 CA Virtual Assurance 安装上安装 Hyper-V 组件。需要提供 CA Virtual Assurance 管理员用户名、密码和端口才能设置或获取信息。更改该设置后无需重新启动 Apache。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -hyperv [locale iso639value]
```

### **-set**

添加或更改 Hyper-V 服务器的配置设置，包括服务器名称、用户名、端口以及协议。

### **-get**

显示 Hyper-V 服务器的配置设置，包括服务器名称、用户名、端口以及协议。

### **-delete**

删除 Hyper-V 服务器，但为了保存历史记录，这些服务器仍保留在管理数据库和性能数据库中。对于每个参数，系统将进行提示。

### **-hyperv**

定义 Hyper-V 服务器。

### **-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

## dpmutil mgmtadb 命令-配置管理数据库

`dpmutil set|get -mgmtadb` 命令用于配置管理数据库，产品套件中的几乎所有组件都使用该管理数据库作为主数据库。

此命令具有以下格式：

```
dpmutil {-set|-get} -mgmtadb [-locale iso639value]
```

### **-set**

定义管理数据库的配置设置。

**-get**

显示管理数据库的配置设置。

**-mgmtadb**

定义要配置的管理数据库。系统将提示您输入数据库服务器、版本、端口和（如果使用 SQL 授权）用于访问主产品表的凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**注意：**更改这些信息后，需要先关闭再打开相关产品服务（CAAIPApache 和 CAAIPTomcat），更改才能生效。

## dpmutil -perfadb 命令—配置性能数据库

dpmutil set|get -perfadb 命令用于为 CA Virtual Assurance 配置性能数据库。该数据库存储 CA Virtual Assurance 收集的所有性能信息。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -perfadb [-locale iso639value]
```

**-set**

定义性能数据库的配置设置。

**-get**

显示性能数据库的配置设置。

**-delete**

删除性能数据库的配置设置。

**-perfadb**

定义想要配置的组件。将提示您输入服务器名称、管理员用户名和密码、数据库类型、数据库实例和数据库端口。性能数据库凭据通常与管理数据库凭据相同。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**注意：**更改这些信息后，需要先关闭再打开相关产品服务（CAAIPApache 和 CAAIPTomcat），更改才能生效。

## dpmutil remotevcaim 命令-配置远程 VC AIM

dpmutil set|get remotevcaim 命令用于配置远程 VC AIM。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -remotevcaim [-locale iso639value]
```

### **-set**

定义远程 VC AIM 的配置设置。

### **-get**

显示远程 VC AIM 的配置设置。

### **-delete**

删除当前已配置的远程 VC AIM。

### **-remotevcaim**

定义要配置的远程 VC AIM。系统将提示您输入安装了 VC AIM 的服务器、安装了 VC 本身的服务器以及用于连接到 VC 服务器的 SNMP 写入权限字符串。

### **-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

## dpmutil -sc 命令—配置服务控制器

dpmutil set|get -sc 命令用于为 CA Virtual Assurance 配置服务控制器。服务控制器是通信量控制器，允许不同的 CA Virtual Assurance 组件相互通信。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -sc [-locale iso639value]
```

### **-set**

定义服务控制器的配置设置。

### **-get**

显示服务控制器的配置设置。

### **-delete**

删除服务控制器的配置设置。

### **-sc**

定义本地服务器的服务控制器。

## 模式

定义用于找到服务控制器的方式的配置设置。支持以下模式：

### SRV 自动查找

使用 DNS SRV 记录查找服务控制器。

### 手工覆盖

基于指定的 URL 标识已安装和正在运行的服务控制器。

示例：https://hostname:port/dpm/sc

### -locale iso639value

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

## dpmutil -scvmmserver 命令—配置 SCVMM 服务器

dpmutil set|get -scvmmserver 命令用于添加、修改、列出或删除配置的 SCVMM 服务器，以进行虚拟机 (VM) 开通。您可以使用该命令在现有的 CA Virtual Assurance 安装上安装 SCVMM 组件。需要提供 CA Virtual Assurance 管理员用户名、密码和端口才能设置或获取信息。更改该设置后无需重新启动 Apache。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -scvmmserver [-locale iso639value]
```

### -set

添加或更改 SCVMM 服务器的配置设置，包括服务器名称、用户名、端口以及协议。

### -get

显示 SCVMM 服务器的配置设置，包括服务器名称、用户名、端口以及协议。

### -delete

删除 SCVMM 服务器，但为了保存历史记录，这些服务器仍保留在管理数据库和性能数据库中。对于每个参数，系统将进行提示。

### -scvmmserver

定义 SCVMM 服务器。

### -locale iso639value

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

## dpmutil -snmp 命令—配置 SNMP

`dpmutil set|get|delete -snmp` 命令用于配置数据中心或服务器级别的 SNMP 设置以用于接收 SNMP 信息。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -snmp [-locale iso639value]
```

### **-set**

添加或更改数据中心或服务器的配置设置，包括：

- 数据中心或服务器级别
- 服务器名称（如果是服务器设置）
- SNMP 名
- 超时
- 重试次数
- 写入权限
- 版本
- 安全用户
- 授权类型
- 授权密码
- 隐私类型
- 隐私密码
- 团体字符串

系统将提示您输入每个参数。您可以配置服务器，以使用服务器和数据中心 SNMP 配置，或仅使用服务器 SNMP 配置。

**-get**

列出数据中心或服务器的 **SNMP** 的配置设置，包括：

- **SNMP** 名
- 版本
- **port**
- 超时
- 重试次数
- 写入权限
- 授权类型
- 隐私类型
- 安全用户
- 团体字符串

您只能列出服务器 **SNMP** 配置，即使服务器使用数据中心默认 **SNMP** 配置也是如此。

**-delete**

删除数据中心或服务器 **SNMP** 配置。系统会提示您删除数据中心或服务器 **SNMP** 配置。对于服务器 **SNMP** 配置，系统将提示您输入服务器名称和 **SNMP** 端口。对于数据中心 **SNMP** 配置，系统将提示您输入 **SNMP** 名称和端口。

**-snmp**

定义 **SNMP** 配置。

**-locale *iso639value***

（可选）指定 **ISO 639\_3166** 组合以覆盖默认的英语输出（例如：法语为 **fr\_FR**）。要使用命令提示符的区域设置，请指定 “**native**”。

## **dpmutil -systemcredential 命令—配置外部系统凭据**

**dpmutil set|get -systemcredential** 命令用于配置用于全局访问和特定于系统的访问的外部系统凭据。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -systemcredential [-locale iso639value]
```

**-set**

定义用于全局访问和特定于系统的访问的外部系统凭据的配置设置。

**-get**

显示用于全局访问和特定于系统的访问的外部系统凭据的配置设置。

**-delete**

根据选择要删除的凭据，删除用于全局访问和特定于系统的访问的外部系统凭据。

**-systemcredential**

定义要配置的用于全局访问和特定于系统的访问的外部系统凭据。系统将提示您输入凭据级别（由整个数据中心继承，还是仅在服务器级别继承）。根据凭据级别，系统可能会提示您输入要向其应用凭据的特定服务器名称。此外，系统将提示您输入凭据名称、凭据信息本身（用户/密码）以及凭据所属的任何组（可选）。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmutil -sysuser 命令—配置在服务启动时使用的本地凭据

dpmutil set|get -sysuser 命令用于为主产品 CAAIPApache 服务配置在服务启动时使用的本地凭据。

此命令具有以下格式：

```
dpmutil {-set|-get} -sysuser [-locale iso639value]
```

**-set**

定义在服务启动时使用的本地凭据的配置设置。

**-get**

显示在服务启动时使用的本地凭据的配置设置。

**-sysuser**

定义要配置的在服务启动时使用的本地凭据。系统将提示您输入用户名和密码。该用户和密码必须有效，以便登录到已配置的 EEM 服务器。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**注意：**更改这些信息后，需要先关闭再打开相关产品服务（CAAIPApache 和 CAAIPTomcat），更改才能生效。



## dpmutil vcdefault 命令-配置默认 VMware vCenter 管理员用户

dpmutil set|get|delete vcdefault 命令用于为 CA Virtual Assurance 配置默认 VMware vCenter 管理员用户。发现 VMware vCenter 服务器时，CA Virtual Assurance 使用默认 VMware vCenter 管理员用户 (vcdefault)。该命令用于设置 vcdefault。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -vcdefault [-locale iso639value]
```

### **-set**

为 VMware vCenter 定义 vcdefault 用户。

### **-get**

显示默认用户或为 VMware vCenter 发现配置的用户。

### **-delete**

删除默认用户或为 VMware vCenter 发现配置的用户。

### **-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmutil vcserver 命令-配置 VMware vCenter

dpmutil set|get vcserver 命令用于添加、修改、列出或删除为虚拟机 (VM) 开通配置的 VMware vCenter 服务器。您可以使用该命令在 CA Virtual Assurance 的现有安装上安装 VMware VirtualCenter 组件。需要提供 CA Virtual Assurance 管理员用户名、密码和端口才能设置或获取信息。更改该设置后无需重新启动 Apache。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -vcserver [-locale iso639value]
```

### **-set**

添加或更改 VMware vCenter 服务器的配置设置，包括服务器名称、用户名、端口以及协议。

### **-get**

显示 VMware vCenter 服务器的配置设置，包括服务器名称、用户名、端口以及协议。

### **-delete**

删除 VMware vCenter 服务器，但是它们仍保留在管理数据库和性能数据库中作为历史记录。系统将提示您输入每个参数。

### **-vcserver**

定义 VMware vCenter 服务器。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## **dpmutil -vioserver 命令--配置 VIO 服务器**

使用 dpmutil -vioserver 命令可添加、修改、列出或删除配置的 VIO 服务器。您可以使用该命令在现有的 CA Virtual Assurance 安装上配置 VIO 服务器。需要提供 CA Virtual Assurance 管理员用户名、密码和端口才能设置或获取信息。

**注意：**重新启动 Apache HTTP 服务器以使更改生效。

此命令具有以下格式：

```
dpmutil {-set|-get|-delete} -vioserver [-locale iso639value]
```

### **-set**

添加或更改 VIO 服务器的配置设置，包括服务器名称、用户名和密码。

### **-get**

显示 VIO 服务器的配置设置，包括服务器名称、用户名。

### **-delete**

删除 VIO 服务器，但为了保存历史记录，这些服务器仍保留在管理数据库和性能数据库中。系统将提示您输入每个参数。

### **-vioserver**

定义 VIO 服务器。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## **通用发现命令**

发现命令用于通过 IP 地址或子网 IP 地址发现计算机。您可以使用 CLI 为发现命令编写脚本并使其自动化，然后基于命令结果运行操作。

## dpmcda cancelnetworkscan 命令—取消网络发现

dpmcda cancelnetworkscan 命令取消正在进行中的网络发现。

此命令具有以下格式：

```
dpmcda cancelnetworkscan [-sc sc_url] -network_name networkname [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-network\_name *networkname***

子网名称。不允许使用重复的网络名称；通过特定网络名称只能发现一个子网。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：取消网络发现

此命令取消正在进行中的网络发现。

```
dpmcda cancelnetworkscan -network_name Network1
```

## dpmcda deletenetwork 命令—删除网络

dpmcda deletenetwork 命令删除特定网络。

此命令具有以下格式：

```
dpmcda deletenetwork [-sc sc_url] -network_name networkname [-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-network\_name *networkname***

子网名称。不允许使用重复的网络名称；通过特定网络名称只能发现一个子网。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：删除网络**

此命令删除网络 `Network1`。

```
dpmcda deletenetwork -network_name Network1
```

## **dpmcda discoverhost 命令—发现主机**

`dpmcda discoverhost` 命令发现主机。

此命令具有以下格式：

```
dpmcda discoverhost [sc sc_url] -host_name hostname [-system_user username  
-system_password password][-ws_user username -ws_password password][-locale  
iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-host\_name *hostnames***

定义一个或多个主机。有效条目：逗号分隔的主机名或 IP 地址。

**-system\_user *username* -system\_password *password***

定义用于增强发现的目标系统凭据（检索硬件特征）。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：发现主机**

本示例发现主机。

```
dpmcda discoverhost -host_name 172.24.255.255 -ws_user wsuser -ws_password
wsuserpassword
```

## dpmcda discovernetwork 命令—发现网络

dpmcda discovernetwork 命令发现网络。

此命令具有以下格式：

```
dpmcda discovernetwork [-sc sc_url] -ipv4_addr ipv4address -network_name
networkname [-dnsserver dnsserver] [-domain domain][-system_user username
-system_password password][-ws_user username -ws_password password][-locale
iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-ipv4\_addr *ipv4address***

定义用于发现的子网 IPv4 地址。子网的有效 IPv4 地址格式:

`xxx.xxx.x.*` 或 `xxx.xxx.x.{1-10}` 或 `xxx.xxx.x.0/24`。

**-network\_name *networkname***

子网名称。不允许使用重复的网络名称; 通过特定网络名称只能发现一个子网。

**-dns\_server *dnsserver***

定义 DNS 服务器的 IP 地址。

**-domain *domain***

定义 DNS 发现的域名。

**-system\_user *username* -system\_password *password***

定义用于增强发现的目标系统凭据（检索硬件特征）。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例: 发现网络**

本示例发现具有 IPv4 地址的网络 NetworkA。

```
dpmcda discovernetwork -ipv4_addr 172.24.255.255 -network_name NetworkA -ws_user wsuser -ws_password wsuserpassword
```

## 收集引擎命令

可以使用 CLI 编写脚本并自动执行“收集引擎”命令，根据命令结果运行操作。使用这些命令需要 SNMP 的专业知识。

**注意：**您必须具有 SNMP 的专业知识才能使用这些命令。您还必须具有专业知识才能为支持的 MIB 配置基于列的度量标准。

### dpmce ceconfig -configuremetric 命令—配置度量标准

dpmce ceconfig -configuremetric 命令用于添加、更改和删除主机上的度量标准。

此命令具有以下格式：

```
dpmce ceconfig -configuremetric [-sc sc_url] -host_name hostname -operation
add|delete -sourceType 3 -metricname metricname -metric_type metrictype
-metric_subtype metricsubtype -metric_instance metricinstance [-metric_method
exact|complement] [-metric_datatype metricdatatype] [-lower_threshold
lowerthreshold] [-upper_threshold upperthreshold] [-metric_enabled
metricenabled] [metric_update yes|no] [-ws_user username -ws_password
password] [-locale iso629value]
```

#### **-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

**示例：**80

#### **-host\_name hostname**

定义主机计算机的名称。

#### **-operation add|delete**

添加或删除别名 (-setalias) 或度量标准 (-configuremetric)。

#### **-sourceType**

定义收集度量标准的源。有效条目：1 = 系统度量标准（性能 Lite 度量标准），3 = 来自 SNMP 接口的集合。

#### **-metricname metricname**

定义度量标准的名称。

**-metric\_type *metrictype***

MIB 名称后跟度量标准的符号名称。

**-metric\_subtype *MIBOID***

定义唯一标识受管对象的 MIB 对象标识符 (OID)。

**-metric\_instance *metricinstance***

用于收集数据的列号。

**-metric\_method *exact|complement|exact\_delta|exact\_complement***

定义用于计算和收集的方法。有效条目：

- **exact** = 使用精确度量标准值。随着度量标准值的增加，该值将有助于提高总使用率。示例：内存。
- **complement** = 以补充方式使用度量标准。该值越高，越有助于降低总使用率。示例：CPU 空闲百分比。
- **exact\_delta** = 使用连续值之间的差。示例：MIB2.Instance.ifInOctets（传入字节数）。

**complement\_delta** = 以补充方式使用连续值之间的差。示例：CPU 总空闲。

**-metric\_datatype *metricdatatype***

定义度量标准的数据类型。有效条目：整型、浮点或双精度。

**-lower\_threshold *lowerthreshold***

定义数据中心的总体使用阈值下限。阈值下限必须始终小于阈值上限。

限制：0-100%

**-upper\_threshold *upperthreshold***

定义数据中心的总体使用阈值上限。阈值上限必须始终大于阈值下限。

限制：1-100%

**-metric\_enabled *yes|no***

定义是否已为收集启用度量标准。有效条目：**yes** 或 **no**。

**-metric\_update *yes|no***

定义是否已在数据库中更新度量标准。有效条目：**yes** 或 **no**。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 **caaipsecurity** 设置自己的会话来避免提示输入凭据。



**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：为数据收集配置度量标准**

本示例为主机添加度量标准配置数据。

```
dpmce ceconfig -configuremetric -host_name S0L1054 -operation add -sourceType 3
-metricname SysEDGE.loadAverage15Min -metric_type SysEDGE.loadAverage15Min
-metric_subtype 1.3.6.1.4.1.546.1.1.7.8.28.0 -metric_method exact
-metric_update yes -ws_user admin -ws_password admin
```

**dpmce ceconfig -configuremetricfilter 命令—配置度量标准筛选**

dpmce ceconfig -configuremetricfilter 命令用于添加、更改和删除度量标准筛选。

此命令具有以下格式：

```
dpmce ceconfig -configuremetricfilter [-sc sc_url] -host_name hostname -operation
add|delete -sourceType 3 -metric_os_type metricOStype -metric_type metrictype
-metric_subtype metricsubtype -metric_instance metricinstance [-metric_method
exact|complement] [-metric_datatype metricdatatype] [-metric_filter_default 0|1]
[-include_for_overall yes|no] [-lower_threshold lowerthreshold]
[-upper_threshold upperthreshold] [-metric_enabled yes|no] [-ws_user username
-ws_password password][-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

***port***

定义服务控制器服务器的侦听端口。

示例：80

**-host\_name *hostname***

定义主机计算机的名称。

**-operation *add|delete***

添加或删除别名 (-setalias) 或度量标准 (-configuremetric)。

**-sourceType**

定义收集度量标准的源。有效条目：1 = 系统度量标准（性能 Lite 度量标准），3 = 来自 SNMP 接口的集合。

**-metric\_os\_type metricOSType**

定义度量标准的操作系统类型。有效条目：WINDOWS、AIX、HPUX、LINUX 或 SOLARIS。

**-metricname metricname**

定义度量标准的名称。

**-metric\_subtype MIBOID**

定义唯一标识受管对象的 MIB 对象标识符 (OID)。

**-metric\_instance metricinstance**

用于收集数据的列号。

**-metric\_method exact|complement|exact\_delta|exact\_complement**

定义用于计算和收集的方法。有效条目：

- **exact** = 使用精确度量标准值。随着度量标准值的增加，该值将有助于提高总使用率。示例：内存。
- **complement** = 以补充方式使用度量标准。该值越高，越有助于降低总使用率。示例：CPU 空闲百分比。
- **exact\_delta** = 使用连续值之间的差。示例：MIB2.Instance.ifInOctets（传入字节数）。
- **complement\_delta** = 以补充方式使用连续值之间的差。示例：CPU 总空闲。

**-metric\_datatype metricdatatype**

定义度量标准的数据类型。有效条目：整型、浮点或双精度。

**-metric\_filter\_default yes|no**

定义是否在默认情况下收集度量标准。有效条目：yes 或 no。

**-include\_for\_overall Yes|No**

定义度量标准是否包含在总体计算中。

**是**

将度量标准包含在总体计算中。

**否**

将度量标准排除在总体计算之外。

**-lower\_threshold *lowerthreshold***

定义数据中心的总体使用阈值下限。阈值下限必须始终小于阈值上限。

限制：0-100%

**-upper\_threshold *upperthreshold***

定义数据中心的总体使用阈值上限。阈值上限必须始终大于阈值下限。

限制：1-100%

**-metric\_enabled *yes|no***

定义是否已为收集启用度量标准。有效条目：*yes* 或 *no*。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 *caaipsecurity* 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 *fr\_FR*）。要使用命令提示符的区域设置，请指定 “*native*”。

**示例：配置度量标准筛选**

本示例添加度量标准筛选。

```
dpmce ceconfig -configuremetricfilter -operation add -sourceType 3
-metric_os_type WINDOWS -metricname "Fake.cpuTotalUserPercent" -metric_subtype
"xxxxxxxxxxxxxxxxxxxx" -metric_method complement_delta -ws_user dcauser
-ws_password dcapassword
```

**dpmce ceconfig -disablemetric 命令—禁用度量标准**

**dpmce ceconfig -disablemetric** 命令用于禁用正在为主机收集的度量标准。

此命令具有以下格式：

```
dpmce ceconfig -disablemetric [-sc sc_url] -host_name hostname -metricname
metricname [-ws_user username -ws_password password][-locale iso629value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *hostname***

定义主机计算机的名称。

**-metricname *metricname***

定义度量标准的名称。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 为主机禁用内存度量标准**

本示例为主机 ServerZ 禁用度量标准 `Memory: AvailableMbytes`。

```
dpmce ceconfig -disablemetric -host_name ServerZ -metricname Memory:
AvailableMbytes
```

## **dpmce ceconfig -disablemetricforoverall 命令—禁用度量标准并将其排除在总体计算之外**

`dpmce ceconfig -disablemetricforoverall` 命令用于禁用当前收集的主机度量标准并将其排除在总体计算之外。

此命令具有以下格式:

```
dpmce ceconfig -disablemetricforoverall [-sc sc_url] -host_name hostname
-metricname metricname [-ws_user username -ws_password password][-locale
iso629value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *hostname***

定义主机计算机的名称。

**-metricname *metricname***

定义度量标准的名称。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “native”。

**示例: 为主机禁用内存度量标准**

本示例禁用度量标准 `Memory: AvailableMbytes` 并将其排除在主机 `ServerZ` 的总体计算之外。

```
dpmce ceconfig -disablemetricforoverall -host_name ServerZ -metricname Memory:
AvailableMbytes
```

## dpmce ceconfig -enablemetric 命令—启用度量标准

`dpmce ceconfig -enablemetric` 命令用于启用主机度量标准。度量标准必须可供收集。

此命令具有以下格式:

```
dpmce ceconfig -enablemetric [-sc sc_url] -host_name hostname -metricname
metricname [-include_for_overall Yes|No][-ws_user username -ws_password
password][-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *hostname***

定义主机计算机的名称。

**-metricname *metricname***

定义度量标准的名称。

**-include\_for\_overall Yes|No**

定义度量标准是否包含在总体计算中。

是

将度量标准包含在总体计算中。

否

将度量标准排除在总体计算之外。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：为主机启用内存度量标准**

本示例启用度量标准 `Memory: AvailableMbytes` 并将其包括在主机 `ServerZ` 的总体计算中。

```
dpmce ceconfig -enablemetric -host_name ServerZ -metricname Memory:
AvailableMbytes -include_for_overall Yes
```

## dpmce ceconfig -enablemetricforoverall 命令—启用度量标准并将其包括在总体计算中

dpmce ceconfig -enablemetricforoverall 命令用于启用当前收集的主机度量标准并将其包括在总体计算中。

此命令具有以下格式：

```
dpmce ceconfig -enablemetricforoverall [-sc sc_url] -host_name hostname
-metricname metricname [-ws_user username -ws_password password][-locale
iso629value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### ***port***

定义服务控制器服务器的侦听端口。

示例：80

### **-host\_name *hostname***

定义主机计算机的名称。

### **-metricname *metricname***

定义度量标准的名称。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：为主机启用内存度量标准以包括在总体计算中

本示例启用度量标准 `Memory: AvailableMbytes` 并将其包括在主机 `ServerZ` 的总体计算中。

```
dpmce ceconfig -enablemetricforoverall -host_name ServerZ -metricname Memory:
AvailableMbytes
```

## dpmce ceconfig -getglobal 命令—检索数据中心间隔

dpmce ceconfig -getglobal 命令用于检索数据中心级别的收集、记录和保留间隔。

此命令具有以下格式：

```
dpmce ceconfig -getglobal [-sc sc_url][-ws_user username -ws_password  
password][-locale iso629value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：检索收集、记录和保留间隔

本示例检索数据中心级别的收集、记录和保留间隔。

```
dpmce ceconfig -getglobal
```

## dpmce ceconfig -getnode 命令—检索主机收集间隔

dpmce ceconfig -getnode 命令用于为主机检索收集、记录和保留间隔。

此命令具有以下格式：

```
dpmce ceconfig -getnode [-sc sc_url] -host_name hostname [-ws_user username  
-ws_password password][-locale iso629value]
```



**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://*hostname:port*/...

***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

***port***

定义服务控制器服务器的侦听端口。

示例：80

**-host\_name *hostname***

定义主机计算机的名称。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：为主机 `ServerZ` 检索收集、记录和保留间隔**

本示例为主机 `ServerZ` 检索间隔。

```
dpmce ceconfig -getnode -host_name ServerZ
```

**dpmce ceconfig -getmetriclist 命令—检索度量标准列表**

`dpmce -getmetriclist` 命令用于为特定主机检索度量标准。

此命令具有以下格式：

```
dpmce ceconfig -getmetriclist [-sc sc_url] -host_system hostname [-ws_user username -ws_password password][-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *hostname***

定义主机计算机的名称。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：检索度量标准列表**

本示例为主机 S0L1054 检索度量标准列表。

```
dpmce ceconfig -getmetriclist -host_name S0L1054
```

## **dpmce ceconfig -refreshmetriclist 命令—重试度量标准收集**

`dpmce ceconfig -refreshmetriclist` 命令用于重试度量标准收集。例如，如果在发现期间未安装代理，则不会为收集检测任何度量标准。如果稍后在系统上安装代理，您可以使用此命令重试度量标准收集，而不必重新发现系统。

此命令具有以下格式：

```
dpmce ceconfig -refreshmetriclist [-sc sc_url] -host_name hostname [-ws_user username -ws_password password][-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *hostname***

定义主机计算机的名称。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：重试度量标准收集**

本示例在多个主机上重试度量标准收集。

```
dpmce ceconfig -refreshmetriclist -host_name S0L1054:engManager200:QALAB01
```

## dpmce ceconfig -setalias 命令—设置别名

`dpmce ceconfig -setalias` 命令为 ESX 主机创建别名，以使其度量标准可以包含在 ESX 服务器总体计算中。

此命令具有以下格式：

```
ceconfig -setalias [-sc sc_url] -operation add|delete -caller_id PMM -metricname  
metricname -alias_host_name aliashostname -parent_host_name parenthostname  
-include_for_overall yes|no [-ws_user username -ws_password password][-locale  
iso629value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-operation *add* | *delete***

添加或删除别名 (`-setalias`) 或度量标准 (`-configuremetric`)。

**-caller\_id *PMM***

定义别名的接口 (产品管理模块)。有效条目: `PMM`。

**-metricname *metricname***

定义度量标准的名称。有效条目: 度量标准名称或 `ALL` (如果要删除与别名主机名关联的所有度量标准)。

**-alias\_host\_name *aliashostname***

定义由 `parent_host_name` (AIM) 管理的 ESX 服务器或 VM。

**-parent\_host\_name *parenthostname***

定义安装有 AIM 的系统。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “`native`”。

**示例: 设置别名**

本示例为 ESX 主机设置别名。

```
ceconfig -setalias -caller_id PMM -metricname SysEDGE.memInUseCapacity
-alias_host_name hostDA12 -parent_host_name engServer200 -include_for_overall
yes -ws_user admin -ws_password admin
```

## dpmce ceconfig -setglobal 命令—设置数据中心间隔

dpmce ceconfig -setglobal 命令用于设置数据中心级别的收集、记录和保留间隔以及阈值。

此命令具有以下格式：

```
dpmce ceconfig -setglobal [-sc sc_url] -recording_interval value  
-retention_sec value -retention_day value  
-lower_threshold lowerthreshold -upper_threshold upperthreshold  
[-ws_user username -ws_password password][-locale iso629value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-recording\_interval *value***

定义记录间隔。如果未定义 retention\_day 或 retention\_sec，则必须提供该值。

单位：秒

### **-retention\_day *value***

定义每日数据平均值存储在性能数据库中的时长（每日汇总数据保留）。如果未定义 recording\_interval 或 retention\_sec，则必须提供该值。

单位：天

### **-retention\_sec *value***

定义轮询数据存储在性能数据库中的时长（轮询数据保留）。定义该值时，请考虑收集的受管系统、服务以及度量标准的数量。存储的轮询数据对象随着时间的推移不断累积且可以影响性能。如果出现性能问题，请减少保留天数。如果未定义 recording\_interval 或 retention\_day，则必须提供该值。

单位：天

**-lower\_threshold lowerthreshold**

定义数据中心的总体使用阈值下限。阈值下限必须始终小于阈值上限。

限制：0-100%

**-upper\_threshold upperthreshold**

定义数据中心的总体使用阈值上限。阈值上限必须始终大于阈值下限。

限制：1-100%

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：为数据中心设置收集和记录间隔**

本示例为数据中心设置收集和记录间隔。

```
dpmce ceconfig -setglobal -recording_interval 120
```

**示例：为数据中心设置阈值**

本示例为数据中心设置阈值上限和下限。

```
dpmce ceconfig -setglobal -lower_threshold 10 -upper_threshold 90
```

## **dpmce ceconfig -setnode 命令—设置主机间隔**

`dpmce ceconfig -setnode` 命令用于为主机设置收集、记录和保留间隔。

此命令具有以下格式：

```
dpmce ceconfig -setnode [-sc sc_url] -host_name hostname -recording_interval  
value -retention_sec value -retention_day value [-ws_user username -ws_password  
password][-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *hostname***

定义主机计算机的名称。

**-recording\_interval *value***

定义记录间隔。如果未定义 `retention_day` 或 `retention_sec`，则必须提供该值。

单位: 秒

**-retention\_sec *value***

定义轮询数据存储在性能数据库中的时长（轮询数据保留）。定义该值时，请考虑收集的受管系统、服务以及度量标准的数量。存储的轮询数据对象随着时间的推移不断累积且可以影响性能。如果出现性能问题，请减少保留天数。如果未定义 `recording_interval` 或 `retention_day`，则必须提供该值。

单位: 天

**-retention\_day *value***

定义每日数据平均值存储在性能数据库中的时长（每日汇总数据保留）。如果未定义 `recording_interval` 或 `retention_sec`，则必须提供该值。

单位: 天

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：为主机 **ServerZ** 设置保留间隔

本示例为主机 **ServerZ** 设置保留间隔。

```
dpmce ceconfig -setnode -host_name ServerZ -retention_day 300
```

## dpmce ceconfig -updatemetric 命令—更新度量标准

dpmce ceconfig -updatemetric 命令用于更新当前收集的主机度量标准、阈值和总体计算属性。

此命令具有以下格式：

```
dpmce ceconfig -updatemetric [-sc sc_url] -host_name hostname -metricname metricname [-lower_threshold lowerthreshold][-upper_threshold upperthreshold] include_for_overall Yes|No [-ws_user username -ws_password password][-locale iso629value]
```

#### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://*hostname:port*/...

#### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### ***port***

定义服务控制器服务器的侦听端口。

示例：80

#### **-host\_name *hostname***

定义主机计算机的名称。

#### **-metricname *metricname***

定义度量标准的名称。

#### **-lower\_threshold *lowerthreshold***

定义数据中心的总体使用阈值下限。阈值下限必须始终小于阈值上限。

限制：0-100%

#### **-upper\_threshold *upperthreshold***

定义数据中心的总体使用阈值上限。阈值上限必须始终大于阈值下限。

限制：1-100%



**-include\_for\_overall Yes|No**

定义度量标准是否包含在总体计算中。

是

将度量标准包含在总体计算中。

否

将度量标准排除在总体计算之外。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：为主机 ServerZ 更新内存度量标准**

本示例更新度量标准 `Memory: AvailableMbytes` 的阈值上限和下限，并将其包括在主机 `ServerZ` 的总体计算中。

```
dpmce ceconfig -updatemetric -host_name ServerZ -metricname Memory:
AvailableMbytes -lower_threshold 10 -upper_threshold 90 -include_for_overall Yes
```

## 事件命令

您可以使用 CLI 为事件命令编写脚本并使其自动化，然后基于命令结果运行操作。

### dpmeventmanager create\_event 命令—创建事件

`dpmeventmanager create event` 命令用于创建事件。

此命令具有以下格式：

```
dpmeventmanager create_event [-sc sc_url] [-status eventstatus] [-component
eventcomponent] [-message eventmessage] [-source eventsources] [-target
eventtarget] [-action eventaction] [-audit {0|1}][-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-status *eventstatus***

（可选）指明事件的状态。

**-component *eventcomponent***

（可选）定义事件对应的产品组件。

示例: 策略、映像等。

**-message *eventmessagesubstring***

（可选）定义要匹配的事件消息子字符串。

**-source *eventsource***

（可选）定义生成事件的计算机。

**-target *eventtarget***

（可选）定义发生事件的目标计算机。

**-action *eventaction***

（可选）定义事件的类别。

**-audit={*0*|*1*}**

（可选）指示事件是否为审核事件。使用 **1**（审核事件）或 **0**（不是审核事件）。

默认: `0`

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

### 示例：为服务器发现创建事件

本示例为 ServerXYZ 服务器发现创建事件。

```
dpmeventmanager create_event -status 1 -component ccm -message "1 New Managed System Added - xyz" -source xyz -action "Inventory Event"
```

## dpmeventmanager get\_events 命令—检索事件

dpmeventmanager get events 命令用于检索事件。

此命令具有以下格式：

```
dpmeventmanager get_events [-sc sc_url] [-status event_status] [-component eventcomponent] [-message eventmessagesubstring] [-source eventsources] [-target eventtarget] [-action eventaction] [-from_start_date_time] [-to_end_date_time] [-audit {0|1}] [-user_name username] [-locale iso629value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-status *eventstatus***

（可选）指明事件的状态。

### **-component *eventcomponent***

（可选）定义事件对应的产品组件。

示例：策略、映像等。

### **-message *eventmessagesubstring***

（可选）定义要匹配的事件消息子字符串。

### **-source *eventsources***

（可选）定义生成事件的计算机。

### **-target *eventtarget***

（可选）定义发生事件的目标计算机。

**-action eventaction**

（可选）定义事件的类别。

**-from start\_date\_time**

（可选）使用指定的月 (*mm*)、日 (*dd*) 或年 (*yyyy*) 指定要检索的事件的开始日期和时间。默认值为所有事件（不管其日期如何）。该参数包括以下内容：

**mm**

包括从该月开始创建的事件。

限制：1-12

**dd**

包括从该日期开始创建的事件。

限制：1-31

**yyyy**

包括从该年开始创建的事件。

限制：1970 或以后

**hh**

包括从该小时开始创建的事件。

限制：0-23

**mm**

包括从该分钟开始创建的事件。

限制：0-59

**ss**

包括从该秒开始创建的事件。

限制：0-59

**-to end\_date\_time**

（可选）使用指定的月 (*mm*)、日 (*dd*) 和年 (*yyyy*) 指定要检索的事件的结束日期和时间。默认值为所有事件（不管其日期如何）。该参数包括以下内容：

**mm**

包括在该月结束前创建的事件。

限制：1-12

**dd**

包括在该日期结束前创建的事件。

限制：1-31

**yy**

包括在该年结束前创建的事件。

限制：01-09

**hh**

包括在该小时结束前创建的事件。

限制：1-24

**mm**

包括在该分钟结束前创建的事件。

限制：1-60

**ss**

包括在该秒结束前创建的事件。

限制：1-60

**-audit={0|1}**

（可选）指示事件是否为审核事件。使用 **1**（审核事件）或 **0**（不是审核事件）。

默认：0

**-user\_name *username***

定义做出触发事件的策略更改的用户的用户名。CLI 自动根据操作系统确定用户名。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 **fr\_FR**）。要使用命令提示符的区域设置，请指定 “**native**”。

**示例：检索特定时间段内的策略事件**

本示例检索从 2009 年 3 月 8 日上午 10:10:10 到现在这一时间段内的所有策略事件。

```
dpmeventmanager get_events -component policy -from "3/08/2009 10:10:10"
```

## 常规外壳命令

您可以使用 CLI 为杂项命令编写脚本并使其自动化，然后基于命令结果运行操作。

### caaipsecurity 命令--指定 Web 服务安全证书的凭据

使用 caaipsecurity 命令可以为您自己的 Web 服务会话指定凭据。在 Web 服务会话中使用 caaipsecurity 时，其他 CLI 命令不会提示您输入凭据。

此命令具有以下格式：

```
caaipsecurity setcurrentsession  
[-prompt {yes|no}]  
[-ws_user username -ws_password password -locale iso639value]
```

```
caaipsecurity deletecurrentsession  
[-prompt {yes|no}]  
[-ws_user username -ws_password password -locale iso639value]
```

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

**默认值：** Yes

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全证书的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

### createdpmdefaultgroup 命令—创建或更新服务

createdpmdefaultgroup 命令使用产品部署正在使用的服务器创建或填充服务。如果服务不存在，将创建服务，并且将服务器添加到服务。如果正在添加不受管理的服务器或 CA Virtual Assurance 未发现的服务器，将自动发现并管理该服务器。如果服务已存在，则只需将服务器添加到现有服务。

此命令具有以下格式：

```
createdpmdefaultgroup
```

**示例：填充服务**

该命令没有参数。

```
createdpmdefaultgroup
```

## **dpmcmd run 命令—运行脚本**

dpmcmd run 命令是用于运行脚本的命令行接口。

此命令具有以下格式：

```
dpmcmd run -cmdline commandline [-pre] [-post][-locale iso639value]
```

**-cmdline *commandline***

定义要运行的脚本文件所在的 Shell 命令或路径。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：运行脚本并创建事件**

本示例运行名为 MyScript 的脚本，并在运行脚本之前和之后创建事件。

```
dpmcmd run -cmdline C:\scripts\MyScript.bat -pre -post
```

## dpmix export | import | verify 命令—迁移系统数据

dpmix 命令行实用工具提供了一种方法，用于将特定的用于后续重新部署的 CA Virtual Assurance 数据对象保存到新安装的或其他 CA Virtual Assurance 系统。例如，可以将操作、规则、作业导入或导出到新系统或现有系统。管理数据库信息在注册表中必须可用，并且必须在安装 Microsoft SQL 客户端后实用工具才可工作。您必须拥有具有管理员权限的用户名才可访问 dpmix。如果输入无效参数或不输入参数，会出现一条错误消息，显示语法信息和可迁移的数据类型列表。

此命令具有以下格式：

```
dpmix {-export|-import|-verify} -f filename [-clean] [-replace] [-noprompt] [-nohelpdesk] [-verbose] -user adminusername -password adminpassword [-auth_file authorizationfilename] -cohesion_server {none|default_cohesion|preserve|name} [-port ccaportnumber] [-default {all|none}] [-service {all|none}] [-server {all|none}] [-action {all|none}] [-rule {all|none}] [-job {all|none}] [-resource {all|none}] [-dpm_ibm_hmc {all|none}] [-serviceprocessor {all|none}] [-vCenter {all|none}] [-networkdevice {all|none}]
```

### -export

保存选定（或默认）的数据类型和关联数据，并将其从本地 CA Virtual Assurance 系统导出到指定的数据文件。

### -import

保存选定（或默认）的数据类型和关联数据，并将其从指定的数据文件导入到本地 CA Virtual Assurance 系统。

### -verify

检查数据文件中数据的语法并列出找到的任何问题，但不更新 CA Virtual Assurance 服务器。将验证以下数据类型：

- 诸如 ServerName 的数据不为空。
- 诸如 PollInterval 的数据为数字。
- 诸如 conditionName 的数据的格式。
- 阈值上限和下限。
- 数据段的编号序列，以确定序列从 1 开始并且是连续的。

版本和内部版本号存在于数据文件中。

### -f filename

定义数据要导入其中或要从其中导出数据的文件的完整路径名。

**注意：**如果仅指定文件名而未指定完全限定文件路径，则默认情况下使用当前目录。



**-clean**

（可选）指定在导入数据之前删除目标对象，或在导出数据之前删除现有的数据文件。该选项与导入或导出一起使用。对于导入，该选项在以下环境中十分有用：数据随着时间的推移累积，并且您想要将系统还原到早期时间点的图像（那时使用导出保存的数据）。对于导出，如果通过导出指定的数据文件的名称已存在，并且您想要删除现有数据文件并使用相同名称导出到新文件，则该选项十分有用。

**重要信息！** 该选项可能会损坏数据，因此会发出警告提示，要求您确认该操作。如果指定了 **-noprompt**，则不会收到该警告提示。

**-replace**

（可选）指定在导入数据时，使用共享相同名称的已导入数据对象更新现有的目标对象。该选项与导入参数一起使用。如果不指定该选项，则默认情况下保留原始对象，并且仅导入新对象。

**-noprompt**

（可选）指定取消通常由程序发出的确认提示。

**-nohelpdesk**

（可选）与 **-import** 参数一起使用，阻止操作的帮助中心相关数据导入 CA Virtual Assurance 系统。如果 CA Virtual Assurance 系统上没有配置帮助中心，该操作可以防止故障。

**-verbose**

（可选）与 **-verify** 参数一起使用，显示有关数据的附加信息。

**-user *adminusername***

定义 CA Virtual Assurance 管理员用户名。

**-password *adminpassword***

定义在 **-user** 中指定的 CA Virtual Assurance 管理员密码。通过忽略该参数，可以避免在命令行输入该敏感值。相反，系统会在不以明文形式将密码回显到控制台的情况下提示您输入密码。对于非交互式的情况（如脚本），可以使用 **-auth\_file** 代替 **-password**。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 **dpmutil set auth** 命令生成的 CA Virtual Assurance 管理员凭证。如果不指定该选项，将使用默认的授权文件。

**-cohesion\_server {none | default\_cohesion | preserve | name}**

指定在发现服务器期间使用的 CCA 服务器。该参数与导入参数一起使用。选项包括：

**none**

指定不使用 CA Application Configuration Manager 进行发现。这是默认值。

**default\_cohesion**

指定使用在 CA Virtual Assurance 中配置的默认 CCA 服务器进行发现。

**preserve**

指定使用为数据文件中每个服务器定义的 CCA 服务器进行发现。该选项会保留 CA Virtual Assurance 系统与 CCA 服务器之间的原始关联。

**注意：**如果 CCA 服务器中的任意一个对正在执行导入的目标系统不可用，则应当在数据文件中修改这些名称以引用有效的 CCA 服务器名称。

**name**

指定使用指定的 CCA 服务器进行发现。

**-port *ccaportnumber***

定义 CCA 源服务器的通信端口。仅在为 -acm\_server 指定了 CCA 服务器名称时使用该选项。

**-default={all | none}**

（可选）指定数据类型选择默认值。

**all**

指定包括所有数据类型参数。这是默认值。

**none**

指定排除所有数据类型参数。

**-service={all | none}**

（可选）指定在导入或导出过程中包括还是排除服务。默认值由有效的 -default 选项设置确定。选项包括：

**all**

指定包括所有服务。

**none**

指定排除所有服务。

**-server={all|none}**

(可选)指定在导入或导出过程中包括还是排除服务器及其启用的度量标准。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有服务器。

**none**

指定排除所有服务器。

**-action {all|none}**

(可选)指定在导入或导出过程中包括还是排除操作。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有操作。

**none**

指定排除所有操作。

**-rule={all|none}**

(可选)指定在导入或导出过程中包括还是排除规则。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有规则。

**none**

指定排除所有规则。

**-job={all|none}**

(可选)指定在导入或导出过程中包括还是排除作业。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有作业。

**none**

指定排除所有作业。

**-resource={all|none}**

(可选) 指定在导入或导出过程中包括还是排除资源组和模板。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有资源组和模板。

**none**

指定排除所有资源组和模板。

**-dpm\_ibm\_hmc={all|none}**

(可选) 指定在导入或导出过程中包括还是排除 IBM 硬件管理控制台 (HMC)。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有 HMC。

**none**

指定排除所有 HMC。

**-serviceprocessor={all|none}**

(可选) 指定在导入或导出过程中包括还是排除系统服务处理器 (SSP)。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有 SSP。

**none**

指定排除所有 SSP。

**-vCenter={all|none}**

(可选) 指定在导入或导出过程中包括还是排除 vCenter 服务器。默认值由有效的 **-default** 选项设置确定。选项包括:

**all**

指定包括所有 vCenter 服务器。

**none**

指定排除所有 vCenter 服务器。

**-networkdevice={all|none}**

（可选）指定在导入或导出过程中包括还是排除网络设备。默认值由有效的 **-default** 选项设置确定。选项包括：

**all**

指定包括所有网络设备。

**none**

指定排除所有网络设备。

**示例：仅导出操作**

本示例仅导出来自 CA Virtual Assurance 系统的操作数据。

```
dpmix export -f c:\data.ini -default none -action all
```

**示例：以替换模式导入**

本示例使用 **-replace** 选项将所有数据项（新的和已定义的）导入到 CA Virtual Assurance 系统。

```
dpmix import -f c:\data.ini -replace
```

**示例：以没有提示的干净模式导入**

本示例在不发出警告提示的情况下首先导入所有类型的数据并删除现有的对象。

```
dpmix import -f c:\data.ini -clean -noprompt
```

**dpmrpt schedulereport 命令—排定报告**

通过 **dpmrpt schedulereport** 命令可以排定报告。

此命令具有以下格式：

```
dpmrpt deliver [-sc sc_url] -host_name hostname -port_number portnumber -cms_name cmsname -cms_port_number cmsportnumber -auth auth -bo_username bousername [-bo_password bopassword] -report_name reportname -encrypted {yes|no} [-start_time starttime] [-end_time endtime] [-server_group reportservice] [-system_name systemname] [-metric_name metricname] [-no_server_group numberofservices] [-no_system numberofsystems] [-pre] [-post]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-host\_name *reportservername***

定义主机报告服务器的名称。

**-port number *portnumber***

定义报告服务器的通信端口。

**-report\_name *reportname***

定义 BusinessObjects 报告名称。

**-cms\_port number *cmsportnumber***

定义 BusinessObjects CMS 的通信端口。

**-auth {*secEnterprise* | *secLDAP* | *secWinAD*}**

定义要使用的报告服务器身份验证协议。没有默认值, 因此您必须指定一个协议。选项包括:

***secEnterprise***

指定安全的企业身份验证协议。

***secLDAP***

指定安全的 LDAP 身份验证协议。

***secWinAD***

指定安全的 Windows AD 身份验证协议。

**-bo\_username *user name***

定义 BusinessObjects 的用户名。

**-bo\_password *password***

(可选) 定义 BusinessObjects 的用户的密码。

**-report\_name *reportname***

定义 BusinessObjects 报告名称。

**-encrypted={Yes|No}**

指定是否加密用户名和密码。

是

指定加密用户名和密码。

否

指定不加密用户名和密码。

**-start\_time starttime**

（可选）指定报告的开始时间。

限制：毫秒

**-end\_time endtime**

（可选）指定报告的结束时间。

限制：毫秒

**-server\_group reportservice**

（可选）为报告定义服务。

**-system\_name systemname**

（可选）为报告定义系统（服务器）的名称。

**-metric\_name metricname**

（可选）定义报告度量标准的名称。

**-no\_server\_group numberservices**

（可选）定义报告中的服务数目。

**-no\_system numbersystems**

（可选）定义报告中系统（服务器）的数目。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**示例：排定报告**

本示例将 BusinessObjects 报告排定在指定的时间运行。

```
dpmrpt schedulereport -host_hostname dcamreporting -port_number 8081 -cms_name  
dcamreporting -cms_port_number 8081 -auth secEnterprise -bo_username  
Administrator -encrypted no -report_name -start_time 86400000 -end_time 86400000
```

## dpmssh run 命令—运行远程外壳命令或脚本

dpmssh run 命令允许使用安全套接字外壳 (SSH) 在目标 Linux 或 UNIX 系统上远程登录并执行外壳命令或脚本以进行安全通信。

此命令具有以下格式：

```
dpmssh run -cmdline commandline -target_host targethostname -target_user targetusername -target_password targetpassword [-pre] [-post]
```

### **-cmdline *commandline***

定义要运行的脚本文件所在的 Shell 命令或路径。

### **-target\_host *targethostname***

定义目标 Linux 或 UNIX 主机服务器的名称。

### **-target\_user *targetusername***

定义目标 Linux 或 UNIX 主机服务器的用户名。

### **-target\_password *targetpassword***

指定目标 Linux 或 UNIX 主机服务器的用户密码。

### **-pre**

（可选）指定是否在执行操作之前生成事件。

### **-post**

（可选）指定是否在执行操作之后生成事件。

### 示例：运行外壳命令并创建事件

本示例针对 /tmp 目录运行 ls 命令。

```
dpmssh run -cmdline "ls /tmp" -target_host linuxserver -target_user myusername -target_password pswd -pre -post
```

### 示例：运行远程脚本

本示例运行名为 MyScript 的脚本。

```
dpmssh run -cmdline "/tmp/MyScript.sh" -target_host unixbox -target_user root -target_password pass1
```

## 帮助中心命令

您可以使用 CLI 为帮助中心命令编写脚本并使其自动化，然后基于命令结果运行操作。



## dpmhd addcomment 命令-使用注释更新票单

dpmhd addcomment 命令用于向现有帮助台票单中添加注释。

此命令具有以下格式：

```
dpmhd addcomment [-sc sc_url] -id ticketid -value comment [-locale iso629value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：<https://hostname:port/...>

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：[localhost.ca.com](https://localhost.ca.com)

### ***port***

定义服务控制器服务器的侦听端口。

示例：80

### **-id *ticketid***

定义在打开票单时，由帮助台系统分配给票单的唯一 ID。

示例：[cr:100400](#)

### **-value *comment***

定义用来更新票单的注释字符串。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：向票单中添加注释

本示例更新帮助台票单 [cr:400335](#)，指出已为 SERVICEA 中的所有虚拟机启动关闭电源操作。

```
dpmhd addcomment -id cr:400335 -value "Powering off SERVICEA"
```

## dpmhd close 命令-关闭帮助台票单

dpmhd close 命令用于关闭帮助台票单。

此命令具有以下格式：

```
dpmhd close [-sc sc_url] -id ticketid [-locale iso629value]
```

**-sc *sc\_url***

(可选) 为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-id *ticketid***

定义在打开票单时, 由帮助台系统分配给票单的唯一 ID。

示例: `cr:100400`

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如: 法语为 `fr_FR`)。要使用命令提示符的区域设置, 请指定 “`native`”。

**示例: 关闭帮助台票单**

本示例关闭通过票单 ID `cr:400335` 标识的帮助台票单。

```
dpmhd close -id cr:400335
```

## **dpmhd getattribute 命令-获取票单的属性值**

`dpmhd getattribute` 命令用于获取现有帮助台票单属性的值。

此命令具有以下格式:

```
dpmhd getattribute [-sc sc_url] -id ticketid -name attr_name [-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-id *ticketid***

定义在打开票单时，由帮助台系统分配给票单的唯一 ID。

示例：cr:100400

**-name *attr\_name***

定义票单属性的名称。支持的属性名称如下所示：

- 说明
- 状态
- 摘要

注意：这些属性名称区分大小写。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：获取票单属性的值**

本示例检索通过票单 ID cr:400335 标识的帮助台票单的 description 属性的值。

```
dpmhd getattribute -id cr:400335 -name description
```

## dpmhd getstatus 命令-获取票单的状态

dpmhd getstatus 命令用于获取现有帮助台票单的状态。

此命令具有以下格式：

```
dpmhd getstatus [-sc sc_url] -id ticketid [-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-id *ticketid***

定义在打开票单时，由帮助台系统分配给票单的唯一 ID。

示例: `cr:100400`

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：检索票单状态**

本示例检索由票单 ID `cr:400335` 标识的帮助台票单的状态。

```
dpmhd getstatus -id cr:400335
```

## dpmhd open 命令—打开帮助台票单

`dpmhd open` 命令用于打开新的帮助台票单。

此命令具有以下格式：

```
dpmhd open [-sc sc_url] -summary summarytext [-description descriptiontext]  
[-affected_user username] [-entity_name entityname] [-type tickettype]  
[-template templatename] [-pre] [-post] [-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-summary *summarytext***

定义有关打开票单的原因的简要说明。

**-description *descriptiontext***

（可选）指定打开票单的原因。

**-entity\_name *entityname***

（可选）定义服务器或服务的名称，用来将票单与帮助台系统中的已知配置项进行匹配。如果配置项主机名与实体名称相同，则该票单与该配置项关联。

**-affected\_user *username***

（可选）定义与该票单关联的最终用户。

**-type *tickettype***

（可选）定义要打开的票单的类型。该参数需要与 **-template** 参数一起使用。如果您未指定类型，将使用为帮助台系统配置的默认票单。支持的票单类型取决于帮助台系统的配置。例如，如果为 ITIL 支持配置了 **Unicenter Service Desk**，则支持“更改顺序”、“突发事件”、“问题”和“请求”票单类型。

**-template *templatename***

（可选）定义打开票单时要使用的模板的名称。模板包含默认票单设置。在使用模板选项时，必须使用票单类型才能找到模板。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

## dpmhd setattribute 命令—设置票单的属性

dpmhd setattribute 命令用于更新现有帮助台票单的属性。

此命令具有以下格式：

```
dpmhd setattribute [-sc sc_url] -id ticketid -name attr_name -value attr_value
[-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-id *ticketid***

定义在打开票单时，由帮助台系统分配给票单的唯一 ID。

示例：cr:100400

**-name *attr\_name***

定义票单属性的名称。支持的属性名称如下所示：

- 说明
- 状态
- 摘要

**注意：**这些属性名称区分大小写。

**-value *attr\_value***

定义用来更新属性的值。

**注意：**如果使用 setattribute 更新票单状态，请输入与状态关联的内部代码。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：更新票单属性**

本示例设置由票单 ID 400335 标识的帮助台票单的 description 属性的值。

```
dpmhd setattribute -id cr:400335 -name description -value "Poweroff server1"
```

## dpmhd setstatus 命令—设置票单的状态

dpmhd setstatus 命令用于更新现有帮助台票单的状态。

此命令具有以下格式：

```
dpmhd setstatus [-sc sc_url] -id ticketid -value statuscode [-locale iso629value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-id *ticketid***

定义在打开票单时，由帮助台系统分配给票单的唯一 ID。

**示例：**cr:100400

**-value *statuscode***

表示用来更新票单的状态代码。例如，对于 **Unicenter Service Desk**，输入与状态关联的内部代码。用于设置 **Unicenter Service Desk** 票单状态的有效代码包括：

**APP**

将状态设置为已核准。

**CL**

将状态设置为已关闭。

**OP**

将状态设置为打开。

**REJ**

将状态设置为已拒绝。

**RSCH**

将状态设置为正在研究。

**WIP**

将状态设置为正在进行。

**-locale *iso639value***

（可选）指定 **ISO 639\_3166** 组合以覆盖默认的英语输出（例如：法语为 **fr\_FR**）。要使用命令提示符的区域设置，请指定 “**native**”。

**示例：更新票单状态**

本示例将由票单 ID **cr:400335** 标识的帮助台票单的状态更新为 “已核准”。

```
dpmhd setstatus -id cr:400335 -value APP
```

## 映像命令

您可以使用 **CLI** 为映像命令编写脚本并使其自动化，然后基于命令结果运行操作。



## dpmimg image 命令-部署 Solaris 映像

使用 `dpmimg image` 命令，可将 Solaris 映像从 JumpStart 服务器部署到 Solaris 客户端，从 JumpStart 安装服务器之外的 JumpStart 引导服务器映像 Solaris 客户端，或者映像 Solaris 10 x86 客户端。

此命令具有以下格式：

```
dpmimg image [-sc sc_url] -target_host targethostname -target_mac macaddress
[-itcm_server itcm domain manager] -auto_deploy {Yes|No} [-deploy_template
templatename] -target_username username [-target_password password] [-auth_file
authorizationfile] [-auth_comp componentID] -img_host imagehostname -img_name
imagename -img_location location [-boot_img_name bootimagename]
[-boot_img_location bootimagelocation] [-boot_img_host bootimagehost]
[-servproc_hostname serviceprochostname] [-servproc_user serviceprocusername]
[-servproc_pw serviceprocpassword] -dhcp_hostname dhcphostname [-itcm_server
SD_adapter_server] [-scalability_server scalabilityservername] [-wait [timeout]]
[-pre] [-post] [[-ws_user <value> -ws_password <value>]] [-prompt
<yes|no>]] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

### ***port***

定义服务控制器服务器的侦听端口。

示例：`80`

### **-target\_host *targethostname***

定义要将映像部署到的目标主机服务器的名称。

### **-target\_mac *macaddress***

定义要部署映像的目标计算机的硬件地址。

### **-itcm\_server *itcmdomainmanager***

（可选）定义软件交付适配器所在 Unicenter DSM 域管理器的名称。当仅配置一个软件交付适配器 Unicenter DSM 域管理器时，该名称是可选的。仅对 CA Virtual Assurance 有效。

**-auto\_deploy {yes|no}**

指定是否自动部署 CA Virtual Assurance 代理。选项包括：

**是**

自动部署 CA Virtual Assurance 代理。

**否**

防止自动部署 CA Virtual Assurance 代理。

**默认值：** no

**-deploy\_template *templatename***

（可选）指定在 CA Virtual Assurance 中创建的通用模板的名称。

**注意：** 请勿将该模板与由 VMware vCenter 创建并管理的模板相混淆。

**-target\_username *targetusername***

定义用于将代理部署到的目标主机服务器（要将映像部署到该服务器）的用户名。

**-target\_password *targetpassword***

（可选）定义用于将代理部署到的目标主机服务器（要将映像部署到该服务器）的用户密码。如果未指定密码，将从授权文件检索密码。

**注意：** 可使用 dpmutil CLI 设置授权文件。

**-auth\_file *authorizationfilename***

（可选）指定用于检索加密密码的授权文件的完整路径名称。该文件包含使用 dpmutil set auth 命令生成的 CA Virtual Assurance 管理员凭据。如果不指定该选项，将使用默认的授权文件。

**-auth\_comp *componentID***

（可选）指定可用于对主机和用户进行分组的组件 ID。

**-img\_host *RSIServer***

定义 RSI 服务器的名称。

**-img\_name *imagename***

定义要部署的映像的名称。

**-img\_location *imagelocation***

定义要部署的映像的位置。

**-boot\_img\_name *bootimagename***

（可选）定义要部署的引导映像的名称。

**-boot\_img\_location *bootimagelocation***

（可选）定义要部署的映像的位置。

**-boot\_img\_host *bootimagehost***

(可选) 定义引导映像所在的服务器的名称。

**-servproc\_hostname *serviceprochostname***

(可选) 定义服务处理器的主机名 (Solaris 10 x86 类型)。

**-servproc\_user *serviceprocusername***

(可选) 定义连接到服务处理器的用户的用户名 (Solaris 10 x86 类型)。

**-servproc\_pw *serviceprocpassword***

(可选) 定义连接到服务处理器的用户的密码 (Solaris 10 x86 类型)。

**-itcm\_server *itcmdomainmanager***

(可选) 定义软件交付适配器所在 Unicenter DSM 域管理器的名称。当仅配置一个软件交付适配器 Unicenter DSM 域管理器时, 该名称是可选的。仅对 CA Virtual Assurance 有效。

**-dhcp\_hostname *dhcphostname***

定义用于连接到 DHCP 服务器的主机名 (Solaris 10 x86 类型)。

**-scalability\_server *servername***

(可选) 指定要用于软件分发的 Unicenter DSM 可扩展性服务器。

**-wait *timeout***

(可选) 表示是否显示作业的状态并直到操作完成、失败或达到超时时长 (如果您指定了超时值) 才返回。如果不指定该选项, CLI 不等待完成即返回。如果无响应, 可选的超时值将定义超时时长 (分钟)。如果输入不带超时值的等待选项, CLI 将使用 `caimgconf.cfg` 文件中的默认等待时间或默认等待 120 分钟。除所有正整数之外, 也可使用下列超时值:

"""

无超时值。

0

无限等待。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

(可选) 指定连接到映像服务模块的 Web 服务用户名。

**-ws\_password wspassword**

（可选）指定连接到映像服务模块的 Web 服务用户的密码。

**-prompt {yes|no}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: Yes

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将映像部署到 Solaris 客户端计算机**

本示例将 Solaris 映像从 JumpStart 服务器部署到 Solaris 客户端计算机，而且不等待映像部署完成。

```
dpmimg image -imgghost sunserver001 -img_name jump_5.10 -auto_deploy Yes
-target_host sunserver002 -target_username rootuser -target_password
rootuserpassword -target_mac 00:00:00:00:00:00 -img_location /export/jump_5.10
```

**示例：将映像部署到 Solaris 10 x86 客户端计算机**

本示例将 Solaris 映像从 JumpStart 服务器部署到 Solaris 10 x86 客户端计算机。

```
dpmimg image -target_host sunserver001 -target_mac 00:00:00:00:00:00
-auto_deploy no -target_username root -target_password rootuserpassword
-img_host sunimageserver001 -img_name sol_10_508_x86 -img_location
/jsimages/SunImageServer/sol_10_508_x86 -servproc_hostname sunserver002
-servproc_user root -servproc_pw rootuserpassword -dhcp_hostname
sunimageserver002 -boot_img_host sunimageserver001 -boot_img_name
sol_10_508_x86_bt -boot_img_location /jsboot/SunImageServer/sol_10_508_x86_bt
-wait 120 -pre -post
```

**示例：使用多个 JumpStart 服务器将 Solaris 映像部署到 Solaris 服务器**

本示例使用多个 JumpStart 服务器将 Solaris 映像部署到 Solaris 服务器 sunserver002。安装 OS 映像位于 sunserver003 上，引导映像位于引导服务器 sunserver004 上。该命令会等待 2 小时，等映像部署之后再运行。

```
dpmimg image -target_host sunserver002 -target_mac 00:00:00:00:00:00
-auto_deploy yes -target_username root -target_password rootuserpassword
-img_host sunserver003 -img_name sol_10_sparc -img_location
/jsimages/sol_10_sparc -boot_img_host sunserver004 -boot_img_name
sol_10_sparc_boot -boot_img_location /jsboot/sol_10_sparc_boot -wait 120 -pre
-post
```

**示例：使用多个 JumpStart 服务器将 Solaris 10 x86 映像部署到 Solaris 10 x86 服务器**

本示例使用多个 JumpStart 服务器将 Solaris 10 x86 映像部署到 Solaris 10 x86 服务器 sunserver002。安装 OS 映像位于 sunserver002 上，引导映像位于引导服务器 sunserver003 上。它将连接到服务处理器服务器 sunserver004 和 DHCP 服务器 dhcpserver001。

```
dpmimg image -target_host sunserver002m -target_mac 00:00:00:00:00:00
-auto_deploy no -target_username root -target_password rootuserpassword
-img_host sunserver002 -img_name sol_10_x86 -img_location /jsimages/sol_10_x86
-servproc_hostname sunserver004 -servproc_user root -servproc_pw
rootuserpassword -dhcp_hostname dhcpserver001 -boot_img_host sunserver003
-boot_img_name sol_10_x86_boot -boot_img_location /jsboot/sol_10_x86_boot -wait
120 -pre -post
```

**dpmimg imgjobcheck 命令-获取 Solaris 映像作业状态**

dpmimg imgjobcheck 命令用于检索特定 CA Virtual Assurance 作业 ID 的 Solaris 映像作业状态。

此命令具有以下格式：

```
dpmimg imgjobcheck [-sc sc_url] -status jobID [-pre] [-post] [[-ws_user <value>
-ws_password <value>]][-prompt <yes|no>]][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-status={*job ID*}**

定义用于获取作业状态的作业 ID。

**-pre**

（可选）指定是否在执行操作之前生成事件。

**-post**

（可选）指定是否在执行操作之后生成事件。

**-ws\_user *wsuser***

（可选）指定连接到映像服务模块的 Web 服务用户名。

**-ws\_password *wspassword***

（可选）指定连接到映像服务模块的 Web 服务用户的密码。

**-prompt {*yes|no*}**

（可选）指定是否提示您输入 Web 服务用户 ID 和密码。

默认值: `Yes`

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：获取 Solaris 开通作业的作业状态**

本示例检索特定 Solaris 开通作业的作业状态。

```
dpmimg imgjobcheck -status 45
```

## 策略命令

可以使用 `dppolicy CLI` 编写策略命令的脚本并使其自动化，然后根据命令的结果执行操作。

### `dppolicy createdefaultactions` 命令—创建默认操作

`dppolicy createdefaultactions` 命令用于创建四个默认的“设置运行状况”操作。这些操作是在安装期间创建的。例如，如果您要在意外删除操作后再次创建这些操作，可使用此命令。

此命令具有以下格式：

```
dppolicy createdefaultactions [-sc sc_url] [-ws_user username -ws_password password] [-locale iso639value]
```

#### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

#### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

#### ***port***

定义服务控制器服务器的侦听端口。

示例：`80`

#### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

#### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

## dpmpolicy exportaction 命令-将操作导出到文件

dpmpolicy exportaction 命令用于将操作导出到文件。

此命令具有以下格式：

```
dpmpolicy exportaction [-sc sc_url] [-action_name actionname] -file filename
[-ws_user username -ws_password password] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-action\_name *actionname***

定义操作的名称。如果不指定该参数，将导出所有操作。

### **-file *filename***

定义文件名。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：将操作导出到文件

本示例将单个操作导出到文件。

```
dpmpolicy exportaction -action_name ACTION_A -file C:\ACTION_A.txt
```



## dpmpolicy exportrule 命令—将规则导出到文件

dpmpolicy exportrule 命令用于将规则导出到文件。

此命令具有以下格式：

```
dpmpolicy exportrule [-sc sc_url] [-rule_name rulename] -file filename [-ws_user username -ws_password password] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-rule\_name *rulename***

定义为创建的规则分配的名称。

### **-file *filename***

定义文件名。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“native”。

### 示例：将规则导出到文件

本示例将单个规则导出到文件。

```
dpmpolicy exportrule -rule_name SERVER_RULE_A -file C:\SERVER_RULE_A.txt
```

### 示例：将所有规则导出到文件

本示例将所有规则导出到文件。

```
dpmpolicy exportrule -file C:\ALL_RULES.txt
```

## dpmpolicy findbestmachine 命令-查找最佳服务器

dpmpolicy findbestmachine 命令用于从指定的服务器列表中查找总体使用率最低的服务器。

此命令具有以下格式：

```
dpmpolicy findbestmachine [-sc sc_url] -machine_list machinelist [-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：查找最佳服务器

本示例从指定的服务器列表中查找总体使用率最低的服务器。

```
dpmpolicy findbestmachine -machine_list "machine1.ca.com, machine2.ca.com, 172.24.36.107"
```

## dpmpolicy findcandidatemachine 命令-查找候选服务器

dpmpolicy findcandidatemachine 命令用于查找与用户定义的规格相匹配的服务器。

此命令具有以下格式：

```
dpmpolicy findcandidatemachine [-sc sc_url] [-machine_list machinelist]  
-constraint Source:OP:Value [-constraint Source:OP:Value] [-ws_user username  
-ws_password password] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

**-constraint={Source:OP:Value},...**

定义候选服务器必须满足的规格。选项包括：

**Source={MEMORY|CPUSPEED|STORAGE|ARCH}**

指定约束类型和关联值。

**MEMORY**

指定内存约束。

限制：MB

**CPUSPEED**

指定 CPU 处理速度约束。

限制：MHz

**STORAGE**

指定存储约束。

限制：MB

**ARCH**

指定硬件约束。

限制：X86、SPARC、PA-RISC、POWERPC 或 X64。

**OP={EQ|NEQ|GT|GTE|LT|LTE}**

指定操作数。操作数包括：

EQ=等于

NEQ=不等于

GT=大于

GTE=大于或等于

LT=小于

LTE=小于或等于

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

### 示例：查找服务器

本示例查找系统内存大于 2000 MB 的服务器。

```
dpmpolicy findcandidatemachine -constraint MEMORY:GT:2000
```

### 示例：在列表中查找服务器

本示例从提供的服务器列表中查找系统内存大于 2000 MB、CPU 不小于 2400 MHz 的服务器。

```
dpmpolicy findcandidatemachine -constraint MEMORY:GT:2000 -constraint  
CPUSPEED:GTE:2400 -machine_list alert.ca.com, 141.202.224.59
```

## dpmpolicy getruleexeccount 命令-检索规则执行计数

dpmpolicy getruleexeccount 命令用于返回规则尝试运行的次数。

此命令具有以下格式：

```
dpmpolicy getruleexeccount [-sc sc_url] -rule_name rulename [-ws_user username  
-ws_password password][-locale iso639value]
```

#### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

#### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

#### **port**

定义服务控制器服务器的侦听端口。

示例：80

#### **-rule\_name *rulename***

定义为创建的规则分配的名称。

#### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

#### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：获取规则执行计数

本示例检索规则的执行次数。

```
dpmpolicy getruleexeccount -rule_name GENERIC_RULE
```

## dpmpolicy importaction 命令-从文件导入操作

dpmpolicy importaction 命令用于从文件导入操作。

此命令具有以下格式：

```
dpmpolicy importaction [-sc sc_url] -file filename [-action_name actionname]  
[-ws_user username -ws_password password] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-file *filename***

定义文件名。

### **-action\_name *actionname***

定义操作的名称。当文件仅包含一个操作时，该选项可用于在导入时重命名该操作。如果文件包含多个操作，将忽略该值。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：从文件导入操作

本示例导入操作并对其进行重命名。

```
dpmpolicy importaction -file C:\ACTION_A.txt -action_name ACTION_B
```

## dpmpolicy importaction 命令—从文件导入规则

dpmpolicy importrule 命令用于从文件导入规则。

此命令具有以下格式：

```
dpmpolicy importrule [-sc sc_url] [-entity_type entitytype -entity_name  
entityname] -file filename [-rule_name rulename] [-ws_user username -ws_password  
password] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-entity\_type={*server*|*service*}**

定义该规则是与服务器还是服务关联。

### **-entity\_name *entityname***

定义服务器或服务的名称。

### **-file *filename***

定义文件名。

### **-rule\_name *rulename***

定义为创建的规则分配的名称。当文件仅包含一个规则时，该选项可用于在导入时重命名该规则。如果文件包含多个规则，将忽略该值。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：从文件导入规则**

本示例将导入规则并对其进行重命名。

```
dpmpolicy importrule -file C:\RULE_A.txt -rule_name RULE_B
```

## dpmpolicy resetruleexeccount 命令-重置运行规则的重试次数

dpmpolicy resetruleexeccount 命令用于将尝试运行规则的次数重置为零。

此命令具有以下格式：

```
dpmpolicy resetruleexeccount [-sc sc_url] -entity_type entitytype -entity_name entityname -rule_name rulename [-ws_user username -ws_password password] [-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-entity\_type={*server*|*service*}**

定义该规则是与服务器还是服务关联。

**-entity\_name *entityname***

定义服务器或服务的名称。

**-rule\_name *rulename***

定义为创建的规则分配的名称。



**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：重置服务器阈值**

本示例重置服务器上运行的规则的阈值。

```
dpmpolicy resetruleexccount -entity_type 0 -entity_name machine1.ca.com
-rule_name SERVER1_RULE
```

**示例：重置服务规则的重试次数**

本示例将服务上运行的规则的重试次数重置为零。

```
dpmpolicy resetruleexccount -entity_type 1 -entity_name SERVICE1 -rule_name
SERVICE1_RULE
```

## dpmpolicy runaction 命令-运行操作

`dpmpolicy runaction` 命令用于运行操作和操作序列。

此命令具有以下格式：

```
dpmpolicy runaction [-sc sc_url] -action_name action name [-event_source event
source] [-event_message event_message] [-rule_name rule name] [-server_name
server name] [-service_name service name] [-propagate] [-vm_name vm name]
[-datacenter datacenter name] [-host_system host system] [-wait_rc] [-pre] [-post]
[-ws_user username -ws_password password] [-locale iso639value] [-D-overwrite
Parameter=value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

**示例：**`80`

**-action\_name *actionname***

定义操作的名称。

**-event\_source *eventsourcesource***

(可选) 定义用来填充 %EVENTSOURCE% 替代变量的值。

**-event\_message *eventmessage***

(可选) 定义用来填充 %EVENTMESSAGE% 替代变量的值。

**-rule\_name *rulename***

(可选) 定义用来填充 %RULENAME% 替代变量的值。

**-server\_name *servername***

(可选) 定义用来填充 %SERVER% 替代变量的值。

**-service\_name *servicename***

(可选) 定义用来填充 %SERVICE% 替代变量的值。

**-propagate**

(可选) 指定您希望针对在 -service\_name 选项中指定的服务中的所有服务器运行操作。

**-vm\_name *vmname***

(可选) 定义用来填充 %VMNAME% 替代变量的值。

**-datacenter\_name *datacentername***

(可选) 定义用来填充 %DATACENTER% 替代变量的值。

**-host\_system *hostsystem***

(可选) 定义用于填充 %HOSTSYSTEM% 替代变量的值 (VMware ESX 主机计算机的名称)。

**-wait\_rc**

(可选) 指定是否等待“运行命令脚本”操作完成。

**-pre**

(可选) 指定是否在执行操作之前生成事件。

**-post**

(可选) 指定是否在执行操作之后生成事件。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全认证的凭据。如果不包含凭据, 系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**-D-overwrite *Parameter=value***

（可选）为指定的参数定义新值。

**示例：运行操作使服务器联机**

本示例运行可使服务器联机的操作。

```
dmpolicy runaction -action_name BringServerOnline -server_name Server001 -
service_name Production -pre -post
```

**示例：运行操作**

本示例运行某个操作。

```
dmpolicy runaction -action_name Action1
```

**示例：针对服务中的所有服务器运行操作**

本示例针对 SERVICE1 服务中的所有服务器运行操作。

```
dmpolicy runaction -action_name ACTION1 -service_name SERVICE1 -propagate
```

**示例：运行需要使用 VM 的操作**

本示例使用 VM 运行某个操作。

```
dmpolicy runaction -action_name vc_action -vm_name Finance\Group\FinVMServer1
-datacenter FINANCE\NYC
```

**示例：使用不同参数值运行操作**

本示例使用与 CA Virtual Assurance 用户界面中所设置值不同的 vm\_name 参数值运行操作。

```
dmpolicy runaction -action_name Create_snapshot -D-vm_name=VM02
```

**dmpolicy setruleexeclimit 命令-设置规则的限制**

dmpolicy setruleexeclimit 命令用于设置确定规则可重试次数的限制。

此命令具有以下格式：

```
dmpolicy setruleexeclimit [-sc sc_url] -entity_type entitytype -entity_name
entityname -rule_name rulename -limit_option option [-limit_value value]
[-ws_user username -ws_password password] [-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-entity\_type={*server*|*service*}**

定义该规则是与服务器还是服务关联。

**-entity\_name *entityname***

定义服务器或服务的名称。

**-rule\_name *rulename***

定义为创建的规则分配的名称。

**-limit\_option=[*limited*|*unlimited*|*disabled*]**

指定规则运行次数的限制选项。选项如下：

***limited***

指定规则运行的次数。

***unlimited***

指定规则可以无限次运行。

***disabled***

指定禁用规则限制选项。

**-limit\_value *value***

（选项）定义限制值。仅当与设置为 *limited* 的限制选项结合使用，该选项才有效。

限制: 大于等于 0

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：设置规则重试次数**

本示例将服务上运行的规则的重试次数限制设置为 40。

```
dpmpolicy setruleexeclimit -entity_type service -entity_name SERVICE1 -rule_name SERVICE1_RULE -limit_option limited -limit_value 40
```

**示例：将规则重试次数设置为无限**

本示例将规则可在服务器上运行的次数设置为无限制。

```
dpmpolicy setruleexeclimit -entity_type server -entity_name machine1.ca.com -rule_name MACHINE1_RULE -limit_option unlimited
```

**示例：禁用限制选项**

本示例禁用该规则的限制选项。

```
dpmpolicy setruleexeclimit -entity_type server -entity_name machine2.ca.com -rule_name MACHINE2_RULE -limit_option disabled
```

## 策略配置

您可以使用 caismutlity CLI 将新功能添加到策略配置用户界面中。

### caismutlity 命令 -- 将功能添加到策略配置 UI 中

使用 caismutlity 命令可提供用于配置用户界面的功能策略。功能策略包括：

- 将现有的 eHealth 模板导入 CA Virtual Assurance 作为策略配置模板。在 CA Virtual Assurance UI 上更新策略配置模板，并将其应用于 SystemEDGE 代理。
- 指示 SystemEDGE 在下列情况下向新的分发服务器报告：
  - 删除了旧的分发服务器。
  - 添加了其他分发服务器。
  - 移至其他分发服务器管理代理所在的其他位置。
- 指示 SystemEDGE 在下列情况下向新的域服务器报告：
  - 删除了旧的域服务器。
  - 在 CA Virtual Assurance 上执行并行升级。

- 在下列情况下将 SystemEDGE 策略和模板以及 SRM 策略导入和导出到域服务器：
  - 在测试计算机上创建的新策略和模板已迁移到生产计算机中。
  - 在 CA Virtual Assurance 上执行并行升级。

此命令具有以下格式：

```
caismutility
[-f server]
[-o sysEDGE|sysEDGESRM]
[-l]
[-d -p policy]
[-importEH dir|file]
[-useDirName]
[-overwrite]
[-c manager agent1]
[-user user]
[-password password]
[-locale locale]
[-import dir|file]
[-type SEPolicy|SETemplate|SRMPolicy|SRMThreshold|SRMTest|auto]
[-export dir]
```

**注意：**所有选项区分大小写。

**-f server**

指定连接到域服务器。

**-o {sysEDGE|sysEDGESRM}**

指定要使用的配置类型。

**-l**

列出可用的对象。

**-d {-p policy}**

将策略传送给代理。

**-importEH {dir|file}**

导入 eHealth 模板。

**-useDirName**

指定父目录的名称。

**-overwrite**

指定覆盖现有模板。

**-c manager {agent1 ...}**

指示代理向新的管理器报告。

**-user user**

指定用户名。

**-password password**

指定密码。

**-locale iso639value**

指定 ISO 639\_3166 组合（例如：法语为 fr\_FR）以覆盖默认的英语输出。要使用命令提示符的区域设置，请指定 “native”。

**-import {dir|file}**

指定要添加到文件存储的策略或模板。

**-type {SEPolicy|SETemplate|SRMPolicy|SRMThreshold|SRMTest|auto}**

指定要导入的文件类型。

**-export dir**

将所有策略和模板复制到指定目录。

**示例：导入一组 eHealth 模板**

此示例导入了一组 eHealth 模板。

```
caismutility -user xxx -password yyy -importEH c:\templates\
```

**示例：将 SystemEDGE 策略导入域服务器中**

此示例将某些 SystemEDGE 策略导入此域服务器中。

```
caismutility -user xxx -password yyy -type SEPolicy -import c:\policies
```

**示例：指示代理向新的管理器报告**

此示例将主机 agent.com 上的代理移至新的管理器 manager.com。

```
caismutility -user xxx -password yyy -o sysEDGE -c manager.com agent.com
```

## 资源管理器命令

可以使用 dpmresourcemgr CLI 编写资源管理器命令的脚本并使其自动化，然后根据命令的结果执行操作。

## dpmresourcemgr addbaremetal 命令-添加物理计算机

addbaremetal 命令用于向数据中心添加物理计算机。

此命令具有以下格式：

```
dpmresourcemgr addbaremetal [-sc sc_url] -server_name hostname -MAC_address macaddress [-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### ***port***

定义服务控制器服务器的侦听端口。

示例：80

### **-host\_name *source host name***

定义正在运行的源服务器的名称。

### **-MAC\_address *macaddress***

定义目标计算机的 MAC 地址。有效条目：12 位十六进制值（例如，A-F、0-9、a-f），格式如下：0013724C2140 或 00:13:72:4C:21:40。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：添加物理计算机

本示例添加 MAC 地址为 00:13:72:4C:21:40 的计算机 baremetal2。

```
dpmresourcemgr addbaremetal -sc https://servicecontroller/dpm/sc -server_name baremetal2 -MAC_address 00:13:72:4C:21:40 -ws_user wsuser -ws_password wsuserpassword
```



## dpmresourcemgr addcomputertonetwork 命令-向网络中添加服务器

dpmresourcemgr addcomputertonetwork 命令用于向现有网络中添加服务器。

此命令具有以下格式：

```
dpmresourcemgr addcomputertonetwork [-sc sc_url] -vlan_id vlanid -mac_list  
macaddresses [-vlan_tagged][-ws_user username -ws_password password][-locale  
iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-vlan\_id *vlanid***

为虚拟网络 (VLAN) 定义整数。有效条目：整数 1-4094，其中 1 是默认网络接口卡 (NIC)。

### **-mac\_list *macaddresses***

定义服务器的 NIC MAC 地址的列表。有效条目：以逗号分隔的 NIC MAC 地址（格式为 xx:xx:xx:xx:xx:xx）。

### **-vlan\_tagged**

将 VLAN 定义为已标记。如果未指定，VLAN 将显示为未标记。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：向网络中添加服务器

本示例向 VLAN 4556 中添加服务器。

```
dpmresourcemgr addcomputertonetwork -vlan 4556 -mac_list 11:22:33:44:55:66  
-ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr addipaddresspool 命令-添加 IP 地址池

dpmresourcemgr addipaddresspool 命令用于使用静态 IP 地址或 DHCP 添加 IP 地址池。

此命令具有以下格式：

```
dpmresourcemgr addipaddresspool [-sc sc_url] -subnet_address address  
-starting_ip ipaddress -ending_ip ipaddress -ip_assignment_type static|dhcp  
[-name name][-owner ownername][-ws_user username -ws_password password][-locale  
iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-subnet\_address *address***

定义网络的 IP 地址。有效条目：xxx.xxx.xxx.xxx 或 xxx.xxx.xxx.xxx/xx。

### **-starting\_ip *ipaddress***

在该范围中定义起始有效 IP 地址，以便在静态 IP 地址池中使用。有效条目：xxx.xxx.xxx.xxx。

### **-ending\_ip *ipaddress***

在 IP 范围中定义结束 IP 地址。有效条目：xxx.xxx.xxx.xxx。

### **-ip\_assignment\_type static | dhcp**

定义服务器获取 IP 地址（静态 IP 地址或 DHCP）的方法。

**-name *name***

定义项的名称。有效条目：1-128 个字母数字字符。

**-owner *ownername***

定义 IP 地址池的所有者。有效条目：1-128 个字母数字字符。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：添加 IP 地址池**

本示例添加 IP 地址池 `engpool`，其中 IP 地址范围为 10.10.21.130 到 10.10.21.140，并将 IP 分配方法设置为 DHCP。

```
dpmresourcemgr addipaddresspool -subnet_address 10.10.21.0 -starting_ip
10.10.21.130 -ending_ip 10.10.21.140 -ip_assignment_type DHCP -ws_user wsuser
-ws_password wsuserpassword
```

**dpmresourcemgr addmachinestosome 命令-向服务中添加服务器**

`addmachinestosome` 命令用于向服务中添加服务器。

此命令具有以下格式：

```
dpmresourcemgr addmachinestosome [-sc sc_url] -service_name servicename
-machine_list machinelist [-ws_user username -ws_password password][-locale
iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例：`80`

**-service\_name servicename**

定义服务的名称。有效条目：服务名称的完整路径。请使用反斜线 \ 分隔父服务和子服务。

**-machine\_list machinelist**

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：向服务中添加服务器**

本示例向名为 `SERVICEA` 的服务中添加一组服务器。

```
dpmresourcemgr addmachinestosome -service_name SERVICEA -machine_list  
"machine4.MyCompany.com, machine5.MyCompany.com, 192.168.255.255" -ws_user  
wsuser -ws_password wsuserpassword
```

## **dpmresourcemgr addtocontainer 命令—将内容添加到容器中**

`dpmresourcemgr addtocontainer` 命令用于将服务器或子容器添加到容器中。

此命令具有以下格式：

```
dpmresourcemgr addtocontainer [-sc sc_url] -container_name containername  
[-machine_list machinelist/-subcontainer_name subcontainername] [-ws_user  
username -ws_password password][-locale iso639value]
```

**-container\_name containername**

定义对象的集合。有效条目：容器名称的完整路径，使用反斜杠 (\) 分隔父容器和子容器。

**-machine\_list machinelist | -subcontainer subcontainername**

定义计算机列表或子容器及其对象。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：将内容添加到容器中**

本示例将名为 ManagedVMs 的子容器添加到名为 Managed 的容器中。

```
dpmresourcemgr addtocontainer -container_name Managed -subcontainer_name ManagedVMs.
```

## dpmresourcemgr createcontainer 命令—创建容器

createcontainer 命令用于创建一个对象来放置其他对象的集合。

此命令具有以下格式：

```
dpmresourcemgr createcontainer [-sc sc_url] -parent_container parentcontainer
-container_name containername -machine_list machinelist [-ws_user username
-ws_password password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-parent\_container *parentcontainer***

定义父服务的完整路径服务名称。

**-container\_name *containername***

定义对象的集合。有效条目：容器名称的完整路径，使用反斜杠 (\) 分隔父容器和子容器。

**-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

#### 示例：创建容器

本示例创建名为 `ManagedVMs` 的容器。

```
dpmresourcemgr createcontainer -parent_container Managed -container_name ManagedVMs
```

## dpmresourcemgr createdefaultservice 命令-创建默认服务

`createdefaultservice` 命令使用在安装期间配置的本地系统和任何外部系统创建默认服务。

此命令具有以下格式：

```
dpmresourcemgr createdefaultservice [-sc sc_url] [-retry retrynumber]
[-discover_local_subnet] [-ws_user username -ws_password password] [-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例：`80`

**-retry retrynumber**

定义创建服务的重试次数。

**-discover\_local\_subnet**

在创建默认服务（如果指定）后自动发现本地子网。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：创建默认服务**

本示例创建尝试连接十次的默认服务，并且会自动发现本地子网。

```
dpmresourcemgr createdefaultservice -retry 10 -discover_local_subnet -ws_user
wsuser -ws_password wsuserpassword
```

**dpmresourcemgr createipnetwork 命令-创建网络**

`dpmresourcemgr createipnetwork` 命令用于指定数据中心中的现有网络以与保留管理器共用。保留管理器最终用户然后便可以在特定网络上保留虚拟机。

此命令具有以下格式：

```
dpmresourcemgr createipnetwork [-sc sc_url] -subnet_address subnetaddress
-vlan_id vlanid -ip_netmask ipnetmask -name networkname -ip_gateway_list
ipaddresses -dns_domainname domainname [-description description] [-dns_list
ipaddresses][ -dns_suffix_list suffixes][ -netbios_server_list
ipaddresses][ -ntp_server_list ipaddresses][ -ws_user username -ws_password
password][ -locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例：`80`

**-subnet\_address address**

定义网络的 IP 地址。有效条目：`xxx.xxx.xxx.xxx` 或 `xxx.xxx.xxx.xxx/xx`。

**-vlan\_id *vlanid***

为虚拟网络 (VLAN) 定义整数。有效条目：整数 1-4094，其中 1 是默认网络接口卡 (NIC)。

**-ip\_mask *ipnetmask***

定义子网掩码。有效条目：格式为 xxx.xxx.xxx.x 或 CIDR 格式为 xxx.xxx.xxx.x/8-28 的标准 IP 网络掩码。前缀长度为 8-28。

**-name *name***

定义项的名称。有效条目：1-128 个字母数字字符。

**-ip\_gateway\_list *ipaddresses***

使用逗号分隔列表定义默认网关和备用网关的 IP 地址。默认网关和备用网关必须位于同一网络上。有效条目：xxx.xxx.xxx.xxx。

**-dns\_domainname *domainname***

定义网络的 DNS 域名。有效条目：每个点分隔地址 1-63 个字母数字字符（例如，xxx.yyy.com）。地址不能以数字开头。

**-description *description***

定义网络，以便于用户理解。有效条目：1-128 个字母数字字符。

**-dns\_list *ipaddresses***

定义首选 DNS 服务器和备用 DNS 服务器以及主 Windows Internet 名称服务 (WINS) 服务器和辅助 Windows Internet 名称服务 (WINS) 服务器的 DNS 列表。有效条目：以逗号分隔的 IP 地址（格式为 xxx.xxx.xxx.xxx）。

**-dns\_suffix\_list *suffixes***

定义网络的 DNS 后缀的列表。有效条目：以逗号分隔的列表（格式为 name.name.com）。

**-netbios\_server\_list *ipaddresses***

定义 NETBIOS 名称服务器 IP 地址的列表。有效条目：以逗号分隔的 IP 地址列表（格式为 xxx.xxx.xxx.xxx）。

**-ntp\_server\_list *ipaddresses***

定义网络时间协议 (NTP) 服务器。有效条目：以逗号分隔的列表（格式为 xxx.xxx.xxx.xxx）。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。



**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：创建网络**

本示例指定现有的数据中心网络 san diego 209 subnet，以供保留管理器最终用户门户在保留虚拟机时使用。

```
dpmresourcemgr createipnetwork -subnet_address 10.10.21.0 -vlan_id 4001 -ip_mask 255.255.255.0 -name "san diego 209 subnet" -ip_gateway_list 10.10.21.124,10.10.21.125 -dns_domainname turbo.com -ws_user wsuser -ws_password wsuserpassword
```

**dpmresourcemgr createresourcegroup 命令—创建资源组**

createresourcegroup 命令用于创建软件包组。

此命令具有以下格式：

```
dpmresourcemgr createresourcegroup [-sc sc_url] -group_name groupname -group_description description -group_type type -group_attribute attribute -package_list packagelist [-ws_user username -ws_password password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

示例：80

**-group\_name *groupname***

定义组的名称。

**-group\_description *description***

定义组的说明。

**-group\_type *type***

定义资源组的类型。对于打包组，设置为 1。对于所有组，设置为零。

**-group\_attribute *attribute***

定义组属性。

**-package\_list *packagelist***

定义各个软件包。

**示例：** -package\_list "<package name A>|<package procedure A>|<procedure type A>|<package name B>|...".

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：创建资源组**

本示例创建资源组。

```
dpmresourcemgr createresourcegroup -group_name "AGENT PACKAGE GROUP"
-group_description "Agent package group" -group_type 1 -group_attribute 0
-package_list "CA Product - Win r11.2|Install|INSTALL|Generic Agent Win32
r5.0|Install|INSTALL"
```

## **dpmresourcemgr createresourcetemplate 命令—创建服务器组模板**

`createresourcetemplate` 命令用于创建软件包模板。

此命令具有以下格式：

```
dpmresourcemgr createresourcetemplate [-sc sc_url] -template_name templatename
-template_description description -template_os_type OStype [-group_list
grouplist] [-package_list packagelist][-ws_user username -ws_password
password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-template\_name *templatename***

定义模板的名称。

**-template\_description *description***

定义模板说明。

**-template\_os\_type *OSType***

定义部署的资源模板的 OS 类型。

**-group\_list *grouplist***

定义要添加到资源模板的组的列表。有效条目: `-group_list "<group name A>|<group type 1=Software Package Group>|<group name B>|... "`

**-package\_list *packagelist***

定义各个软件包。如果组类型设置为 1(PACKAGE)，则定义该选项。

有效条目: `-package_list "<package name A>|<package procedure A>|<procedure type A>|<package name B>|... "`

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：创建资源模板

本示例创建软件包模板。

```
dpmresourcemgr createresourcetemplate -template_name "Generic template"
-template_description "Generic template description" -template_os_type 0
-group_list "Package group A|1|Package group B|1" -package_list "CAProduct - Win
r11.2|Install|INSTALL|Generic Agent Win32 r5.0|Install|INSTALL"
```

## dpmresourcemgr createservice 命令-创建服务

createservice 命令用于创建服务。服务可以拥有多个子服务。

此命令具有以下格式：

```
dpmresourcemgr createservice [-sc sc_url] -service_name servicename
[-machine_list machinelist] [-lower_threshold lowerthreshold] [-upper_threshold
upperthreshold] [-lag lag] [-priority priority] [-ws_user username -ws_password
password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-service\_name *servicename***

定义服务的名称。有效条目：服务名称的完整路径。请使用反斜线 \ 分隔父服务和子服务。

### **-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

### **-lower\_threshold *lowerthreshold***

（可选）指定服务的总体使用率阈值下限。阈值下限必须小于阈值上限。有效条目：1-100 (%)

### **-upper\_threshold *upperthreshold***

（可选）指定服务的总体使用率阈值上限。阈值上限必须始终大于阈值下限。有效条目：1-100 (%)

**-lag lagvalue**

（可选）指定延迟值。有效条目：0 或更大（秒）。

**-priority priorityvalue**

（可选）指定服务的优先级。有效条目：1 或更高。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：创建服务**

本示例创建名为 `SERVICE1` 的服务。

```
dpmresourcemgr createservice -service_name SERVICE1 -machine_list
"ComputerList.MyCompany.com, 172.31.255.255, Computer1.MyCompany.com"
-lower_threshold 20 -upper_threshold 80 -lag 1 -priority 1 -ws_user wsuser
-ws_password wsuserpassword
```

**dpmresourcemgr deletebaremetal 命令-删除物理计算机**

`deletebaremetal` 命令用于从数据中心删除物理计算机。

此命令具有以下格式：

```
dpmresourcemgr deletebaremetal [-sc sc_url] -server_name hostname -MAC_address
macaddress [-ws_user username -ws_password password][-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例：`80`

**-host\_name source host name**

定义正在运行的源服务器的名称。

**-MAC\_address macaddress**

定义目标计算机的 MAC 地址。有效条目：12 位十六进制值（例如，A-F、0-9、a-f），格式如下：0013724C2140 或 00:13:72:4C:21:40。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：删除物理计算机**

本示例从数据中心删除 EngServer1 服务器。

```
dpmresourcemgr deletebaremetal -server EngServer1 -MAC_address 11:22:33:44:55:66  
-ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr deletecontainer 命令—删除容器

`deletecontainer` 命令用于删除容器及其内容。

此命令具有以下格式：

```
dpmresourcemgr deletecontainer [-sc sc_url] -container_name containername  
[-recursive] [-ws_user username -ws_password password][-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

**示例：**`80`

**-container\_name containername**

定义对象的集合。有效条目：容器名称的完整路径，使用反斜杠 (\) 分隔父容器和子容器。

**-recursive**

如果设置了该选项，将对子服务或子容器递归运行该命令。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：删除容器**

本示例创建名为 `ManagedVMs` 的容器。

```
dpmresourcemgr deletecontainer -container_name ManagedVMs.
```

**dpmresourcemgr deleteipnetworks 命令-删除网络**

`dpmresourcemgr deleteipnetworks` 命令用于从 CA Virtual Assurance 中删除网络，但仅当没有当前或现有保留使用该网络时才执行此操作。

此命令具有以下格式：

```
dpmresourcemgr deleteipnetworks [-sc sc_url] -subnet_address_list subnetaddress
[-ws_user username -ws_password password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例：`80`

**-subnet\_address\_list *addresses***

按 IP 地址定义网络列表。有效条目：逗号分隔的 IP 地址，格式为 `xxx.xxx.xxx.xxx` 或 `xxx.xxx.xxx.xxx/xx`。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

#### 示例：删除网络

本示例删除供 CA Virtual Assurance 使用的 san diego 209 subnet。

```
dpmresourcemgr deleteipnetworks -subnet_address_list 10.10.21.0,10.10.21.130  
-ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr deletesystem 命令-删除系统

`deletesystem` 命令用于从数据中心删除服务器。

此命令具有以下格式：

```
dpmresourcemgr deletesystem [-sc sc_url] -machine_list machinelist [-post]  
[-ws_user username -ws_password password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例：`80`

**-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

**-post**

（可选）指定是否在执行操作之后生成事件。



**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：删除系统**

本示例删除名为 ServerB 的服务器。

```
dpmresourcemgr deletesystem -machine_list "machine4.MyCompany.com,
machine5.MyCompany.com, 192.168.255.255" -ws_user wsuser -ws_password
wsuserpassword
```

**dpmresourcemgr getdiscoverednetworks 命令-获取发现的网络**

`getdiscoverednetworks` 命令用于检索所有发现的网络的列表。

此命令具有以下格式：

```
dpmresourcemgr getdiscoverednetworks [-sc sc_url] [-ws_user username -ws_password
password] [-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

**示例：**`80`

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

**示例：检索所有网络**

本示例检索所有发现的网络的列表。

```
dpmresourcemgr getdiscoverednetworks -ws_user wsuser -ws_password wsuserpassword
```

**dpmresourcemgr getipaddresspools 命令-列出 IP 地址池**

dpmresourcemgr getipaddresspools 命令用于按所有者或分配类型列出现有的 IP 地址池。

此命令具有以下格式：

```
dpmresourcemgr getipaddresspools [-sc sc_url] -subnet_address subnetaddress  
[-owner ownername] [-ws_user username -ws_password password] [-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-subnet\_address *address***

定义网络的 IP 地址。有效条目：xxx.xxx.xxx.xxx 或 xxx.xxx.xxx.xxx/xx。

**-owner *ownername***

定义 IP 地址池的所有者。有效条目：1-128 个字母数字字符。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：更新 IP 地址池**

本示例列出 IP 地址为 10.10.21.30 的网络的现有 IP 地址池。

```
dpmresourcemgr getipaddresspools -subnet_address 10.10.21.130 -owner Reservation
Manager -ws_user wsuser -ws_password wsuserpassword
```

**dpmresourcemgr getipnetworks 命令-列出网络**

dpmresourcemgr getipnetworks 命令用于列出使用 createipnetwork 命令在 CA Virtual Assurance 中指定的网络。

此命令具有以下格式：

```
dpmresourcemgr getipnetworks [-sc sc_url][-ws_user username -ws_password
password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

***hostname***

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

***port***

定义服务控制器服务器的侦听端口。

**示例：**80

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：列出网络**

本示例列出供 CA Virtual Assurance 使用的网络。

```
dpmresourcemgr getipnetworks -ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr getserverlist 命令-获取服务器的列表

getserverlist 命令用于从数据中心检索服务器的列表。

此命令具有以下格式：

```
dpmresourcemgr getserverlist [-sc sc_url] -service_name servicename [-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-service\_name *servicename***

定义服务的名称。有效条目：服务名称的完整路径。请使用反斜线 \ 分隔父服务和子服务。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：获取服务器的列表

本示例检索属于服务 SERVICE2 的服务器的列表。

```
dpmresourcemgr getserverlist -service_name SERVICE2 -ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr getservicelist 命令-获取服务的列表

getserverlist 命令用于从服务器检索服务的列表。

此命令具有以下格式：

```
dpmresourcemgr getservicelist [-sc sc_url][-parent_service_name
parentservice][-server_name servername][-recursive][-ws_user username
-ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

### ***port***

定义服务控制器服务器的侦听端口。

示例：`80`

### ***-parent\_service\_name parentservice***

定义父服务。有效条目：父服务的完整路径服务。

### ***-server\_name servername***

定义服务器的名称。如果不指定，该命令将返回所有已定义服务的列表。

### ***-recursive***

如果已设置，该函数将递归式执行到子服务。

### ***-ws\_user username -ws\_password password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### ***-locale iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：获取服务的列表

本示例检索属于父服务 `SERVICE2` 和服务器 `EngServer1` 的服务的列表。

```
dpmresourcemgr getservicelist -parent service_name SERVICE2 -server_name
EngServer1 -ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr getsubnetdefaults 命令-列出子网默认值

dpmresourcemgr getsubnetdefaults 命令用于列出当前网络默认值。

此命令具有以下格式：

```
dpmresourcemgr getsubnetdefaults [-sc sc_url] -subnet_address address [-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### ***port***

定义服务控制器服务器的侦听端口。

示例：80

### **-subnet\_address *address***

定义网络的 IP 地址。有效条目：xxx.xxx.xxx.xxx 或 xxx.xxx.xxx.xxx/xx。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：列出网络默认值

本示例更新 IP 地址为 10.10.21.130 的网络的默认值。

```
dpmresourcemgr getsubnetdefaults -subnet_address 10.10.21.130 -ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr removecomputerfromipnetwork 命令-从网络中删除服务器

dpmresourcemgr removecomputerfromipnetwork 用于从网络中删除服务器。

此命令具有以下格式：

```
dpmresourcemgr removecomputerfromipnetwork [-sc sc_url] -vlan_id vlanid  
-mac_list macaddresses [-ws_user username -ws_password password][-locale  
iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-vlan\_id *vlanid***

为虚拟网络 (VLAN) 定义整数。有效条目：整数 1-4094，其中 1 是默认网络接口卡 (NIC)。

### **-mac\_list *macaddresses***

定义服务器的 NIC MAC 地址的列表。有效条目：以逗号分隔的 NIC MAC 地址（格式为 xx:xx:xx:xx:xx:xx）。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：从网络中删除服务器

本示例从 VLAN 4556 中删除服务器 11:22:33:44:55:66。

```
dpmresourcemgr removecomputerfromipnetwork -vlan 4556 -mac_list  
11:22:33:44:55:66 -ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr removefromcontainer 命令—从容器中删除内容

`removefromcontainer` 命令用于从容器中删除计算机列表或子容器。

此命令具有以下格式：

```
dpmresourcemgr removefromcontainer [-sc sc_url] -container_name containername
[-machine_list machinelist /-subcontainer_name subcontainername] [-ws_user
username -ws_password password] [-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

### ***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

### ***port***

定义服务控制器服务器的侦听端口。

示例：`80`

### **-container\_name *containername***

定义对象的集合。有效条目：容器名称的完整路径，使用反斜杠 (\) 分隔父容器和子容器。

### **-machine\_list *machinelist* | -subcontainer *subcontainername***

定义计算机列表或子容器及其对象。有效条目：子容器的完整路径容器。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：从容器中删除内容

本示例从容器 `Managed` 中删除名为 `ManagedVMs` 的子容器。

```
dpmresourcemgr removefromcontainer -container_name Managed -subcontainer_name
ManagedVMs.
```



## dpmresourcemgr removeipaddresspool 命令-删除 IP 地址池

dpmresourcemgr removeipaddresspool 命令用于删除 IP 地址池。

此命令具有以下格式：

```
dpmresourcemgr removeipaddresspool [-sc sc_url] -name name|-starting_ip  
ipaddress [-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-name *name* | -starting\_ip *ipaddress***

定义名称或起始 IP 地址范围。有效条目：1-128 个字母数字字符或  
xxx.xxx.xxx.xxx。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：删除 IP 地址池

本示例删除以地址 10.10.21.130 开头的 IP 地址池。

```
dpmresourcemgr removeipaddresspool -starting_ip 10.10.21.130 -ws_user wsuser  
-ws_password wsuserpassword
```

## dpmresourcemgr removemachinesfromservice 命令-从服务中删除服务器

removemachinesfromservice 命令用于从服务中删除服务器。

此命令具有以下格式：

```
dpmresourcemgr removemachinesfromservice [-sc sc_url] -service_name servicename  
-machine_list machinelist [-ws_user username -ws_password password][-locale  
iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-service\_name *servicename***

定义服务的名称。有效条目：服务名称的完整路径。请使用反斜线 \ 分隔父服务和子服务。

### **-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：从服务中删除服务器

本示例从服务 SERVICE2 中删除两个服务器。

```
dpmresourcemgr removemachinesfromservice -service_name SERVICE2 -machine_list  
"machine1.MyCompany.com, machine2.MyCompany.com" -ws_user wsuser -ws_password  
wsuserpassword
```

## dpmresourcemgr removeresourcegroup 命令—删除资源组

removeresourcegroup 用于命令删除资源组。

此命令具有以下格式：

```
dpmresourcemgr removeresourcegroup [-sc sc_url] -group_name groupname  
-group_type grouptype [-ws_user username -ws_password password][-locale  
iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-group\_name *groupname***

定义组的名称。

### **-group\_type *type***

定义资源组的类型。对于打包组，设置为 1。对于所有组，设置为零。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

### 示例：删除资源组

本示例删除名为 Agent Package Group 的资源组。

```
dpmresourcemgr removeresourcegroup -group_name "AGENT PACKAGE GROUP" -group_type  
1
```

## dpmresourcemgr removeservice 命令-删除服务

removeservice 命令用于从数据中心删除服务。

此命令具有以下格式：

```
dpmresourcemgr removeservice [-sc sc_url] -service_name servicename  
[-recursive][-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-service\_name *servicename***

定义服务的名称。有效条目：服务名称的完整路径。请使用反斜线 \ 分隔父服务和子服务。

### **-recursive**

如果已设置，该命令将递归式执行到子服务。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “native”。

### 示例：删除服务

本示例删除名为 TESTGROUP 的服务。

```
dpmresourcemgr removeservice -service_name TESTGROUP -ws_user wsuser  
-ws_password wsuserpassword
```

## dpmresourcemgr removeresourcetemplate 命令—删除资源模板

removeresourcetemplate 命令用于删除软件包模板。

此命令具有以下格式：

```
dpmresourcemgr removeresourcetemplate [-sc sc_url] -template_name templatename  
-template_os_type OStype [-ws_user username -ws_password password][-locale  
iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-template\_name *templatename***

定义模板的名称。

### **-template\_os\_type *OStype***

定义部署的资源模板的 OS 类型。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“native”。

### 示例：删除资源模板

本示例删除名为 Generic template 的软件包模板。

```
dpmresourcemgr removeresourcetemplate -template_name "Generic template"  
-template_os_type 0
```

## dpmresourcemgr retrieveresourcegroup 命令—检索资源组

retrieveresourcegroup 命令用于检索软件包组。

此命令具有以下格式：

```
dpmresourcemgr retrieveresourcegroup [-sc sc_url] [-group_name groupname]  
-group_type type [-ws_user username -ws_password password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-group\_name *groupname***

定义组的名称。

### **-group\_type *type***

定义资源组的类型。对于打包组，设置为 1。对于所有组，设置为零。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“native”。

### 示例：检索资源组

本示例检索任何资源类型的所有软件包组。

```
dpmresourcemgr retrieveresourcegroup -group_type 0
```

### 示例：检索所有软件包组

本示例检索所有软件包组信息。

```
dpmresourcemgr retrieveresourcegroup -group_type 1
```

### 示例：检索特定软件包组

本示例检索特定软件包组信息。

```
dpmresourcemgr retrieveresourcegroup -group_name "AGENT PACKAGE GROUP"
-group_type 1
```

## dpmresourcemgr retrieveresourcetemplate 命令—检索资源模板

retrieveresourcetemplate 命令用于检索软件包模板。

此命令具有以下格式：

```
dpmresourcemgr retrieveresourcetemplate [-sc sc_url] [-template_name
templatename] -template_os_type OStype [-ws_user username -ws_password
password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-template\_name *templatename***

定义模板的名称。

### **-template\_os\_type *OStype***

定义部署的资源模板的 OS 类型。对于所有 OS 类型，设置为 -1。

### **-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

### **-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：检索资源模板**

本示例检索任何 OS 类型的软件包模板信息。

```
dpmresourcemgr retrieveresourcetemplate -template_name "Generic template"
-template_os_type -1
```

**dpmresourcemgr setcontainername 命令—设置容器名称**

setcontainername 命令用于更改现有容器的名称。

此命令具有以下格式：

```
dpmresourcemgr setcontainername [-sc sc_url] -container_name containername
-container_new_name containernewname [-ws_user username -ws_password
password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-container\_name *containername***

定义对象的集合。有效条目：容器名称的完整路径，使用反斜杠 (\) 分隔父容器和子容器。

**-container\_new\_name *containernewname***

定义容器的新名称。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。



### 示例：设置容器名称

本示例将容器的名称从 ManagedVM1 更改为 ManagedVM3。

```
dpmresourcemgr setcontainername -container_name ManagedVM -container_new_name ManagedVM3。
```

## dpmresourcemgr setcontainerservice 命令—将容器设置为服务

setcontainerservice 命令用于将容器转换为服务。

此命令具有以下格式：

```
dpmresourcemgr setcontainerservice [-sc sc_url] -container_name containername  
-machine_list machinelist -lower_threshold lowerthreshold -upper_threshold  
upperthreshold -lag lag -priority priority [-ws_user username -ws_password  
password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

### **-lower\_threshold *lowerthreshold***

（可选）指定服务的总体使用率阈值下限。阈值下限必须小于阈值上限。有效条目：1-100 (%)

### **-upper\_threshold *upperthreshold***

（可选）指定服务的总体使用率阈值上限。阈值上限必须始终大于阈值下限。有效条目：1-100 (%)

### **-lag *lagvalue***

（可选）指定延迟值。有效条目：0 或更大（秒）。

### **-priority *priorityvalue***

（可选）指定服务的优先级。有效条目：1 或更高。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：将容器转换为服务**

本示例将名为 `Managed1` 的容器转换为服务。

```
dpmresourcemgr setcontainerservice -container_name Managed1 -machine_list
"ComputerList.MyCompany.com, 172.31.255.255, Computer1.MyCompany.com"
-lower_threshold 20 -upper_threshold 80 -lag 1 -priority 1
```

## **dpmresourcemgr setmanagementip 命令--设置主机的管理 IP**

`dpmresourcemgr setmanagementip` 命令用于设置主机的管理 IP 地址。

**注意：**如果存在服务器名称参数的任何重复的 *主机名*，控制台消息将改为请求使用 `system_uuid` 参数。发生此情况时，您可以从数据库表 `[aom2].[dbo].[_CA_ComputerSystem]` 的 `[_Name]` 列的相应记录获取 UUID；或者可从 `"...\productname\bin\caaipaomwsclient.exe /enum=CA_ComputerSystem"` 输出结果中获取。

此命令具有以下格式：

```
dpmresourcemgr setmanagementip [-sc sc_url] -server_name hostname | -system_uuid
uuid -ip_address ipaddress [-ws_user username -ws_password password][-locale
iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

**示例：**`80`

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

## dpmresourcemgr setmanagementstatus 命令-设置主机的管理状态

`setmanagementstatus` 命令用于将主机设置为受管或未受管状态。

此命令具有以下格式：

```
dpmresourcemgr setmanagementstatus [-sc sc_url] -server_name hostname
-management_status 0|1 [-ws_user username -ws_password password][-locale
iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例：`80`

**-host\_name source host name**

定义正在运行的源服务器的名称。

**-management\_status 0 | 1**

定义服务器的管理状态。有效条目：`0` = 未受管状态；`1` = 受管状态。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：将主机设置为受管状态**

本示例将名为 **ServerA** 的主机的管理状态设置为受管状态。

```
dpmcda setmanagementstatus -sc https://localhost/dpm/sc -host_name ServerA
-management_status 1
```

**示例：将主机设置为未受管状态**

使用该命令可以将名为 **ServerB** 的主机的管理状态设置为未受管状态。

```
dpmcda setmanagementstatus -host_name ServerB -management_status 0 -ws_user
wsuser -ws_password wsuserpassword
```

## **dpmresourcemgr setserviceproperties 命令-设置服务属性**

**setserviceproperties** 命令用于创建服务属性。

此命令具有以下格式：

```
dpmresourcemgr setserviceproperties [-sc sc_url] -servicename servicename
-lower_threshold lowerthreshold -upper_threshold upperthreshold -lag lag
-priority priority [-ws_user username -ws_password password][-locale
iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-service\_name *servicename***

定义服务的名称。有效条目：服务名称的完整路径。请使用反斜线 \ 分隔父服务和子服务。

**-lower\_threshold *lowerthreshold***

（可选）指定服务的总体使用率阈值下限。阈值下限必须小于阈值上限。有效条目：1-100 (%)

**-upper\_threshold *upperthreshold***

（可选）指定服务的总体使用率阈值上限。阈值上限必须始终大于阈值下限。有效条目：1-100 (%)

**-lag *lagvalue***

（可选）指定延迟值。有效条目：0 或更大（秒）。

**-priority *priorityvalue***

（可选）指定服务的优先级。有效条目：1 或更高。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

**示例：设置服务属性**

在本示例中，为服务 `Service12` 设置服务属性。

```
dpmresourcemgr setserviceproperties -servicename Service12 -lower_threshold 2
-upper_threshold 4 -log 2 -priority 2 -ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr updateresourcegroup 命令—更新资源组

`updateresourcegroup` 命令用于更新软件包组。

此命令具有以下格式：

```
dpmresourcemgr updateresourcegroup [-sc sc_url] -group_name groupname
-group_description description -group_type group_type -group_attribute attribute
[-package_list packagelist][-ws_user username -ws_password password][-locale
iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例：`80`

**-group\_name *groupname***

定义组的名称。

**-group\_description *description***

定义组的说明。

**-group\_type *type***

定义资源组的类型。对于打包组，设置为 1。对于所有组，设置为零。

**-group\_attribute *attribute***

定义组属性。

**-package\_list *packagelist***

定义各个软件包。如果组类型设置为 1(PACKAGE)，则定义该选项。

有效条目：-package\_list "<package name A>|<package procedure A>|<procedure type A>|<package name B>|... "。

**-ws\_user *username* -ws\_password *password***

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定“native”。

**示例：更新资源组**

本示例更新资源组。

```
dpmresourcemgr updateresourcegroup -group_name "AGENT PACKAGE GROUP"
-group_description "Agent package group" -group_type 1 -group_attribute 0
-package_list "Generic Agent Win32 r5.0|install|INSTALL"
```

## dpmresourcemgr updateresourcetemplate 命令—更新资源模板

updateresourcetemplate 命令用于更新软件包模板。

此命令具有以下格式：

```
dpmresourcemgr updateresourcetemplate [-sc sc_url] -template_name templatename
-template_description templatedescription -template_os_type OSType [-group_list
grouplist][-package_list package_list][-ws_user username -ws_password
password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例: `https://hostname:port/...`

***hostname***

定义安装了服务控制器的完全限定主机。

示例: `localhost.ca.com`

***port***

定义服务控制器服务器的侦听端口。

示例: `80`

**-template\_name *templatename***

定义模板的名称。

**-template\_description *description***

定义模板说明。

**-template\_os\_type *OSType***

定义部署的资源模板的 OS 类型。

**-group\_list *grouplist***

定义要添加到资源模板的组的列表。有效条目: `-group_list "<group name A>|<group type 1=Software Package Group>|<group name B>|..."`

**-package\_list *packagelist***

定义各个软件包。如果组类型设置为 1(PACKAGE)，则定义该选项。

有效条目: `-package_list "<package name A>|<package procedure A>|<procedure type A>|<package name B>|..."`

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定“`native`”。

### 示例：更新软件包模板

本示例更新软件包模板。

```
dpmresourcemgr updateresourcetemplate -template_name "Generic template"
-template_description "Generic template description" -template_os_type 0
-group_list "Package group A|1|Package group B|1" -package_list "Generic Agent
Win32 r5.0|Install|INSTALL
```

## dpmresourcemgr updateservice 命令-更新服务

updateservice 命令用于更改服务特征。

此命令具有以下格式：

```
dpmresourcemgr updateservice [-sc sc_url] -service_name servicename
-machine_list machinelist -lower_threshold lowerthreshold -upper_threshold
upperthreshold -lag lag -priority priority [-ws_user username -ws_password
password][-locale iso639value]
```

### **-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

### **hostname**

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

### **port**

定义服务控制器服务器的侦听端口。

示例：80

### **-service\_name *servicename***

定义服务的名称。

### **-machine\_list *machinelist***

（可选）指定可用服务器的列表。有效条目：以逗号分隔的计算机名称。

### **-lower\_threshold *lowerthreshold***

（可选）指定服务的总体使用率阈值下限。阈值下限必须小于阈值上限。有效条目：1-100 (%)

### **-upper\_threshold *upperthreshold***

（可选）指定服务的总体使用率阈值上限。阈值上限必须始终大于阈值下限。有效条目：1-100 (%)



**-lag lagvalue**

（可选）指定延迟值。有效条目：0 或更大（秒）。

**-priority priorityvalue**

（可选）指定服务的优先级。有效条目：1 或更高。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 `fr_FR`）。要使用命令提示符的区域设置，请指定 “`native`”。

**示例：更新服务**

本示例更新服务 `TESTGROUP1` 的阈值、延迟和优先级。

```
dpmresourcemgr updateservice -service_name TESTGROUP1 -machine_list
"machine4.MyCompany.com, machine5.MyCompany.com, 192.168.255.255"
-lower_threshold 10 -upper_threshold 85 -lag 0 -priority 2 -ws_user wsuser
-ws_password wsuserpassword
```

**dpmresourcemgr updateipnetwork 命令-更新网络**

`dpmresourcemgr updateipnetwork` 命令用于更新现有网络的参数。

此命令具有以下格式：

```
dpmresourcemgr updateipnetwork [-sc sc_url] -subnet_address subnetaddress
[-vlan_id vlanid] [-name name] [-description description] [-ws_user username
-ws_password password][-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

示例：`https://hostname:port/...`

**hostname**

定义安装了服务控制器的完全限定主机。

示例：`localhost.ca.com`

**port**

定义服务控制器服务器的侦听端口。

示例：`80`

**-subnet\_address address**

定义网络的 IP 地址。有效条目：`xxx.xxx.xxx.xxx` 或 `xxx.xxx.xxx.xxx/xx`。

**-vlan\_id vlanid**

为虚拟网络 (VLAN) 定义整数。有效条目：整数 1-4094，其中 1 是默认网络接口卡 (NIC)。

**-name name**

定义项的名称。有效条目：1-128 个字母数字字符。

**-description description**

定义网络，以便于用户理解。有效条目：1-128 个字母数字字符。

**-ws\_user username -ws\_password password**

(可选) 指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 `caaipsecurity` 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

(可选) 指定 ISO 639\_3166 组合以覆盖默认的英语输出 (例如：法语为 `fr_FR`)。要使用命令提示符的区域设置，请指定 “native”。

**示例：更新网络**

本示例向网络 `san diego 209 subnet` 中添加说明。

```
dpmresourcemgr updateipnetwork -subnet_name 10.10.21.130 -vlan_id 4001 -name "san diego 209 subnet" -description "5th floor lab" -ws_user wsuser -ws_password wsuserpassword
```

## **dpmresourcemgr updatesubnetdefaults 命令-更新网络默认值**

`dpmresourcemgr updatesubnetdefaults` 命令用于更新现有网络默认值。

此命令具有以下格式：

```
dpmresourcemgr updatesubnetdefaults [-sc sc_url] -subnet_address subnetaddress  
-ip_gateway_list ipaddresses -dns_domainname domainname -dns_list ipaddresses  
-dns_suffix_list suffixes -netbios_server_list ipaddresses -ntp_server_list  
ipaddresses [-ws_user username -ws_password password][-locale iso639value]
```

**-sc *sc\_url***

（可选）为服务控制器主机定义 URL。

示例：https://hostname:port/...

***hostname***

定义安装了服务控制器的完全限定主机。

示例：localhost.ca.com

***port***

定义服务控制器服务器的侦听端口。

示例：80

**-subnet\_address *address***

定义网络的 IP 地址。有效条目：xxx.xxx.xxx.xxx 或 xxx.xxx.xxx.xxx/xx。

**-ip\_gateway\_list *ipaddresses***

使用逗号分隔列表定义默认网关和备用网关的 IP 地址。默认网关和备用网关必须位于同一网络上。有效条目：xxx.xxx.xxx.xxx。

**-dns\_domainname *domainname***

定义网络的 DNS 域名。有效条目：每个点分隔地址 1-63 个字母数字字符（例如，xxx.yyy.com）。地址不能以数字开头。

**-dns\_list *ipaddresses***

定义首选 DNS 服务器和备用 DNS 服务器以及主 Windows Internet 名称服务 (WINS) 服务器和辅助 Windows Internet 名称服务 (WINS) 服务器的 DNS 列表。有效条目：以逗号分隔的 IP 地址（格式为 xxx.xxx.xxx.xxx）。

**-dns\_suffix\_list *suffixes***

定义网络的 DNS 后缀的列表。有效条目：以逗号分隔的列表（格式为 name.name.com）。

**-netbios\_server\_list *ipaddresses***

定义 NETBIOS 名称服务器 IP 地址的列表。有效条目：以逗号分隔的 IP 地址列表（格式为 xxx.xxx.xxx.xxx）。

**-ntp\_server\_list *ipaddresses***

定义网络时间协议 (NTP) 服务器。有效条目：以逗号分隔的列表（格式为 xxx.xxx.xxx.xxx）。

**-ws\_user *username* -ws\_password *password***

（可选）指定要用于 Web 服务安全检查和凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale iso639value**

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：更新网络默认值**

本示例更新网络 san diego 209 subnet 的默认值。

```
dpmresourcemgr updatesubnetdefaults -subnet_address 10.10.21.130  
-ip_gateway_list 10.10.33.0,10.10.34.0 -dns_domainname turbo.com -dns_list  
10.10.55.21 -netbios_server_list 10.10.21.130 -ntp_server_list 10.10.33.135  
-ws_user wsuser -ws_password wsuserpassword
```

## dpmresourcemgr updateipaddresspool 命令-更新 IP 地址池

dpmresourcemgr updateipaddresspool 命令用于编辑对现有池进行 IP 分配的方法。

此命令具有以下格式：

```
dpmresourcemgr updateipaddresspool [-sc sc_url] -name name|-starting_ip  
ipaddress -ip_assignment_type static|dhcp [-ws_user username -ws_password  
password][-locale iso639value]
```

**-sc sc\_url**

（可选）为服务控制器主机定义 URL。

**示例：**https://hostname:port/...

**hostname**

定义安装了服务控制器的完全限定主机。

**示例：**localhost.ca.com

**port**

定义服务控制器服务器的侦听端口。

**示例：**80

**-name name | -starting\_ip ipaddress**

定义名称或起始 IP 地址范围。有效条目：1-128 个字母数字字符或 xxx.xxx.xxx.xxx。

**-ip\_assignment\_type static | dhcp**

定义服务器获取 IP 地址（静态 IP 地址或 DHCP）的方法。

**-ws\_user username -ws\_password password**

（可选）指定要用于 Web 服务安全认证的凭据。如果不包含凭据，系统会提示您输入凭据。通过使用 caaipsecurity 设置自己的会话来避免提示输入凭据。

**-locale *iso639value***

（可选）指定 ISO 639\_3166 组合以覆盖默认的英语输出（例如：法语为 fr\_FR）。要使用命令提示符的区域设置，请指定 “native”。

**示例：更新 IP 地址池**

本示例将以 10.10.21.130 开头的 IP 地址池的 IP 分配方法更改为 DHCP。

```
dpmresourcemgr updateipaddresspool -starting_ip 10.10.21.130 -ip_assignment_type
DHCP -ws_user wsuser -ws_password wsuserpassword
```

## 远程监控命令

远程监控为可以用来使任务实现自动化的 RM AIM 提供了命令行接口 (rmonwatch)。例如，可以使用它将远程监控系统分配给远程监控配置集。

此命令具有以下格式：

```
rmonwatch [options] getVersion | {add|del} configfile | getTable tablename
```

### rmonwatch add 命令-将行添加到 MIB 表中

使用该 add 命令可将行添加到 rmonWbemSysTable 和 rmonWbemCredTable MIB 表中。

该 add 命令可读取包含关于系统和凭据条目的详细信息的配置文件。该文件使用 .ini 格式，其中每部分均指定系统或凭据条目。由于各部分的名称必须是唯一的，因此附加了 \_<n>，其中 <n> 从 1 开始。每部分均包含各个 MIB 表的可配置属性的变量和赋值。

**示例：配置文件示例 (rmonwatch.cf)**

```
[System_1]
SystemName = sys01-01
RowStatus = active
ConfigSet = default
Credentials = test
MaxInstances = 100
[Credentials_1]
Credentials = test
RowStatus = active
UserName = Administrator
Password = IE7vse8zkBzurAwz5880vyTpc9v
```

凭据条目的 Password 属性必须以加密形式指定。使用 CA SystemEDGE se\_enc 实用工具按照以下方式加密密码：

```
se_enc -s -i in.txt -o out.txt
```

将需要加密的密码放到 in.txt 中（不要添加任何新行）。在调用命令之后，从 out.txt 中复制已加密的密码，并将其作为 Password 属性的值。

该 add 命令使用以下语法：

```
rmonwatch [options] add configfile
```

rmonwatch 使用以下选项：

**--version**

显示程序版本号并退出。

**--help**

显示 rmonwatch 帮助

**-h hostname | -h ipAddr**

（可选）指定运行 RM AIM 的 CA SystemEDGE 主机。

**默认值：** localhost

**-p port**

（可选）指定 CA SystemEDGE SNMP 端口。

**默认：** 161

**-c community**

（可选）为 SNMP 版本 1 和 2c 指定 SNMP 读/写团体字符串。

**默认值：** public

示例：使用该 `add` 命令添加在 `rmonwatch.cf` 中定义的表行

```
rmonwatch -c admin add rmonwatch.cf
RMONWBEM AIM Watch 程序, 版本 1.0.0
### 处理凭据条目 "test"
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.12.1.3.4.116.101.115.116 -s Administrator
1.3.6.1.4.1.546.16.22.12.1.3.4.116.101.115.116 Administrator
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.12.1.4.4.116.101.115.116 -s IE7vse8zkBzurAwz5880vyTpc9v=
1.3.6.1.4.1.546.16.22.12.1.4.4.116.101.115.116 IE7vse8zkBzurAwz5880vyTpc9v=
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.12.1.2.4.116.101.115.116 -i 1
1.3.6.1.4.1.546.16.22.12.1.2.4.116.101.115.116 1
### 处理系统条目 "sys01-01"
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.11.1.3.8.115.121.115.48.49.45.48.49 -s "default"
1.3.6.1.4.1.546.16.22.11.1.3.8.115.121.115.48.49.45.48.49 default
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.11.1.4.8.115.121.115.48.49.45.48.49 -s "test"
1.3.6.1.4.1.546.16.22.11.1.4.8.115.121.115.48.49.45.48.49 test
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.11.1.5.8.115.121.115.48.49.45.48.49 -i 1
1.3.6.1.4.1.546.16.22.11.1.5.8.115.121.115.48.49.45.48.49 1
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.11.1.6.8.115.121.115.48.49.45.48.49 -i 100
1.3.6.1.4.1.546.16.22.11.1.6.8.115.121.115.48.49.45.48.49 100
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.11.1.2.8.115.121.115.48.49.45.48.49 -i 1
1.3.6.1.4.1.546.16.22.11.1.2.8.115.121.115.48.49.45.48.49 1
```

## rmonwatch del 命令-从 MIB 表中删除行

使用该 `del` 命令可从 `rmonWbemSysTable` 和 `rmonWbemCredTable` MIB 表中删除行。

该 `del` 命令可读取包含关于系统和凭据条目的详细信息的配置文件。该文件使用 `.ini` 格式，其中每部分均指定系统或凭据条目。由于各部分的名称必须是唯一的，因此附加了 `_<n>`，其中 `<n>` 从 1 开始。每部分均包含各个 MIB 表的可配置属性的变量和赋值。

**示例：配置文件示例 (rmonwatch.cf)**

```
[System_1]
SystemName = sys01-01
RowStatus = active
ConfigSet = default
Credentials = test
MaxInstances = 100
[Credentials_1]
Credentials = test
RowStatus = active
UserName = Administrator
Password = password01
```

该删除命令使用以下语法：

```
rmonwatch [options] del configfile
```

rmonwatch 使用以下选项：

**--version**

显示程序版本号并退出。

**--help**

显示 rmonwatch 帮助

**-h hostname | -h ipAddr**

（可选）指定运行 RM AIM 的 CA SystemEDGE 主机。

**默认值：**localhost

**-p port**

（可选）指定 CA SystemEDGE SNMP 端口。

**默认：**161

**-c community**

（可选）为 SNMP 版本 1 和 2c 指定 SNMP 读/写团体字符串。

**默认值：**public

**示例：使用该 del 命令删除在 rmonwatch.cf 中定义的表行**

```
rmonwatch -c admin del rmonwatch.cf
```

**注意：**删除行仅需要配置文件中索引属性的规格。

RMONWBEM AIM Watch 程序，版本 1.0.0

### 处理凭据条目 “test”

```
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.12.1.2.4.116.101.115.116 -i 6
1.3.6.1.4.1.546.16.22.12.1.2.4.116.101.115.116 6
```



```
### 处理系统条目“sys01-01”  
E:\sysedge\bin\snmpset -h localhost -p 161 -v 1 -c admin -m 0 -t 5 -r 3 -o  
1.3.6.1.4.1.546.16.22.11.1.2.8.115.121.115.48.49.45.48.49 -i 6  
1.3.6.1.4.1.546.16.22.11.1.2.8.115.121.115.48.49.45.48.49 6
```

## rmonwatch getTable 命令-查看 rmonWbem MIB 表

使用该 `getTable` 命令可以查看以下 MIB 表的内容。表名称必须包括以粗体显示的术语（不区分大小写）。

- `rmonWbemCSetTable`
- `rmonWbemSysTable`
- `rmonWbemCredTable`
- `rmonWbemConfTable`
- `rmonWbemQueryTable`
- `rmonWbemInstTable`

该 `getTable` 命令使用以下语法：

```
rmonwatch [options] getTable {cset | sys | cred | conf | query | inst}
```

rmonwatch 使用以下选项：

### **--version**

显示程序版本号并退出。

### **--help**

显示 rmonwatch 帮助

### **-h hostname | -h ipAddr**

（可选）指定运行 RM AIM 的 CA SystemEDGE 主机。

**默认值：**localhost

### **-p port**

（可选）指定 CA SystemEDGE SNMP 端口。

**默认：**161

### **-c community**

（可选）为 SNMP 版本 1 和 2c 指定 SNMP 读/写团体字符串。

**默认值：**public

示例：对系统表运行 `getTable` 命令。

```
rmonwatch getTable sys
RMONWBEM AIM Watch 程序, 版本 1.0.0
E:\sysedge\bin\walktree -h localhost -p 161 -v 1 -c public -m 0 -t 30 -r 3 -o
1.3.6.1.4.1.546.16.22.11
rmonWbemSysSystemName <sys01-01> .sys01-01
rmonWbemSysSystemName <sys02-01-8> .sys02-01-8
rmonWbemSysSystemName <sys03-01-7> .sys03-01-7
rmonWbemSysRowStatus active .sys01-01
rmonWbemSysRowStatus active .sys02-01-8
rmonWbemSysRowStatus active .sys03-01-7
rmonWbemSysDescr <nul> .sys01-01
rmonWbemSysDescr <nul> .sys02-01-8
rmonWbemSysDescr <nul> .sys03-01-7
rmonWbemSysContact <nul> .sys01-01
rmonWbemSysContact <nul> .sys02-01-8
rmonWbemSysContact <nul> .sys03-01-7
rmonWbemSysQualifiers .sys01-01
rmonWbemSysQualifiers .sys02-01-8
rmonWbemSysQualifiers .sys03-01-7
rmonWbemSysConfigSet <default> .sys01-01
rmonWbemSysConfigSet <metricNet,metricFS,metricDisk> .sys02-01-8
rmonWbemSysConfigSet <default> .sys03-01-7
rmonWbemSysCredentials <test> .sys01-01
rmonWbemSysCredentials <win2008> .sys02-01-8
rmonWbemSysCredentials <win7> .sys03-01-7
rmonWbemSysProtocol dcom .sys01-01
rmonWbemSysProtocol dcom .sys02-01-8
rmonWbemSysProtocol dcom .sys03-01-7
rmonWbemSysMaxInstances <100> .sys01-01
rmonWbemSysMaxInstances <100> .sys02-01-8
rmonWbemSysMaxInstances <100> .sys03-01-7
rmonWbemSysIPAddress <192.168.0.10> .sys01-01
rmonWbemSysIPAddress <192.168.0.11> .sys02-01-8
rmonWbemSysIPAddress <192.168.0.12> .sys03-01-7
rmonWbemSysMACAddress <00:12:3F:64:1D:AE> .sys01-01
rmonWbemSysMACAddress <00:50:56:97:7D:BA> .sys02-01-8
rmonWbemSysMACAddress <00:25:64:A9:98:86> .sys03-01-7
rmonWbemSysAdded <2010-5-17,17:26:35.0,+2:0> .sys01-01
rmonWbemSysAdded <2010-5-17,17:26:37.0,+2:0> .sys02-01-8
rmonWbemSysAdded <2010-5-17,17:26:40.0,+2:0> .sys03-01-7
rmonWbemSysLastBoot <2010-5-17,7:52:25.0,+2:0> .sys01-01
rmonWbemSysLastBoot <2010-5-1,20:53:22.0,+2:0> .sys02-01-8
rmonWbemSysLastBoot <2010-5-1,20:53:22.0,+2:0> .sys03-01-7
rmonWbemSysLastPoll <0d 00:10:41> .sys01-01
rmonWbemSysLastPoll <0d 00:11:11> .sys02-01-8
rmonWbemSysLastPoll <0d 00:11:11> .sys03-01-7
rmonWbemSysLastSuccess <0d 00:10:41> .sys01-01
rmonWbemSysLastSuccess <0d 00:11:11> .sys02-01-8
rmonWbemSysLastSuccess <0d 00:11:11> .sys03-01-7
```

```

rmonWbemSysLastError    success    .sys01-01
rmonWbemSysLastError    success    .sys02-01-8
rmonWbemSysLastError    success    .sys03-01-7
rmonWbemSysQueryNumber  <14>    .sys01-01
rmonWbemSysQueryNumber  <17>    .sys02-01-8
rmonWbemSysQueryNumber  <14>    .sys03-01-7
rmonWbemSysQueryFailed  <0>    .sys01-01
rmonWbemSysQueryFailed  <0>    .sys02-01-8
rmonWbemSysQueryFailed  <0>    .sys03-01-7

```

## rmonwatch getVersion 命令-查看版本信息

使用该 `getVersion` 命令可以查看 `rmonWbemVersion` MIB 变量的值。

该 `getVersion` 命令使用以下语法：

```
rmonwatch [options] getVersion
```

`rmonwatch` 使用以下选项：

### **--version**

显示程序版本号并退出。

### **--help**

显示 `rmonwatch` 帮助

### **-h hostname | -h ipAddr**

（可选）指定运行 RM AIM 的 CA SystemEDGE 主机。

**默认值：** localhost

### **-p port**

（可选）指定 CA SystemEDGE SNMP 端口。

**默认：** 161

### **-c community**

（可选）为 SNMP 版本 1 和 2c 指定 SNMP 读/写团体字符串。

**默认值：** public

### 示例：getVersion 命令返回的结果

```

RMONWBEM AIM Watch 程序, 版本 1.0.0
E:\sysedge\bin\snmpget -h localhost -p 161 -v 1 -c public -m 0 -t 5 -r 3 -o
1.3.6.1.4.1.546.16.22.1.0
1.3.6.1.4.1.546.16.22.1.0 SysEDGE RMONWBEM AIM, VERSION:1.0.0 BUILD:10137
REQUIRES:5.0.0

```

## 高级 SNMP 参数

对于 `rmonwatch add`、`del`、`getTable` 和 `getVersion` 命令，以下参数是相同的。仅当使用版本为 2c 或 3 的 SNMP，或使用非默认超时并重试时，才需要这些参数。

### **-v snmpVersion**

（可选）指定 SNMP 版本。可能的值为：

1

2c

3

默认：1

### **-u secName**

（可选）指定 SNMPv3 安全用户的名称。

### **-s secLevel**

（可选）指定 SNMPv3 的安全级别。可能的值为（无默认值）：

1 = noAuthNoPriv

2 = AuthNoPriv

3 = AuthPriv

### **-n contextName**

（可选）指定 MIBMuxed 代理的实例名称（无默认值）。

### **-a authPassword**

（可选）标识针对 AuthNoPriv 或 AuthPriv 安全级别选择 SNMPv3 时所需的身份验证密码（无默认值）。

### **-A authProtocol**

（可选）指定身份验证协议。可能的值为：

MD5 = 身份验证协议 HMAC-MD5

SHA = 身份验证协议 HMAC-SHA

默认值：MD5

### **-x privPassword**

（可选）指定针对安全级别 3 (AuthPriv) 选择 SNMPv3 时所需的隐私（加密）密码。

**-X encryptProtocol**

（可选）指定针对隐私使用加密协议（无默认值）。可能的值为：

DES-数据加密标准

AES-使用 128 位加密密钥的高级加密标准 (AES128)

**-m FIPSmode**

（可选）指定 FIPS 模式。可能的值为：

0 = 非 FIPS

1 = FIPS 共存

2 = 仅 FIPS

默认：0

**-t timeout**

（可选）指定 SNMP 设置并获得命令超时（秒）。

默认：5

**--timeout=walktreetimeout**

（可选）为 walktree 命令指定 SNMP 超时。

默认：30

**-r snmpretries**

（可选）指定重试 SNMP 命令的次数。

默认：3

## 日志文件

本节提供了可用 CA Virtual Assurance 日志文件及其配置设置的摘要。

## 可用的日志文件

CA Virtual Assurance 中的各种组件会生成日志文件，当您在产品使用过程中遇到问题时，可以检查这些日志文件。您可能还需要这些日志文件来帮助您与技术支持交流。列出的日志文件位于以下目录中：

*Install\_Path\CA\VPM\log*

可用的日志文件及其包含的信息如下所示：

### CA Virtual Assurance 外部组件管理器

#### **DPMExtCompMgr.log**

包含外部组件管理器的日志记录信息。

### CLI 日志文件

#### **dpmimgcli.log (dpmimg.exe)**

包含发出的映像命令的日志记录信息。

#### **dpmvccli.log (dpmvc.exe)**

包含发出的 vCenter 命令的日志记录信息。

#### **dpmlparcli.log (dpmlpar.exe)**

包含发出的 LPAR 命令的日志记录信息。

#### **dpmzonecli.log (dpmzone.exe)**

包含发出的 Solaris Zone 命令的日志记录信息。

#### **dpmcmdcli.log (dpmcmd.exe)**

包含发出的常规用途外壳命令的日志记录信息。

#### **dpmresourcemgrcli.log (dpmresourcemgr.exe)**

包含发出的资源管理器命令的日志记录信息。

### 事件管理器

#### **DPMEventMgr.log**

包含事件管理器的日志记录信息。

### 映像

#### **dpm-imaging.log**

包含映像服务的日志记录信息。

## 启动

### **dpm-schedadapter.log**

包含启动服务的运行时日志记录信息。

### **SchedInitialize.log**

包含启动服务的初始化日志记录信息。

## 性能监视器

### **coll\_eng.log**

包含性能监视器的日志记录信息。

## 策略引擎

### **capolicyeng.log**

包含策略引擎的日志记录信息。

## 资源管理器

### **ResourceMgr.log**

包含资源管理器的日志记录信息。

## 服务控制器

### **servcon.log**

包含服务控制器的日志记录信息。

## VirtualCenter 适配器

### **dpm-vcadapter.log**

包含 VirtualCenter 适配器的本机运行时组件的日志记录信息。

### **vcadapter\_java.log**

包含 VirtualCenter 适配器的 Java 运行时组件的日志记录信息。

## 日志文件设置

您可以通过使用记事本或类似的文本编辑器编辑 *dpmlogging.cfg* 文件来配置每个日志文件。该文件位于以下目录中：

*Install\_Path\conf*

您可以更改以下属性的设置：

### 最大日志大小(字节)

定义日志的大小。

### 最大回滚数

定义在达到最大大小之后应保存的日志文件数。

### 日志记录级别

定义想要捕获日志的事件的级别。

可用的语法和日志记录级别位于 *dpmlogging.cfg* 文件中。



## 第 4 章：性能度量标准

---

以下各节介绍了默认情况下适用于 CA Virtual Assurance 代理和 SystemEDGE 的性能度量标准。

此部分包含以下主题：

- [CA HP UNIX 性能度量标准](#) (p. 769)
- [CA IBM AIX 性能度量标准](#) (p. 770)
- [CA IBM LPAR 性能度量标准](#) (p. 771)
- [CA IBM HACMP 性能度量标准](#) (p. 776)
- [CA Linux 性能度量标准](#) (p. 777)
- [CA Microsoft Hyper-V 性能指标](#) (p. 778)
- [CA Oracle Solaris 性能度量标准](#) (p. 778)
- [CA Oracle Solaris Zones 性能度量标准](#) (p. 779)
- [CA SystemEDGE 性能指标](#) (p. 784)
- [CA SystemEDGE 增强的度量标准](#) (p. 787)
- [CA VMware vCenter 服务器性能指标](#) (p. 794)
- [CA Citrix XenServer 性能度量标准](#) (p. 805)
- [CA Windows Microsoft 群集性能度量标准](#) (p. 808)
- [CA Windows 性能度量标准](#) (p. 809)
- [Cisco Unified Computing System \(UCS\) 性能度量标准](#) (p. 810)
- [远程监控性能指标](#) (p. 812)

### CA HP UNIX 性能度量标准

以下列表提供有关针对 HP UNIX 服务器监控的度量标准的信息。

#### **CA 内存组：可用物理内存百分比**

系统上的可用物理内存百分比。

#### **CPU：百分比总数**

系统上 CPU 使用率的百分比总数。

#### **CA CPU 组：用户使用百分比总数**

系统中用户活动导致的总 CPU 使用率。

#### **CA CPU 组：系统使用百分比总数**

系统中 OS 活动导致的总 CPU 使用率。

#### **CA CPU 组：十五分钟内的平均负载**

系统在 15 分钟内的平均负载。

#### **CA CPU 组：空闲使用百分比总数**

系统空闲进程使用的 CPU 百分比。

**CA 磁盘组：每秒字节数**

磁盘每秒读取和写入的字节数。

**CA 磁盘组：每秒读取数（平均值）**

磁盘每秒平均读取数。

**CA 磁盘组：每秒写入数（平均值）**

磁盘每秒平均写入数。

**CA 网络组：每秒接收的 TCP 操作数据包**

网络适配器每秒接收的数据包的数目。

**CA 网络组：每秒发送的 TCP 操作数据包**

网络适配器每秒发送的数据包的数目。

**CA 文件系统组：可用空间**

文件系统的可用空间量（字节）。

**CA 文件系统组：已用空间**

文件系统的已用空间量（字节）。

## CA IBM AIX 性能度量标准

以下列表提供有关针对 IBM AIX 服务器监控的度量标准的信息。

**CA 内存组：可用物理内存百分比**

系统上的可用物理内存百分比。

**CPU：百分比总数**

系统上 CPU 使用率的百分比总数。

**CA CPU 组：用户使用百分比总数**

系统中用户活动导致的总 CPU 使用率。

**CA CPU 组：系统使用百分比总数**

系统中 OS 活动导致的总 CPU 使用率。

**CA CPU 组：十五分钟内的平均负载**

系统在 15 分钟内的平均负载。

**CA 磁盘组：每秒字节数**

磁盘每秒读取和写入的字节数。

**CA 磁盘组：每秒读取数（平均值）**

磁盘每秒平均读取数。

**CA 磁盘组：每秒写入数（平均值）**

磁盘每秒平均写入数。

**CA 网络组：每秒接收的 TCP 操作数据包**

网络适配器每秒接收的数据包的数目。

**CA 网络组：每秒发送的 TCP 操作数据包**

网络适配器每秒发送的数据包的数目。

**CA 文件系统组：可用空间**

文件系统的可用空间量（字节）。

**CA 文件系统组：已用空间**

文件系统的已用空间量（字节）。

**CA CPU 组：空闲使用百分比总数**

系统空闲进程使用的 CPU 百分比。

## CA IBM LPAR 性能度量标准

下表提供了有关为 IBM 逻辑分区监控的性能度量标准的信息。

### CA IBM pSeries 逻辑分区性能度量标准

下表提供了 IBM pSeries 逻辑分区性能度量标准的说明。

**CPU 使用率（百分比）**

逻辑分区（需要在 LPAR 上安装 CA Virtual Assurance 代理）的 CPU 使用率的百分比。

**内存使用率（百分比）**

逻辑分区（需要在 LPAR 上安装 CA Virtual Assurance 代理）的内存 (RAM) 使用率的百分比。

**一分钟平均负载**

逻辑分区（需要在 LPAR 上安装 CA Virtual Assurance 代理）的一分钟平均负载。

**五分钟平均负载**

逻辑分区（需要在 LPAR 上安装 CA Virtual Assurance 代理）的五分钟平均负载。

**十五分钟平均负载**

逻辑分区（需要在 LPAR 上安装 CA Virtual Assurance 代理）的十五分钟平均负载。

**CPU 分配**

分配给逻辑分区的物理 CPU 数。

**内存分配**

分配给逻辑分区的逻辑内存块 (LMB) 数。

**不可用分区**

未正确配置该分区向赞助商提供信息（例如，坏掉或故障的设备），但赞助商知晓该信息。如果显示了分区，且需要性能信息，则该 LPAR 上必须安装有 CA Virtual Assurance 代理。受管节点显示为该度量标准的实例。

**授权容量（百分比）**

已使用的物理处理器的逻辑分区容量的百分比。

**物理忙碌**

将处理器忙碌状态表示为授权容量百分比乘以集期望的 CPU 授权的计算。

**已使用的物理处理器**

逻辑分区已使用的处理器单元数。

## CA IBM pSeries 性能度量标准

下表提供了 IBM pSeries 逻辑分区性能度量标准的说明。

**平均 CPU 使用率（百分比）**

服务器上逻辑分区的平均 CPU 使用率。

**CPU 使用率（百分比）**

服务器的总 CPU 使用率。

**平均内存使用率（百分比）**

服务器上逻辑分区的平均 RAM 使用率。

**内存使用率（百分比）**

服务器的总 RAM 使用率。

**一分钟平均负载**

服务器上所有逻辑分区的平均一分钟平均负载。

### 五分钟平均负载

服务器上所有逻辑分区的平均五分钟平均负载。

### 十五分钟平均负载

服务器上所有逻辑分区的平均十五分钟平均负载。

### 不可用系统

无赞助商可从中收集信息的逻辑分区 (LPAR) 的受管系统。

### 已使用的物理处理器

已使用的物理处理器数。

## 处理器池支持

本节介绍了处理器池的概念，并提供有关适用于 IBM LPAR 的 CPU 度量标准的信息。

### 处理器池

*处理器池*是可以跨不同的逻辑分区共享的一组物理处理器。自引入基于 POWER5 的 IBM Power 系统以来，就可以使用共享处理器池。基于 POWER5 的服务器支持一个与物理共享处理器池相同的共享处理器池。

基于 POWER6 及更高版本的服务器支持多个共享处理器池。“*多个共享处理器池 (MSPP)*”功能在 Power6 和更高版本的服务器上受支持。使用此功能可创建多个处理器池，从而使 CPU 资源分配更为灵活。所有支持多个共享处理器池功能的 Power 系统均至少具备一个共享处理器池，最多有 64 个共享处理器池。默认的共享处理器池的 ID 始终为 0。系统（仅具有默认的共享处理器池）的默认行为是 POWER5 服务器（仅定义了物理共享处理器池）的当前行为。

## 池容量

每个共享处理器池都具有与其关联的最大容量。“*最大池容量*”定义共享处理器池中的分区组可以利用的处理器容量上限。最大池容量必须等于或大于共享处理器池中的额定池容量。

系统管理员可以为共享处理器池中的任何无上限 LPAR 保留处理器容量。“*保留的池容量*”是保留以供共享处理器池中所有无上限的逻辑分区使用的处理单元数量。

如果处理器池中的某些分区不使用其授权容量，相同池中的其他无上限 LPAR 将根据 LPAR 的权重分配附加容量。

共享处理器池的“*额定池容量*”用于定义保证可用于处理器池中分区组的处理器容量。额定池容量等于共享处理器池中分区的授权容量和保留的池容量之和。分区的额定容量为进行使用情况统计和监控提供了基础。消耗其所有额定容量的分区将报告 100% 的利用率。根据其虚拟处理器配置，无上限分区可以从共享处理器池消耗未使用的处理容量，在此情况下它将报告超过 100% 的利用率。

## 默认的共享处理器池

默认的共享处理器池与用户定义的池具有相同属性，但是这些属性具有固定值，并且不受系统管理员直接控制。最大池容量等于物理共享处理器池的全部容量，而保留的池容量为 0。

通过系统管理员操作（如打开专用处理器分区的电源，或动态地将物理处理器移入或移出物理共享处理器池），可以间接更改默认池的最大容量。

## 有上限 LPAR 和无上限 LPAR

使用共享处理器的逻辑分区可具有有上限或无上限共享模式。

“*有上限的逻辑分区*”无法使用比分配到的处理单元更大的处理器能力。为有上限分区分配了最大容量，确保无法超过此容量且不会影响物理系统的整体行为。

“*无上限的逻辑分区*”可以使用比分配到的处理容量更大的处理器能力。限制因素是已分配给逻辑分区的虚拟处理器数目或共享处理器池的最大处理单元数目。保证无上限分区具有最低资源级别（具有使用更多资源的潜力），可以为分区创建最低服务级别。

如果多个无上限 LPAR 同时需要额外的处理器容量，服务器可将未使用的处理容量分发给所有无上限 LPAR。此分发过程由每个 LPAR 的无上限权重决定。默认的无上限权重值是 128，最高权重值是 255。当您将无上限权重设置为 0 时，不会将未使用的容量分发给逻辑分区。

## CPU 度量标准

本节将为支持处理器池的环境的 CPU 使用率提供重要的 CPU 度量标准。

## 消耗度量标准

当前分配给分区的处理单元数目表示其授权。如果分区是无上限共享分区，消耗的处理单元数可能会超出分配的数量。两个对应的度量标准分别是物理消耗和授权消耗。

### 物理消耗（对象为物理处理器）

表示当前使用的处理单元数量。物理消耗相当于分区的额定容量乘以其授权消耗率。

**示例：**如果带有 0.50 处理单元额定容量的分区具有 150% 的额定容量消耗率，则它将消耗  $0.50 * 1.5 = 0.75$  的物理处理器。

### 授权消耗（对象为额定容量）

表示当前使用的处理单元相对于分配给分区的处理单元数目的百分比。消耗其所有额定容量的分区将报告 100% 的利用率。因此，无上限共享分区的授权消耗率可能超过 100%。

## 容量度量标准

共享处理器池可以提供对处理器容量的控制，该处理器容量可由特定的分区集通过物理共享处理器池使用。保证处理器池中的每个分区具有处理器授权。可以从保留的池容量向无上限分区分配附加容量。如果处理器池的某些分区未使用其授权容量，则会放弃未使用的容量。相同池中的其他无上限分区会根据其无上限权重分得附加容量。下列度量标准描述了共享处理器池中处理器容量的使用率：

### 额定池容量

表示共享处理器池中分区的授权容量和保留的池容量之和。

### 可用池容量

表示共享池中可用的物理处理器（应用程序），即共享池中未使用的部分。

### 示例：可用的物理处理器和 CPU 利用率之间的关系

“lparstat”命令用于显示共享池中可用的物理处理器（应用程序）。可以从已启用共享处理器池使用授权的任何逻辑分区运行 lparstat 命令。共享池的大小是 8 个 CPU（页面大小）。每个抽样间隔的可用处理器均显示在“应用程序”列下。系统配置如下所示：type=Shared、mode=Uncapped、smt=4、lcpu=4、mem=1024 MB、psize=8、ent=0.10。对于可用处理器数为 7.92，CPU 利用率为 1.0% 的情况，计算方式为： $((8.0 - 7.92) / 8.0)$ 。

## CA IBM HACMP 性能度量标准

以下部分提供用于 IBM HACMP 监控的度量标准的相关信息。

### IBM HACMP 群集度量标准

以下列表说明 IBM HACMP 群集性能指标。

**群集状态 (hacmpAimStatClusterState)**

表示 HACMP 群集的状态。

**群集子状态 (hacmpAimStatClusterSubState)**

表示 HACMP 群集的子状态。

**群集服务状态 (hacmpAimStatServiceState)**

表示操作系统服务状态。

### IBM HACMP 节点度量标准

以下列表说明了 IBM HACMP 节点性能度量标准。

**节点状态 (hacmpAimStatNodeState)**

指示节点的状态。

**总内存 (KB) (hacmpAimStatNodeTotalMemory)**

识别节点的总内存 (KB)。

**内存使用情况 (KB) (hacmpAimStatNodeUsedMemory)**

识别节点的总内存使用量 (KB)。

**内存使用率 (%) (hacmpAimStatNodeUsedMemory)**

识别节点的总内存使用率 (%)。

**CPU 使用率 (%) (hacmpAimStatNodeMemoryUsage)**

识别服务器的 CPU 使用率 (%)。

### IBM HACMP 资源组度量标准

以下项目说明了 IBM HACMP 资源组性能度量标准。

**资源组状态 (hacmpAimStatResourceGroupState)**

表示资源组的状态。



## IBM HACMP 网络度量标准

以下列表说明了 IBM HACMP 网络性能度量标准。

### 网络状态 (hacmpAimStatNetworkState)

表示网络状态。

### 网络接口状态 (hacmpAimStatNetworkInterfaceState)

表示网络接口状态。

## CA Linux 性能度量标准

以下度量标准是针对 Linux 服务器进行监控的。

### CA CPU 组：用户使用百分比总数

进程运行用户指令所用时间的百分比。

### CA CPU 组：系统使用百分比总数

处理器运行操作系统指令所用时间的百分比。

### CA CPU 组：十五分钟内的平均负载

15 分钟内的平均负载。

### CA 磁盘组：每秒字节数

从物理磁盘读取和向其写入数据的速率。

### CA 磁盘组：每秒读取数

从物理磁盘读取数据的速率。

### CA 磁盘组：每秒写入数

向物理磁盘写入数据的速率。

### CA 网络组：每秒接收的 TCP 操作数据包

网络接口接收数据包的速率。

### CA 网络组：每秒发送的 TCP 操作数据包

网络接口发送数据包的速率。

### CA 文件系统组：可用空间

文件系统中的可用存储空间（千字节）。

### CA 文件系统组：已用空间

文件系统中已用的存储空间（千字节）。

## CA Microsoft Hyper-V 性能指标

以下列表提供有关 Microsoft Hyper-V 的监控指标的信息。

### 重要 VM 的 Hyper-V 主机号

重要 VM 的数量。如果该值超过指定的数目，则会出现严重警告。

**OID:** 1.3.6.1.4.1.546.16.56.2.1.2.1.16.1

### Hyper-V 主机 CPU 使用率（百分比）

服务器的总 CPU 使用率。如果该值超过 80%，将会出现警告。

**OID:** 1.3.6.1.4.1.546.16.56.2.1.2.1.9

### Hyper-V 主机内存使用率（百分比）

服务器的总 RAM 使用率。如果该值超过 60%，将会出现警告。

**OID:** 1.3.6.1.4.1.546.16.56.2.1.2.1.10

### VM CPU 使用率（百分比）

VM 的 CPU 总使用率。如果该值超过 80%，将会出现警告。

**OID:** 1.3.6.1.4.1.546.16.56.2.2.2.1.21

### VM 运行状况

VM 的运行状况。如果运行状况显示出现重要故障，则会出现重要警告。如果运行状况显示出现严重故障，则会出现严重警告。

**OID:** 1.3.6.1.4.1.546.16.56.2.2.2.1.6

## CA Oracle Solaris 性能度量标准

以下列表提供有关针对 Oracle Solaris 服务器监控的度量标准的信息。

### CA CPU 组：用户使用百分比总数

进程运行用户指令所用时间的百分比。

### CA CPU 组：系统使用百分比总数

处理器运行操作系统指令所用时间的百分比。

### CA CPU 组：十五分钟内的平均负载

15 分钟内的平均负载。

### CA 磁盘组：每秒字节数

从物理磁盘读取和向其写入数据的速率。

**CA 磁盘组：每秒读取数**

从物理磁盘读取数据的速率。

**CA 磁盘组：每秒写入数**

向物理磁盘写入数据的速率。

**CA 网络组：每秒接收的 TCP 操作数据包**

网络接口接收数据包的速率。

**CA 网络组：每秒发送的 TCP 操作数据包**

网络接口发送数据包的速率。

**CA 文件系统组：可用空间**

文件系统中的可用存储空间（千字节）。

**CA 文件系统组：已用空间**

文件系统中已用的存储空间（千字节）。

## CA Oracle Solaris Zones 性能度量标准

以下列表提供了有关为 Solaris Zones 服务器监控的性能度量标准的信息。

### Solaris Zones 服务器度量标准

下表提供了 Solaris Zones 服务器性能度量标准的说明。

**CPUs 数**

服务器上的物理 CPU 总数。

**vCPU 数目**

非全局区域上的 CPU 数。

**资源池数**

服务器上的资源池数。

**区域总数**

服务器上的区域总数，包括全局区域。

**区域状态**

全局区域的状态。

## Solaris Zones 服务器磁盘度量标准

下表提供了 Solaris Zones 服务器磁盘性能度量标准的说明。

### 读取 KB/每秒

每秒读取的 KB 数。

### 写入 KB/每秒

每秒写入的 KB 数。

### 已读取 KB

已读取的 KB 数。

### 已写入 KB

已写入的 KB 数。

### 已使用的磁盘空间

已使用磁盘空间的总数。

### 可用的磁盘空间

可用磁盘空间的总数。

### DiskBusyPct

磁盘忙碌（事务正在进行中）的时间的百分比。

### DiskWaitPct

事务等待服务（队列非空）的时间的百分比。

## Solaris Zones 服务器网络度量标准

下表提供了 Solaris Zones 服务器网络性能度量标准的说明。

### 已接收的 KB

已接收的 KB 数。

### 已传输的 KB

已传输的 KB 数。

### 接收的 KB/每秒

基于轮询间隔每秒接收的 KB 数。

### 传输的 KB/每秒

基于轮询间隔每秒传输的 KB 数。

### 状态

PNET 的已连接状态。

## Solaris Zones 服务器内存度量标准

下表提供了 Solaris Zones 服务器内存性能度量标准的说明。

### 已配置的内存 (KB)

已配置的服务器内存，以 KB 为单位。

### 可用内存 (KB)

可用的服务器内存，以 KB 为单位。

### 内存使用率（百分比）

服务器当前的内存使用率。

### 已使用的内存 (KB)

已使用的服务器内存，以 KB 为单位。

## Solaris Zones 服务器 CPU 度量标准

下表提供了 Solaris Zones 服务器 CPU 性能度量标准的说明。

### CPU 使用率（百分比）

服务器当前的 CPU 使用率。

### CPU 空闲秒数（百分比）

基于轮询间隔 CPU 空闲状态所花费的平均时间的百分比。

### CPU 已使用的秒数（百分比）

基于轮询间隔使用 CPU 所花费的平均时间的百分比。

### CPU 等待秒数（百分比）

分配给服务器的 CPU 份额数。

## Solaris Zones 度量标准

下表提供了 Solaris Zones 性能度量标准的说明。

### 区域 CPU 计数

分配给区域的 CPU 数。

### 项目 CPU 份额

分配给项目的 CPU 数。

## Solaris Zones CPU 度量标准

下表提供了 Solaris Zones CPU 性能度量标准的说明。

### **CPU 使用率（百分比）**

区域当前的 CPU 使用率百分比。

### **CPU 空闲秒数（百分比）**

基于轮询间隔 CPU 处于空闲状态所花费时间的平均百分比。

### **CPU 已使用的秒数（百分比）**

基于轮询间隔使用 CPU 所花费时间的平均百分比。

### **CPU Sys 秒数（百分比）**

基于轮询间隔系统使用 CPU 所花费时间的平均百分比。

### **CPU 等待秒数（百分比）**

基于轮询间隔处于 CPU 等待状态所花费时间的平均百分比。

### **已配置的 CPU 份额**

分配给区域的 CPU 份额数。

### **已配置的 CPU 最高值（百分比）**

已配置的 CPU 最高保留阈值的百分比。

### **已配置的 CPU 最低值（百分比）**

已配置的 CPU 最低保留阈值的百分比。

## Solaris Zones 内存度量标准

下表提供了 Solaris Zones 内存性能度量标准的说明。

### **锁定的内存 (MB)**

已配置的区域内存，以 MB 为单位。

### **已使用的内存 (KB)**

已使用的区域内存，以 MB 为单位。

### **内存使用率（百分比）**

区域的当前内存使用率。

如果为该区域设置了 **PhyMemory** 限制，其将用于内存使用率的计算。在这种情况下，则以本地区域级别给定使用率，并可达到 100%。

## Solaris Zones 最大交换度量标准

下表提供了 Solaris Zones 最大交换性能度量标准的说明。

### 最大交换量 (KB)

用于交换空间的可用内存，以 Kb 为单位。

## Solaris Zones 共享内存度量标准

下表提供了 Solaris Zones 共享内存性能度量标准的说明。

### 共享内存 (KB)

共享内存量，以 KB 为单位。

## Solaris Zones 资源池度量标准

下表提供了 Solaris Zones 资源池性能度量标准的说明。

### CPU 份额（百分比）

资源池的总 CPU 份额的百分比。

## Solaris Zones 项目度量标准

下列提供了 Solaris Zones 项目性能度量标准的说明。

### 内存容量

最大内存限制，以 MB 为单位。

### 共享的最大内存

允许由进程用来在该项目中运行的内存总量，以 MB 为单位。

### 任务计数

项目的任务计数。

### CPU 使用

项目正在使用的 CPU 量。

### 内存使用率

项目正在使用的内存量。

## CA SystemEDGE 性能指标

以下是受 SystemEDGE 代理监控的性能指标。

### **activeMemory (AIX、HP-UX、Linux、Solaris)**

在固定时间段内采样的活动实际内存总量 (KB)。采样时间间隔取决于系统。在 SunOS 中，内核会每 5 秒钟更新一次该参数。当内存中运行的进程实际使用该内存时，内存处于活动状态。该值应当小于或等于 memInUse。较大的值可能表示系统内存不足。

### **cpuTotalIdle (AIX、HP-UX、Linux、Solaris、Windows)**

所有 CPU 在空闲模式下花费的总秒数。NT 系统不支持该变量。该值可以打包 32 位计数器。

### **cpuTotalIdlePercent (AIX、HP-UX、Linux、Solaris、Windows)**

系统 CPU 以空闲模式执行的时间百分比（抽样期间内）。

### **cpuTotalSys (AIX、HP-UX、Linux、Solaris、Windows)**

所有 CPU 在内核或系统模式下花费的总秒数。该值可以打包 32 位计数器。

### **cpuTotalSysPercent (AIX、HP-UX、Linux、Solaris、Windows)**

系统 CPU 执行内核或操作系统的时间百分比（抽样期间内）。

### **cpuTotalUser (AIX、HP-UX、Linux、Solaris、Windows)**

所有 CPU 在用户模式下花费的总秒数。该值可以打包 32 位计数器。

### **cpuTotalUserPercent (AIX、HP-UX、Linux、Solaris、Windows)**

系统 CPU 以用户模式执行的时间百分比（抽样期间内）。

### **loadAverage15Min (AIX、HP-UX、Linux、Solaris)**

过去 15 分钟的平均负载。平均负载代表指定时间范围内在运行队列中的作业平均数。报告的值是平均负载乘以 100。例如，值 50 表示平均负载为 0.50。

### **loadAverage1Min (AIX、HP-UX、Linux、Solaris、Windows)**

过去 1 分钟的平均负载。平均负载代表指定时间范围内在运行队列中的作业平均数。报告的值是平均负载乘以 100。例如，值 50 表示平均负载为 0.50。

### **loadAverage5Min (AIX、HP-UX、Linux、Solaris、Windows)**

过去 5 分钟的平均负载。平均负载代表指定时间范围内在运行队列中的作业平均数。报告的值是平均负载乘以 100。例如，值 50 表示平均负载为 0.50。



**memCapacity (AIX、HP-UX、Linux、Solaris、Windows)**

正在使用的系统活动内存百分比。通过将 activeMem(10) 除以内存 (System.3) 并转换为百分比，可近似得出该对象的值。因为虚拟内存允许系统使用较高的 memCapacity 值进行操作，所以不应当单独使用该值来判断加载系统的方式。管理器也应当监控 swapCapacity。

**memInUse (AIX、HP-UX、Linux、Solaris、Windows)**

正在使用的系统内存量。

**memInUseCapacity (AIX、HP-UX、Linux、Solaris、Windows)**

正在使用的系统内存百分比。通过将 memInUse(9) 除以内存 (System.3) 并转换为百分比，可近似得出该对象的值。因为虚拟内存允许系统使用较高的 memCapacity 值进行操作，所以不应当单独使用该值来判断加载系统的方式。管理器也应当监控 swapCapacity。

**numOpenFiles (AIX、HP-UX、Linux、Solaris、Windows)**

系统范围内打开文件的总数。

**numPageFaults (AIX、HP-UX、Solaris、Windows)**

自上次初始化内核以来，已经发生的页面错误总数。当运行进程尝试访问当前不在物理内存中的虚拟存储页面时，就会发生页面错误。该尝试会导致硬件页面错误。高比率的页面错误可能表示系统超载或系统内存不足。

**numPageInsPerScan (AIX、HP-UX、Linux、Solaris)**

自上次初始化内核以来，已调入的总页数。页面调入是一种操作，由在其中从辅助存储读取页面内容的虚拟存储系统执行。高比率的更改可能表示系统超载。

**numPageOutsPerScan (AIX、HP-UX、Linux、Solaris、Windows)**

自上次扫描以来，已调出的总页数。页面调出是一种操作，由在其中将页面内容写入辅助存储的虚拟存储系统执行。高比率的更改可能表示系统超载。

**numProcs (AIX、HP-UX、Linux、Solaris、Windows)**

表槽当前已分配且正在使用的进程数。在较旧的基于 BSD 的系统上，该指标非常重要，因为它对任何一个时刻可能在系统上运行的进程总数设定了上限值。

**numSwitches (AIX、HP-UX、Linux、Solaris、Windows)**

自上次初始化内核以来，已经发生的上下文切换总数。每当进程放弃 CPU 而由另一个进程代替时，都会发生上下文切换。该计数器反映了系统活动的级别；高比率的上下文切换表示系统负载。

**numSyscalls (AIX、HP-UX、Solaris、Windows)**

自上次初始化内核以来，已经发生的系统调用总数。当进程调用一个必须以内核模式或特权模式执行的函数时，会发生系统调用。它是系统负载的指标，并且在被许多用户共享的服务器或计算机上可能较高。

**numZombieProcs (AIX、HP-UX、Linux、Solaris、Windows)**

在系统上找到的僵尸进程的当前数量。该 MIB 对象通过将所有进程与 processStateStr 僵尸进程相加而得。数量越少越好。在 SystemEDGE 4.1 和更高版本中受到支持。

**runQLen (AIX、HP-UX、Linux、Solaris、Windows)**

在固定时间段内采样时，排定程序的运行队列的长度。采样时间间隔取决于系统。在 SunOS 中，内核会每 5 秒钟更新一次该参数。系统平均负载通过 runQLen 的值进行计算；平均负载定义为针对某些时间间隔的 runQLen 平均值。

**swapCapacity (AIX、HP-UX、Linux、Solaris、Windows)**

系统的使用中的交换空间总量百分比。通过将 swapInUse(13) 除以 totalSwapSpace(29) 并乘以 100 可计算出该对象的值。在此包括该值可便于轮询、监控和历史记录采样。该对象是所有交换分区和区域的聚合值。

**swapInUse (AIX、HP-UX、Linux、Solaris、Windows)**

系统当前使用的交换空间量 (KB)。当内存短缺时使用交换空间。通常，整个进程被换出到辅助存储。将该值与 totalSwap（在 Concord 的 kernelConfig 组中）进行比较可帮助表明是否需要额外的交换空间。

**totalSwapSpace (AIX、HP-UX、Linux、Solaris、Windows)**

系统交换空间总量 (KB)。交换空间是辅助存储（主要为磁盘）上的区域，用于交换和分页。交换是一种内存管理策略，当内存短缺时会将整个进程写入辅助存储中。当进程开始时，其所有页面都在虚拟内存中；只有正处于使用状态的页面才会从磁盘或交换空间引入。

## CA SystemEDGE 增强的度量标准

使用以下度量标准增强 SystemEDGE，以支持迁移 NSM 系统和性能代理。SystemEDGE 代理每 30 秒从 OS 中读取一次所需的变量，并以检索其他统计信息的相同方式将这些变量存储到内部缓存中。

新变量使用与 CA Systems Performance LiteAgent 中相同的名称显示，但位于不同的 OID。下表将所需的 CA Systems Performance LiteAgent 变量与 SystemEDGE 变量相对应：

### Linux

| NSM 项                     | SystemEDGE 项                                                              | 说明                        |
|---------------------------|---------------------------------------------------------------------------|---------------------------|
| CPU                       | cpuGroup.<br>cpuTotalUtilPercent<br>cpuStatsEntry.<br>cpuStatsUtilPercent | % 总数—用户和系统 CPU 的使用率之和。    |
| ForkRequestssec           | kernelperf.forkRequestssec                                                | 分支数目/秒。                   |
| FreePhysical Memory       | performance.memoryStats.<br>freePhysicalMemory                            | 未使用的物理（实际）内存百分比。          |
| RequestsPage Inssec       | performance.memoryStats.<br>requestsPageInssec                            | 每秒调入的页数。                  |
| RequestsPage Outssec      | performance.memoryStats.<br>requestsPageOutssec                           | 每秒调出的页数。                  |
| SystemUsageFree Memory    | performance.memoryStats.<br>systemUsageFreeMemory                         | 未分配的虚拟内存总量。（千字节）          |
| SystemUsageTotal Memory   | performance.memoryStats.<br>systemUsageTotalMemory                        | 系统上可用的虚拟内存总量。（千字节）        |
| SystemUsageUsed Memory    | performance.memoryStats.<br>systemUsageUsedMemory                         | 系统使用已用内存—使用着的虚拟内存的 KB 总数。 |
| ScanRatePage Scanssec     | performance.memoryStats.<br>scanRatePageScanssec                          | 内核检查可用内存页的频率。             |
| RequestsPageFree Bytessec | performance.memoryStats.<br>requestsPageFreeBytessec                      | 通过终止进程释放到内核的空间量。          |
| RequestsPageFrees         | performance.<br>memoryStats.requestsPageFrees                             | 自系统启动以来释放到内核的页面总数。        |
| RequestsPage Fressec      | performance.memoryStats.<br>requestsPageFreessec                          | 通过终止进程释放到内核的页面数。          |

| NSM 项                       | SystemEDGE 项                                                    | 说明                                                 |
|-----------------------------|-----------------------------------------------------------------|----------------------------------------------------|
| RequestsPageInBytessec      | performance.memoryStats.requestsPageInBytessec                  | 每秒页面调入的数据量。                                        |
| RequestsPageOutBytessec     | performance.memoryStats.requestsPageOutBytessec                 | 每秒页面调出的数据量。                                        |
| RequestsPageReclaimBytessec | performance.memoryStats.requestsPageReclaimBytessec             | 从可用列表回收的空间量（以页面为单位）。                               |
| RequestsPageReclaimssec     | performance.memoryStats.requestsPageReclaimssec                 | 每秒回收的页数。                                           |
| InterfaceTrafficIncomingsec | performance.network.ifEntry.<index>.interfaceTrafficIncomingsec | 每个接口卡每秒输入的数据包数。                                    |
| InterfaceTrafficOutgoingsec | performance.network.ifEntry.<index>.interfaceTrafficOutgoingsec | 每个接口卡每秒输出的数据包数目。                                   |
| InputErrors                 | performance.network.ifEntry.<index>.inputErrors                 | 输入错误的数目。这些错误通常表示故障硬件，可能是生成错误数据包的其他计算机，或是错误连接器或中断器。 |
| inputpackets                | performance.network.ifEntry.<index>.inputpackets                | 接口上传输的输入数据包数目。                                     |
| outputpackets               | performance.network.ifEntry.<index>.outputpackets               | 接口上传输的输出数据包数目。                                     |
| blkss                       | diskStatsTable.<index>.diskStatsBlkss                           | 从该设备传输和传输到该设备的字节数（以 512 字节为单位）。                    |
| rws                         | diskStatsTable.<index>.diskStatsRWs                             | 每秒从该设备传输和传输到该设备的数据量（读和写）。                          |
| AllocatedSpace              | devTable.<index>.devAllocatedSpace                              | 文件系统上的可用 MB 总数。                                    |
| FreeSpace                   | devTable.<index>.devFreeSpace                                   | 文件系统上的可用 MB 数。该项可能不包括为超级用户保留的空间。                   |

| NSM 项             | SystemEDGE 项                                   | 说明                                                                                                                                                         |
|-------------------|------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| InodesFree        | devTable. <index>. devFfiles                   | 文件系统上的可用 <b>inode</b> 数（与文件系统上允许的文件和设备数相对应）。某些系统上的 <b>inode</b> 表是动态分配的，因此可用条目的数量很少。                                                                       |
| InodesUsed        | devTable. <index>. devInodesUsed               | 文件系统中正在使用的 <b>inode</b> 或文件数。                                                                                                                              |
| SpaceUsed         | devTable. <index>. devCapacity                 | 文件系统正在使用的文件系统空间的百分比。对于大部分文件系统，该项不包括为超级用户保留的任何额外空间。因此，与报告分区上所有空间（其中一些空间可能对普通用户不可用）的“文件系统”组不同，该值可以有差异。该差异通常适用于 /opt、/var 或 /home 等文件系统，而不适用于 / 或 /tmp 等文件系统。 |
| UsedSpace         | devTable. <index>. devUsedSpace                | 文件系统中正在使用的 <b>MB</b> 数。                                                                                                                                    |
| DiskTime, 忙碌      | diskStatsTable.<index>. diskStatsUtilization   | 上一个度量期间该磁盘的使用率（使用率百分比）。该项也可以表示为：<br>$(\text{磁盘忙时间} / \text{已用时间}) * 100$ % 忙时间—设备忙于处理请求的时间部分。                                                              |
| AvgDiskBytesRead  | diskStatsTable.<index>. diskStatsAvgBytesRead  | 在读取操作期间从磁盘传输的平均字节数。                                                                                                                                        |
| AvgDiskBytesWrite | diskStatsTable.<index>. diskStatsAvgBytesWrite | 在写操作期间传输到磁盘的平均字节数。                                                                                                                                         |
| DiskReadTime      | diskStatsTable.<index>. diskStatsReadTime      | 选定磁盘驱动器忙于处理读请求所用时间的百分比。                                                                                                                                    |

| NSM 项               | SystemEDGE 项                                  | 说明                                              |
|---------------------|-----------------------------------------------|-------------------------------------------------|
| DiskWriteTime       | diskStatsTable.<index>.diskStatsWriteTime     | 选定磁盘驱动器忙于处理写请求所用时间的百分比。                         |
| AvgDiskQueue Length | diskStatsTable.<index>.diskStatsQueueLength   | 上个度量期间服务队列中等待的平均操作数。                            |
| DiskReadBytessec    | diskStatsTable.<index>.diskStatsReadBytessec  | 在读操作期间从磁盘传输字节的速率。                               |
| DiskWriteBytessec   | diskStatsTable.<index>.diskStatsWriteBytessec | 在写操作期间向磁盘传输字节的速率。                               |
| AvgDisksecTransfer  | diskStatsTable.<index>.diskStatsServiceTime   | 上一个度量期间在该磁盘上进行的操作的平均服务时间（毫秒）。也可以表示为：磁盘忙时间/传输次数。 |

## Solaris

| NSM 项                   | SystemEDGE 项                                   | 说明                        |
|-------------------------|------------------------------------------------|---------------------------|
| ForkRequestssec         | kernelperf. forkRequestssec                    | 分支数目/秒。                   |
| FreePhysicalMemory      | performance.memoryStats.freePhysicalMemory     | 未使用的物理（实际）内存百分比。          |
| RequestsPage Insec      | performance.memoryStats.requestsPageInsec      | 每秒调入的页数。                  |
| RequestsPage Outsec     | performance.memoryStats.requestsPageOutsec     | 每秒调出的页数。                  |
| SystemUsageFree Memory  | performance.memoryStats.systemUsageFreeMemory  | 未分配的虚拟内存总量。（千字节）          |
| SystemUsageTotal Memory | performance.memoryStats.systemUsageTotalMemory | 系统上可用的虚拟内存总量。（千字节）        |
| SystemUsageUsed Memory  | performance.memoryStats.systemUsageUsedMemory  | 系统使用已用内存—使用着的虚拟内存的 KB 总数。 |
| ScanRatePage Scanssec   | performance.memoryStats.scanRatePageScanssec   | 内核检查可用内存页的频率。             |
| PagesIntransit          | kernelperf. pagesIntransit                     | 传送中阻挡的页面错误总数。             |
| PagesIntransitsec       | kernelperf. pagesIntransitsec                  | 传送中每秒阻挡的页面错误数。            |

| NSM 项                       | SystemEDGE 项                                        | 说明                                                                     |
|-----------------------------|-----------------------------------------------------|------------------------------------------------------------------------|
| RequestsPageFreeBytessec    | performance.memoryStats.requestsPageFreeBytessec    | 通过终止进程释放到内核的空间量。                                                       |
| RequestsPageFrees           | performance.memoryStats.requestsPageFrees           | 自系统启动以来释放到内核的页面总数。                                                     |
| RequestsPageFreessec        | performance.memoryStats.requestsPageFreessec        | 通过终止进程释放到内核的页面数。                                                       |
| RequestsPageInBytessec      | performance.memoryStats.requestsPageInBytessec      | 每秒页面调入的数据量。                                                            |
| RequestsPageOutBytessec     | performance.memoryStats.requestsPageOutBytessec     | 每秒页面调出的数据量。                                                            |
| RequestsPageReclaimBytessec | performance.memoryStats.requestsPageReclaimBytessec | 从可用列表回收的空间量（以页面为单位）。                                                   |
| RequestsPageReclaimssec     | performance.memoryStats.requestsPageReclaimssec     | 每秒回收的页数。                                                               |
| SystemUsageLockedMemory     | performance.memoryStats.systemUsageLockedMemory     | 锁定内存的千字节总数。                                                            |
| InputErrors                 | performance.network.ifEntry.<index>.inputErrors     | 输入错误的数目。这些错误通常表示故障硬件，可能是生成错误数据包的其他计算机，或是错误连接器或中断器。                     |
| blkss                       | diskStatsTable.<index>.blkss                        | 从该设备传输和传输到该设备的字节数（以 512 字节为单位）。                                        |
| rws                         | diskStatsTable.<index>.rws                          | 每秒从该设备传输和传输到该设备的数据量（读和写）。                                              |
| avwait                      | diskStatsTable.<index>.diskStatsAvWait              | 传送请求在设备队列中空闲等待的平均时间（毫秒）。                                               |
| InodesFree                  | devTable.<index>.devFfiles                          | 文件系统中的可用 inode 数（与文件系统上允许的文件和设备数相对应）。某些系统上的 inode 表是动态分配的，因此可用条目的数量很少。 |



| NSM 项      | SystemEDGE 项                   | 说明                     |
|------------|--------------------------------|------------------------|
| InodesUsed | devTable.<index>.devInodesUsed | 文件系统中正在使用的 inode 或文件数。 |

**Windows**

| NSM 项              | SystemEDGE 项                                  | 说明                                              |
|--------------------|-----------------------------------------------|-------------------------------------------------|
| AvgDiskBytesRead   | diskStatsTable.<index>.diskStatsAvgBytesRead  | 在读取操作期间从磁盘传输的平均字节数。                             |
| AvgDiskBytesWrite  | diskStatsTable.<index>.diskStatsAvgBytesWrite | 在写操作期间传输到磁盘的平均字节数。                              |
| AvgDisksecTransfer | diskStatsTable.<index>.diskStatsServiceTime   | 上一个度量期间在该磁盘上进行的操作的平均服务时间（毫秒）。也可以表示为：磁盘忙时间/传输次数。 |
| AvgDiskQueueLength | diskStatsTable.<index>.diskStatsQueueLength   | 上个度量期间该磁盘服务队列中等待的平均操作数。                         |
| DiskReadTime       | diskStatsTable.<index>.diskStatsReadTime      | 选定磁盘驱动器忙于处理读请求所用时间的百分比。                         |
| DiskWriteTime      | diskStatsTable.<index>.diskStatsWriteTime     | 选定磁盘驱动器忙于处理写请求所用时间的百分比。                         |
| DiskReadBytessec   | diskStatsTable.<index>.diskStatsReadBytessec  | 在读操作期间从磁盘传输字节的速率。                               |
| DiskWriteBytessec  | diskStatsTable.<index>.diskStatsWriteBytessec | 在写操作期间向磁盘传输字节的速率。                               |

**Solaris/Windows**

| NSM 项                       | SystemEDGE 项                                                             | 说明                     |
|-----------------------------|--------------------------------------------------------------------------|------------------------|
| CPU                         | cpuGroup.cpuTotalUtilPercent<br>cpuStatsEntry.<br>cpuStatsUtilPercent    | % 总数—用户和系统 CPU 的使用率之和。 |
| InterfaceTrafficIncomingsec | performance.network.<br>ifEntry. <index>.<br>interfaceTrafficIncomingsec | 每个接口卡每秒输入的数据包数。        |



| NSM 项                           | SystemEDGE 项                                                             | 说明                                                                                                                                                         |
|---------------------------------|--------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| InterfaceTraffic<br>Outgoingsec | performance.network.<br>ifEntry. <index>.<br>interfaceTrafficOutgoingsec | 每个接口卡每秒输出的数据包数目。                                                                                                                                           |
| inputpackets                    | performance. network.<br>ifEntry.<br><index>.inputpackets                | 接口上传输的输入数据包数目。                                                                                                                                             |
| outputpackets                   | performance. network.<br>ifEntry.<br><index>.outputpackets               | 接口上传输的输出数据包数目。                                                                                                                                             |
| blkss                           | diskStatsTable.<index>.<br>diskStatsBlkss                                | 从该设备传输和传输到该设备的字节数（以 512 字节为单位）。                                                                                                                            |
| rws                             | diskStatsTable.<index>.<br>diskStatsRWs                                  | 每秒从该设备传输和传输到该设备的数据量（读和写）。                                                                                                                                  |
| AllocatedSpace                  | devTable. <index>.<br>devAllocatedSpace                                  | 文件系统中的可用 MB 总数。                                                                                                                                            |
| FreeSpace                       | devTable. <index>.<br>devFreeSpace                                       | 文件系统上的可用 MB 数。该项可能不包括为超级用户保留的空间。                                                                                                                           |
| SpaceUsed                       | devTable.<index>.<br>devCapacity                                         | 文件系统正在使用的文件系统空间的百分比。对于大部分文件系统，该项不包括为超级用户保留的任何额外空间。因此，与报告分区上所有空间（其中一些空间可能对普通用户不可用）的“文件系统”组不同，该值可以有差异。该差异通常适用于 /opt、/var 或 /home 等文件系统，而不适用于 / 或 /tmp 等文件系统。 |
| UsedSpace                       | devTable.<index>.<br>devUsedSpace                                        | 文件系统中正在使用的 MB 数。                                                                                                                                           |

| NSM 项               | SystemEDGE 项                                  | 说明                                                                              |
|---------------------|-----------------------------------------------|---------------------------------------------------------------------------------|
| DiskTime, 忙碌        | diskStatsTable.<index>.diskStatsUtilization   | 上一个度量期间该磁盘的使用率（使用率百分比）。该项也可以表示为：<br>(磁盘忙时间/已用时间) * 100。<br>% 忙时间—设备忙于处理请求的时间部分。 |
| AvgDiskBytesRead    | diskStatsTable.<index>.diskStatsAvgBytesRead  | 在读取操作期间从磁盘传输的平均字节数。                                                             |
| AvgDiskBytesWrite   | diskStatsTable.<index>.diskStatsAvgBytesWrite | 在写操作期间传输到磁盘的平均字节数。                                                              |
| AvgDiskQueue Length | diskStatsTable.<index>.diskStatsQueueLength   | 上个度量期间服务队列中等待的平均操作数。                                                            |
| DiskReadBytessec    | diskStatsTable.<index>.diskStatsReadBytesSec  | 在读取操作期间从磁盘传输字节的速率。                                                              |
| DiskWriteBytessec   | diskStatsTable.<index>.diskStatsWriteBytesSec | 在写操作期间向磁盘传输字节的速率。                                                               |
| AvgDisksecTransfer  | diskStatsTable.<index>.diskStatsServiceTime   | 上一个度量期间在该磁盘上进行的操作的平均服务时间（毫秒）。该项可以表示为：磁盘忙时间/传输次数。                                |

## CA VMware vCenter 服务器性能指标

以下列表提供有关 VMware vCenter 服务器和 VMware ESX 主机的监控性能指标的信息。

### VMware VC ESX 主机

以下列表说明 VMware VC ESX 主机性能指标。

#### CPU 数

系统上的 CPU 总数。

#### 已注册的 VM 数

已注册的 VM 总数。

**运行的 VM 数**

运行的 VM 总数。

**系统正常运行时间**

系统已处于联机状态的总时间（毫秒）。

## VMware VC ESX 主机 CPU

以下列表说明 VMware VC ESX 主机 CPU 性能指标。

**CPU 使用率（百分比）**

服务器当前的 CPU 使用率。

**每个 CPU 的 CPU 使用率（百分比）**

服务器上每个 CPU 的当前 CPU 使用率。

**CPU 空闲秒数（百分比）**

基于轮询间隔 CPU 空闲状态所花费的平均时间的百分比。

**CPU 使用秒数（百分比）**

基于轮询间隔使用 CPU 所花费的平均时间的百分比。

**已配置的 CPU 份额**

分配给服务器的 CPU 份额数。

**已配置的 CPU 最大值（百分比）**

已配置的 CPU 最高保留阈值的百分比。

**已配置的 CPU 最小值（百分比）**

已配置的 CPU 最低保留阈值的百分比。

## VMware VC ESX 主机磁盘

以下列表说明 VMware VC ESX 主机磁盘性能指标。

**总线重置次数**

总线重置总次数。

**总线重置次数/秒**

每秒的总线重置次数。

**已中止的命令数**

已中止的命令总数。

**已中止的命令数/秒**

每秒中止的命令数。

**读取 KB/每秒**

每秒读取的 KB 数。

**写入 KB/每秒**

每秒写入的 KB 数。

**已读取 KB**

已读取的 KB 数。

**已写入 KB**

已写入的 KB 数。

**读取次数**

读取次数。

**写入次数**

写入次数。

**读取次数/秒**

每秒读取次数。

**写入次数/秒**

每秒写入次数。

## VMware VC ESX 主机内存

以下列表说明 VMware VC ESX 主机内存性能指标。

**可用内存 (KB)**

可用的服务器内存 (KB)。

**已配置的内存 (KB)**

已配置的服务器内存，以 KB 为单位。

**可用内存 (KB)**

可用的服务器内存，以 KB 为单位。

**可用物理内存 (KB)**

可用的物理内存 (KB)。

**内存使用率（百分比）**

服务器当前的内存使用率。

**已使用的内存 (KB)**

已使用的服务器内存，以 KB 为单位。

**内存膨胀 (KB)**

内存控制使用的内存量。

## VMware VC ESX 主机网络

以下列表说明 VMware VC ESX 主机网络性能指标。

**接收的数据包数**

接收的数据包总数。

**传输的数据包数**

传输的数据包数。

**传输的数据包数/秒**

轮询间隔内每秒传输的数据包数。

**接收的数据包数/秒**

轮询间隔内每秒接收的数据包数。

**已接收的 KB**

已接收的 KB 数。

**已传输的 KB**

传输的总字节数 (KB)。

**接收的 KB/每秒**

基于轮询间隔每秒接收的 KB 数。

**传输的 KB/每秒**

基于轮询间隔每秒传输的 KB 数。

## VMware VC ESX 主机每个 CPU

以下列表说明 VMware VC ESX 主机每个 CPU 的性能指标。

**每个 CPU 的 CPU 使用率（百分比）**

服务器上每个 CPU 的当前 CPU 使用率。

## VMware VC VM

以下列表说明 VMware VC VM 性能指标。

### 已连接的用户数

此 VM 上的已连接的用户数。

### VM 系统正常运行时间

系统已处于联机状态的总时间（毫秒）。

### VM 系统心跳

VM 的响应状态指示器。

### VM 系统 OS 状态

OS 的操作模式。

### VM 系统虚拟 CPU 数

已分配的虚拟 CPU 的数量。

### VM 系统 VM 状态

VM 的状态。

## VMware VC VM CPU

以下列表说明 VMware VC VM CPU 性能指标。

### CPU 份额（百分比）

VM 的 CPU 性能统计的百分比。

### CPU 超额秒数（百分比）

在轮询间隔内使用 CPU 的平均超额时间百分比。

### CPU 使用秒数（百分比）

在轮询间隔内使用 CPU 的平均时间百分比。

### CPU 就绪秒数（百分比）

在轮询间隔内 CPU 处于就绪状态的平均时间百分比。

### CPU 等待秒数（百分比）

在轮询间隔内 CPU 处于等待状态的平均时间百分比。

### 已配置的 CPU 份额

分配给 VM 的 CPU 份额数。

**活动的 CPU 份额**

VM 的 CPU 性能统计信息 (MHz)。

**CPU 份额百分比**

VM 的 CPU 性能统计的百分比。

**已配置的 CPU 最大值（百分比）**

已配置的 CPU 最高保留阈值的百分比。

**已配置的 CPU 最小值（百分比）**

已配置的 CPU 最低保留阈值的百分比。

**CPU 关联**

VM CPU 关联的状态。

**VM 就绪时间百分比**

VM 已就绪，但无法按照计划在物理 CPU 上运行的时间百分比。

**VM CPU 分配百分比**

VM CPU 分配百分比。

## VMware VC VM 磁盘

以下列表说明 VMware VC VM 磁盘性能指标。

**总线重置次数**

总线重置总次数。

**总线重置次数/秒**

每秒的总线重置次数。

**已中止的命令数**

已中止的命令总数。

**已中止的命令数/秒**

每秒中止的命令数。

**读取 KB/每秒**

每秒读取的 KB 数。

**写入 KB/每秒**

每秒写入的 KB 数。

**已读取 KB**

已读取的 KB 数。

**已写入 KB**

已写入的 KB 数。

**读取次数**

读取次数。

**写入次数**

写入次数

**读取次数/秒**

每秒读取次数。

**写入次数/秒**

每秒写入次数。

**份额**

用于资源调度的磁盘份额数。

## VMware VC VM 内存

以下列表说明 VMware VC VM 内存性能指标。

**可用内存 (KB)**

可用的 VM 内存 (KB)。

**已配置的内存 (KB)**

配置的 VM 内存 (KB)。

**可用内存 (KB)**

可用的 VM 内存 (KB)。

**已使用的内存 (KB)**

已使用的 VM 内存 (KB)。

**内存使用率（百分比）**

VM 的当前内存使用率。

**开销内存 KB**

开销的内存量 (KB)。

**共享内存 (KB)**

当前共享的内存量 (KB)。

**交换内存 KB**

当前交换的内存量 (KB)。



**内存大小 KB**

VM 的内存总量 (KB)。

**配置的内存份额**

VM 专为资源竞争配置的内存份额数。

**配置的最小内存 KB**

配置的内存最小保留阈值 (KB)。

**配置的最大内存 KB**

配置的内存最大保留阈值 (KB)。

**内存膨胀 (KB)**

内存控制使用的内存量。

**内存膨胀目标 KB**

内存控制可以使用的内存量。

**分配的内存份额**

已分配 VM 内存份额的状态。

**最小内存**

VM 的当前内存使用量下限 (MB)。

**最大内存**

VM 的当前内存使用量上限 (MB)。

**实际内存**

虚拟机的 VM 内存大小 (MB)。

**活动内存**

VM 活动内存状态 (MB)。

**内存百分比**

VM 内存的状态（百分比）。

**内存控制**

VM 内存控制 (MB)。

**内存开销**

VM 内存开销 (MB)。

## VMware VC VM 网络

以下列表说明 VMware VC VM 网络性能指标。

### 接收的数据包数

接收的数据包数。

### 传输的数据包数

传输的数据包数。

### 传输的数据包数/秒

轮询间隔内每秒传输的数据包数。

### 接收的数据包数/秒

轮询间隔内每秒接收的数据包数。

### 已接收的 KB

已接收的 KB 数。

### 已传输的 KB

已传输的 KB 数。

### 接收的 KB/每秒

基于轮询间隔每秒接收的 KB 数。

### 传输的 KB/每秒

基于轮询间隔每秒传输的 KB 数。

## VMware VC 群集

以下列表说明 VMware VC 群集性能指标。

### CPU 公平性

分布式 CPU 分配的公平性。

### 内存公平性

分布式内存分配的公平性。

### 有效 CPU MHz

可用的 CPU 资源量 (MHz)。

### 有效内存 KB

可用的内存资源量 (KB)。

**故障转移容错**

可以接受的故障次数。

**VMotion 比率**

DR 的迁移阈值。

## VMware VC 数据存储

以下列表说明 VMware VC 数据存储性能指标。

**可用空间百分比**

数据存储上的可用空间百分比。

**可用空间 KB**

数据存储上的可用空间 (KB)。

**总空间 KB**

数据存储上的总空间 (KB)。

**已用空间百分比**

数据存储上的已用空间百分比。

**已用空间 KB**

数据存储上的已用空间 (KB)。

## VMware VC 计算资源

以下列表说明 VMware VC 计算资源性能指标。

**总体状态**

计算资源的状态。

**有效 CPU MHz**

可用于运行 VM 的 CPU 资源量 (MHz)。

**有效内存 KB**

可用于运行 VM 的内存资源量 (KB)。

**总 CPU MHz**

所有服务器中的 CPU 资源量 (MHz)。

**总内存 KB**

所有服务器中的内存量 (KB)。

## VMware VC 资源池

以下列表说明 VMware VC 资源池性能指标。

### 总体状态

资源池的状态。

### 内存总体使用率（百分比）

资源池的总体内存使用率（百分比）。

### CPU 总体使用率（百分比）

资源池的总体 CPU 使用率（百分比）。

### CPU 扩展保留

在具有可扩展保留(1)的资源池中，如果父资源池具有无限的资源，则资源池的保留的增长可以超出指定值。不可扩展(0)保留称为固定保留。

### CPU 最大

资源池的最大 CPU (MHz)。

### CPU 最小

资源池的最小 CPU (MHz)。

### CPU 份额

分配给资源池的 CPU 份额。

### 内存扩展保留

在具有可扩展保留(1)的资源池中，如果父资源池具有无限的资源，则资源池的保留的增长可以超出指定值。不可扩展(0)保留称为固定保留。

### 最大内存

资源池的最大内存 (MB)。

### 最小内存

资源池的最小内存 (MB)。

### 内存份额

分配给资源池的内存份额。

### CPU 最大使用率

当前 CPU 使用率的上限。

### CPU 使用率（百分比）

资源池 CPU 使用百分比。

**已使用 CPU 保留**

用于资源池的保留 CPU。

**用于 VM 的 CPU 保留**

该资源池或任何其子资源池中用于运行 VM 的 CPU 保留。

**用于池的 CPU 未保留**

子资源池的未保留 CPU。

**用于 VM 的 CPU 未保留**

子虚拟内存的未保留 CPU。

**内存最大使用量**

资源池的最大内存使用量 (MB)。

**内存使用量百分比**

资源池内存使用量百分比。

**已使用内存保留**

为所有子资源池保留的内存。

**用于 VM 的内存保留**

在该资源池或任何其子资源池中用于运行 VM 的保留内存。

**用于池的内存未保留**

未为资源池保留的内存。

**用于 VM 的内存未保留**

未为子 VM 保留的内存。

## VMware VC Server

以下列表说明 VMware vCenter 服务器性能指标。

**合计 CPU 使用率（百分比）**

vCenter 服务器中所有主机的合计 CPU 使用率。

**合计内存使用率（百分比）**

vCenter 服务器中所有主机的合计内存使用率。

## CA Citrix XenServer 性能度量标准

以下列表提供了用于 Citrix XenServers 监控的性能度量标准的相关信息。

## Citrix Xen 主机

以下列表提供了 Citrix XenServer 主机性能度量标准的说明。

### 总内存 (MB)

服务器的可用总内存 (MB)。

### 可用内存 (MB)

服务器的可用内存 (MB)。

### 内存使用 (MB)

服务器的内存使用 (MB)。

### 内存使用率 (%)

服务器的内存使用率 (%)。

### CPU 使用率 (%)

服务器的 CPU 使用率 (%)。

## Citrix Xen 主机 pNic

以下列表提供了 Citrix XenServer 主机 pNic 性能度量标准的说明。

### 读取速率 (KB/秒)

物理网络接口的读取速率 (KB/秒)。

### 写入速率 (KB/秒)

物理网络接口的写入速率 (KB/秒)。

### 速度 (KB/秒)

物理网络接口的速度 (KB/秒)。

## Citrix Xen 主机存储

以下列表提供了 Citrix XenServer 主机存储性能度量标准的说明。

### 虚拟分配 (MB)

存储库中所有虚拟磁盘映像的总大小 (MB)。

### 物理使用量 (MB)

存储库的物理使用量 (MB)。

**SR 容量 (MB)**

存储库的总容量 (MB)。

**SR 可用空间 (MB)**

存储库的可用空间 (MB)。

## Citrix Xen VM

以下列表提供了 Citrix XenServer VM 性能度量标准的说明。

**VM vCPU 使用率 (%)**

该 VM 的 CPU 使用率 (%)。

**内存使用 (MB)**

该 VM 的内存使用量 (MB)。

**内存使用率 (%)**

该 VM 的内存使用率 (%)。

**来宾实际内存 (MB)**

来宾操作系统的实际内存 (MB)。

**来宾总内存 (MB)**

来宾操作系统的总内存 (MB)。

**来宾可用内存 (MB)**

来宾操作系统的可用内存 (MB)。

## Citrix Xen VM vDisk

以下列表提供了 Citrix XenServer VM vDisk 性能度量标准的说明。

**读取速率 (KB/秒)**

虚拟磁盘的读取速率 (KB/秒)。

**写入速率 (KB/秒)**

虚拟磁盘的写入速率 (KB/秒)。

**虚拟大小 (MB)**

虚拟磁盘的大小 (MB)。

**物理使用量 (MB)**

存储库中虚拟磁盘的实际物理使用量 (MB)。

## Citrix Xen VM vNic

以下列表提供了 Citrix XenServer VM vNic 性能度量标准的说明。

### 读取速率 (KB/秒)

虚拟网络接口的读取速率 (KB/秒)。

### 写入速率 (KB/秒)

虚拟网络接口的写入速率 (KB/秒)。

## CA Windows Microsoft 群集性能度量标准

以下列表提供有关针对 Windows Microsoft 群集服务器监控的度量标准的信息。

### 运行时间百分比: Windows MSCS 群集

群集已由 CA Virtual Assurance 评估且处于“运行”状态的时间百分比。

### 运行时间百分比: Windows MSCS 节点

群集节点已由 CA Virtual Assurance 评估且处于“运行”状态的时间百分比。

### 运行时间百分比: Windows MSCS 资源组

群集资源组已由 CA Virtual Assurance 评估且处于“运行”状态的时间百分比。

### 运行时间百分比: Windows MSCS 资源

群集资源已由 CA Virtual Assurance 评估且处于“运行”状态的时间百分比。

### 运行时间百分比: Windows MSCS 网络接口

群集网络接口已由 CA Virtual Assurance 评估且处于“运行”状态的时间百分比。

### 运行时间百分比: Windows MSCS 网络

群集网络已由 CA Virtual Assurance 评估且处于“运行”状态的时间百分比。



## CA Windows 性能度量标准

以下列表提供有关针对 Windows 服务器监控的度量标准的信息。

**物理磁盘：磁盘每秒读取的字节数**

从物理磁盘读取数据的速率。

**物理磁盘：磁盘每秒写入的字节数**

向物理磁盘写入数据的速率。

**物理磁盘：磁盘时间百分比**

磁盘驱动器忙于服务读取或写入请求所用时间的百分比。

**处理器时间百分比**

处理器运行非空闲线程所用时间的百分比。

**内存：可用 MB**

可用于在计算机上运行的进程的物理内存量（兆字节）。

**内存：页/秒**

从磁盘读取或向磁盘写入页面以解决硬页面错误的速率。

**内存：正在使用的已确认字节百分比**

正在使用的可能物理内存总量的百分比，该物理内存已在分页文件中保留了空间（如果必须写入到磁盘）。

**网络接口：每秒字节总数**

每个网络适配器发送与接收字节（包括消息成帧字符）的速率。

**网络接口：当前带宽**

估计的网络接口当前带宽（位/秒）。例如，10000 表示 10 Mbps。

**网络接口：每秒数据包数**

在网络接口发送与接收数据包的速率。

## Cisco Unified Computing System (UCS) 性能度量标准

以下列表提供有关针对 Cisco UCS 监控的度量标准的信息。

### **caBladeServerCpuLoad**

该刀片服务器的处理器不空闲的平均百分比（一分钟内）。

### **caBladeServerMemorySize**

特定刀片服务器的总可用读/写内存 (MB)（通常是 RAM）。

### **caBladeServerCurrentAllocatedMemory**

刀片服务器当前已分配的读/写内存量 (MB)（正在使用的内存）。

### **caBladeServerTotalLocalStorageSize**

本地磁盘存储大小 (MB)（不包括联网驱动器）。

### **caBladeServerTotalUsedLocalStorage**

当前已分配和占用的本地磁盘存储大小 (MB)。

### **caBladeIfHCInUcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCInMulticastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCInBroadcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCOutUcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCOutMulticastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCOutBroadcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCInOctets**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfHCOutOctets**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

### **caBladeIfInOctets**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfOutOctets**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfInUnknownProtos**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfInErrors**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfOutErrors**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfMtu**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfInUcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfInMulticastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfInBroadcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfInDiscards**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfOutUcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfOutMulticastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfOutBroadcastPkts**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caBladelfOutDiscards**

类似于 RFC 2863 的 ifVHCPacketGroup 对象。

**caUcsChassisFanAmbientTemp**

摄氏度。

**caUcsChassisPsuAmbientTemp**

PSU 感测到的温度（摄氏度）。

**caUcsChassisFabricExtenderAmbientTemp**

摄氏度。

#### **caUcsCpuTemp**

CPU 温度传感器当前可用的温度读数。

#### **caUcsMotherboardIoSensorTemp**

摄氏度。

#### **caUcsMotherboardRearSensorTemp**

摄氏度。

## 远程监控性能指标

一下列表说明远程监控性能指标（包括即用型）。指标名称与 CIM 属性名一致（添加 CPU\_、Disk\_、Net\_ 等前缀）以表示正查询的 CIM 类。

#### **CPU\_PercentIdle**

在上一个时间间隔中，处理器（总计）空闲时间的百分比。

#### **Disk\_PercentIdle**

在上一个时间间隔中，该磁盘空闲时间的百分比。

#### **FSys\_FreeMBDecrease**

在上一个时间间隔中，该磁盘可用空间的减少量 (MB)。

#### **FSys\_FreeMB**

在上一个轮询中，该磁盘的可用空间 (MB)。

#### **Mem\_PercentUsed**

在上一个轮询中，用于已确认的内存的确认限制百分比。

#### **Net\_IPAddress**

系统的 IP 地址；该条目为必需项。

#### **Net\_MACAddress**

系统的 MAC 地址；该条目为必需项。

#### **Net\_QueueLength**

在上一个轮询中，此接口的输出队列长度（数据包数）。

#### **Proc\_PercentCPU**

在上一个时间间隔中，Top1 进程所使用的的时间的百分比。

#### **Proc\_PercentMemory**

在上一个轮询中，Top1 进程所使用的物理内存的百分比。

#### **Event\_SystemError**

在上一个时间间隔中，写入系统事件日志的错误事件的数目。

**Srvc\_StoppedAuto**

在上一个轮询中，使用自动启动类型的停止服务的数目。

**Sys\_LastBootTime**

上次启动系统的时间；该条目为必需项。

**Sys\_LastLocalTime**

在上一个轮询中，系统的本地日期和时间。

**Sys\_OSInfo**

有关安装的操作系统的信息。

**Sys\_PhysMemKB**

物理内存的总计大小 (KB)。

**BIOS\_SerialNumber**

BIOS 序列号。

**BIOS\_Version**

BIOS 版本。

**Disk\_QueueLength**

在上一个时间间隔中，为该磁盘排队的读写请求的平均数。

**Disk\_ReadPerSec**

在上一个时间间隔中，从该磁盘读取的平均时间（秒）。

**Disk\_WritePerSec**

在上一个时间间隔中，向该磁盘写入的平均时间（秒）。

**Mem\_FreePages**

在上一个轮询中，空闲页面表项的数目。

**Mem\_NonPagedMB**

在上一个轮询中，非分页池（OS 使用的不可分页的内存）的大小 (MB)。

**Mem\_NonPagedMB\_3GB**

在上一个轮询中，非分页池（OS 用的不可分页的内存）的大小 (MB)。

**Mem\_PagedMB**

在上一个轮询中，分页池（OS 使用的可分页的内存）的大小 (MB)。

**Mem\_PagedMB\_3GB**

在上一个轮询中，分页池（OS 使用的可分页的内存）的大小 (MB)。

**Mem\_PagingPerSec**

在上一个时间间隔中，页面读写到磁盘以解决页面错误的每秒比率（页面数）。

**Mem\_FreeMB**

在上一个轮询中，可用来运行进程的物理内存 (MB)。

**Net\_PercentBusy**

在上一个时间间隔中，用于通过该接口发送与接收的带宽百分比。

**Sys\_Has3GBSwitch**

可以分配给进程的最大内存 (KB)。

**Sys\_Is64bit**

OS 体系结构（例如 32 位、64 位 Intel 或 64 位 AMD）。

**Sys\_OSType**

OS 类型。

**Disk\_AvgDiskBytesPerRead**

在上一个时间间隔的读取操作期间，从该磁盘传输的平均字节数。

**Disk\_AvgDiskBytesPerWrite**

在上一个时间间隔的写入操作期间，传输到该磁盘的平均字节数。

**Disk\_AvgDiskReadQueueLength**

在上一个时间间隔中，为该磁盘排队的读取请求的平均数。

**Disk\_AvgDiskWriteQueueLength**

在上一个时间间隔中，为该磁盘排队的写入请求的平均数。

**Disk\_DiskReadsPersec**

在上一个时间间隔中，每秒从该磁盘读取的比率。

**Disk\_DiskWritesPersec**

在上一个时间间隔中，每秒写入到该磁盘的比率。

**Disk\_PercentDiskReadTime**

在上一个时间间隔中，该磁盘忙于满足读取请求的时间的百分比。

**Disk\_PercentDiskWriteTime**

在上一个时间间隔中，该磁盘忙于满足写入请求的时间的百分比。

**Disk\_SplitIOPerSec**

在上一个时间间隔中，单个 I/O 请求每秒向该磁盘拆分为多个请求的比率。

**FSys\_PercentFreeSpace**

在上一个轮询中，该磁盘上可用空间的百分比。

**Net\_PacketsOutboundErrors**

在上一个轮询中，此接口上错误的出站数据包的数目。

**Net\_PacketsReceivedDiscarded**

在上一个轮询中，此接口上丢弃的进站数据包的数目。

**Net\_PacketsReceivedErrors**

在上一个轮询中，此接口上错误的进站数据包的数目。

**Net\_PacketsReceivedNonUnicastPersec**

在上一个时间间隔中，该接口上每秒接收广播和多播数据包的比率。

**Net\_PacketsReceivedUnicastPersec**

在上一个时间间隔中，该接口上每秒接收单播数据包的比率。

**Net\_PacketsSentNonUnicastPersec**

在上一个时间间隔中，该接口上每秒请求以发送的广播和多播数据包的比率。

**Net\_PacketsSentUnicastPersec**

在上一个时间间隔中，该接口上每秒请求以发送的单播数据包的比率。





# 索引

## 符号

- ! 命令—调用命令或子进程 (Cmdlet) - 48
- !! 命令—调用命令或子进程并自动捕获输出 (Cmdlet) - 47
- ? 命令—将输出写入到 stdout (Cmdlet) - 49
- ?? 命令—将列表中的输出写入到 stdout (Cmdlet) - 49

## A

- accept 命令—从 stdin 读取字符串并将其分配给变量 (Cmdlet) - 50
- arrdump 命令—显示数组 (Cmdlet) - 51
- ASSERT 命令—比较数学表达式 (Funclet) - 52
- AutoShell - 19
- AutoShell 功能 - 106
- AutoShell 可加载模块的命令参考 - 155
- AutoShell 类 - 130
- AutoShell 核心参考 - 43

## B

- base64Decode—解码 base64 编码的字符串 (函数) - 107
- base64Encode—对字符串执行 base64 编码 (函数) - 107

## C

- CA Cisco UCS 命令 - 605
- CA Citrix XenServer 性能度量标准 - 822
- CA HP UNIX 性能度量标准 - 785
- CA Hyper-V CLI 命令 - 454
- CA IBM AIX 性能度量标准 - 786
- CA IBM HACMP 性能度量标准 - 792
- CA IBM LPAR AutoShell 命令 - 292
- CA IBM LPAR CLI 命令 - 523
- CA IBM LPAR 性能度量标准 - 787
- CA IBM pSeries 性能度量标准 - 788
- CA IBM pSeries 逻辑分区性能度量标准 - 787
- CA Linux 性能度量标准 - 794
- CA Microsoft Cluster Server CLI 命令 - 560
- CA Microsoft Hyper-V AutoShell 命令 - 194
- CA Microsoft Hyper-V 性能指标 - 794

- CA Oracle Solaris Zones AutoShell 命令 - 318
- CA Oracle Solaris Zones 性能度量标准 - 796
- CA Oracle Solaris 性能度量标准 - 795
- CA Solaris Zones CLI 命令 - 578
- CA SystemEDGE 性能指标 - 801
- CA SystemEDGE 增强的度量标准 - 804
- CA Technologies 产品引用 - 3
- CA VMware vCenter Server CLI 命令 - 373
- CA VMware vCenter 服务器 AutoShell 命令 - 156
- CA VMware vCenter 服务器性能指标 - 810
- CA Windows Microsoft 群集性能度量标准 - 825
- CA Windows 性能度量标准 - 825
- caaipsecurity 命令--指定 Web 服务安全检  
查的凭据 - 685
- caismutility 命令 -- 将功能添加到策略配置  
UI 中 - 724
- cat、type 命令—显示文本文件 (Cmdlet) - 53
- cd、chdir 命令—更改目录 (cmdlet、  
funclet) - 54
- Cisco Unified Computing System (UCS) 性能度  
量标准 - 826
- Citrix Xen VM - 823
- Citrix Xen VM vDisk - 824
- Citrix Xen VM vNic - 824
- Citrix Xen 主机 - 822
- Citrix Xen 主机 pNic - 823
- Citrix Xen 主机存储 - 823
- copy、cp 命令—复制文件和目录 (Funclet) -  
55
- CPU 度量标准 - 791
- createdpmdefaultgroup 命令—创建或更新  
服务 - 686
- curDir—检索当前目录 (函数) - 108

## D

- DBG\_PROMPT 命令—中断脚本执行以进行  
检查 (Cmdlet) - 57
- DBG\_PROMPT\_LINE 命令—中断脚本执行以  
进行带行号的检查 (Cmdlet) - 58
- del、rm 命令—删除文件 (Funclet) - 59

---

dir、ls 命令—获取文件和目录信息 (Funclet) - 61

dpmcda cancelnetworkscan 命令—取消网络发现 - 658

dpmcda deletenetwork 命令—删除网络 - 659

dpmcda discoverhost 命令—发现主机 - 659

dpmcda discovernetwork 命令—发现网络 - 660

dpmce ceconfig -configuremetric 命令—配置度量标准 - 662

dpmce ceconfig -configuremetricfilter 命令—配置度量标准筛选 - 664

dpmce ceconfig -disablemetric 命令—禁用度量标准 - 667

dpmce ceconfig -disablemetricforoverall 命令—禁用度量标准并将其排除在总体计算之外 - 668

dpmce ceconfig -enablemetric 命令—启用度量标准 - 669

dpmce ceconfig -enablemetricforoverall 命令—启用度量标准并将其包括在总体计算中 - 670

dpmce ceconfig -getglobal 命令—检索数据中心间隔 - 671

dpmce ceconfig -getmetriclist 命令—检索度量标准列表 - 673

dpmce ceconfig -getnode 命令—检索主机收集间隔 - 672

dpmce ceconfig -refreshmetriclist 命令—重试度量标准收集 - 674

dpmce ceconfig -setalias 命令—设置别名 - 675

dpmce ceconfig -setglobal 命令—设置数据中心间隔 - 676

dpmce ceconfig -setnode 命令—设置主机间隔 - 678

dpmce ceconfig -updatemetric 命令—更新度量标准 - 679

dpmcmd run 命令—运行脚本 - 686

dpmeventmanager create\_event 命令—创建事件 - 681

dpmeventmanager get\_events 命令—检索事件 - 682

dpmhd addcomment 命令—使用注释更新票单 - 696

dpmhd close 命令—关闭帮助台票单 - 697

dpmhd getattribute 命令—获取票单的属性值 - 698

dpmhd getstatus 命令—获取票单的状态 - 699

dpmhd open 命令—打开帮助台票单 - 700

dpmhd setattribute 命令—设置票单的属性 - 701

dpmhd setstatus 命令—设置票单的状态 - 702

dpmhv-addVMNic (cmdlet) 命令—将网络接口控制器添加到 VM 中 - 194

dpmhv-addVMSCSIController (cmdlet) 命令—将 SCSI 控制器添加到 VM 中 - 195

dpmhv-changeVMState (cmdlet) 命令—更改 VM 的状态 - 196

dpmhv-createTemplateFromVM (Funclet) 命令—从 VM 创建模板 - 198

dpmhv-createVirtDisk (Funclet) 命令—创建虚拟磁盘 - 199

dpmhv-createVirtFloppy (Cmdlet) 命令—创建虚拟软盘 - 200

dpmhv-createVM (Funclet) 命令—创建 VM - 201

dpmhv-createVMFromTemplate (Funclet) 命令—使用模板创建 VM - 205

dpmhv-createVMFromTemplateEx (Funclet) 命令—使用模板创建 VM - 210

dpmhv-destroyTemplate (Cmdlet) 命令—删除模板 - 218

dpmhv-destroyVM (Cmdlet) 命令—删除 VM - 219

dpmhv-expandVirtDisk (Funclet) 命令—扩展虚拟磁盘 - 220

dpmhv-exportVM (Funclet) 命令—导出 VM - 221

dpmhv-exportVMEx (Funclet) 命令—导出 VM - 222

dpmhv-getHostSwitches (Funclet) 命令—显示 Hyper-V 主机上的主机开关 - 224

dpmhv-getJobInfo (Funclet) 命令—显示作业信息 - 227

dpmhv-getJobStatus (Funclet) 命令—显示作业状态 - 232

---

dpmhv-getPhysDisks (Funclet) 命令-显示所有物理磁盘 - 232

dpmhv-getVersion (Funclet) 命令-显示 Web 服务版本 - 236

dpmhv-getVMInfo (Funclet) 命令-显示所有 VM - 236

dpmhv-getVMPProperties (Funclet) 命令-显示 VM 属性 - 241

dpmhv-getVMState (Funclet) 命令-显示 VM 状态 - 242

dpmhv-importVM (Funclet) 命令-导入 VM - 244

dpmhv-importVMEx (Funclet) 命令-复制和导入 VM - 245

dpmhv-setSysprepProperties (Cmdlet) 命令-创建虚拟磁盘 - 247

dpmhv-setVMClustered (Cmdlet) 命令-在群集中设置 VM - 252

dpmhv-setVMDisk (Funclet) 命令-在驱动器控制器上创建驱动器或磁盘 - 253

dpmhv-setVMPProperties (Cmdlet) 命令-设置 VM 属性 - 255

dpmhv-SetVMVirtFloppy (Cmdlet) 命令-将虚拟软盘映像连接到 VM 软盘驱动器 - 257

dpmhv-ShowClusterSharedVolumes (Funclet) 命令-显示所有群集共享卷 - 258

dpmhv-ShowDirectories (Funclet) 命令-显示所有目录 - 259

dpmhv-ShowHosts (Funclet) 命令-显示所有 Hyper-V 服务器 - 261

dpmhv-ShowLogicalDisks (Funclet) 命令-显示所有逻辑磁盘 - 268

dpmhv-ShowSCVMMHardwareProfiles (Funclet) 命令-显示所有 SCVMM 硬件配置文件 - 271

dpmhv-ShowSCVMMOSProfiles (Funclet) 命令-显示所有来宾 OS 配置文件 - 276

dpmhv-ShowTemplates (Funclet) 命令-显示 Hyper-V 主机上的模板 - 280

dpmhv-ShowVMDisks (Funclet) 命令-显示所有虚拟磁盘 - 282

dpmhv-showVMFloppy (Funclet) 命令-显示软盘驱动器的属性 - 284

dpmhv-showVMNICs (Funclet) 命令-显示所有网络接口卡 - 286

dpmhv-showVMSCSI (Funclet) 命令-显示所有 SCSI 控制器 - 288

dpmhyperv AddVMNic 命令-将网络适配器控制器添加到 VM 中 - 455

dpmhyperv AddVMSCSIController 命令-为 VM 创建 SCSI 控制器 - 456

dpmhyperv AgentGet 命令-返回代理值 - 457

dpmhyperv AgentSet 命令-设置代理值 - 459

dpmhyperv ChangeVMState 命令-更改 VM 的状态 - 460

dpmhyperv CreateTemplateFromVM 命令-从 VM 创建模板 - 462

dpmhyperv CreateVirtDisk 命令-创建虚拟磁盘映像 - 463

dpmhyperv CreateVirtFloppy 命令-创建虚拟软盘映像 - 465

dpmhyperv CreateVM 命令-创建 VM - 466

dpmhyperv CreateVMFromTemplate 命令-利用模板创建 VM - 473

dpmhyperv CreateVMFromTemplate 命令-通过 SCVMM 服务器模板创建 VM - 469

dpmhyperv DeleteTemplate 命令-删除模板 - 522

dpmhyperv DeleteVM 命令-删除 VM - 479

dpmhyperv ExpandVirtDisk 命令-扩展虚拟磁盘的大小 - 481

dpmhyperv ExportVM 命令-导出 VM - 482

dpmhyperv ExportVMEx 命令-导出 VM - 484

dpmhyperv GetHostSwitches 命令-获取主机的交换机 - 486

dpmhyperv GetJobErrorInfo 命令-获取作业的错误信息 - 487

dpmhyperv GetJobInfo 命令-获取作业的信息 - 488

dpmhyperv GetJobState 命令-获取作业的状态 - 489

dpmhyperv GetServerInfo 命令-获取所有受管服务器 - 490

dpmhyperv GetVersion 命令-获取 PMM 版本 - 491

dpmhyperv GetVMPProperties 命令-获取 VM 属性 - 492

dpmhyperv GetVMState 命令-获取 VM 状态 - 493

dpmhyperv ImportVM 命令-导入 VM - 495

- 
- dpmhyperv ImportVMEx 命令-复制并导入 VM - 496
- dpmhyperv SetClustered 命令-在群集中添加或删除 VM - 497
- dpmhyperv SetSysPrepProperties 命令-设置 Sysprep 属性 - 499
- dpmhyperv SetVMDisk 命令-将驱动器或磁盘分配给驱动器控制器 - 502
- dpmhyperv SetVMProperties 命令-设置 VM 的属性 - 504
- dpmhyperv SetVMVirtFloppy 命令-将虚拟软盘映像分配给 VM 软盘驱动器 - 507
- dpmhyperv ShowClusterSharedVolumes 命令-显示所有群集共享卷 - 508
- dpmhyperv ShowDirectories 命令-显示 Hyper-V 服务器上的目录属性 - 509
- dpmhyperv ShowLogicalDisks 命令-显示所有逻辑磁盘 - 510
- dpmhyperv ShowPhysDisks 命令-显示主机的物理磁盘 - 511
- dpmhyperv ShowSCVMMGuestOSProfiles 命令-显示来宾操作系统配置文件 - 513
- dpmhyperv ShowSCVMMHardwareProfiles 命令-显示所有 SCVMM 硬件配置文件 - 512
- dpmhyperv ShowTemplates 命令-显示所有 VM 模板 - 514
- dpmhyperv ShowVirtFloppy 命令-显示虚拟软盘驱动器列表 - 515
- dpmhyperv ShowVMDisks 命令-显示所有虚拟磁盘 - 516
- dpmhyperv ShowVMNics 命令-显示所有网络接口卡 - 517
- dpmhyperv ShowVMs (Funclet) 命令-获取所有 VM - 291
- dpmhyperv ShowVMs 命令-显示所有 VM - 518
- dpmhyperv ShowVMSCSIControllers 命令-显示所有 SCSI 控制器 - 519
- dpmhyperv ValidateAgentSNMPAccess 命令-验证对代理的 SNMP 访问 - 520
- dpmhyperv ValidateServerAccess 命令-检查服务器访问权限 - 521
- dpmimg image 命令-部署 Solaris 映像 - 704
- dpmimg imgjobcheck 命令-获取 Solaris 映像作业状态 - 708
- dpmix export|import|verify 命令-迁移系统数据 - 687
- dpmlpar cycle 命令-循环逻辑分区 - 524
- dpmlpar delete 命令-删除逻辑分区 - 527
- dpmlpar getresources 命令-获取 LPAR 资源 - 529
- dpmlpar image 命令-创建 IBM AIX LPAR - 531
- dpmlpar imgjobcheck 命令-检索 IBM AIX 映像作业的状态 - 543
- dpmlpar list 命令-列出 HMC 资源 - 544
- dpmlpar setresources 命令-更新内存资源 - 556
- dpmlpar setresources 命令-更新处理器资源 - 557
- dpmlpar setresources 命令-减少内存资源 - 552
- dpmlpar setresources 命令-减少处理器资源 - 554
- dpmlpar setresources 命令-添加内存资源 - 547
- dpmlpar setresources 命令-添加处理器资源 - 549
- dpmlpar-cycle 命令-循环使用逻辑分区 (Funclet) - 292
- dpmlpar-delete 命令-删除逻辑分区 (Funclet) - 295
- dpmlpar-getresources 命令-获取 LPAR 资源 (Funclet) - 296
- dpmlpar-image 命令-创建 IBM AIX LPAR (Funclet) - 298
- dpmlpar-imgjobcheck 命令-检索 IBM AIX 映像作业的状态 (Funclet) - 308
- dpmlpar-list 命令-列出 HMC 资源 (Funclet) - 309
- dpmlpar-setresources 命令-减去内存资源 (Funclet) - 315
- dpmlpar-setresources 命令-减去处理器资源 (Funclet) - 316
- dpmlpar-setresources 命令-增加内存资源 (Funclet) - 311
- dpmlpar-setresources 命令-增加处理器资源 (Funclet) - 313
- dpmmscs bringgrouponline 命令-使资源组联机 - 571
- dpmmscs bringresourceonline 命令-使资源联机 - 572

---

dpmmcs getclusterinfo 命令—获取群集的信息 - 560

dpmmcs getgroupresourcesinfo 命令—获取群集组的资源信息 - 561

dpmmcs getnetworkinterfacesinfo 命令—获取网络接口信息 - 562

dpmmcs getnetworksinfo 命令—获取网络信息 - 563

dpmmcs getnodesinfo 命令—获取群集节点信息 - 564

dpmmcs getresourcegroupsinfo 命令—获取群集资源组信息 - 565

dpmmcs getresourcesdependenciesinfo 命令—获取资源的依存关系信息 - 566

dpmmcs getresourcetypesinfo 命令—获取群集资源类型信息 - 567

dpmmcs getservicesinfo 命令—获取群集服务信息 - 568

dpmmcs movegroup 命令—将资源组移动到其他节点 - 569

dpmmcs pauseservice 命令—暂停群集服务 - 570

dpmmcs resumeservice 命令—恢复群集服务 - 573

dpmmcs startservice 命令—启动群集服务 - 574

dpmmcs stopservice 命令—停止群集服务 - 575

dpmmcs takegroupoffline 命令—使资源组脱机 - 576

dpmmcs takeresourceoffline 命令—使资源脱机 - 577

dpmpolicy createdefaultactions 命令—创建默认操作 - 710

dpmpolicy exportaction 命令—将操作导出到文件 - 711

dpmpolicy exportrule 命令—将规则导出到文件 - 712

dpmpolicy findbestmachine 命令—查找最佳服务器 - 713

dpmpolicy findcandidatemachine 命令—查找候选服务器 - 714

dpmpolicy getruleexeccount 命令—检索规则执行计数 - 716

dpmpolicy importaction 命令—从文件导入规则 - 718

dpmpolicy importaction 命令—从文件导入操作 - 717

dpmpolicy resetruleexeccount 命令—重置运行规则的重试次数 - 719

dpmpolicy runaction 命令—运行操作 - 720

dpmpolicy setruleexeclimit 命令—设置规则的限制 - 722

dpmresourcemgr addbaremetal 命令—添加物理计算机 - 727

dpmresourcemgr addcomputertonetwork 命令—向网络中添加服务器 - 728

dpmresourcemgr addipaddresspool 命令—添加 IP 地址池 - 729

dpmresourcemgr addmachinestoservice 命令—向服务中添加服务器 - 730

dpmresourcemgr addtocontainer 命令—将内容添加到容器中 - 731

dpmresourcemgr createcontainer 命令—创建容器 - 732

dpmresourcemgr createdefaultservice 命令—创建默认服务 - 733

dpmresourcemgr createipnetwork 命令—创建网络 - 734

dpmresourcemgr createresourcegroup 命令—创建资源组 - 736

dpmresourcemgr createresourcetemplate 命令—创建服务器组模板 - 737

dpmresourcemgr createservice 命令—创建服务 - 739

dpmresourcemgr deletebaremetal 命令—删除物理计算机 - 740

dpmresourcemgr deletecontainer 命令—删除容器 - 741

dpmresourcemgr deleteipnetworks 命令—删除网络 - 742

dpmresourcemgr deletesystem 命令—删除系统 - 743

dpmresourcemgr getdiscoverednetworks 命令—获取发现的网络 - 744

dpmresourcemgr getipaddresspools 命令—列出 IP 地址池 - 745

dpmresourcemgr getipnetworks 命令—列出网络 - 746



---

dpmresourcemgr getserverlist 命令-获取服务器的列表 - 747

dpmresourcemgr getservicelist 命令-获取服务的列表 - 748

dpmresourcemgr getsubnetdefaults 命令-列出子网默认值 - 749

dpmresourcemgr  
removecomputerfromipnetwork 命令-从网络中删除服务器 - 750

dpmresourcemgr removefromcontainer 命令-从容器中删除内容 - 751

dpmresourcemgr removeipaddresspool 命令-删除 IP 地址池 - 752

dpmresourcemgr  
removemachinesfromservice 命令-从服务中删除服务器 - 753

dpmresourcemgr removeresourcegroup 命令-删除资源组 - 754

dpmresourcemgr removeservice 命令-删除服务 - 755

dpmresourcemgr removesourcetemplate 命令-删除资源模板 - 756

dpmresourcemgr retrieveresourcegroup 命令-检索资源组 - 757

dpmresourcemgr retrieveresourcetemplate 命令-检索资源模板 - 758

dpmresourcemgr setcontainername 命令-设置容器名称 - 759

dpmresourcemgr setcontainerservice 命令-将容器设置为服务 - 760

dpmresourcemgr setmanagementip 命令-设置主机的管理 IP - 761

dpmresourcemgr setmanagementstatus 命令-设置主机的管理状态 - 762

dpmresourcemgr setserviceproperties 命令-设置服务属性 - 763

dpmresourcemgr updateipaddresspool 命令-更新 IP 地址池 - 772

dpmresourcemgr updateipnetwork 命令-更新网络 - 769

dpmresourcemgr updateresourcegroup 命令-更新资源组 - 765

dpmresourcemgr updateresourcetemplate 命令-更新资源模板 - 766

dpmresourcemgr updateservice 命令-更新服务 - 767

dpmresourcemgr updatesubnetdefaults 命令-更新网络默认值 - 770

dpmrpt schedulereport 命令-排定报告 - 693

dpmssh run 命令-运行远程外壳命令或脚本 - 695

dpmucs associateserviceprofile-将服务配置文件与刀片服务器相关联 - 607

dpmucs cloneserviceprofile-克隆服务配置文件 - 608

dpmucs createmacpool 命令-创建 MAC 地址池 - 609

dpmucs createorg 命令-创建组织 - 611

dpmucs createserverpool 命令-创建服务器池 - 612

dpmucs createserviceprofile 命令-创建服务配置文件 - 614

dpmucs createtemplatefromserviceprofile 命令-从服务配置文件创建模板 - 615

dpmucs createuuidpool 命令-创建 UUID 池 - 617

dpmucs createwwpool 命令-创建 WWNN 池 - 618

dpmucs deletemacpool 命令-删除 MAC 地址池 - 620

dpmucs deleteorg 命令-删除组织 - 621

dpmucs deleteserverpool 命令-删除服务器池 - 623

dpmucs deleteserviceprofile-删除服务配置文件 - 624

dpmucs deleteuuidpool 命令-删除 UUID 池 - 625

dpmucs deletewwpool 命令-删除 WWNN 池 - 626

dpmucs deployimage 命令-部署 Cisco UCS 映像 - 628

dpmucs disassociateserviceprofile 命令-解除关联服务配置文件 - 630

dpmucs failoverserviceprofile 命令-将服务配置文件与用于故障转移的刀片服务器相关联 - 631

dpmucs getblades 命令-获取刀片服务器信息 - 632

---

dpmucs getchassis 命令—获取机箱信息 - 634

dpmucs getpolicy 命令—从 UCS 管理器获取策略 - 635

dpmucs getserviceprofilestatus 命令—获取服务配置文件状态 - 636

dpmucs getswitches 命令—获取交换机信息 - 637

dpmucs gettopsystem 命令—获取系统信息 - 638

dpmucs listserviceprofiles 命令—列出服务配置文件 - 639

dpmucs powercycle 命令—重新启动刀片服务器 - 640

dpmucs renamepool 命令—重命名池 - 642

dpmucs serviceprofileupdate 命令—执行服务配置文件操作 - 643

dpmutil -amq 命令—配置 ActiveMQ 消息代理 - 646

dpmutil ecs 命令—配置 Windows 排定程序 - 647

dpmutil -hd 命令—配置帮助台 - 647

dpmutil -hmc 命令—配置 HMC 或 IVM 服务器 - 648

dpmutil mgmtldb 命令—配置管理数据库 - 649

dpmutil -perfdb 命令—配置性能数据库 - 650

dpmutil remotevcaim 命令—配置远程 VC AIM - 651

dpmutil -sc 命令—配置服务控制器 - 651

dpmutil -scvmmserver 命令—配置 Hyper-V 服务器 - 649

dpmutil -scvmmserver 命令—配置 SCVMM 服务器 - 652

dpmutil -snmp 命令—配置 SNMP - 653

dpmutil -systemcredential 命令—配置外部系统凭据 - 654

dpmutil -sysuser 命令—配置在服务启动时使用的本地凭据 - 655

dpmutil vcdefault 命令—配置默认 VMware vCenter 管理员用户 - 656

dpmutil vcserver 命令—配置 VMware vCenter - 656

dpmutil -vioserver 命令—配置 VIO 服务器 - 657

dpmutil 语法 - 645

dpmvc addesxhost 命令—添加 ESX 主机 - 374

dpmvc addvmvdisk 命令—添加虚拟磁盘 - 376

dpmvc addvmvnic 命令—添加虚拟 NIC - 378

dpmvc clone 命令—克隆 VM - 381

dpmvc createdatastore 命令—创建数据存储 - 389

dpmvc cycle 命令—循环 VM - 391

dpmvc datastore 命令—获取数据存储属性 - 394

dpmvc delete 命令—销毁 VM - 396

dpmvc distributedswitch 命令—管理虚拟分布式交换机 - 399

dpmvc entermaintenancemode 命令—设置为维护模式 - 403

dpmvc exitmaintenancemode 命令—退出维护模式 - 405

dpmvc faulttolerance 命令—指定容错操作 - 407

dpmvc getavailablescsidisks 命令—获取可用的 SCSI 磁盘 - 409

dpmvc gethosthba 命令—获取主机总线适配器 - 410

dpmvc getresources 命令—获取 VM 资源 - 411

dpmvc getshares 命令—获取 VM 份额 - 413

dpmvc getsnapshots 命令—获取 VM 快照 - 415

dpmvc imgjobcheck 命令—获取克隆作业状态 - 417

dpmvc migrate 命令—迁移 VM - 419

dpmvc removedatastore 命令—删除数据存储 - 422

dpmvc removevmvdisk 命令—删除虚拟磁盘 - 423

dpmvc removevmvnic 命令—删除虚拟 NIC - 425

dpmvc setcpu 命令—设置 VM 的 CPU 数目 - 427

dpmvc setmemory 命令—设置 VM 的内存大小 - 428

dpmvc setresources 命令—设置 VM 资源 - 430

dpmvc setshares 命令—设置 VM 份额 - 433

dpmvc snapshot 命令—管理快照 - 437

---

dpmvc templatetovm 命令-将模板转换成 VM - 442

dpmvc vapp 命令-管理 vApp - 188, 445

dpmvc virtualswitch 命令-管理虚拟交换机 - 449

dpmvc vmtotemplate 命令-将 VM 转换成模板 - 452

dpmvc-addesxhost 命令-添加 ESX 主机 (Funclet) - 156

dpmvc-addvmvdisk 命令-添加虚拟磁盘 (Funclet) - 157

dpmvc-addvmvnic 命令-添加虚拟 NIC (Funclet) - 158

dpmvc-clone 命令-克隆 VM (Funclet) - 160

dpmvc-createdatastore 命令-创建数据存储 (Funclet) - 166

dpmvc-cycle 命令-循环使用 VM (Funclet) - 167

dpmvc-datastore 命令-获取数据存储属性 (Funclet) - 168

dpmvc-delete 命令-销毁 VM (Funclet) - 169

dpmvc-distributedswitch 命令-管理虚拟分布式交换机 (Funclet) - 170

dpmvc-entermaintenancemode 命令-设置为维护模式 (Funclet) - 173

dpmvc-exitmaintenancemode 命令-退出维护模式 (Funclet) - 174

dpmvc-faulttolerance 命令-指定容错操作 (Funclet) - 175

dpmvc-getavailablescsidisks 命令-获取可用 SCSI 磁盘 (Funclet) - 176

dpmvc-gethosthba 命令-获取主机总线适配器 (Funclet) - 176

dpmvc-getresources 命令-获取 VM 资源 (Funclet) - 177

dpmvc-getshares 命令-获取 VM 份额 (Funclet) - 178

dpmvc-getsnapshots 命令-列出 VM 的快照 (Funclet) - 179

dpmvc-imgjobcheck 命令-获取克隆作业状态 (Funclet) - 180

dpmvc-migrate 命令-迁移 VM (Funclet) - 180

dpmvc-removedatastore 命令-删除数据存储 (Funclet) - 181

dpmvc-removevmvdisk 命令-删除虚拟磁盘 (Funclet) - 182

dpmvc-removevmvnic 命令-删除虚拟 NIC (Funclet) - 182

dpmvc-setresources 命令-设置 VM 资源 (Funclet) - 183

dpmvc-setshares 命令-设置 VM 份额 (Funclet) - 184

dpmvc-snapshot 命令-管理快照 (Funclet) - 185

dpmvc-templatetovm 命令-将模板转换为 VM (Funclet) - 187

dpmvc-virtualswitch 命令-管理虚拟交换机 (Funclet) - 191

dpmvc-vmtotemplate 命令-将 VM 转换为模板 (Funclet) - 193

dpmvc-wait 命令-等待作业的时间 (Funclet) - 193

dpmzone associateproject 命令-将任务与项目关联 - 578

dpmzone clonezone 命令-克隆区域 - 579

dpmzone createandinstallzone 命令--创建和安装区域 - 580

dpmzone createpool 命令-创建资源池 - 583

dpmzone createproject 命令-创建项目 - 584

dpmzone createzone 命令-创建区域 - 586

dpmzone customcommand 命令-运行自定义命令 - 588

dpmzone deleteproject 命令-删除项目 - 589

dpmzone deletezone 命令-删除区域 - 590

dpmzone installzone 命令-安装区域 - 591

dpmzone movezone 命令-移动区域 - 592

dpmzone rebootzone 命令-重新启动区域 - 593

dpmzone renamezone 命令-重命名区域 - 594

dpmzone setpoolscheduler 命令-设置池排定程序 - 595

dpmzone setprocessorset 命令-设置处理器集的最大 CPU 数目 - 596

dpmzone showinterfaces 命令-显示网络接口 - 597



---

dpmzone showpools 命令—显示资源池 - 598

dpmzone showprojects 命令—列出项目的属性 - 598

dpmzone showresourceset 命令—显示资源集 - 599

dpmzone showtasks 命令—列出项目中的任务 - 600

dpmzone showzoneinterfaces 命令—显示区域的网络接口 - 600

dpmzone showzoneprocess 命令—显示区域中的进程 - 601

dpmzone showzoneprojects 命令—显示区域中的项目 - 602

dpmzone showzones 命令—显示区域 - 602

dpmzone startzone 命令—启动区域 - 603

dpmzone stopzone 命令—停止区域 - 604

dpmzone uninstallzone 命令—卸载区域 - 605

dpmzone-associateproject 命令—将任务与项目关联 (Funclet) - 318

dpmzone-clonezone 命令—克隆区域 (Funclet) - 319

dpmzone-createandinstallzone 命令--创建并安装区域 - 320

dpmzone-createpool 命令—创建资源池 (Funclet) - 322

dpmzone-createproject 命令—创建项目 (Funclet) - 323

dpmzone-createzone 命令—创建区域 (Funclet) - 324

dpmzone-customcommand 命令—运行自定义命令 (Funclet) - 326

dpmzone-deleteproject 命令—删除项目 (Funclet) - 327

dpmzone-deletezone 命令—删除区域 (Funclet) - 327

dpmzone-installzone 命令—安装区域 (Funclet) - 328

dpmzone-movezone 命令—移动区域 (Funclet) - 328

dpmzone-rebootzone 命令—重新启动区域 (Funclet) - 329

dpmzone-renamezone 命令—重命名区域 (Funclet) - 329

dpmzone-setpoolscheduler 命令—设置池排定程序 (Funclet) - 330

dpmzone-setprocessorset 命令—设置处理器集的最大 CPU 数目 (Funclet) - 331

dpmzone-showinterfaces 命令—显示网络接口 (Funclet) - 332

dpmzone-showpools 命令—显示资源池 (Funclet) - 332

dpmzone-showprojects 命令—列出项目的属性 (Funclet) - 332

dpmzone-showresourceset 命令—显示资源集 (Funclet) - 333

dpmzone-showtasks 命令—列出项目中的任务 (Funclet) - 333

dpmzone-showzoneinterfaces 命令—显示区域的网络接口 (Funclet) - 334

dpmzone-showzoneprocess 命令—显示区域中的进程 (Funclet) - 334

dpmzone-showzoneprojects 命令—显示区域中的项目 (Funclet) - 335

dpmzone-showzones 命令—显示区域 (Funclet) - 335

dpmzone-startzone 命令—启动区域 (Funclet) - 336

dpmzone-stopzone 命令—停止区域 (Funclet) - 336

dpmzone-uninstallzone 命令—卸载区域 (Funclet) - 337

## E

exit、quit 命令—终止 AutoShell 会话 (Cmdlet) - 64

external 命令—声明本地外部函数 (Cmdlet) - 64

## G

get-autoShellClassInfo 命令—显示本地 AutoShell 类的方法和属性 (Cmdlet) - 65

gete—获取环境变量（函数） - 108

get-help 命令-获取帮助信息 (Funclet) - 66

get-remoteResult 命令-从远程目标获取结果 (Funclet) - 67

get-webServiceInfo 命令-显示 WSDL 信息 (Cmdlet) - 68

---

## H

help 命令-请求帮助信息 (Cmdlet) - 68

## I

IBM HACMP 节点度量标准 - 793

IBM HACMP 网络度量标准 - 793

IBM HACMP 资源组度量标准 - 793

IBM HACMP 群集度量标准 - 792

ism-applyPolicies 命令-将策略应用于一组主机 (Funclet) - 338

ism-applyPoliciesAndTemplates 命令-将策略和模板应用于一组主机 (Funclet) - 341

ism-applyPoliciesAndTemplatesForService 命令-将策略和模板应用于所有系统 (Funclet) - 344

ism-applyPoliciesForService 命令-将策略应用于所有系统 (Funclet) - 342

ism-applyPolicyForService 命令-将策略应用于所有系统 (Funclet) - 337

ism-applyPolicyTemplates 命令-将策略模板应用于一组主机 (Funclet) - 340

ism-applyPolicyTemplatesForService 命令-将策略模板应用于所有系统 (Funclet) - 343

ism-applyPolicyWait 命令-应用策略并等待 (Funclet) - 339

ism-cancelJob 命令-取消作业 (Funclet) - 345

ism-copyPolicy 命令-复制策略 (Funclet) - 345

ism-copyPolicyTemplate 命令-复制策略模板 (Funclet) - 366

ism-createJobTemplate 命令-创建作业模板 (Funclet) - 346

ism-deleteJobTemplate 命令-删除作业模板 (Funclet) - 346

ism-deletePolicy 命令-删除策略 (Funclet) - 347

ism-deletePolicyTemplate 命令-删除策略模板 (Funclet) - 366

ism-deployPackageToHost 命令-将软件包部署到主机 (Funclet) - 348

ism-deployTemplate 命令-从模板部署软件包 (Funclet) - 349

ism-deployTemplateForService 命令-在服务中部署模板软件包 (Funclet) - 350

ism-deployTemplateToHost 命令-在主机上部署模板软件包 (Funclet) - 352

ism-encryptString 命令-加密字符串 (Funclet) - 353

ism-listComputerSystems 命令-显示所有计算机系统 (Funclet) - 353

ism-listDeployedSystems 命令-列出已部署软件包的所有主机 (Funclet) - 354

ism-listJobs 命令-列出所有作业 (Funclet) - 354

ism-listJobStatus 命令-出作业状态 (Funclet) - 355

ism-listJobTemplates 命令-列出所有作业模板 (Funclet) - 355

ism-listPackages 命令-列出所有软件包 (Funclet) - 355

ism-listPolicies 命令-列出所有策略 (Funclet) - 356

ism-listPolicyStatus 命令-列出策略状态 (Funclet) - 356

ism-listPolicyTemplate 命令-列出所有策略模板 (Funclet) - 371

ism-listPolicyTemplateStatus 命令-列出策略模板状态 (Funclet) - 367

ism-listServiceSystems 命令-列出服务中的所有计算机系统 (Funclet) - 357

ism-listSysEdgeConfig 命令-列出 SystemEDGE 配置 (Funclet) - 357

ism-listSystemPolicies 命令-列出所有策略 (Funclet) - 360

ism-listSystemPoliciesForService 命令-列出服务中的所有策略 (Funclet) - 358

ism-listTemplateEulas 命令-显示模板中的 EULA (Funclet) - 358

ism-removePolicyTemplate 命令-删除策略模板 (Funclet) - 367

ism-renamePolicy 命令-重命名策略 (Funclet) - 359

ism-renamePolicyTemplate 命令-重命名策略模板 (Funclet) - 368

ism-setMaintenanceMode 命令-设置主机的维护模式 (Funclet) - 359

ism-setMaintenanceModeForService 命令-设置服务中的维护模式 (Funclet) - 360

---

ism-setSysEdgeLogLevel 命令-设置  
SystemEDGE 记录级别 (Funclet) - 361

ism-setSysEdgeLogLevelForService 命令-设置  
服务中的 SystemEDGE 记录级别 (Funclet) -  
362

ism-updateAllExceptionSystems 命令-应用最  
新策略 (Funclet) - 362

ism-updateAllOldPolicies 命令-更新所有旧策  
略 (Funclet) - 363

ism-updateAllPolicies 命令-更新所有策略  
(Funclet) - 363

ism-updateAllUpToDateSystems 命令-更新所  
有最新系统 (Funclet) - 363

ism-updateExceptionSystems 命令-应用最新  
策略 (Funclet) - 364

ism-updateExceptionSystemsTemplate 命令  
—应用最新策略模板 (Funclet) - 369

ism-updateOldSystems 命令-更新旧策略  
(Funclet) - 364

ism-updateOldSystemsTemplate 命令—更新  
所有旧系统模板 (Funclet) - 369

ism-updatePolicies 命令-更新策略的配置  
(Funclet) - 365

ism-updateSystemsTemplate 命令—应用系  
统模板 (Funclet) - 370

ism-updateUpToDateSystems 命令-重新应用  
策略 (Funclet) - 365

ism-updateUpToDateSystemsTemplate 命令  
—重新应用系统模板 (Funclet) - 370

## M

memoRead—将文本文件读入字符串 (函数)  
- 109

memoWrit—将字符串写入文件 (函数) - 110

mkdir 命令-创建目录 (Cmdlet) - 69

mv、ren 命令-移动文件和目录 (Funclet) - 71

## N

new-comObject 命令-实例化 COM 对象  
(Funclet) - 73

new-SSHSession 命令-创建一个新的安全  
SSH 会话 (Funclet) - 75

new-webService 命令-创建 Web 服务对象  
(Funclet) - 77

## O

objdump 命令-显示对象的可枚举属性  
(Cmdlet) - 78

OSRedirect 类 - 131

OSRedirect.clear 方法 - 132

OSRedirect.errorOccurred 方法 - 133

OSRedirect.errout 方法 - 134

OSRedirect.hasCompleted 方法 - 135

OSRedirect.onCompleted 方法 - 135

OSRedirect.onError 方法 - 136

OSRedirect.OSRedirect 构造函数 - 132

OSRedirect.output 方法 - 137

OSRedirect.receivedErrOutput 方法 - 137

OSRedirect.receivedOutput 方法 - 138

OSRedirect.receivedResult 方法 - 139

OSRedirect.result 方法 - 140

## P

platform—查询操作系统类型 (函数) - 111

push-client 命令-在远程 Windows 系统上安  
装 Autoshell 客户端 (Funclet) - 78

push-winRemote 命令-在多个远程 Windows  
系统上进行复制并执行 (Funclet) - 80

pute—设置环境变量 (函数) - 111

PW\_GET 命令-输入隐藏数据 (Funclet) - 82

pwd 命令-输出工作目录的名称 (Cmdlet) - 83

## Q

qout—将后跟换行符的输出写入 stdout (函  
数) - 112

qqout—将输出写入 stdout (函数) - 113

query-service 命令—查询 Windows 服务的  
状态 (Funclet) - 83

## R

regCreateKey—创建注册表项 (函数) - 114

regCreateSubkeys—从数组创建子注册表项  
(函数) - 115

regDeleteKey—删除注册表项或注册表项层  
次结构 (函数) - 116

regDeleteVal—删除注册表值 (函数) - 117

regGetKeyValues—获取注册表项值信息 (函  
数) - 118

---

regGetSubKey—检索注册表项的子注册表项（函数） - 120

regGetVal—获取注册表值（函数） - 121

regIsKey—检查注册表项是否存在（函数） - 122

regIsVal—检查注册表值是否存在（函数） - 124

regSetKeyValues—从数组设置注册表项值（函数） - 125

regSetVal—设置注册表值（函数） - 126

rem-client 命令—从远程 Windows 系统中删除 Autoshell 客户端 (Funclet) - 84

RemoteTarget 类 - 141

RemoteTarget.abort 方法 - 144

RemoteTarget.createRemoteContext 方法 - 144

RemoteTarget.destroyRemoteContext 方法 - 145

RemoteTarget.errorOccurred 方法 - 146

RemoteTarget.execute 方法 - 147

RemoteTarget.getError 方法 - 148

RemoteTarget.getHostName 方法 - 149

RemoteTarget.hasCompleted 方法 - 150

RemoteTarget.onError 方法 - 151

RemoteTarget.output 方法 - 152

RemoteTarget.receivedOutput 方法 - 152

RemoteTarget.receivedResult 方法 - 153

RemoteTarget.RemoteTarget 构造函数 - 142

RemoteTarget.result 方法 - 154

RemoteTarget.wasAborted 方法 - 155

rmdir 命令—删除目录 (Funclet) - 86

rmonwatch add 命令-将行添加到 MIB 表中 - 773

rmonwatch del 命令-从 MIB 表中删除行 - 775

rmonwatch getTable 命令-查看 rmonWbem MIB 表 - 777

rmonwatch getVersion 命令-查看版本信息 - 779

run-client 命令—在远程 Windows 系统上运行 Autoshell 客户端 (Funclet) - 88

run-local 命令—在本地系统上执行脚本 (Funclet) - 89

run-remote 命令—在远程系统上执行脚本 (Funclet) - 90

run-SSHCommand 命令—通过 SSH 运行远程命令 (Funclet) - 94

run-SSHShell 命令—运行交互式 SSH 命令行 (Funclet) - 96

run-winRemote 命令—在多个远程 Windows 系统上执行命令 (Funclet) - 98

## S

set alternate to 命令—设置备用输出文件 (Cmdlet) - 101

set console 命令—停止控制台输出 (Cmdlet) - 102

set result display on/off 命令—打开或关闭自动结果显示 (Cmdlet) - 103

setProcExitCode—设置 AutoShell 退出代码（函数） - 127

shellType—查询 AutoShell 类型（函数） - 128

Solaris Zones CPU 度量标准 - 799

Solaris Zones 内存度量标准 - 800

Solaris Zones 共享内存度量标准 - 800

Solaris Zones 服务器 CPU 度量标准 - 798

Solaris Zones 服务器内存度量标准 - 798

Solaris Zones 服务器网络度量标准 - 797

Solaris Zones 服务器度量标准 - 796

Solaris Zones 服务器磁盘度量标准 - 797

Solaris Zones 度量标准 - 799

Solaris Zones 项目度量标准 - 801

Solaris Zones 资源池度量标准 - 800

Solaris Zones 最大交换度量标准 - 800

start-java 命令—启动 Java 虚拟机 (Cmdlet) - 105

start-service 命令—在远程或本地系统上启动 Windows 服务 (Funclet) - 103

stop-service 命令—在远程或本地系统上停止 Windows 服务 (Funclet) - 104

## T

typeName—获取表达式的类型名称（函数） - 129

## V

VMware VC ESX 主机 - 811

VMware VC ESX 主机 CPU - 811

VMware VC ESX 主机内存 - 813

VMware VC ESX 主机网络 - 813

---

VMware VC ESX 主机每个 CPU - 814  
VMware VC ESX 主机磁盘 - 812  
VMware VC Server - 822  
VMware VC VM - 814  
VMware VC VM CPU - 815  
VMware VC VM 内存 - 817  
VMware VC VM 网络 - 818  
VMware VC VM 磁盘 - 816  
VMware VC 计算资源 - 820  
VMware VC 资源池 - 820  
VMware VC 数据存储 - 819  
VMware VC 群集 - 819

## W

wait 命令—等待按键 (Cmdlet) - 105  
weak external 命令—声明本地外部函数 (Cmdlet) - 106

## 四划

从 CIM 对象检索祖先项和子项 - 23  
从键盘读取数据（标准输入） - 33  
日志文件 - 781  
日志文件设置 - 784

## 五划

可用的日志文件 - 782  
处理器池 - 789  
处理器池支持 - 789  
本指南涵盖的内容 - 15

## 六划

关于 AutoShell - 19  
字符串化 - 31  
收集引擎命令 - 662  
有上限 LPAR 和无上限 LPAR - 790  
有效 AutoShell 用户 - 21  
池容量 - 790  
约定 - 16  
访问 AutoShell - 22  
访问操作系统 - 34

## 七划

批处理模式 - 29  
运行脚本 - 35  
远程批处理模式 - 30

远程监控命令 - 773  
远程监控性能指标 - 828

## 八划

事件命令 - 681  
使用 OSRedirect 类 - 34  
使用 RemoteTarget 类 - 38  
使用 run-local 命令 - 36  
使用 run-remote 命令 - 38  
使用交互模式 - 28  
使用远程交互模式 - 29  
命令行指令 (CLI) - 373  
命令行脚本 - 373  
性能度量标准 - 785

## 九划

将数据写入控制台 (stdout) - 32  
帮助中心命令 - 696  
映像命令 - 703  
相关出版物 - 15

## 十划

容量度量标准 - 792  
根据类别列出的命令、函数和类 - 44  
消耗度量标准 - 791  
读者 - 15  
调用 AutoShell - 26  
资源管理器命令 - 726  
通用发现命令 - 658  
通用信息模型 (CIM) 对象 - 22  
部署和策略配置 AutoShell 命令 - 337  
高级 SNMP 参数 - 780

## 十一划

常规 AutoShell 命令 - 46  
常规外壳命令 - 685  
检索 CIM 对象 - 24  
检索 CIM 类 - 22

## 十二划

策略命令 - 710  
策略配置 - 724  
联系技术支持 - 3

---

## 十三划

简介 - 15

## 十四划

管理命令行实用工具 - 645

管理器 Shell 和客户端 Shell - 21

## 十六划

默认的共享处理器池 - 790