CA IT Process Automation Manager (CA ITPAM)

Web Services Utilities User Guide

ca technologies

LEGAL NOTICE

This publication is based on current information and resource allocations as of its date of publication and is subject to change or withdrawal by CA at any time without notice. The information in this publication could include typographical errors or technical inaccuracies. CA may make modifications to any CA product, software program, method or procedure described in this publication at any time without notice.

Any reference in this publication to non-CA products and non-CA websites are provided for convenience only and shall not serve as CA's endorsement of such products or websites. Your use of such products, websites, and any information regarding such products or any materials provided with such products or at such websites shall be at your own risk.

Notwithstanding anything in this publication to the contrary, this publication shall not (i) constitute product documentation or specifications under any existing or future written license agreement or services agreement relating to any CA software product, or be subject to any warranty set forth in any such written agreement; (ii) serve to affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (iii) serve to amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this publication remain at CA's sole discretion.

The information in this publication is based upon CA's experiences with the referenced software products in a variety of development and customer environments. Past performance of the software products in such development and customer environments is not indicative of the future performance of such software products in identical, similar or different environments. CA does not warrant that the software products will operate as specifically set forth in this publication. CA will support only the referenced products in accordance with (i) the documentation and specifications provided with the referenced product, and (ii) CA's then-current maintenance and support policy for the referenced product.

Certain information in this publication may outline CA's general product direction. All information in this publication is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this document "AS IS" without warranty of any kind, including, without limitation, any implied warranties of merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill or lost data, even if CA is expressly advised of the possibility of such damages.

COPYRIGHT LICENSE AND NOTICE:

TITLE AND PUBLICATION DATE:

CA IT PAM Web Services Utilities User Guide
Publication Date: October 13, 2010

CA PRODUCT REFERENCES

This document references the following CA products:

- CA IT Process Automation Manager (CA IT PAM)

FEEDBACK

Please email us at impcdfeedback@ca.com to share your feedback on this publication. Please include the title of this publication in the subject of your email response. For technical assistance with a CA product, please contact CA Technical Support at http://ca.com/support. For assistance with support specific to Japanese operating systems, please contact CA at http://www.casupport.jp.

Contents

Chapter 1: CA IT PAM Utilities

The CA IT Process Automation Manager Web Service Utilities are field developed utilities designed to simplify interaction and integration with CA IT Process Automation Manager (CA ITPAM). The package supports interaction at two levels:

■ Command line interface (CLI) that can be invoked interactively or programmatically

■ Java web service client wrapper that eliminates the need to build a web service stub from the "out of the box" WSDL and that works with more complex structures that can be included as a library in any Java application.

In a relatively short amount of time you should be able to initiate, control and track the progress of processes defined to IT PAM from the command line or external application.

Before You Begin

Before attempting use the utility you should:

■ Be familiar with the CA IT PAM architecture and management interfaces

■ Have a working knowledge the terms "library", "object" and "operator" in the context of CA IT PAM

■ Have some experience defining, launching and troubleshooting CA IT PAM processes

■ Understand the concepts of "datasets", "attachments", and "parameters" as applied to IT PAM processes and datasets

■ Verify that CA IT PAM is installed and available. To do this launch a URL similar to the following from your browser:

http://<Server Name or IP>:<Port>/itpam/soap?wsdl

The CA IT PAM WSDL should be returned and displayed. If the service fails to respond, confirm the service address, port and verify that the IT PAM services have been started on the server. If the service is still not responding, review the product documentation and other published troubleshooting guides to resolve the issue. Finally, if you are unable to communicate with the web service, contact CA Support for further assistance.

As with any new software component, you should use a lab environment to test and explore the utility prior to moving it into a production environment.

Command Line Interface

The command line interface (CLI) can be useful in a variety of scenarios. For example:

■ Providing simple scripts that end users can run to initiate self-service processes

■ Simple integration with external applications that may only be capable of executing a command and accepting a return code

■ Scripting unit and regression testing of new defined or modified processes

The CLI provides a command driven, text based interface that is simple to use while still supporting the capabilities of the IT PAM web service.

Setup and Configuration

It is not necessary to install any CA IT PAM components locally on system(s) where the command line interface will be used. Only a Java Runtime Environment (JRE) version 1.5 or 1.6, the Web Service Utilities package includes the necessary binary and Java library files are required. To setup and configure the utility do the following:

1. Extract the contents of the package to any directory (for example, C:\Program Files\CA\CA IT PAM Utilities)

2. Open the "caitpamwsucli.cmd" file with any text editor

3. Edit the following line to set the location of the installed JRE

   ```
   set JRE_PATH=<Path to Installed Java JRE 1.5 or 1.6 (ex. C:\Program Files\Java\jre5>
   ```

4. Save the file

The utility is ready for use.

Syntax and General Usage

As with most command line interfaces, the CLI is invoked by calling or executing the executable file followed by the appropriate switches and arguments. For convenience, help with syntax and usage can also be output to the console or redirected to a file by executing the following command:

```
caitpamwscli.cmd [/? or /h or /help]
```

To interact with the IT PAM web service to initiate a process or request information, arguments that specify the "action" to perform as well as qualifying parameters must be specified. Arguments are specified as name/value pairs joined with an equal sign (name=value). An argument value that contains spaces must be enclosed in double quotes. For example:

```
name="value with spaces"
```

Argument name value pairs must be separated with spaces. For example:

```
name1=value1 name2=value2
```

Spaces between an argument name and value are not allowed. For example, the following is valid:

```
name=value
```

While the following will be rejected:

```
name = value
```

Commands may be run silently (typically when invoked in scripts or triggered by external application) or interactively in a console session.

Silent Execution

To support integration and scripting, commands run silently by default. Commands that produce output will write to STDOUT when successful and no output when an error occurs. Error messages will be written to the log file only. Environment variable "ERRORLEVEL" will be set to one of the following success/error codes and the success/error code will be returned:

```
0          Completed successfully
9001       Attempt to initialize CLI utilities failed
9002       Unexpected Java or web service exception
9003       IT PAM entity not found or inaccessible
9004       Method not supported for action/entities specified
9005       Required argument or parameter not specified
9006       Unsupported date format specified
9007       Host specified not found or inaccessible
9008       Insufficient user permissions
9009       Unexpected web service exception thrown by web service
host
```

Important: When "wrapping" the command in Windows batch scripts remember to "call" the command instead of simply executing to capture the "ERRRORLEVEL". For example:

```
call caitpamwscli.cmd <Switches and arguments>
```

Interactive Execution

Commands can also been executed interactively by specifying the switch, "/i" as the first argument:

```
caitpamwscli.cmd /i <Additional switches and arguments...>
```

When invoked interactively, all output, including error messages and usage tips will be displayed on the console.

Service Configuration and Authentication

Actions that communicate with the IT PAM web service will require users to specify the following values:

- user=<IT PAM user name>

- password=<IT Pam user's password>

- uri=<IT PAM web service URI (ex. http://<SERVER>:<PORT>/itpam/soap)

- version=<IT PAM version (ex. 2.1 or 2.2)

For security reasons, the user name and password are not stored and must be specified each time a command is invoked.

For convenience, the action "setServiceConfig" has been provided to store the "uri" and "version" values. This eliminates the need to specify the arguments each time.

```
caitpamwscli.cmd /i user=<User> password=<Password>
action=setServiceConfig uri=<URI> version=<Version>
```

The current saved values may be retrieved using the "action" argument "getServiceConfig".

```
caitpamwscli.cmd /i user=<User> password=<Password>
action=getServiceConfig
```

Saved values would only be recognized by the instance of "caitpamwscli.cmd" executing the command. For example, if the utility is installed on "MachineA" and "MachineB", executing the command on "MachineA" would have no impact on the configuration of "MachineB" and vice versa. Likewise, if two instances of the utility are installed on "MachineA" - once in "<Drive:\(Path)\Test\CA IT PAM Utilities" and again in "<Drive:\(Path)\Production\CA IT PAM Utilities" - executing the command from the "Test" directory instance would update the configuration for that instance only. The configuration of the instance installed in the "Production" directory would not be affected.

The saved values may be superseded by specifying the arguments when executing a command.

Supported Actions

An "action" argument must be specified for each command. Additional required and optional arguments vary dependent upon the action specified. For convenience, help with syntax and usage for supported actions can also be output to the console or redirected to a file by executing the following command:

```
caitpamwscli.cmd </? or /h or /help> action=<Supported Action>
```

Current list of supported actions is included in the output of the general usage command (see Syntax and General Usage).

abortProcess

Abort processing of process specified (IT PAM versions 2.2 and higher).

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=abortProcess processId=<Process ID>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a message indicating it completed successfully.

approvePendingInteraction

Flags pending interaction specified as approved to continue.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=approvePendingInteraction interactionId=<Interaction ID>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command does not generate output.

deleteArchivedInstances

Purges archived process instances in the specified date range specified (CA IT PAM versions 2.2 and higher).

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=deleteArchivedInstances startDate=<Date> endDate=<Date>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

Supported date formats include:

■ yyyy-MM-dd

■ yyyy/MM/dd

■ MM-dd-yyyy

■ MM/dd/yyyy

The command does not generate output.

deleteAttachments

Deletes attachments specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=deleteAttachments attachmentId=<Attachment ID
1>…attachmentId=<Attachment ID N>
```

Arguments "uri" and "version" are optional only if a default web service URI and version has been set (see Service Configuration and Authentication).

At least one "attachmentId" argument must be specified. Any number of "attachmentId" arguments may be specified.

The command does not generate output.

exportObject

Export objects from IT PAM library to file specified (IT PAM versions 3.0 and higher).

```
caitpamwsu.cmd    [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>][version=<Web Service Version>]
action=exportObject  sourceLibraryPath=<Source ITPAM library path
to search for content to be exported  (include object name to
export a specific object)> targetPath=<Target file system path
for resulting object export (Note: Content may exported to
existing directory on the IT PAM server or an accessible UNC
path)> targetFileName=<Target file name for resulting object
export file> [targetPathIsLocal=<Flag set to true if target
path/file name specified is local(relative to the client) or
false if relative to the server (default is false)]
[levelsToSearch=<Integer set levels relative to source library
path to search/include for content to be exported (default is 0)]
[absolutePath=<Flag set to true if library paths in resulting
export file should be absolute or false if paths should be
relative (default is false)>] [overwriteTargetFile=<Flag set to
true if any existing export file with the same name in the same
location should be overwritten (default is false)>]
[objectType=<String set to a valid object type applied as a
filter that determines which objects in the source path to
include in the exported content (Note: Multiple name/value pairs
may be specified) [outputFormat=<Properties (name/value
pairs)(default) or XML>]
```

Supported object type filter strings:

- Agenda

- Calendar

- Commander

- CustomIcon

- CustomOperator

- CustomSensor

- Dataset

- InteractionRequestForm

- LogViewer

- Package

- Process

- ProcessWatch

- Resources

- StatePolicy

- System

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command does not return output to the calling application or console. The IT PAM export file is written directly to the file system location specified.

generateEvent

Generate an event consumable by waiting IT PAM process operator(s) specified (CA IT PAM versions 3.0 and higher).

```
caitpamwsu.cmd    [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=generateEvent eventName=<Event Name> [eventType=<Event
Type>] [eventDestination=<Event Destination>] [eventSource=<Event
Source>] [eventExpirationDuration=<Event Duration in Seconds
(default is 0, expires immediately>] [eventSingleSubscriber=<True
or False to flag event as "consumable" by a single listener
only>] [eventPayload=<Event Pay Load (content to be used by
waiting process operator(s) - creates a simple parameter in the
waiting process operator(s) dataset named "payload" set to the
value specified)>] [param.<Parameter Name1>=<Parameter
Value1>...param.<Parameter NameN>=<Parameter ValueN>] creates an
value map parameter in the waiting process operator(s) dataset
named "payload" and adds elements for each parameter specified
(see IT PAM for information on referencing individual member
elements of value maps)]
```

Important: Event payload ("eventPayload") will be superseded (ignored) if a parameter or parameters are specified.

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of event data for event generated in either "Properties" format (name/value pairs (default)) or "XML" format.

getAttachments

Returns properties of attachments specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=deleteAttachments contentId=<Content ID (Attachment Name)>
contentType=<Content Type> [outputFormat=<Properties (Name/Value
Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of attachments that match the specified criteria in either "Properties" format (name/value pairs (default)) or "XML" format.

getMatchingEvents

Returns event data for events matching criteria specified (IT PAM versions 3.0 and higher).

```
caitpamwsu.cmd    [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getMatchingEvents eventName=<Event Name)>
[eventType=<Event Type>] [eventDestination=<Event Destination>]
[eventSource=<Event Source>] [eventExpression=<Expression that
will be compared to event "payload" parameter(s)(ex.
payload.paramName=="paramValue")>] [allowPartialMatch=<Specify
true to return data for events that partially match the specified
criteria (default is false)>] [enablePatternMatch=<Specify true
if criteria specified should be interpreted as regex-type
patterns (default is false)> [outputFormat=<Properties
(name/value pairs)(default) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of event data for events matching the criteria specified in either "Properties" format (name/value pairs (default)) or "XML" format.

getPendingInteration

Returns properties of pending interaction specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getPendingInteraction interactionId=<Interaction ID>
[outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs the properties of the specified interaction in either "Properties" format (name/value pairs (default)) or "XML" format.

getPendingInterations

Returns all pending interactions and related properties.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getPendingInteraction interactionId=<Interaction ID>
[outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of all pending interactions and related properties in either "Properties" format (name/value pairs (default)) or "XML" format.

getProcessLogs

Returns process log entries for process specified filtered by optional criteria specified (CA IT PAM versions 2.2 and higher).

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getProcessLogs processId=<Process ID> [level=<Minimum
Severity Level of Log Entries> [category=<Log Category
1>…category=<Log Category N>] [outputFormat=<Properties
(Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

All log messages for the process specified are returned by default.

Specifying a valid value for "level" will filter messages and return only messages with the severity level specified and higher. Valid levels are "Error", "Warning", "Notice" and "Normal". Invalid values for "level" will be ignored.

Specifying one or more "category" arguments will filter messages and return only messages associated with the category/categories specified. Valid categories are "Process", "Operator", "Handler", "Response", "Other" and "Custom". Invalid values for "category" arguments will be ignored.

The command outputs a listing of log entries for the process specified matching the optional level and category filters specified in either "Properties" format (name/value pairs (default)) or "XML" format.

getProcessParameters

Returns parameter names and values for process specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getProcessParameters processId=<Process ID>
[outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of parameters names and values for the specified process in either "Properties" format (name/value pairs (default)) or "XML" format.

getProcessPendingInteractions

Returns pending interactions and related properties for process specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getProcessPendingInteractions processId=<Process ID>
[outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of pending interactions and related properties for the specified process in either "Properties" format (name/value pairs (default)) or "XML" format.

getProcessStatus

Returns current status and related properties for process specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getProcessStatus processId=<Process ID>
[outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs the current status and a listing of related properties for the specified process in either "Properties" format (name/value pairs (default)) or "XML" format..

getServerStatus

Returns pending interactions and related properties for process specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getServerStatus
```

```
Arguments "uri" and "version" are optional only if a default web
service URI and version have been set (see Service Configuration
and Authentication).
```

The command outputs a string indicating current server status.

getServiceConfig

Returns current stored values for IT PAM web service URI and version.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getServiceConfig
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs the current stored values as name/value pairs.

getStartRequestForms

Returns start request forms and related properties located in the specified search path.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getStartRequestForms [searchPath=<Library Path to Search
(Default is root ("/"))>] [recursive=<True or False (Default is
False))>] [outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs a listing of start request forms and related properties located in the search path specified in either "Properties" format (name/value pairs (default)) or "XML" format.

getStartRequestProcessReferenced

Returns fully qualified process name referenced by the specified start request.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getStartRequestProcessReferenced startRequestName=<Start
Request Name> startRequestPath=<Start Request Library Path>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs the fully qualified process name referenced by the specified start request.

getStartRequestStatus

Returns current status and related properties for the specified start request.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=getStartRequestStatus startRequestId=<Start Request ID>
[outputFormat=<Properties (Name/Value Pairs) or XML>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command outputs the current status and related properties for the specified start request in either "Properties" format (name/value pairs (default)) or "XML" format.

importObject

Creates/updates an ITPAM object defined by the source file in the ITPAM library relative to the specified target path and folder specified (CA IT PAM versions 2.2 and higher). Note that, if an existing object with same name in the same location exists, it will be overwritten.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=importObject importSourcePathFileName=<Source Path and
File Name of Object Definition XML File>
importTargetLibraryPathName=<Target Library Path and Folder Name>
importMakeAvailable=<True or False> importMakeCurrent=<True or
False>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command does not generate output.

rejectPendingInteraction

Flags pending user interaction as rejected.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=rejectPendingInteraction interactionId=<Interaction ID>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

Note: Beginning with CA IT PAM v2.2, attempts to reject a pending interaction are blocked if the "Show approval page" option for the "User Interaction" and an appropriate message will result.

The command does not generate output.

resumeProcess

Resume processing of previously suspended process specified (CA IT PAM versions 2.2 and higher).

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=resumeProcess processed=<Process ID>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command does not generate output.

setServiceConfig

Set the default web service URI and version to eliminate the need to specify "uri" and "version" arguments for every command issues (see Service Configuration and Authentication).

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=setServiceConfig uri=<IT PAM Web Service URI> version=<IT
PAM Web Service Version>
```

The command does not generate output.

startProcess

Start process specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=startProcess processName=<Fully Qualified Process Name>
[startDate=<Date>] [startTime=<Time>] [param..<Parameter Name
1>=<Parameter Value 1>…param..<Parameter Name N>=<Parameter Value
N>] [attachmentParameterName=<Attachment Parameter Name>
attach.<Attachment Name 1>=<Attachment Location
1>…attach.<Attachment Name N>=<Attachment Location N>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

Default start date and time is now. Start date is required if start time is specified. Supported date formats are:

- yyyy-MM-dd

- yyyy/MM/dd

- MM-dd-yyyy

- MM/dd/yyyy

Supported time formats are:

- hh:mm

- hh:mm:ss

Parameter ("param..") arguments are optional. Any number of parameter arguments may be specified.

Attachment ("attach.") arguments are optional. Any number of attachments may be specified. If attachment arguments ("attach.") are specified an attachment parameter name ("attachmentParameterName") and only one attachment parameter name must be specified.

The command does not generate output.

startRequest

Initiate start request specified.

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=startRequest startRequestName=<Start Request Name>
startRequestPath=<Start Request Library Path> [startDate=<Date>]
[startTime=<Time>] [param..<Parameter Name 1>=<Parameter Value
1>…param..<Parameter Name N>=<Parameter Value N>]
[attachmentParameterName=<Attachment Parameter Name>
attach.<Attachment Name 1>=<Attachment Location
1>…attach.<Attachment Name N>=<Attachment Location N>]
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

Default start date and time is now. Start date is required if start time is specified. Supported date formats are:

- yyyy-MM-dd

- yyyy/MM/dd

- MM-dd-yyyy

- MM/dd/yyyy

Supported time formats are:

- hh:mm

- hh:mm:ss

Parameter ("param..") arguments are optional. Any number of parameter arguments may be specified.

Attachment ("attach.") arguments are optional. Any number of attachments may be specified. If attachment arguments ("attach.") are specified an attachment parameter name ("attachmentParameterName") and only one attachment parameter name must be specified.

The command does not generate output.

suspendProcess

Suspend processing of specified process specified (CA IT PAM versions 2.2 and higher).

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=suspendProcess processed=<Process ID>
```

Arguments "uri" and "version" are optional only if a default web service URI and version have been set (see Service Configuration and Authentication).

The command does not generate output.

Action File

Required and optional values may be stored in an "action file" to eliminate the need to specify command line arguments. By reducing the number of arguments that must be entered the effort to use the command line interface is significantly reduced and while the reliability of the outcome is increased. The benefits are compounded when "action files" are created and referenced for repetitive activities. Arguments values specified in "action files" may be supplemented or superseded by arguments passed when a command referencing an "action file" is executed.

Using Action Files

An "action file" is a simple text file similar to a Windows "INI" or Java "Properties" file. Each entry is a name/value pair corresponding to any name/value pair argument that would be specified for a supported action via the command line (see Supported Actions).

To illustrate, assume the following:

- CA IT PAM user is "user01"

- CA IT PAM user's password is "pwd01"

- CA IT PAM web service URI is "http://server:8080/itpam/soap"

- CA IT PAM version is "2.2"

- Process to be started is the "Example 1" process in the "/Examples" folder in the library (or fully qualified name "/Examples/Example 1"

- Value for one parameter, "EmailAddress", defined and used in the "Example 1" process should be set to user02@domain.com

- The process also references an attachment parameter, "EmailAttachments", that accepts links uploaded attachments to the process. For the example, the file "C:\Temp\Instructions.txt" should be uploaded as an attachment.

Refer to CA IT PAM product documentation for more information on process parameters and attachments.

The first step is to refer to the syntax for the "action" to be performed to get the argument names supported (since the above example requires that a process be started, see "startProcess" in the Supported Actions section of this document or execute the command "caitpamwscli.cmd /help action=startProcess").

```
caitpamwscli.cmd [/i] user=<User Name> password=<Password>
[uri=<Web Service URI>] [version=<Web Service Version>]
action=startProcess processName=<Fully Qualified Process Name>
[startDate=<Date>] [startTime=<Time>] [param..<Parameter Name
1>=<Parameter Value 1>…param..<Parameter Name N>=<Parameter Value
N>] [attachmentParameterName=<Attachment Parameter Name>
attach.<Attachment Name 1>=<Attachment Location
1>…attach.<Attachment Name N>=<Attachment Location N>]
```

Simply add the argument names and corresponding values as name/value pairs, one per line to a text file:

```
user=user01
password=pwd01
uri=http://server:8080/itpam/soap
version=2.2
action=startProcess
```

```
processName=/Examples/Example 1
param..EmailAddress=user02@domain.com
attachmentParameterName=EmailAttachments
attach.EmailAttachment01=C:\Temp\Instruction.txt
```
Assuming the "action file" content was saved to "C:\Temp\Example.txt" the process could be started by executing the following command:

```
caitpamwscli.cmd actionFile=C:\Temp\Example.txt
```

Although using the "action file" made it possible to start the process passing required parameter values with a single argument, it should be clear that storing credentials in a simple text file would not be recommended. However, as noted early, command line arguments can be specified to supplement or supersede "action file" entries.

To avoid storing credentials, remove the entries from the "action file". In the example the revised content of "C:\Temp\Example.txt" would be as follows:

```
uri=http://server:8080/itpam/soap
version=2.2
action=startProcess
processName=/Examples/Example 1
param..EmailAddress=user02@domain.com
attachmentParameterName=EmailAttachments
attach.EmailAttachment01=C:\Temp\Instruction.txt
```

The command to start the processing with the revised "action file" would be:

```
caitpamwscli.cmd actionFile=C:\Temp\Example.txt user=user01
password=pwd01
```

Finally, let's assume that, as recommended, the web service configuration arguments now included in the "action file" referenced a test environment. When moved to production, the file would need to be edited to reference the production URI and version.

To eliminate the need to reference to the web service URI and version in "action files" based on the environment (i.e., test versus production) use the "setServiceConfig" action to store the values globally for all commands issued using a specific instance of "caitpamwscli.cmd" (see Service Configuration and Authentication):

```
caitpamwscli.cmd action=setServiceConfig
uri=http://server:8080/itpam/soap version=2.2
```

The "action file" content would be revised as follows:

```
action=startProcess
processName=/Examples/Example 1
param..EmailAddress=user02@domain.com
attachmentParameterName=EmailAttachments
```

```
attach.EmailAttachment01=C:\Temp\Instruction.txt
```

The same "action file" and command:

```
caitpamwscli.cmd actionFile=C:\Temp\Example.txt user=user01
password=pwd01
```

…could not be used in both the test and production environments (assuming "user01" was defined in both environments with the same password, otherwise the values for "user" and "password" would need to be changed).

Argument Super-Session

Argument values can be specified in several ways:

■   As arguments when the "caitpamwscli.cmd" is issued (see Supported Actions)

■   As entries in an "action file" (see Action File)

■   As stored configuration values in the case of "uri" and "version" arguments (see Service Configuration and Authentication)

■   Default values when applicable (see Supported Actions)

Argument value conflicts (such as multiple values for the same argument name specified in two or more ways) will be resolved in the following order:

■   Value passed as an argument when the "caitpamwscli.cmd" is executed will supersede all other values

■   Value specified as an entry in an action file will supersede store configuration or default values

■   Stored configuration values or default values will be used only when applicable and not superseded by an "action file" entry or argument specified as argument when the "caitpamwscli.cmd" is issued

For example, assume the following:

■   Web service URI was set to http://Production:8080/itpam/soap using the "setServiceConfig" action (see Service Configuration and Authentication)

■   Action file "C:\Temp\QA.txt" contains the entry "uri=http://QA:8080/itpam/soap" (see Action File)

…and the command:

```
caitpamwscli.cmd actionFile=C:\Temp\QA.txt
uri=http://TEST:8080/itpam/soap
```

…is issued. Since the argument passed when the command is executed supersedes all other values, the targeted web service URI will be "http://TEST:8080/itpam/soap".

Change the command to:

```
caitpamwscli.cmd actionFile=C:\Temp\QA.txt
```

…and the targeted web service URI will be "http://QA:8080/itpam/soap" since the "action file" entry supersedes stored configuration or default values.

Remove the entry "uri=http://QA:8080/itpam/soap" from the action file "C:\Temp\QA.txt" and issue the command:

```
caitpamwscli.cmd actionFile=C:\Temp\QA.txt
```

…and the target web service URI will be "http://Production:8080/itpam/soap" (the stored value set using the "setServiceConfig" action).

Troubleshooting

Every effort was made to produce clear, specific exception messages. When executing interactively (see Interactive Execution), exception messages will be displayed on the console. In addition, all messages generated are logged to the "caitpamwscli.log" file in the "log" sub-directory relative to the location of the "caitpamwscli.cmd" executable file.

Typically, only informational, warning, error and fatal messages will be logged and in most these messages should be sufficient to identify the cause of the problem and make corrections. However, unexpected exceptions are always a possibility and it may be necessary to gather more verbose debug tracing information. To increase or decrease the minimum severity of messages logged add the following argument to any command or "action file":

```
logLevel=<Minimum Severity of Messages to Logged>
```

…where severity is one of the following:

- FATAL
- ERROR
- WARN
- INFO
- DEBUG
- ALL

When severity "ALL" is specified, all messages will be written to the log file and displayed on the console. If no value or an invalid value is specified, the level will be set to the default (INFO).

The log file will rollover when the size reaches 2mb. The current and previous 9 log files will be retained.

Web Service Client

The CA IT PAM Utilities bundle also includes the Java libraries required to instantiate a web service client and to interact with the web service directly. To use the web service client in your Java project do the following:

1. Copy all the library (JAR) files from the CA IT PAM Utilities "lib" directory to a location that can be conveniently accessed and referenced from your project

2. Add all the library (JAR) files to the class path (build path) for your project

3. Include the following classes as needed:

   > com.ca.dso.itpam.webservice.client.ItpamWebServiceClientFactory

   > com.ca.dso.itpam.webservice.client.ItpamWebService Client

To assist those less familiar with Java, steps to create an example project follow using the popular Eclipse IDE.

Sample Eclipse Project

Possibly the best way to become familiar with the CA IT PAM web service client is to experiment with some sample code. The "doc" sub-directory of the CA IT PAM Utilities package contains a "ready to use" project for the popular Eclipse IDE. To successfully use the sample you should:

■ Be familiar with the Eclipse development environment

■ Have Eclipse 3.4 or higher installed

■ Eclipse workspace you intend to use should be configured go use a Java 1.5 or 1.6 JRE

■ Have a test environment with an instance of CA IT Process Automation Manager installed and running (see Before You Begin)

Assuming the above…

1. Locate the "CA IT PAM Web Service Client Sample" folder in the "doc" sub-directory of the CA IT PAM Utilities package

2. Copy "CA IT PAM Web Service Client Sample" folder and contents to the Eclipse workspace folder you intend to use

3. Launch Eclipse and open the workspace

4. From the Eclipse menu select File, Import to launch the "Import" wizard

5. On the first panel of the wizard, expand the "General" node in the tree displayed, select "Existing Projects into Workspace" then click Next.

6. On the "Import Projects" panel click the Browse button associated with the "Select root directory" text box

7. Navigate to the workspace folder containing the expanded "CA IT PAM Web Service Sample" project folder

8. The "CA IT PAM Web Service Sample" project should appear in the "Projects" list – simple check the associated check box then click Finish.

The "CA IT PAM Web Service Sample" project should now appear in the Eclipse "Project Explorer" view.

The sample code will need to be modified slightly to set values specific to your CA IT PAM test environment. To do this:

1. Expand the "CA IT PAM Web Service Sample" node in the Eclipse "Project Explorer" tree

2. Expand the "src" folder node

3. Expand the "com.ca.dso.itpam.webservice.sample" package node

4. Double-click on the "Sample.java" node

   The "Sample.java" class should appear in an Eclipse editor window. Locate the lines listed below:

```
String strItpamUri = "http://<Server>:<Port>/itpam/soap";
String strItpamVersion = "<Version>";
String strItpamUser = "<User Name>";
String strItpamPassword = "<User's Password>";
```

5. Replace the highlighted entries with the values that apply to your test environment then same the changes.

You are now ready to test.

From the Eclipse menu select Run, Run Configurations. When you expand the "Java Applications" node in the explorer tree on the left side of the "Run Configurations" a run configuration called "Sample" should be listed. Select the "Sample" run configuration node then click the Run button on the "Run Configurations" dialog box (bottom right). The sample code should execute and the following message should appear in the Eclipse "Console" view (just below the editing pane).

```
Server status ok.
```

Assuming you were successful and are familiar with Java, JavaDoc is included for all exposed public methods. You may view the JavaDoc via the Eclipse IDE or, alternatively as HTML by using your browser to open the "index.html" in the "doc\CA IT PAM Web Service Client JavaDoc" sub-directory. Try your hand at invoking other web service methods by adding code to the sample project provided.

Known Limitations

CA IT PAM web services have and will continue to evolve. Since the CA IT PAM Utilities are designed to work with multiple releases, dependent upon the release you have, implementation of certain features may behave differently or not be supported. Consult the "Release Notes.mht" document distributed with the package for details.

Summary

When an integrated solution requires triggering and interacting with CA Process Automation Manager processes you can use the CA IT PAM Utilities command line interface and/or the web services client to reduce the effort and time to value