CA SiteMinder®

SDK Release Notes

r6.0 SP6/r6.x QMR 6

Fourth Edition

Contact CA

Contact CA Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At http://ca.com/support, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, complete our short customer survey, which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Welcome

Welcome to the CA SiteMinder® SDK Release Notes. These release notes contain information about platform support, software requirements, installation, known issues, and published fixes.

Chapter 2: Operating System Support

To learn about operating system support for the SDK, search the Platform Support Matrices on the Technical Support site.

The SDK platform support matrix is typically included with the SiteMinder general support matrix. Search for the SiteMinder Platform Matrix for r6 SP6.

Note: Be sure to verify that previously supported platforms are still supported.

Chapter 3: Software Requirements

Verify that you have the required JRE version installed. For the required version, see the SiteMinder Platform Matrix for r6 SP6 on the Technical Support site. You can download the latest JRE version from the Oracle Sun Developer Network SDN.

SiteMinder SDK r6.0 SP 6 supports Policy Server v6.0, v6.0 SP 1, v6.0 SP 2, v6.0 SP 3, v6.0 SP 4, r6.0 SP 5, and r6.0 SP6. Applications developed with SDK r6.0 SP 6 cannot be run against Policy Server versions older than v6.0.

Applications developed with a previous version of the SDK and that work with the SiteMinder Policy Server v5.0 and v5.5 continue to work with the Policy Server v6.0, v6.0 SP 1, v6.0 SP 2, v6.0 SP 3, v6.0 SP 4, r6.0 SP 5, and r6.0 SP 6.

Chapter 4: Installation

This section contains the following topics:

Install the SDK on Windows

No special accounts or privileges are required to install the SiteMinder SDK. Instructions for installing a first version of the SDK and upgrading from an existing version are the same.

Do not install the SiteMinder SDK in the installation path where the Policy Server or Web Agent is installed, because there might be different versions of the same support libraries.

To install the SDK on Windows

1. Close all programs.
2. Download the SiteMinder SDK from the CA Technical Support site.
3. Navigate to the win32 directory and run the following program:

   ```
   nete-sdk-6.0-sp6-win32.exe
   ```

Follow the screen prompts provided by the wizard.

Install the SDK on UNIX in GUI Mode

No special accounts or privileges are required to install the SiteMinder SDK. Instructions for installing a first version of the SDK and upgrading from an existing version are the same.

Do not install the SiteMinder SDK in the installation path where the Policy Server or Web Agent is installed, because there might be different versions of the same support libraries.

On UNIX, the installation executable file is nete-sdk-6.0-sp6-*platform*.bin.

To Install the SDK in UNIX GUI Mode

1. Close all programs.
2. Download the SiteMinder SDK from the CA Technical Support site.
3. In a UNIX shell, navigate to the directory that corresponds to your platform (solaris, aix, linux, or hpux).
4. Enter the following command:

   `sh ./nete-sdk-12.0-sp2-platform.bin`

   platform

   Replace platform with sol, aix, linux, or hp.

   For example, on Solaris platforms, the command is:

   `sh ./nete-sdk-12.0-sp2-sol.bin`
5. Follow the prompts provided by the installation wizard.

Install the SDK on UNIX in Console Mode

No special accounts or privileges are required to install the SiteMinder SDK. Instructions for installing a first version of the SDK and upgrading from an existing version are the same.

Do not install the SiteMinder SDK in the installation path where the Policy Server or Web Agent is installed, because there might be different versions of the same support libraries.

On UNIX, the installation executable file is nete-sdk-6.0-sp6-*platform*.bin.

To Install the SDK in UNIX Console Mode

1. Close all programs.
2. Download the SiteMinder SDK from the CA Technical Support site.
3. In a UNIX shell, navigate to the directory that corresponds to your platform (solaris, aix, linux, or hpux).

4. Enter the following command:

   ```
   sh ./nete-sdk-6.0-sp6-platform.bin -i console
   ```

   platform

   Replace platform with sol, aix, linux, or hp.

   For example, on Solaris platforms, the command is:

   ```
   sh ./nete-sdk-6.0-sp6-sol.bin -i console
   ```

Follow the screen prompts provided by the installation wizard.

Uninstall the SDK from Windows

Remove the SDK from your system when you no longer require it.

To uninstall the SiteMinder SDK from Windows

1. From the Control Panel, double-click Add/Remove Programs.
2. Select SiteMinder SDK r6.0 SP 6 and click Change/Remove.
3. Follow the screen prompts, and click Close when done.
4. Alternatively, you can run the following command:

   ```
   nete-sdk-uninstall.exe
   ```

The SDK is removed from your system.

Uninstall the SDK from UNIX

Remove the SDK from your system when you no longer require it.

Note: When you are uninstalling the SDK from a UNIX system, verify that the JRE is in the PATH variable. If the JRE is not in the PATH variable, you receive an error message. In this case, install a JVM before running the uninstall program.

To uninstall the SiteMinder SDK from the UNIX console

1. In a console window, navigate to the install_config_info/nete-sdk-uninstall directory within the SDK installation

   Example: /export/netegrity/sdk/install_config_info/nete-sdk-uninstall

2. Run the following command:

   ```
   ./uninstall -i console
   ```

3. When prompted, press Enter to remove the SDK.
4. Alternatively, you can run the following command:

```
nete-sdk-uninstall.sh -l console
```

Chapter 5: General Considerations

This section contains the following topics:

Policy Management API

The Policy Management API was changed significantly beginning with SiteMinder v4.5. For example, the OID was introduced as the unique identifier of an object and functions, such as Sm_PolicyApi_Init() and Sm_PolicyApi_Release(), were added.

Single Sign-On Support in Custom Agents

When single sign-on support is enabled, custom agents can accept the SMSESSION single sign-on cookie created by a standard SiteMinder v4.x, v5.x, or v6.x Web Agent. To accept an SMSESSION cookie created by a custom agent, the standard agent must be upgraded as follows:

QMR	Supported SiteMinder Agent
4.xQMR4	SiteMinder v4.x agents
5.xQMR1	SiteMinder v5.x and v6.x agents
6.xQMR6	SiteMinder v5.x and v6.x agents

To enable a SiteMinder v4.x, v5.x, or v6.x standard agent with the appropriate QMR upgrade to accept SMSESSION cookies created by a custom agent, the standard agent's Agent configuration file (LocalConfig.conf with IIS servers or WebAgent.conf with other servers) or central configuration object (for v5.x or higher) must contain the following entry:

```
AcceptTPCookie="yes"
```

Set AcceptTPCookie as follows:

- With 4.xQMR4 agents and above, add AcceptTPCookie="yes" directly in the standard agent's Agent configuration file.
- With 5.xQMR1 agents and above, add the entry to the standard agent's Agent Configuration Object if the AllowLocalConfig parameter for that object is set to no. If AllowLocalConfig is set to yes, you can set AcceptTPCookie in the standard agent's Agent configuration file.

Custom Agents and SiteMinder v6.0 SP 6

When you build a custom agent with SDK r6.0 SP 6, you must run the custom agent only against v6.0, v6.0 SP 1, v6.0 SP 2, v6.0 SP 3, v6.0 SP 4, r6.0 SP 5, or r6.0 SP 6 Policy Server.

Agents built with the SiteMinder Agent API from SDK 5.5 can be used with SiteMinder Policy Servers v5.x, v6.0, v6.0 SP 1, v6.0 SP 2, v6.0 SP3, v6.0 SP 4, r6.0 SP 5, and r6.0 SP 6.

5.08 LDAP SDK Integration

Beginning with SiteMinder v5.5 SP 1, the Policy Server is integrated with the iPlanet Directory SDK for C 5.08 (5.08 LDAP SDK). If your custom code had been linked with the older version of the LDAP SDK, you can relink with the new SDK.

Chapter 6: Known Issues

This section contains the following topics:

Verify HP-UX Patch Level to Install a SiteMinder Component

The SiteMinder installers for r6.0 SP6 require Java 1.6. If you want to install a SiteMinder component on a system that uses the HP-UX operating environment, do the following:

1. Go to the HP Software Depot web site.
2. Search the Software Depot for the following item:

   `JDK, JRE, and Plug-In 6.0.x Downloads and Documentation`
3. Verify that your system contains the correct prerequisites.

Patch Required for Custom Agents Running on HP-UX

Before you install a custom agent created with the Agent API on an HP-UX machine, install the following HP patch on the agent machine:

PHSS_24303 Id & linker tools cumulative patch

This patch is available at the HP web site.

Configuration Issue with the smpolicyapi Sample on Unix/Linux Platforms

Building the smpolicyapi sample program on Unix/Linux systems requires access to several Policy Server shared libraries and the NETE_PS_ROOT environment variable set to point to them. To accomplish this, install the Policy Server on the system with the SDK and use nete_ps_env.ksh to set the environment.

Hierarchical Realms Not Supported in the Java Policy Management API

Hierarchical realms are not supported in the Java Policy Management API for the SDK r6.0 SP 2 release.

SiteMinder May Not Enforce the Requirement that a Given Resource Filter Be Unique for an Agent that is a Member of Different Agent Groups (10911)

If a given resource is defined in different realms of the same domain, and the resource is protected by the same agent, unpredictable behavior can result.

SiteMinder does not prevent you from setting up a situation such as the following:

- Agent1 is contained in agentgroup1 and agentgroup2.
- Realm1 is created under a domain with the resource filter \sales\. The realm is associated with agentgroup1.
- Realm2 is created under the same domain with the same resource filter. The realm is associated with agentgroup2.
- Because Agent1 is defined for both agentgroup1 and agentgroup2, the resource filter \sales\ resource appears in different realms, but is protected by the same agent.

This situation can occur when adding or modifying a realm.

Workaround: The sample application PolicyApiSample.java contains a method named checkRealmResourceFilter() that checks for this situation. You can copy the method and use it in your Java applications, or use it as a model for your C applications.

Constants Missing from Java Policy Management API (13348)

The Java class com.netegrity.sdk.policyapi.SmPasswordPolicy is missing the following constants:

Constant Name	Constant Value
Sm_PasswordPolicy_StopPriorityChaining	0x00000080
Sm_PasswordPolicy_ExpireDisablePassword	0x00000100
Sm_PasswordPolicy_FailuresDisablePassword	0x00000200
Sm_PasswordPolicy_ForceCase	0x00000400
Sm_PasswordPolicy_CaseSelect	0x00000800
Sm_PasswordPolicy_CaseBits	0x00000c00
Sm_PasswordPolicy_StripLeadingWhiteSpace	0x00001000
Sm_PasswordPolicy_StripTrailingWhiteSpace	0x00002000
Sm_PasswordPolicy_StripFlankingWhiteSpace	0x00003000
Sm_PasswordPolicy_StripEmbeddedWhiteSpace	0x00004000
Sm_PasswordPolicy_WhiteSpaceBits	0x00007000
Sm_PasswordPolicy_PreProcessBits	0x00007c00

Workaround: To set these values, use the literal values directly instead of referencing the constant name.

Attributes Terminated with a Space Character Cause SSO Problems in Java (13712)

When decodeSSOToken() returns, each attribute (byte array) returned in the AttributeList parameter is terminated with a space character.

Workaround: Before you use the session specification and session id attributes in a login() call to validate the session, trim the terminating space from each byte array.

Extra ServerDef Object Created with Java AgentAPI.getConfig() (14841)

If the Java AgentAPI.getConfig() method is used to read the configuration file, it creates an InitDef object having one more ServerDef object than is present. The extra object does not point to any Policy Server and is merely an empty object.

Java SmPolicyApiImpl.search() Cannot Search for an Object of Type Domain (15133)

In package com.netegrity.sdk.policyapi, the method PolicyApiImpl.search() returns an empty list when the object type to search for is type Domain, even if valid search parameters are given.

Sort Order in a Java SmDmsCursor Object Cannot Be Empty (15317)

In the Java DMS API, any call (such as search(), getGroups(), and getMembers()) that uses an SmDmsCursor object with an empty sort order fails.

General Password Validation Errors in C (16787)

Password validation error IDs are reported in the nMsgId parameter of Sm_PolicyApi_GetPasswordMsg(). Specific error IDs are enumerated in Sm_PolicyApi_PasswordMsgId_t. Any error ID that is not enumerated in Sm_PolicyApi_PasswordMsgId_t is considered a general password validation error. General password validation error IDs are reported in nMsgId when Sm_PolicyApi_GetPasswordMsg() returns -38.

Incorrect Authentication Pages No Longer Displayed When SecurID Token is in New Token Mode (23219)

An ACE HTML Forms authentication scheme that returns any of the following access event reason codes (Sm_Api_Reason_t) must have the redirect URL as the last element of the authentication scheme parameter list:

- Sm_Api_Reason_New_PIN_Sys_Tokencode
- Sm_Api_Reason_New_PIN_Select

- Sm_Api_Reason_New_User_PIN_Tokencode
- Sm_Api_Reason_New_PIN_Accepted

A semicolon (;) separates the URL from the first part of the parameter list.

Executing an Application Built with the Policy Management API on Solaris (28334)

To avoid a core dump, applications built with the Policy management API must not link with internal Policy Server libraries, for example, libsmutilities.so. This restriction does not apply to libraries libsmpolicyapi45.so and libsmplatform.so.

On Windows, the Java Method getConfig() Fails (39780)

In the Java class AgentAPI, the method getConfig() fails because it cannot locate the specified agent. This error only occurs on Windows platforms.

Chapter 7: Fixes in SiteMinder SDK r6.0 SP 6

The fixes described in this chapter were made in the r6.0 SP6 release of SiteMinder.

SDK Allows Addition of Duplicate Realm (92503)

Symptom:

The SDK allows you to add a duplicate realm with a conflicting resource filter.

Solution:

This is no longer an issue. If you try adding a duplicate realm, you receive an error message.

STAR Issue: 18447116

Sub-Realms Using Identical Names Return NoRealm with SmPolicyApi.getRealmRealms Method (109613, 100718)

Symptom:

If I have two sub-realms with the same name, the SmPolicyApi.getRealmRealms Method returns NoRealm, even if the sub-realms have different parent realms.

Solution:

Updated the JavaDoc to indicate that sub-realms require unique names.

STAR Issue: 18829270:01

Java API Could Not Retrieve New Password Value (109827, 114394)

Symptom:

The Java Authentication API could not retrieve a new password value. A new method has been added to return this value.

Solution:

This issue is fixed.

The UserCredentialsContext class contains the following new method: getNewPassword.

STAR Issue: 18921774:03

SmCertMap Methods Documentation (110146)

Symptom:

The description of the setFlags and setMapAttr methods required additional detail.

Solution:

This issue is fixed.

STAR Issue: 18905311-01

Pure Java Agent API Ignores and Overwrites Session ID (111401)

Symptom:

The pure Java API was not correctly setting the Session ID when passed by the SessionDef parameter.

Solution:

This issue is fixed.

STAR Issue: 19043761

Pure Java API Returns Truncated updatedSSOToken (112208)

Symptom:

If the decodeSSOToken method was used with the updateToken parameter, the pure Java API returned a truncated updatedSSOToken.

Solution:

This issue is fixed.

STAR Issue: 19013393

DMS API Throws Exception (112275)

Symptom:

The DMS API getgroups method fails with a “Failed to create a sorting LDAP control" error message.

Solution:

This is no longer an issue.

STAR Issue: 18891341

Enhanced DMS Performance (127882)

Symptom:

When using most of the DMS API calls, the Policy Server verifies that the user performing the operations is an administrator of the specified directory. The Policy Server does this by retrieving all the domains and their associated user directories. The problem arises when multiple threads try to verify the administrators at the same time. The contention for the locks on the cache slows the operation considerably.

Solution:

The solution to this problem was to add alternative methods that do no verify the administrator for each directory. The administrator is verified only once at login. The alternative methods do not fetch all the domains and user directories. The alternative methods were implemented by adding the NoAdminCheck suffix to the parallel method. For example, getDmsContext/getDmsContextNoAdminCheck.

STAR Issue: 19755226;1

Chapter 8: Fixes in SiteMinder SDK r6.0 SP 5

The fixes described in this chapter were made in the r6.0 SP5 release of SiteMinder.

Ability To Override Target URL Configuration for SAML1x (10159)

Symptom:

For SAML1.x TARGET is a required parameter for SSO. Previously, the SP-side had no way to override this configuration parameter, or validate that the target is a protected SiteMinder resource.

Solution:

Two functions have been added to the Policy Management API:

- Sm_PolicyApi_AddTargetConfigToSAML1xScheme()
- Sm_PolicyApi_GetTargetConfigFromSAML1xScheme()

Policy Server Restart Was Required for a Trusted Host (95465)

Symptom:

A trusted host created by the C API could not be used without a Policy Server restart.

Solution:

The Policy Server is fixed to write the buffered server commands before the store is released. A restart is no longer required.

STAR issue: 17966870

Custom Agents Can Lose Key Synchronizations (87269)

Symptom:

A custom agent made a call to the DoManagement function, but did not get changed agent keys from the Policy Server. As a result, the custom agent failed to decrypt the SM SESSION cookie for single sign-on.

Solution:

When the Policy Server routinely purges old agent commands, it now will always keep the most recent agent key change command, however old it might be.

STAR issue: 17596204-1

Realm Filter Not Unique Error Occurs (85423)

Symptom:

"Realm Filter Not Unique" error occurs during the creation of policy store objects.

Solution:

This is no longer an issue.

STAR Issue: 17107802

Method getOidString() Is Deprecated

Symptom:

The method getOidString() is deprecated.

Solution:

Use the method policyApi.getDomainObject() instead.

Agent Not Notified When ACO Modified through SDK (82796)

Symptom:

When the Agent Configuration Object (ACO) is modified through the SDK, the agent is not notified.

Solution:

This is no longer an issue.

STAR Issue: 17753390-1

The Pure Java API Supports Larger Buffer Sizes (79220/82496)

Symptom:

The JNI version will increase the size of the buffer to match the the size of the response from the server. The pure java agent API passes a fixed size ByteBuffer. The Pure java agent API's limit of 32784 Bytes is very small.

Star Issue 17428858-01

Solution:

The pure Java API now supports larger buffer sizes for its transport layer.

The decodeSSOToken Function Decodes the SSO SMSESSION Cookie Correctly (82495/79193)

Symptom:

When you use the Cookie Provider SSO that shares the static agent key and extract SMSESSION cookies from the IE HTTP session, pure Java API AgentAPI.decodeSSOToken() is unable to decode the SMSESSION cookies with return value -1.

Solution:

The pure Java API decodeSSOToken function now decodes the SSO SMSESSION cookie correctly.

STAR Issue: 17568332-1

An Error Occurs When Listing User Information (81006)

Symptom:

An error occurs when using the CLI to list information for more than 507 users.

Solution:

This is no longer an issue.

STAR Issue: 17723135

Pure Java Agent API Tunnel Size Is Too Small (79220)

Symptom:

The pure Java agent API tunnel size is too small.

Solution:

This is no longer an issue.

STAR Issue: 17428858-01

Pure Java Agent API Function decodeSSOToken Fails (79193)

Symptom:

The pure Java agent API function decodeSSOToken fails to decode the SSO SMSESSION cookie.

Solution:

This is no longer an issue.

STAR Issue: 17568332-1

An Error Occurs on Calls to Pure Java Agent API Login (77035)

Symptom:

An error occurs on calls to pure Java agent API Login.

Solution:

This is no longer an issue.

STAR Issue: 17555709;01

Calls to Pure Java Agent API Function GetConfig Fail (66144)

Symptom:

Calls to Pure Java Agent API function GetConfig fail when SmHost.conf has RO file permission.

Solution:

This is no longer an issue.

The Pure Java Agent API isProtected Method Did Not Set realmdef Members to Null (55246)

The pure Java Agent API isProtected method now sets realmdef members to null for unprotected resources.

The Pure Java Agent API login Method Did Not Accept a null or "" Value for the clientIP Parameter (55247)

The pure Java method Agent API login now accepts a null or "" value for the clientIP parameter.

The authenticateUser() method Failed for the Java Policy Management API (52937)

The Java Policy Management API method authenticateUser() now properly returns the authentication status of the user.

The Agent API connect() Method in the Perl CLI Did Not Return an Error Code (54815)

The Agent API method connect() returns the correct status (SM_AGENTAPI_FAILURE) when connection to policy server is not successful.

Java API SmTrustedHost.writeProperties(hash) Threw NullPointerException (54842)

The Java Policy Management API function SmTrustedHost.writeProperties(hash) now properly handles null pointers.

Domain Scope User Could Not Get Realm Using a Perl Script (54465)

Domain Scope user can now successfully call function GetRealm() in Perl. This behavior is now in synch with the behavior available through the SiteMinder Administrative UI.

SmPolicyApi Methods getRealmRealms and getRealmRules Were Not Working Properly (47895)

The Java Policy Management API method getRealmRealms() now returns only the realms under the specified realm instead of all realms under the given domain. Similarly, the method getRealmRules now returns only the rules under the specified realm instead of all rules under all realms under the given domain.

AgentTLI Failed in memcpy() Method Called from CSmSerializable::Serialize (53299) (53206)

Add additional length checking for strings passed to the Java API to help ensure that overly long strings do not crash the TLI layer. If a string longer than the maximum permitted length is passed in, an error is returned.

SmPolicyApiImpl.doExport() Corrupted UNICODE Characters (52969)

The Java API functions doExport() and doImport() now properly handle UTF-8 encoding of data.

C++ Comments in the Code Were Not Compatible with Some C Compilers (52130) (52048)

C-style comments replace C++-style comments in the C API header files SmApi.h and SmAgentAPI.h. This change eliminates problems that occur when a C compiler does not support C++-style comments.

The Java Policy Management API Did Not Contain Functions to Search for User Groups and User Attributes (52769)

Added capabilities to the Java Policy Management API to return user groups and user attributes.

No Validation For Invalid IP Address When Calling SmPolicy.setIPRestriction (48918)

An exception is thrown when an invalid IP address is set for com.netegrity.sdk.policyapi.SmPolicy.setIPRestriction (java.util.Vector ipRestriction) provided by the Java SDK. This validation check helps ensure that the policy can be edited in the Administrative UI when an invalid IP address is passed in custom code.

Passing a Null SessionDef to the Java Agent API authorize() Method Caused the JVM to Crash (48962)

Passing a null SessionDef to the Java AgentAPI authorize() method no longer causes the JVM to crash.

SmPolicy.getIPRestriction() Did Not Return All Restrictions (51509)

The Java Policy Management method SmPolicy.getIPRestriction() now properly returns all IP restrictions for a SiteMinder policy.

Custom Code Calling user.addToGroup() Caused smpolicysrv to Fail (48023)

Custom code calling user.addToGroup() no longer causes smpolicysrv to fail.

SmPolicyApi.getPolicyLinks() Triggered a Full Policy Store Cache Refresh (48122)

To improve performance, the Java API function SmPolicyApi.getPolicyLinks() triggers a full policy store cache refresh only when necessary.

SmPolicyApi.getUserDirSearchOrder() Returned Oids Instead of Names (48221)

Problem:

The Java Policy Management SDK returns OID strings instead of object names for some methods.

Solution:

The Java Policy Management SDK returns OID strings only for objects that cannot be identified by name, specifically certificate and directory mappings.

Chapter 9: Fixes in SiteMinder SDK v6.0 SP 4

This section contains the following topics:

Speed Increased for Fetching Domain Objects (47124)

A problem in the Policy Server domain property fetch code has been fixed. This fix means that the performance of the Java Policy Server API v6.x is comparable to the performance of the Java Policy Server API 4.x.

Java Logger Class Is Added to the Standard Installation (47077)

The Java sample application Active Response failed to build. This kitting issue has been resolved.

Data that the Web Agent SDK 6.0 on Linux Sends to the Policy Server Is Fixed (44051)

In the call to SmAgentApiInit(), the send() function was transmitting corrupt data to the Policy Server. The client saw a security handshake error. This issue has been corrected.

Sm_Api_Reason_MaxDefined Is Updated (45062)

Sm_Api_Reason_MaxDefined has been updated in SmAgentApi.h to reflect additional reason codes for federation.

An Error Code Is No Longer Returned When Calling GetHostConfig (46531)

Problem:

When you pass a host configuration object name to GetHostConfig(), the call fails.

Solution:

When you pass a host configuration object name or OID to GetHostConfig(), the call is successful.

Return Code SM_AGENTAPI_UNRESOLVED Is Available (46536)

This return code is now listed in SmAgentApi.h.

Chapter 10: Fixes in SiteMinder SDK v6.0 SP 3

This section contains the following topics:

Update to search( ) Method (43246)

In the Java Policy Management API, the search() method of the class SmPolicyApiImpl now finds the Domain, Agent Config, and Host Config objects as expected. Earlier the search() API failed for these objects.

Update to getGlobalObjectNames() Method (42993)

In the Java Policy Management API, the getGlobalObjectNames() method of the SmPolicyApiImpl class now returns the list of Agent or Host configuration objects as required if the property is set as SmAgentConfig.PropAgentConfigs or SmHostConfig.PropHostConfigs.

Modified Linkage of Agent API Library on Solaris (41741, 41586)

This SiteMinder Agent API fix modifies the linkage for the Agent API shared library on the Solaris platform. As a result, the shared library is self-contained with respect to other third-party libraries used in the same process.

Password Changes through API Validated (43063, 41644)

The Policy Server has been fixed to validate any password changes performed through the DMS API. As a result, password changes conform to policy restrictions set through the Administrative UI.

Two Additional Policy API Methods Supported in a Custom Authentication Scheme (41008, 40306)

The policy server authentication service now supports calling the Policy API initialization Sm_PolicyApi_Init() and Policy API release Sm_PolicyApi_Release() methods inside the respective hook functions SmAuthInit() and SmAuthRelease() in a custom authentication scheme.

Chapter 11: Fixes in SiteMinder SDK v6.0 SP 2

This section contains the following topics:

Policy Server Authentication Service Calling Policy API Init and Release Functions (33966, 31579)

The Policy Server authentication service now accommodates calling the Policy Management API Sm_PolicyApi_Init() and Sm_PolicyApi_Release() functions at any location inside a custom authentication scheme.

Updated Java API Method AgentApi.init() (38813)

The Java-based API method AgentApi.init() is corrected to accommodate conditions when a null ServerDef object is added to an InitDef object that is used as a parameter to the init() method.

Chapter 12: Fixes in SiteMinder SDK v6.0 SP 1

This section contains the following topics:

AgentAPI.login() Method No Longer Crashes the Java Server When Incorrect Parameters are Passed (7517)

This crash used to occur when the spec field of the SessionDef object was null and the UserCredentials object was empty.

Sm_AgentApi_UpdateAttributes() Now Recalculates Active Responses that Originally Had No Data (C) (30912)

In the past, this C function did not recalculate an active response that initially held no data. (That is, the active response attribute was 0 or null.) Now, an active response attribute, whose return value has a length greater than zero or whose TTL value is set, is considered valid.

Chapter 13: Fixes in SiteMinder SDK v6.0

This section contains the following topics:

Missing C Password Message IDs (25854)

The following password message IDs are now enumerated in Sm_PolicyApi_PasswordMsgId_t:

- Sm_PolicyApi_PasswordMsgId_PasswordSystemPIN
- Sm_PolicyApi_PasswordMsgId_PasswordUserMaxNumPIN
- Sm_PolicyApi_PasswordMsgId_PasswordUserMinMaxNumPIN
- Sm_PolicyApi_PasswordMsgId_PasswordUserMaxAlphaPIN
- Sm_PolicyApi_PasswordMsgId_PasswordUserMinMaxAlphaPIN
- Sm_PolicyApi_PasswordMsgId_PasswordAcceptPIN
- Sm_PolicyApi_PasswordMsgId_PasswordContentLowerAlpha
- Sm_PolicyApi_PasswordMsgId_PasswordContentUpperAlpha
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoLowerAlpha
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoUpperAlpha
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoDigits
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoPunctuation
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoNonPrintable
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoNonAlphaNum
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoAlphaNum
- Sm_PolicyApi_PasswordMsgId_PasswordContentMatchRegExp
- Sm_PolicyApi_PasswordMsgId_PasswordContentNoMatchRegExp
- Sm_PolicyApi_PasswordMsgId_PasswordUserMinNumPIN
- Sm_PolicyApi_PasswordMsgId_PasswordUserDigitsPIN
- Sm_PolicyApi_PasswordMsgId_PasswordUserAlphaNumPIN

Chapter 14: International Support

An *internationalized* product is an English product that runs correctly on local language versions of the required operating system and required third-party products, and supports local language data for input and output. Internationalized products also support the ability to specify local language conventions for date, time, currency and number formats.

A *translated* product (sometimes referred to as a *localized* product) is an internationalized product that includes local language support for the product's user interface, online help and other documentation, as well as local language default settings for date, time, currency, and number formats.

SiteMinder has been internationalized and localized to the extent indicated in the platform support matrix for SiteMinder r6.0 SP6.