ADSS OCSP Service

Table of Contents

How to enable support for Opera browsers that send OCSP requests and receive a response saying "Bad Request"

Background:

This problem occurs because by default Apache Tomcat disallows the use of the "%2F" character (i.e. URL encoded value of '/' character) for security reasons - attackers can get access to protected resources if this character is allowed. Click here for more details on this topic. See the below link for details on this topic.

Some versions of Opera generate OCSP GET request that contain the "%2F" character as part of URL encoded value of OCSP request. The information below shows how to configure Apache Tomcat to allow the use of "%2F" character so that requests can be processed successfully.

Solution:

Look for the below given text in the "service.bat" file at the location: "[ADSS Server installation directory]/tomcat/bin"

```
"%EXECUTABLE%" //US//%SERVICE_NAME% ++JvmOptions
"-Djava.io.tmpdir=%CATALINA_BASE%\temp;-Djava.util.logging.manager=org.apache.juli.Cla
ssLoaderLogManager;-Djava.util.logging.config.file=%CATALINA_BASE%\conf\logging.proper
ties" --JvmMs %6 --JvmMx %7
```

And modify this by appending ";-Dorg.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true" as shown below:

```
"%EXECUTABLE%" //US//%SERVICE_NAME% ++JvmOptions
"-Djava.io.tmpdir=%CATALINA_BASE%\temp;-Djava.util.logging.manager=org.apache.juli.Cla
ssLoaderLogManager;-Djava.util.logging.config.file=%CATALINA_BASE%\conf\logging.proper
ties;-Dorg.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true" --JvmMs %6
--JvmMx %7
```

Uninstall the "Ascertia-ADSS-Service" using the script "uninstall_service.bat" and install it again using the script "install_service.bat" (use an Admin privilege command prompt Window). This will solve the problem by adding a required registry entry in windows registries.

How to avoid a malformed request error when sending multiple certificate status requests in an OCSP request message?

In order for the OCSP Service to process multiple certificate IDs in a single OCSP request object the OCSP Service Advanced Settings need to be configured accordingly.
Click here to learn more on how to configure the ADSS OCSP Service to process up to "n" certificate IDs in a single OCSP request.

Why does the first OCSP request after a restart take a long time to respond?

The first OCSP transaction takes additional time because the OCSP response signing key details are being retrieved from HSM / database. For the subsequent OCSP calls, the OCSP response signing key alias is cached and thus the responses are processed much faster. This is an expected behaviour and it only happens when the OCSP service is restarted.

How to configure ADSS Server OCSP service for optimum performance?

In an environment where the number of incoming OCSP requests are very high (e.g. over 500 requests per second), the OCSP responder should be configured to minimise internal processing overheads.

Click here to learn more about Optimising ADSS OCSP Server performance.

How to bypass CRL expiry check in OCSP Service in a special case of Business Continuity Management

Background:

The ADSS OCSP Service checks the status of a certificate by looking into the latest CRL available to it but before checking the certificate status it first checks the expiry of the CRL and if CRL is expired it returns the 'Unknown' status. There are some scenarios where it is required to bypass the CRL expiry check e.g. In Algerian PKI Once a TSP (Trust Service Provider) goes out of operations due to any reason, the TSP CA Key will be destroyed and a BCM (Business Continuity Management) Server will take over, the BCM Server have the TSP CA CRL and until the CRL is valid the BCM Server will be able to respond the OCSP calls properly, once the TSP CRL gets expired then it will start returning unknown status.

Since the CA key will be destroyed and no new CRL would be published, the BCM Server will have to rely on the last CRL (that's also expired) to entertain requests for all the certificates that are still valid and not revoked. But due to the CRL expiry check, it will always return 'Unknown' status for all the certificates issued by this particular CA. Due to this, the BCM Server will not be able to continue the business processes that was its sole purpose and valid certificates of the CA could not be used for any transactions.

Solution:

To handle such a situation, a new feature is introduced in the ADSS Server where this CRL expiry check can by bypassed if a special configuration is enabled in ADSS Advanced Settings. Note that this feature should only be used in special scenarios as discussed above and its highly recommended to not enable it in other cases. In order to enable/disable this feature, navigate to (ADSS Console > Global Settings > Advanced Settings) and select the "General" category from the "Property Type" drop-down. Enable the "BYPASS_CRL_EXPIRY" property as shown in the image below: