Trust Manager

Table of Contents

How to replace an expired CA certificate?

There can be three scenarios to replace an expired certificate:

1. If a CA certificate is configured inside the Trust Manager module, then the operator can simply browse to read in a new replacement certificate, provided that the public key of the certificate has not changed. Once done, the settings can be applied by restarting all the ADSS Server instances through the Server Manager module.
2. If a new key pair has been generated for the CA, then the operator needs to register a new CA within the Trust Manager module with a different CA friendly name.
3. If a CA certificate is used as a local CA inside the ADSS Server Certification Service, then the operator needs to renew the certificate, by reviewing the details of CA certificate in Key Manager > Certificate. From here, depending upon certificate details, two choices can be made:
    a. Either a self signed certificate is created, or
    b. A PKCS#10 request is sent to an external Certificate Authority for certification.

If a new key pair is used to generate a new CA within the ADSS Key Manager module, then the existing local CA within the ADSS Manage CAs module needs to be updated accordingly. Once the new certificate has been applied, the Server Manager module should be used to reload all the system configuration data.

Impact on the system:

When a CA certificate has expired then that CA certificate cannot be trusted to verify any certificates issued by it or its sub-CAs, hence all such certificates will be reported as not trusted. Moreover, the certificates issuance by the local CA will also stop functioning, if the expired CA has been configured as local CA.

How to ensure trust anchors are synchronised when the Core/ Console/ Service instances are running on different machines?

Background:

Whenever a CA is added, updated or deleted from the ADSS Trust Manager module, then the “adss.keystore” and "jssecacerts" files are automatically updated by the ADSS server. Now in a scenario, where ADSS Server is running in the load balancing environment, and the core and console instances are running on different machines. Here if the core instance becomes either down or unavailable, then upon initiating any of 'new', 'edit' or 'delete' operation, only the local instance of "adss.keystore" and "jssecacerts" files is updated with the CA status accordingly.

However, each ADSS server instance needs to be synchronised accordingly to have the same configuration details.Therefore, a warning message is shown to restart all the ADSS Server instances to synchronise these files with other ADSS Server instances as well.

Impact on System:

ADSS server may behave inconsistently.

How to link a CA to the relevant TSAs in the ADSS Trust Manager module?

Timestamp authority (TSA) addresses are primarily stored within Global Settings > Timestamping module. These timestamp authorities can be used for various purposes e.g.

- Archiving in the LTANS Service,
- Timestamping signatures within the Signing Service,
- Enhancing signatures in the Signing and Verification Services.
- Creating the PAdES-LTV signatures in Go>Sign Service, etc.

The signer certificates may have been issued by different CAs. Hence each CA may have its own relevant TSA authority, to create timestamped signatures for certificates issued by that CA. All the CAs may not be authorised to get the services of a single TSA. It is therefore required to

associate only the relevant TSAs to a CA certificate, which is configurable within the Trust Manager module. The list of configured TSAs within the ADSS Trust Manager module is actually a subset of the TSAs configured within Global Settings > Timestamping module. For more details, see the online admin guide section Trust Manager > Advanced Settings