ADSS Server v8.3.3 –

Regular Release Installation Guide

ASCERTIA LTD

MARCH 2024

Document Version- 1.0.0

CONTENTS

TABLES

FIGURES

1 Introduction

The ADSS Server's regular release provides a more systematic and comprehensive update package that entails applying the updates to the existing base version of the ADSS Server. The update process often requires clients to undertake several steps, such as file removal, data backup, and more. With multiple instances of the server in operation, this process can become complex and prone to errors for our clients. Running the ADSS Server installer, which will automatically handle all necessary tasks—upgrading the database, backing up code, removing outdated files, and installing new ones. This approach aims to simplify the update process, reduce the likelihood of errors, and ensure more efficient and reliable system maintenance.

1.1 Scope

This manual describes how to install one or more instances of ADSS Server Regular Release.

1.2 Intended Readership

This manual is intended for ADSS Server administrators responsible for installation and initial configuration. It is assumed that the reader has a basic knowledge of digital signatures, certificates and information security.

1.3 Conventions

The following typographical conventions are used in this guide to help locate and identify information:

- Bold text identifies menu names, menu options, items you can click on the screen, file names, folder names, and keyboard keys.
- `Courier New` font identifies code and text that appears on the command line.
- **`Bold Courier New`** identifies commands that you are required to type in.

1.4 Technical Support

If Technical Support is required, Ascertia has a dedicated support team. Ascertia Support can be reached/accessed in the following ways:

Website	https://www.ascertia.com
Email	support@ascertia.com
Knowledge Base	https://www.ascertia.com/products/knowledge-base/adss-server/
FAQs	https://ascertia.force.com/partners/login

In addition to the free support services detailed above, Ascertia provides formal support agreements with all product sales. Please contact sales@ascertia.com for more details.

When sending support queries to Ascertia Support team send ADSS Trust Monitor logs. Use the Ascertia's trace log export utility to collect logs for last two days or from the date the problem arose. It will help the support team to diagnose the issue faster. Follow the instructions on how to run the trace log export utility

2 System Requirements

The following table lists the system requirements for ADSS Server:

Components	Requirements
ADSS Server	ADSS Server is a Java EE 17 application, supported on these platforms: **Operating System** The following 64-bit operating systems are supported: • Windows Server 2022, 2019 and 2016 • Linux (RedHat v7.x, v8.x, CentOS v7.x, v8.x, Ubuntu v20.x, v22.x) **Hardware** A modern multi-core CPU such as the Xeon E3-xxxx or E5-xxxx or E55xx or E56-xx or similar are recommended, with 16 GB RAM (min 8GB RAM) and 200 GB disk space. Additional RAM may be required to power signing or LTANS archive services. Roughly 0.5 GB to 1 GB of disk space is required to keep the trace logs per 100,000 service transactions. **Database** ADSS Server saves its configuration and transactional data in a database. The following databases are supported: • Microsoft SQL Server 2022, 2019, 2017, 2016 (Express, Standard, Web or Enterprise Edition) • Azure SQL Database (Database-as-a-service) • Oracle 19c (Standard Edition, Enterprise Edition) • PostgreSQL v14.x, v13.x, v12.x and v11.x • MySQL v8.x, Percona-XtraDB-Cluster v8.0.23. About 1GB of database space is required to store the service logs of 100,000 transactions for each service, unless these are regularly auto archived or customised. **Note:** PostgreSQL contrib extension must be installed while setting up PostgreSQL database.
Optional Database Server	The database can be run on a separate server if preferred. This is recommended for high performance environments to allow all server resources to be directed to ADSS Server services. **Hardware:** A modern multi-core CPU such as the Xeon E3-xxxx or Xeon E5-xxxx or E55xx or E56-xx or similar range are recommended, with 16 GB RAM, typically 5-10 GB or more of disk space will be required depending on usage and transactional data / log retention requirements.
Client systems (systems sending service requests to ADSS Server)	Any reasonable system. ADSS Client SDK for Java API requires JRE v1.7 or above. ADSS Client SDK for .NET requires Microsoft .NET Framework 4.5 and Microsoft .NET Core 8.0.
Operator Browsers	The following browsers are supported for ADSS Server Operators: • Google Chrome 70.x or above • Mozilla Firefox 60.x or above

Components	Requirements
	• Microsoft Edge 35.x or above
Mobile Devices OS	For authorised remote signing, the mobile apps (iOS and Android) of Go>Sign Mobile will require the following OS versions: • iOS 9.0 or above • Android 6 (Marshmallow) or above
Optional HSMs	If required, the following Hardware Security Modules are supported: • Thales SafeNet Luna and ProtectServer HSMs • Entrust nShield Solo or Connect HSMs • Utimaco CP5 SE & CryptoServer SE Gen2 HSMs • Microsoft Azure Key Vault HSM • Amazon AWS Cloud HSM (Supported when ADSS Server deployed on Linux)
Optional DMZ proxy machine	A DMZ proxy server can be configured if required. The following DMZ proxy machines are supported: • Windows Server - Microsoft IIS, Apache or IBM HTTP Server • Linux - Apache or IBM HTTP Server Use a reasonable CPU, 2GB RAM,100 MB disk space

Table 1 - ADSS Server System Requirements

2.1 Typical Deployment Scenario

A typical ADSS Server installation schematic looks like this:

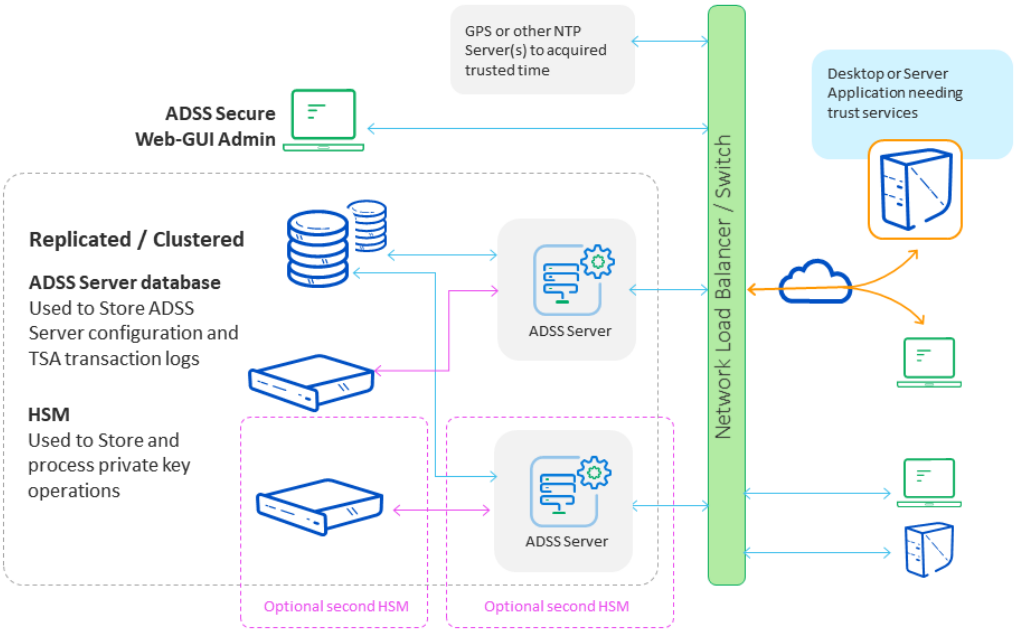


Figure 1 - Typical ADSS Server Deployment Scenario

ADSS Server and the database it uses can both be installed on the same machine. 12GB RAM is recommended for such a scenario. For high performance environments, it is recommended to install them on separate systems.

The details shown above are the minimum system requirements; these may need to be revised to meet specific usage requirements. For high throughput systems consider using multiple load-balanced ADSS Servers in a network load-balanced resilient arrangement. Multiple physical CPUs can be added although additional licenses are required for these. Virtualized systems are also supported.

ADSS Server can also be installed on the same system as the business application it services.

2.2 HSM Support for Key Wrapping

If you wish to use ADSS Server with its HSM based user key generation wrapping and export under a static or dynamic KEK then be careful with the specifications of the HSMs you order or try to reuse.

The best thing to do is to run the ADSS Server PKCS#11 Test Utility to check if the HSM supports the mechanisms needed for this and indeed other functions. HSM vendors are known to change the mechanisms that are supported in this area, and some exclude such mechanisms from the allowable list when in FIPS 140-2. If in doubt check with Ascertia support and also check with your HSM vendor that the AES_CBC_ENCRYPT_DATA mechanism is supported for key wrapping and export.

3 Pre-Installation Checks

The below-mentioned points must be ensured before installing the regular release for ADSS Server v8.3.2:

- The ADSS Server v8.3.1 must be installed. The regular release can only be applied to ADSS Server v8.3.1 and NOT to the prior versions.
- The operator must take the database backup of ADSS Server v8.3.1. To view more details regarding the backup procedure, navigate to the **ADSS-Server-Backup-and-Restore-Procedure-for-Application-and-Database.pdf** document at available in the **[ADSS-Server-Home]/docs** folder.
- The ADSS Server instances must also be stopped before installing the regular release package.

The more comprehensive details of pre-installation checks have been described in details in Section 3 of the **ADSS-Server-Installation-Guide** document. To view details, navigate to the **[ADSS-Server-Home]/docs** folder.

4 ADSS Server Installation

ADSS Server is a Java 17 EE application that has rich functionality. The ADSS Server license file contains a list of services/modules licensed for you, so not all services may be available within your ADSS Server deployment.

ADSS Server is shipped with a customized distribution of Apache Tomcat and Java and Ascertia continues to periodically upgrade these to the latest available versions. Operators and administrators should not attempt upgrade to these separately because it will lead to a system configuration that is not supported by Ascertia. If an upgrade is required, raise it to Ascertia at support@ascertia.com.

ADSS Server can be installed in either of these modes:

- GUI based - for Windows/X11 platforms
- Command Line (Non-GUI based) - for remote installation on UNIX platforms

4.1 Installation Process

ADSS Server regular release installer must be unzipped to a suitable directory (later referred as **[ADSS-Server-Home]**). ADSS Server regular release installation directory path **MUST NOT** contain space characters otherwise the installer will not be launch.

To start the installation, navigate to **[ADSS-Server-Home]/setup** directory. Either using a command line or Windows GUI interface.

Windows

Run the **install.bat** file under administrative privileges, as shown below, (otherwise ADSS Server services will not be registered in Windows Services Panel) to launch the installer.

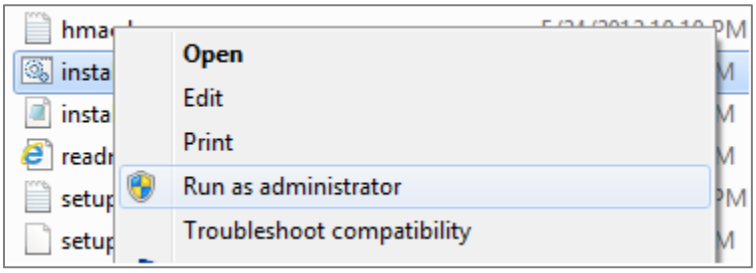


Figure 2 - Windows Example Installer Run as administrator

UNIX

To install ADSS Server Regular Release on UNIX systems the installer must be launched under **root** user privileges (otherwise ADSS Server daemons will not be registered in /etc/systemd/system). Click here to read how to change the owner and group once the installation has completed. Use the following command to mark install.sh file as executable before launching:

```
$ chmod + x install.sh
```

The following command will kick off the installer in GUI mode:

```
$ sh install.sh
```

The following command will run the installer in **Headless Mode (Non-GUI)**:

```
$ sh install.sh headless
```

The installation wizard will guide you through the various steps to ensure a complete and correct deployment of ADSS Server is achieved. These are detailed next in the upcoming sections. Three services will be registered in Windows Services Panel or /etc/systemd/system on UNIX.

ADSS Server is not installed as a single Windows NT service or a Unix daemon. A standard installation of ADSS Server is comprised of three components:

ADSS Core, ADSS Console and ADSS Service.

Each of these components uses a separate JVM. For a standard installation of ADSS Server all three components is installed on one machine. For a custom installation, it is possible to install the components on separate machines. It is possible to install multiple ADSS Core, and ADSS Console instances for high availability, together with multiple ADSS Service instances to load-balance the service requests for higher throughput.

4.2 Installation Steps

ADSS Server uses dynamic master key to ensure protection and security of data. To generate a dynamic master key, ADSS Server provides multiple mechanisms to its users that are categorized into different key types. These includes:

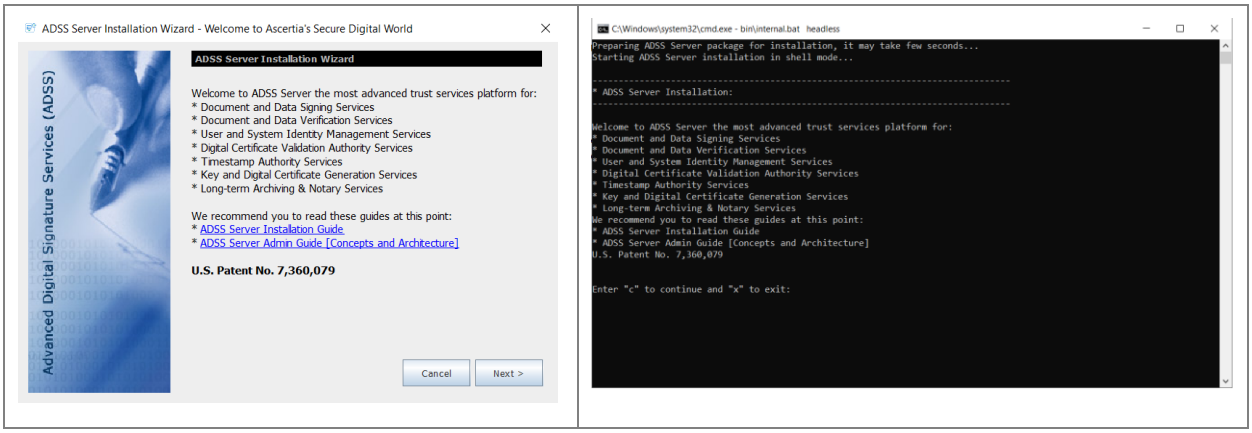
- Software based key – Auto Startup
- Software based key with M of N controls – Manual Startup
- Hardware based key – Manual Startup

The desired key type will be selected according to the requirement. The details of each key type is explained below:

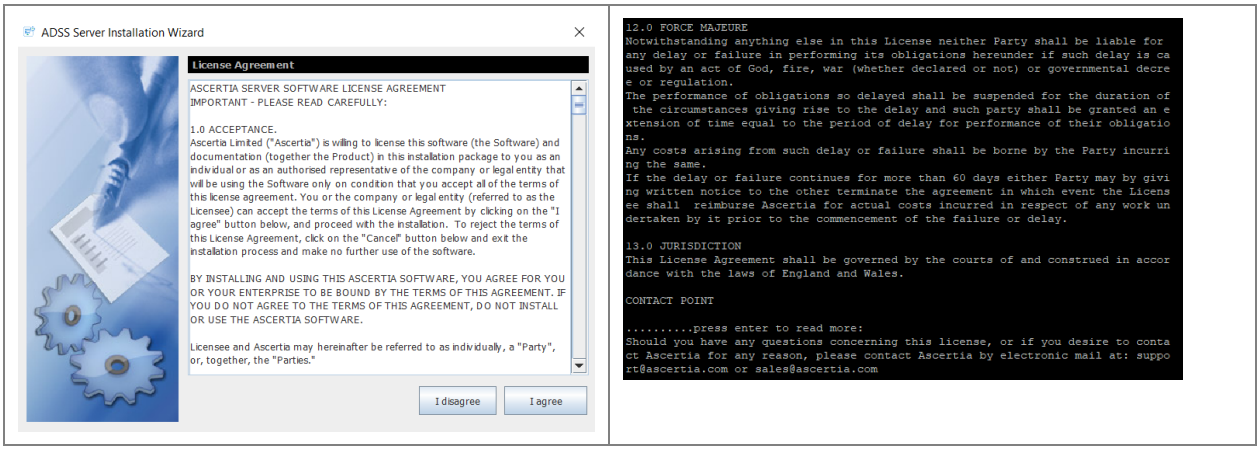
4.2.1 Software Based Key – Auto Startup

In this scheme, master key is generated using a software crypto source and protected by ADSS Server. Master key can be renewed after regular intervals in order to ensure security. Here, the master key will be protected by ADSS Server itself hence ADSS will be started without any operator's intervention.

Running the **install.bat/sh** file located at **[ADSS-Server-Home]/setup** directory shows the following screen:

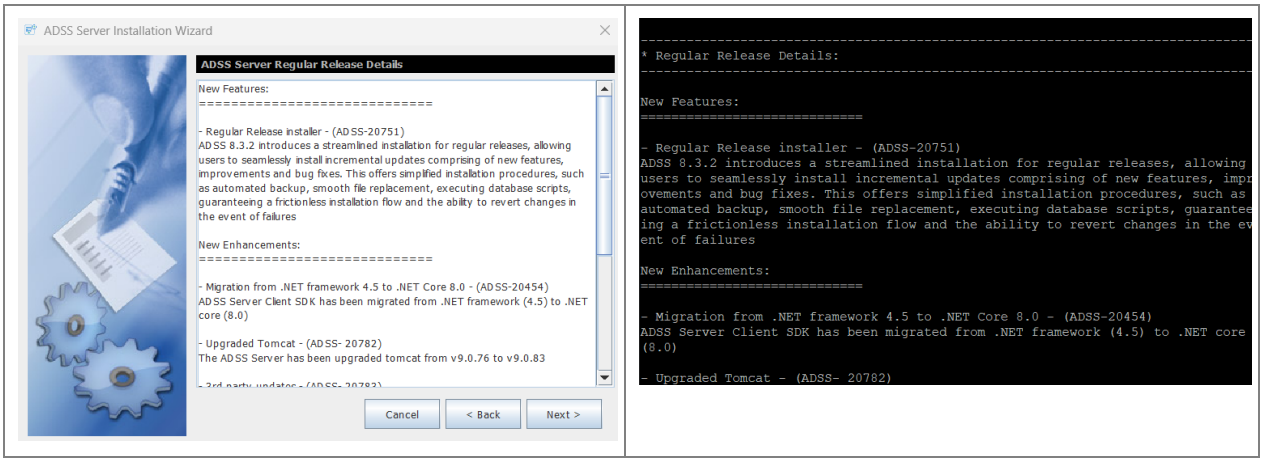


Clicking **Next >** shows the following screen:

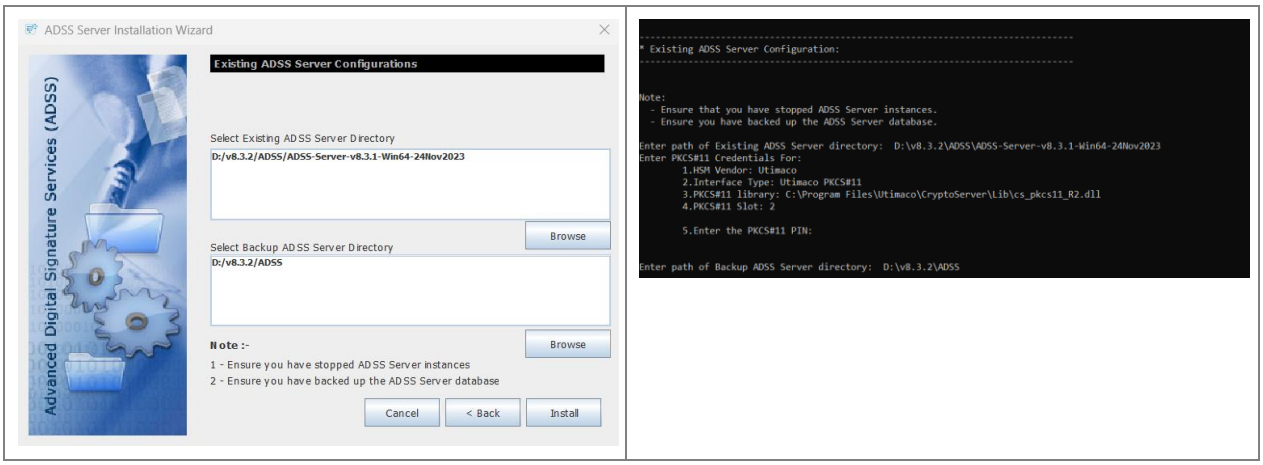


If you agree with the displayed terms and conditions, then click "**I agree**" to continue the installation process otherwise click "**I disagree"** to stop the installation process.

Clicking **I agree** shows the following screen:



This screen shows the details of the regular release. To continue the installation process click on the **Next** button, it will display the following screen:



The above screen allows the user to get the existing ADSS Server Directory and Backup ADSS Server Directory from the file system. After the directories are uploaded, the regular installer determines that the

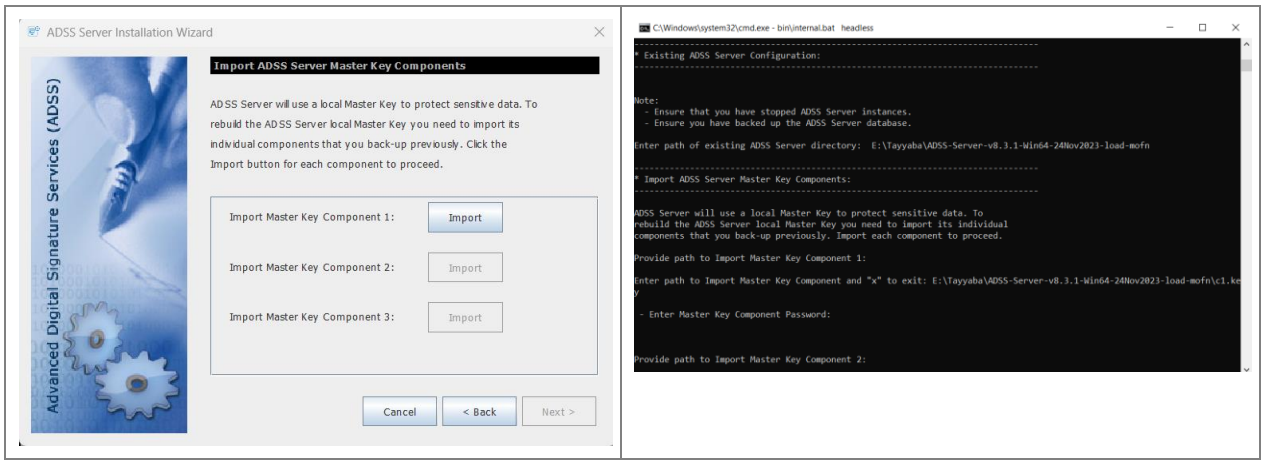
uploaded ADSS Server file is either ‘Software based key – Auto Startup’, ‘Software based key with M of N controls’, or ‘Hardware based key’.

If the uploaded ADSS Server file is ‘Software based key – Auto Startup’, then the installation will be continued as per the traditional mechanism. However, if the uploaded ADSS Server file supports ‘Software based key with M of N controls’ mechanism, then refer to the section below:

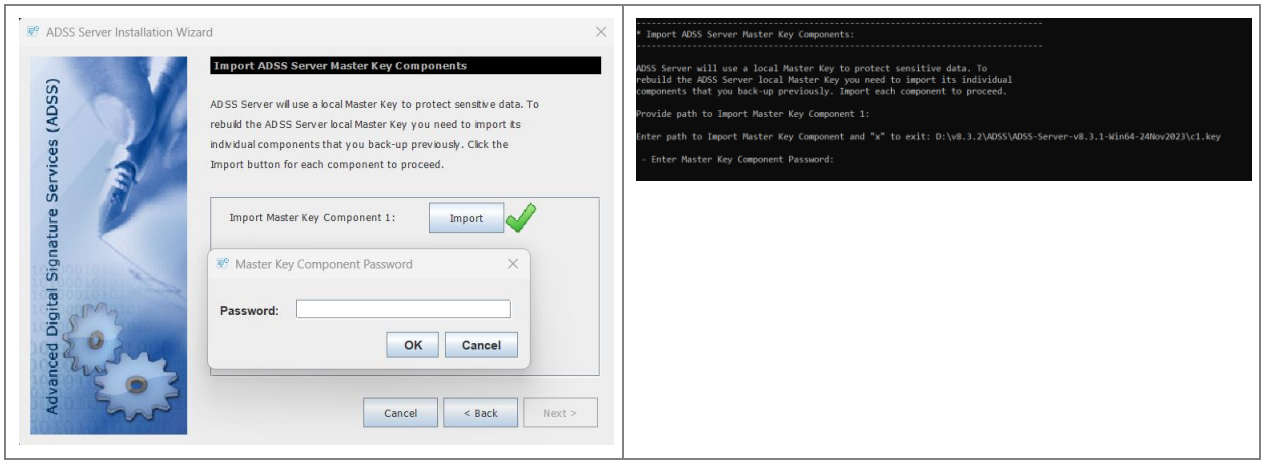
4.2.2 Software Based Key with M of N Controls – Manual Startup

In this mode, the Master Key is generated using a software crypto provider and split according to M of N rule. The minimum value of M will be 2 and N will be 3. The maximum value of M of N will be 16.

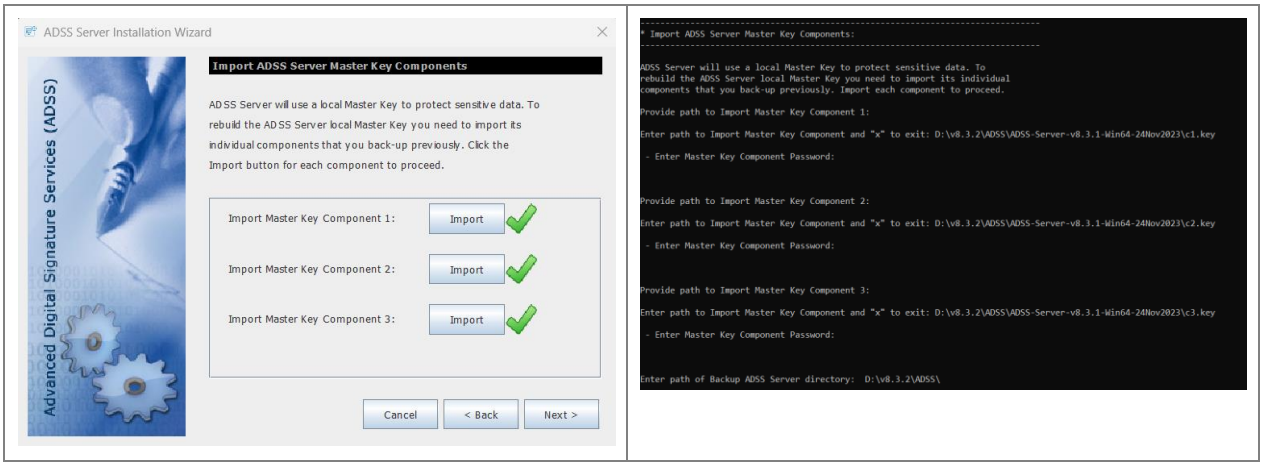
The below screen allows the user to upload the generated master keys for the uploaded ADSS Server directories:



While importing the master key, the operator will be asked to set a password for it:



It is recommended to set a different password for each master key. Once all the master keys are successfully imported, the below screen will be displayed:

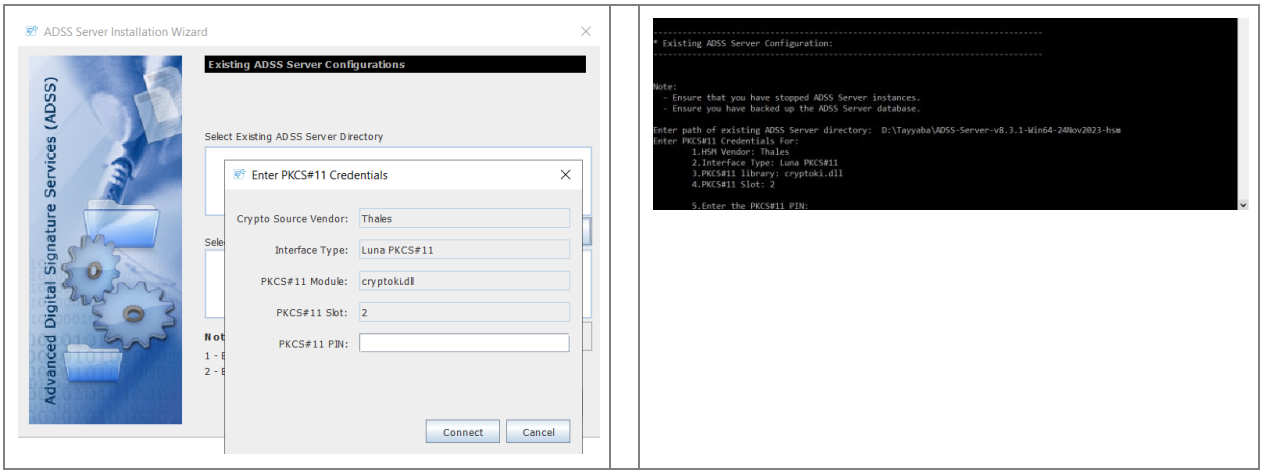


After importing the master keys, click on the **Next** button, the installation will continue as per the traditional mechanism.

If the uploaded ADSS Server file supports 'Hardware based key' mechanism, then refer to the section below:

4.2.3 Hardware Based Key – Manual Startup

In this mode, a key is created inside an HSM that is used as Master Key for ADSS Server. Below screen will be displayed during the installation process:



The above screen will allow the operator to PKCS#11 PIN in order to establish the connection with the HSM.

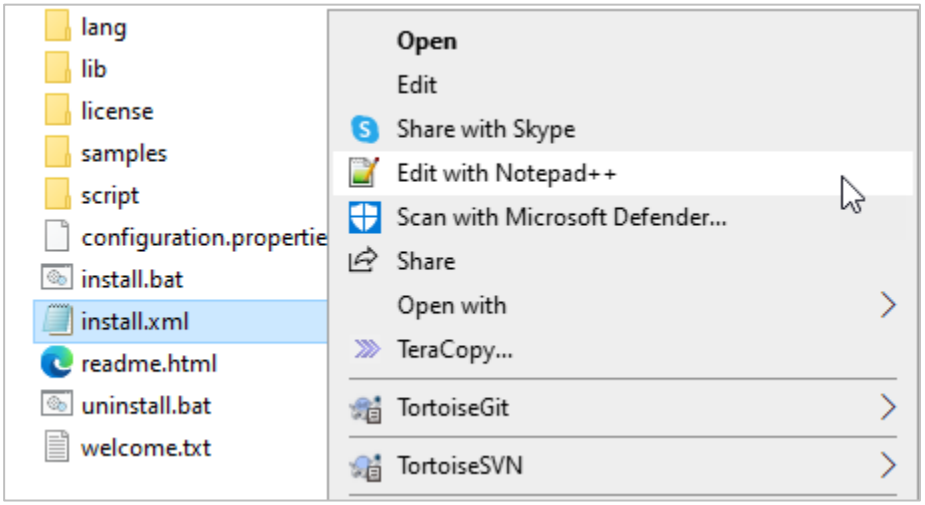
Afterwards, the installation will be continued as per the traditional mechanism. To view more details regarding the ADSS Server installation, refer to ADSS-Server-Installation-Guide located at **[ADSS-Server-Home]/docs** directory.

4.3 Installation using Silent Mode

For Installation in Silent Mode, we have an 'install.xml' file located [ADSS-Server-Installation-Directory]/setup folder. It is an xml file that can be opened in Notepad (or Notepad ++). The file contains all the necessary options that are required for the Installation of ADSS Server, but in XML format. The operator can edit the file, fill in all the required fields as per its requirement (same as GUI or Headless mode) and proceed with the Installation.

In order to install ADSS Server in Silent Mode, follow the steps below:

1. Navigate to [ADSS-Server-Installation-Directory]/setup
2. Open ‘install.xml’ file with Notepad++



3. Once the file is open, it displays the following screen:

```
<?xml version="1.0" encoding="UTF-8"?>
<Wizard>
    <Welcome visible="true"/>
    <LicenseAgreement visible="true">
        <Agree>TRUE</Agree> <!-- possible values are TRUE/FALSE -->
    </LicenseAgreement>
    <InstallationMode visible="true">
        <Mode masterKey="generate">FIRST_TIME</Mode> <!-- possible values are FIRST_TIME/LOAD_BALANCE/UPGRADE/EXISTING_DATABASE/REGULAR_RELEASE, Possible
        values for masterKey are generate,import -->
    </InstallationMode>
    <GenerateMK visible="true">
        <AssuranceLevel>1</AssuranceLevel> <!-- possible values 1/2/3, These denote to 1=Software based key - Auto Startup,2=Software based key with M of N
        controls - Manual Startup,3=Hardware based key - Manual Startup Default 1-->
        <MofN>2-3</MofN>                        <!-- (M of N rule 2 of 3, only required for Software based key with M of N controls) Minimum and
        Default value 2-3 and Maximum value 16-16 -->
        <HsmVendor></HsmVendor>                 <!-- possible values are UTIMACO/THALES/ENTRUST_NCIPHER/OTHER only required for Hardware based key-->
        <InterfaceType></InterfaceType>         <!-- possible values When HsmVendor is UTIMACO (UTIMACO/UTIMACO_CPS), THALES(SAFENET/SAFENET_CC),
        ENTRUST_NCIPHER (THALES/THALES_CC) and for OTHER (OTHER) only required for Hardware based key-->
        <Pkcs11Module></Pkcs11Module>           <!-- PKCS#11 driver library file name/complete path for this hardware device -->
        <Pkcs11Slot></Pkcs11Slot>               <!-- appropriate PKCS#11 slot no -->
        <Pkcs11Pin></Pkcs11Pin>                 <!-- PIN or password for the slot chosen -->
        <Pkcs11UserId></Pkcs11UserId>           <!-- Specify the User ID for the Crypto User, only required for UTIMACO_CPS Interface Type-->
        <Pkcs11UserPin></Pkcs11UserPin>         <!-- Specify the PIN for above Crypto User, only required for UTIMACO_CPS Interface Type-->
        <MasterKeyAlias></MasterKeyAlias>       <!-- Specify the Master key Alias(value of Key_alias found in conf/adss_startup.properties file), Only
        required when Mode is LOAD_BALANCE/EXISTING_DATABASE or masterKey='import'-->
        <MasterKeyKAKPrivate></MasterKeyKAKPrivate>     <!-- Specify the Master key KAK private(value of Kak_pr found in conf/adss_startup.properties file) ,
        Only required when Mode is LOAD_BALANCE/EXISTING_DATABASE or masterKey='import'-->
        <MasterKeyKAKPublic></MasterKeyKAKPublic>       <!-- Specify the Master key KAK Public(value of Kak_pu found in conf/adss_startup.properties file) ,
        Only required when Mode is LOAD_BALANCE/EXISTING_DATABASE or masterKey='import'-->
    </GenerateMK>
    <ManagingMK visible="true">
        <MbkPath1 password=""></MbkPath1> <!-- Same attributes are require either for backup (FIRST_TIME/Prior 6.0 UPGRADE) or for import
        (LOAD_BALANCE/EXISTING_DATABASE) -->
        <MbkPath2 password=""></MbkPath2>
        <MbkPath3 password=""></MbkPath3>
        <!-- N number of paths will required here, only for 2 AssuranceLevel -->
        <!--<MbkPath4 password=""></MbkPath4>-->
        <!--<MbkPath5 password=""></MbkPath5>-->
    </ManagingMK>
    <InstallDirectoryPath visible="true"/>
    <PreferredIpVersion visible="true">
        <IpVersion>IPv4</IpVersion> <!-- possible values are IPv4/IPv6 -->
        <!-- Provide IPv4/IPv6 address accordingly, If IP address is not defined then installer will pick System's IP address-->
        <IpAddress></IpAddress>
    </PreferredIpVersion>
    <HostName visible="true">
    <!-- Provide hostname accordingly, If host name is not defined then installer will pick System's host name-->
        <Host></Host>
    </HostName>
    <!-- upgrading screen-->
    <ExistingInstallation visible="true">
        <!--<Path>EXISTING_ADSS_INSTALLATION_DIR</Path>-->
        <BackupDirectory>
<!--        Backup directory path in case of regular release install path-->
<!--            <Path>BACKUP_DIRECTORY</Path>-->
        </BackupDirectory>
    </ExistingInstallation>
    <LicenseOption visible="true">
        <Option>EVALUATION</Option> <!-- possible values are COMMERCIAL/EVALUATION -->
    </LicenseOption>
    <LicenseFilePath visible="true">
        <!--<Path>../core/license.xml</Path>--><!--license file path-->
    </LicenseFilePath>
    <LicenseType visible="true">
        <Type>FULL_ADSS_SERVER</Type> <!-- possible values are
        FULL_ADSS_SERVER/DOCUMENT_SIGNING_VERIFICATION/EID_VALIDATION/TIMESTAMPING_ARCHIVING/PKI_INFRASTRUCTURE -->
    </LicenseType>
    <SampleData visible="true">
        <Insert>TRUE</Insert>
    </SampleData>
    <DatabaseConfiguration visible="true">
        <DatabaseType>MSSQL</DatabaseType> <!-- possible values are MSSQL,AZURESQL,PGSQL,ORACLE,MYSQL -->
        <Configuration>TYPICAL</Configuration> <!-- possible values are TYPICAL/ADVANCED -->
    </DatabaseConfiguration>
    <TypicalDatabaseConfiguration visible="true">
        <MachineName></MachineName>
        <Port>1433</Port>
        <Authentication></Authentication> <!-- possible Values are SERVER_AUTH, WINDOW_AUTH-->
        <DatabaseName></DatabaseName>
        <UserId></UserId>
        <Password></Password>
    </TypicalDatabaseConfiguration>
    <AdvancedDatabaseConfiguration visible="true">
        <UserId></UserId>
        <Password></Password>
        <JDBCURL></JDBCURL>
        <!-- Use the advanced JDBC URL for one of the PostgreSQL, SQL Server, Azure SQL, Oracle and MySQL accordingly -->
        <!--<JDBCURL>jdbc:postgresql://<DATABASE_MACHINE>:5432/<DATABASE_NAME></JDBCURL>-->
        <!--<JDBCURL>jdbc:sqlserver://<DATABASE_MACHINE>:1433;databaseName=<DATABASE_NAME></JDBCURL>-->
        <!--<JDBCURL>jdbc:sqlserver://<DATABASE_MACHINE>;database=<DATABASE_NAME></JDBCURL>-->
        <!--<JDBCURL>jdbc:oracle:thin:@<DATABASE_MACHINE>:1521:<DATABASE_NAME></JDBCURL>-->
        <!--<JDBCURL>jdbc:mariadb://<DATABASE_MACHINE>:3306/<DATABASE_NAME></JDBCURL>-->
    </AdvancedDatabaseConfiguration>
    <InstallMode visible="true">
        <Mode>TYPICAL</Mode> <!-- possible values are TYPICAL/CUSTOM -->
    </InstallMode>
    <CustomInstallation visible="true">
        <Core>TRUE</Core>
        <Console>FALSE</Console>
        <Signing>FALSE</Signing>
        <Verification>FALSE</Verification>
        <Certification>FALSE</Certification>
        <OCSP>FALSE</OCSP>
        <TSA>FALSE</TSA>
        <XKMS>FALSE</XKMS>
        <LTANS>FALSE</LTANS>
        <SCVP>FALSE</SCVP>
        <Decryption>FALSE</Decryption>
        <TSLMonitor>FALSE</TSLMonitor>
        <CRLMonitor>FALSE</CRLMonitor>
        <GoSign>FALSE</GoSign>
        <OCSPMonitor>FALSE</OCSPMonitor>
        <OCSPRepeater>FALSE</OCSPRepeater>
        <RA>FALSE</RA>
        <RAS>FALSE</RAS>
        <SAM>FALSE</SAM>
        <CSP>FALSE</CSP>
        <HMAC>FALSE</HMAC>
        <SPOC>FALSE</SPOC>
        <NPKD>FALSE</NPKD>
    </CustomInstallation>
    <MemoryConfiguration visible="true">
        <x86>
            <CoreLimit>1024</CoreLimit>
            <ConsoleLimit>1024</ConsoleLimit>
            <ServicesLimit>1536</ServicesLimit>
        </x86>
        <x64>
            <CoreLimit>1024</CoreLimit>
            <ConsoleLimit>1024</ConsoleLimit>
            <ServicesLimit>2048</ServicesLimit>
        </x64>
    </MemoryConfiguration>
    <ProgressFrame visible="true"/>
    <ServiceConfiguration visible="true">
        <InstallAsService>TRUE</InstallAsService>
        <StartService>TRUE</StartService>
        <InstallDefaultAdminPFX>TRUE</InstallDefaultAdminPFX>
        <ComputeHMAC>TRUE</ComputeHMAC>
    </ServiceConfiguration>
    <UpgradeSummary visible="true"/>
</Wizard>
```

4. The above image shows the contents of the file. The operator need to fill in the required information according to its requirement in the respective xml fields.

For installation in Silent Mode, we need to set the <visible =" "> attribute value to FALSE (wherever true) in xml file.

Fill out the information in xml file as mentioned below:

a. Set License Agreement tag to 'true' if you agree with the displayed terms and conditions. Otherwise set 'false' to stop the installation process.

b. As explained earlier, there are various installation modes available in ADSS Server. To set the mode of installation and master key, we have <InstallationMode> tag in install.xml file. An operator can either generate new or import an existing Master Key and select the required installation option in via this tag. Possible values are FIRST_TIME, LOAD_BALANCE/UPGRADE, EXISTING_DATABASE and REGULAR_RELEASE.

c. The next step is to set the type of master key that is to be generated. We will set all the required properties of master key in <GenerateMK>. The list of properties inside this tag are explained below:

   i. Use <AssuranceLevel> tag to set the type of master key to be generated. Here, possible values are 1,2 and 3 where:
      - 1 represent Software based key - Auto Startup,
      - 2 represents Software based key with M of N controls - Manual Startup,
      - 3 represents Hardware based key - Manual Startup

      All these key types have been explained in the section 4.1.1. Default value is 1.

   ii. Use <MofN> tag to define the M of N rule. It will only be used when Software based key with M of N controls is selected. Default value is 2-3 and maximum value is 16-16.

   iii. Use <HsmVendor> tag to define the required HSM. It will only be used in case of Hardware based key – Manual Startup. Possible values are UTIMACO, THALES, ENTRUST_NCIPHER and OTHER.

   iv. Use <InterfaceType> tag to define the interface type of the selected HSM. It will only be used in case of Hardware based key – Manual Startup. Possible values are UTIMACO (UTIMACO/UTIMACO_CP5), THALES(SAFENET/SAFENET_CC), ENTRUST_NCIPHER (THALES/THALES_CC) and for OTHER (OTHER).

   v. The <Pkcs11Module> tag is used to define the PKCS#11 driver library file name or the complete path for the selected hardware device.

   vi. The <Pkcs11Slot> tag is used to set an appropriate PKCS#11 slot number.

   vii. The <Pkcs11Pin> tag is used to set PIN or password for the chosen PKCS#11 slot.

   viii. The <Pkcs11UserId> tag is used to specify the User ID for the crypto user. The User ID will only be set in case where UTIMACO_CP5 Interface Type is selected.

   ix. The < Pkcs11UserPin> tag is used to specify the PIN for the above crypto user. The User PIN will only be set in case where UTIMACO_CP5 Interface Type is selected.

   x. The <MasterKeyAlias> tag is used to specify the Master Key Alias. The value of Key_alias can be found in conf/adss_startup.properties file. It is only required in the case where Installation Mode is LOAD_BALANCE/EXISTING_DATABASE or masterKey='import'.

   xi. The <MasterKeyKAKPrivate> tag is used to specify the Master Key KAK Private. The value of KAK private can be found in conf/adss_startup.properties file. It is only required in the case where Installation Mode is LOAD_BALANCE/EXISTING_DATABASE or masterKey='import'.

   xii. The <MasterKeyKAKPublic> tag is used to specify the Master Key KAK Public. The value of KAK public can be found in conf/adss_startup.properties file. It is only

required in the case where Installation Mode is LOAD_BALANCE, EXISTING_DATABASE or masterKey='import'.

d. As mentioned in section 4.4.1, once the master key is generated, we need to take the backup of the master key in the form of three components (or more if Assurance Level '2' is selected) and encrypt them with the provided password. For this <ManagingMK> tag will be used. Below tags will be used to set the password of master key components:
    i. The <MbkPath1 password=""> tag will be used to set the password for first key component.
    ii. The <MbkPath2 password=""> tag will be used to set the password for second key component.
    iii. The <MbkPath3 password=""> tag will be used to set the password for third key component.
e. The <InstallDirectoryPath> tag will enable the operators to see the directory where the master key components are saved on the file system.
f. The <PreferredIpVersion> tag is used to set the required IP version (IPv4 or IPv6) for ADSS Server Installation.
    i. The <IpVersion> tag is used to set the required IP version.
g. The <HostName> tag is used to define the host name for ADSS Server. If the host name is not defined, then the installer will pick up system's default host name.
h. The <ExistingInstallation> tag is used in order to upgrade the current version of ADSS Server. The operator will need to provide the current directory path inside the existing installation tag.
    i. The <BackupDirectory> tag (located under Existing Installation) is used to specify the Backup directory path in case of regular release installation.
i. The <LicenseOption> tag allows the operator to set the type of license to be used in ADSS Server installation. The possible values are COMMERCIAL or EVALUATION.
j. The <LicensePath> tag allows the operator to set the path from where the license will be uploaded from the files system for ADSS Server installation.
k. The <LicenseType> tag allows the operator to set the features to be used (Full ADSS Server or some specific modules) with the current ADSS Server installation. The possible values are FULL_ADSS_SERVER, DOCUMENT_SIGNING_VERIFICATION, EID_VALIDATION, TIMESTAMPING_ARCHIVIN and PKI_INFRASTRUCTURE.
l. The <SampleData> tag is used to insert sample data in order to allow immediate testing of ADSS Server.
    i. We need to set the <Insert> tag (under Sample Data tag) to TRUE in order to insert the required sample data.
m. The <DatabaseConfiguration> tag allows you to set the required configurations relating to the database type and configuration modes. It contains the following tags:
    i. The <DatabaseType> tag allows the type of the database that is to be used for ADSS Server Installation. The possible values are MSSQL, AZURESQL, PGSQL, ORACLE and MYSQL.
    ii. The <Configuration> tag allows the operator set the type of configuration for the selected database type. Possible values are TYPICAL and ADVANCED:
        - Use <TypicalDatabaseConfiguration> tag to set the values for Machine Name, Port, Authentication type (Server or Windows), Database Name, User ID and Password.
        - Use <AdvancedDatabaseConfiguration> tag to set the values for User ID and Password. Also we need to set the advanced JDBC URL for one of the PostgreSQL, SQL Server, Azure SQL, Oracle and MySQL accordingly.
n. The <InstallMode> tag allows the operator to set the installation mode i.e. whether the operator wants to install all the components of ADSS Server i.e. TYPICAL or install all the components of ADSS Server with custom memory parameters i.e. CUSTOM.
o. The <CustomInstallation> tag allows the operator to set the selected ADSS Server components and service modules with custom memory parameters.

p. The <MemoryConfiguration> tag allows the operator to set memory limit for each instance of ADSS Server i.e. Core Limit, Console Limit and Service Limit for both x86 an x64 systems.
q. The <ProgressFrame> tag allows the operator to track the progress of installation.
r. Under Service Configuration tag, make sure that the values of <InstallAsService>, <StartService>, <InstallDefaultAdminPFX> and <ComputeHMAC> are set to TRUE.
s. Save and exit the xml file.

Launch the install.bat or install.sh file, the ADSS Server will be installed with the required set of configurations of install.xml file.

4.4 Launching ADSS Server Admin Console

To access ADSS Server Admin Console, open a web browser (where you imported the Administrator PFX above) and type the following URL:

https://{Machine-Name}:8774/adss/console

Where machine-name is one of:

- localhost (in a case when ADSS Server is accessed on the local system where it is deployed)
- A local network system name (e.g. adss-server-machine1)
- An IP Address
- A URL (e.g. globaltrustfinder.com)

Initially you will be presented with a default TLS Client Authentication certificate that is pre-configured in ADSS Server. It is recommended that you change this default certificate by creating/importing a new certificate from ADSS Server Admin Console after login. Click here for more information. Note you can also use certificates issued by third parties.

Initially you will be presented with a temporary TLS Server Authentication certificate that is pre-configured in ADSS Server. This is the default administrator certificate. You should change it by creating a new certificate using the ADSS Server admin console. Refer to the ADSS Server Knowledge Base for more details. Ascertia recommends creating at least two operators.

A popup dialog may be shown; listing TLS Client Authentication certificates installed in the browser (including the one installed during ADSS Server installation) and asking you to choose appropriate certificate. Choose the certificate with a common name of “ADSS Default Admin” to login the ADSS Server Console.

Before launching the admin console, make sure that you have installed/imported adss_default_admin.pfx from [ADSS-Server-Home]/setup/certs/ directory in your web browser.

4.5 Uninstalling ADSS Server

To start the uninstallation, navigate to **[ADSS-Server-Home]/setup** directory. Either using a command line or Windows GUI interface.

Windows

To uninstall ADSS Server on Windows platform, go to **[ADSS-Server-Home]/setup** directory and run **uninstall.bat** file as administrator. This process will stop and then delete the registered ADSS Server components from Windows Service Panel.

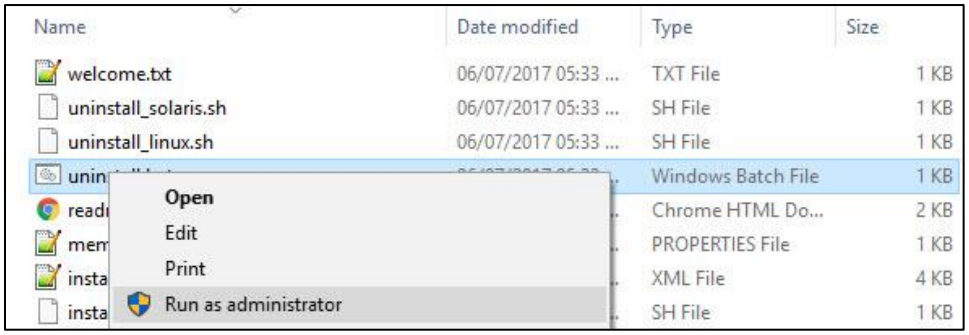


Figure 3 - Windows Example Uninstall Run as administrator

UNIX

To uninstall ADSS Server on a UNIX platform, go to **[ADSS-Server-Home]/setup** and run **`sh uninstall_linux.sh or uninstall_solaris.sh`** command accordingly under root user to delete registered ADSS Server components from **/etc/systemd/system**.

Use the following command to mark uninstall.sh file as executable before launching:

```
$ chmod + x uninstall_linux.sh
            Or
$ chmod + x uninstall_solaris.sh
```

The following command will run the uninstaller:

```
$ sh uninstall_linux.sh
          Or
$ sh uninstall_solaris.sh
```

On both Windows and UNIX platforms the uninstall procedure will not delete the directory structure and contents of ADSS Server, nor remove the database and its contents.

It is very important to securely delete any HSM held keys if the system is uninstalled because of a decommissioning exercise. This is not within the scope of ADSS Server and the relevant manufacturer will provide the necessary instructions to achieve this.

4.6 ADSS Server Service Interface URLs

Once ADSS Server is installed you can use ADSS Server service interfaces to process requests from your business applications. Interface URLs are documented in each service's **Interface URLs** page in Admin Guide. Each service has a unique URL and there are also variants for each depending upon which protocol the client wishes to use e.g. OASIS DSS or HTTP protocol for signature operations.

4.7 Troubleshooting

If any of the ADSS Server Core, Console or Service components fail to start after installation or there is a failure during installation wizard then ensure the following:

- Allow 1150 connections on your database server to allow ADSS Server to function at the recommended capacity. Note the connection pooling ensures these are maximum values and will not be created unless capacity demands it.
- The appropriate ADSS Server package for Windows/UNIX was chosen to install on the relevant machine.
- There should be no space character anywhere in the ADSS Server installation directory path.
- ADSS Server should be installed with administrator/root user privileges.
- In case of Windows platform check the following services are found in Windows Services Panel.
    - Ascertia-ADSS-Core
    - Ascertia-ADSS-Console
    - Ascertia-ADSS-Service
- In case of UNIX platform check that following service daemons are found within /etc/systemd/system
    - tomcatd-ADSS-core
    - tomcatd-ADSS-console
    - tomcatd-ADSS-service
- If ADSS Server does not start automatically after installation, then manually start ADSS Server;

  start following services from Windows Services Panel on Windows OS:
    - Ascertia-ADSS-Core
    - Ascertia-ADSS-Console
    - Ascertia-ADSS-Service

  On UNIX, use these commands to start the services:
    - systemctl restart tomcatd_core_linux.service
    - systemctl restart tomcatd_console_linux.service
    - systemctl restart tomcatd_service_linux.service
- If a certificate is not shown, then it is because of one of the following reasons:
    - The browser settings are such that the certificate is automatically selected.
    - There was a problem importing TLS client authentication certificate into the browser.

If Technical Support is required, Ascertia has a dedicated support team providing debugging, integration assistance and general customer support. Ascertia Support can be accessed as described in Chapter 1.

The installation procedure produces a log file called install.log, which is located in **[ADSS-Server-Home]/setup** directory. Any errors during installation will be recorded in this file.

Finally, consult the logs directory as each service produces its own unique log file.

4.8 Failure Event

In the event of a failure, an automatic restoration process for files is in place to ensure minimal disruption and data loss. However, it's important to note that this automated recovery pertains solely to files and does not extend to the database; hence, the database will not be restored.

5 Post-Installation Notes

The post-installation notes have been described in details in Section 3 of the **ADSS-Server-Installation-Guide** document. To view details, navigate to the **[ADSS-Server-Home]/docs** folder.

******* End of Document *******