ADSS Server v8.3.14 – Installation Guide

ASCERTIA LTD

DECEMBER 2025

Document Version- 1.0.0

© Ascertia Limited. All rights reserved.

This document contains commercial-in-confidence material. It must not be disclosed to any third party without the written authority of Ascertia Limited.

CONTENTS

1 INTRODUCTION 4
1.1 SCOPE 4
1.2 INTENDED READERSHIP 4
1.3 CONVENTIONS 4
1.4 TECHNICAL SUPPORT 4
2 SYSTEM REQUIREMENTS 5
2.1 TYPICAL DEPLOYMENT SCENARIO 6
2.2 HSM SUPPORT FOR KEY WRAPPING 7
3 PRE-INSTALLATION CHECKS 8
3.1 DOCUMENTATION 8
3.2 HARDWARE, NETWORK & OPERATING SYSTEM 8
3.3 MEMORY REQUIREMENTS 9
3.4 DATABASE CONFIGURATIONS 9
3.5 HARDENING ADSS SERVER INSTALLATION 12
3.6 SEPARATION OF DATA/PARTITIONING 16
3.7 USING DEFAULT/CUSTOM PORTS 16
3.8 HIGH AVAILABILITY (HA) REQUIREMENTS 16
3.9 DISABLE ANTI-VIRUS 16
3.10 ADSS SERVER OPERATOR ACCOUNTS & PRIVILEGES 16
3.11 NOTIFICATIONS & ALERTS 17
3.12 HARDWARE SECURITY MODULE 18
3.13 ADSS SERVER PROFILES 18
3.14 PKI BASED DEPLOYMENTS 18
3.15 CHECKING RESOURCE LIMITS 18
3.16 INSTALLATION OF RNG-TOOLS ON LINUX SYSTEM 18
4 ADSS SERVER INSTALLATION 19
4.1 INSTALLATION PROCESS 19
4.2 INSTALLATION USING SILENT MODE 54
4.3 LAUNCHING ADSS SERVER ADMIN CONSOLE 58
4.4 UNINSTALLING ADSS SERVER 59
4.5 ADSS SERVER SERVICE INTERFACE URLS 60
4.6 TROUBLESHOOTING 60
5 POST-INSTALLATION NOTES 62
5.1 SECURE INTER-SERVICE COMMUNICATION OF ADSS SERVER 62
5.2 SECURE DEPLOYMENT OF ADSS SERVER 62
5.3 ADSS SERVER OPERATORS 62
5.4 HSM CONFIGURATION 62
5.5 LOCAL CA CONFIGURATION 62
5.6 EXTERNAL TRUST SERVICES 63
5.7 NTP CONFIGURATION 63
5.8 DATABASE LOG ARCHIVING FREQUENCY 63
5.9 ALERT CONFIGURATIONS 63
5.10 LICENSING 63
5.11 PREPARE THE BACKUP STRATEGY 63
5.12 TRACE LOG SIZING GUIDE 64
5.13 ADSS SERVER CLIENTS 64
5.14 ADVANCE SETTINGS CONFIGURATION 64

TABLES

TABLE 1 - ADSS SERVER SYSTEM REQUIREMENTS 6
TABLE 2 - DATABASE CONNECTION PARAMETERS - TYPICAL.....33
TABLE 3 - DATABASE CONNECTION PARAMETERS – ADVANCED.....35

FIGURES

FIGURE 1 - TYPICAL ADSS SERVER DEPLOYMENT SCENARIO 7
FIGURE 2 - WINDOWS SERVICE PANEL ADSS SERVER PROCESS OWNER VIEW 9
FIGURE 3 - CHANGING THE DATABASE OWNER 13
FIGURE 4 - WINDOWS SERVICE PANEL ADSS SERVER PROCESS OWNER VIEW 14
FIGURE 5 - ADSS SERVER ROLE BASED ACCESS CONTROL EXAMPLE 17
FIGURE 6 - WINDOWS EXAMPLE INSTALLER RUN AS ADMINISTRATOR 19
FIGURE 7 - ADSS SERVER INSTALLATION WIZARD SUCCESS SCREEN 38
FIGURE 8 - WINDOWS EXAMPLE UNINSTALL RUN AS ADMINISTRATOR 59

1 Introduction

1.1 Scope

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

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	Ascertia Community Portal

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 17 EE application, supported on these platforms:
	
	Operating System
	The following 64-bit operating systems are supported:
	• Windows Server 2025, 2022, 2019 and 2016
	• Linux
	  ○ RedHat v7.x, v8.x, v9.0, v9.1, v9.3, v9.4, v9.5
	  ○ CentOS v7.x
	  ○ Ubuntu v20.x, v22.x, v24.x
	  ○ AlmaLinux v9.3, v9.4
	
	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, 21c, 23ai (Standard Edition, Enterprise Edition)
	• MySQL v8.0.x
	• Percona-XtraDB-Cluster v8.0.23
	• PostgreSQL v12, v13, v14, v15 and v16
	
	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
	• For further information on the database servers tested with Ascertia ADSS Server, please refer to the platform support report for ADSS Server which can be found here.

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

Components	Requirements
	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 • 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) **Note:** For further information on the database servers tested with Ascertia ADSS Server, please refer to the platform support report for ADSS Server which can be found here.
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:

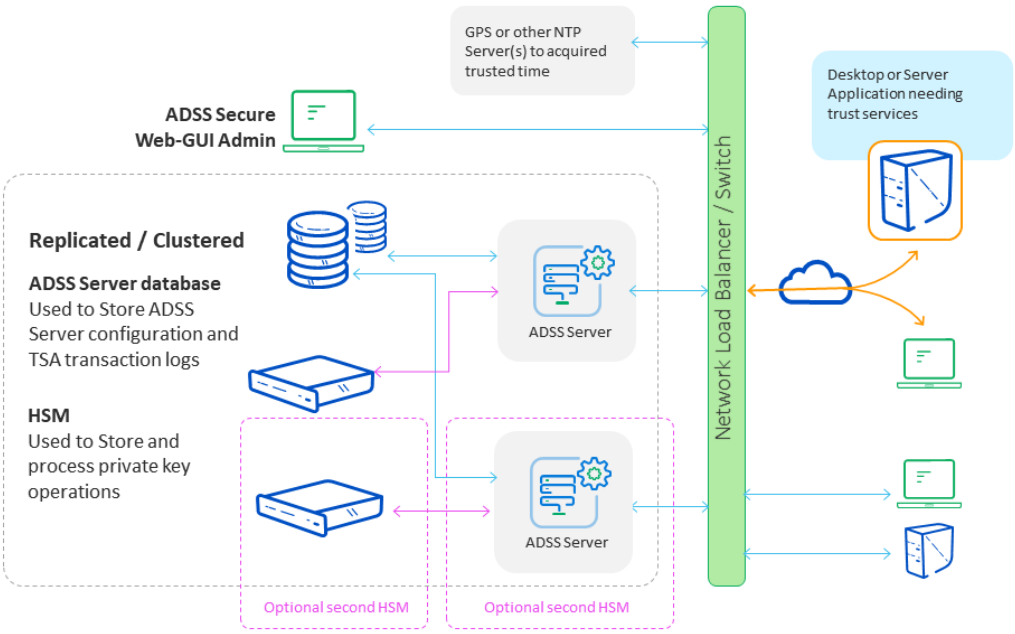


Figure 1 - Typical ADSS Server Deployment Scenario

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

Most of the installation failures or problems are due to a failure to complete all of the steps that are required before commencing the deployment. These pre and post check lists are intended for administrators to use in consultation with system administrators, storage administrators, network administrators, database administrators, and third-party hardware and software vendors to coordinate and plan the tasks for the ADSS Server installation. Planning and preparation are essential to ensure that your installation proceeds smoothly.

Ascertia recommends the following check list is followed before beginning the actual installation of ADSS Server.

3.1 Documentation

Review the documentation thoroughly and ensure all components are in place. At a basic level this means a database and if a production system, Hardware Security Module (HSM). The ADSS Server Installation Guides, Database Guides and Quick Guides are all in the /Docs folder of the downloaded zip file. The ADSS Server Admin Guide is available in the product's web-admin screen 'help' section and also here:

Welcome (ascertia.com)

Before proceeding with the installation, it is advisable to make yourself familiar with the services you intend to use. For example, Signing Service and Verification Service, which depend upon TSA and certificate revocation services.

3.2 Hardware, Network & Operating System

Review the required hardware, network, and operating system and ensure that they are in place before proceeding for ADSS Server Installation. For more see the section System Requirements.

3.2.1 Permissions & Service Owners

The following operating system privileges are required for ADSS Server installation:

- Linux or Solaris deployments require sufficient privilege to create the necessary service daemons for the Tomcat instances. For Linux or Solaris systems create the necessary user and group who will own and run ADSS Server instances. Note the user does not require any special permissions and an ordinary user account is sufficient. After completing the initial installation, you can update the owner and group settings. For step-by-step instructions on running ADSS Server as a non-root user, refer to the technical note titled ***'How to run ADSS Server services as a non-root user daemon on UNIX/Linux'*** available in the Ascertia Community Portal knowledge base.
- Windows administrator privilege is required to create Windows Services during the installation. The installation must be performed as a local administrator. Windows deployments will install the services to run under Local System. It is recommended a suitable low-level privilege user be assigned ownership of the services. To change the Windows Service owner of the ADSS Server modules, you will need to ensure they are stopped first. Use the standard dialogue box of the service to change the owner as shown here for the Console service:

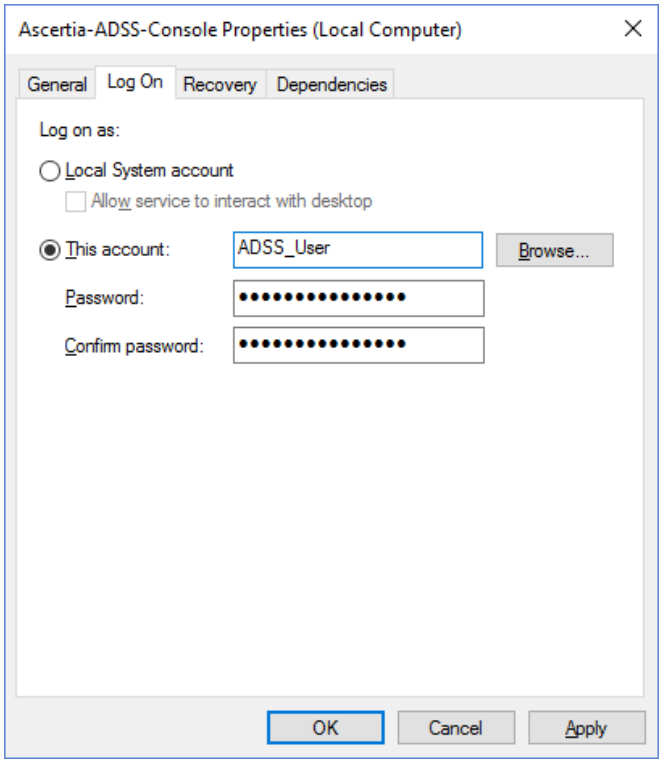


Figure 2 - Windows Service Panel ADSS Server Process Owner View

3.2.2 Hostname & IP Address

It is important to ensure the ADSS Server host system is correctly configured with regard to hostname, and resolution to IP of such. During the installation ADSS Server installer will use the given system hostname as the identifier and subsequently attempt to retrieve the IP address that this name resolves to. Therefore, either a suitable DNS or local hosts file entry must exist to achieve this. Take care when using a combination of long and short host names, i.e. host with and without fully qualified domain attached. Ascertia recommends using the long, fully qualified host name when deploying ADSS Server.

3.3 Memory Requirements

Review the memory requirements/disk space for the database as well as Tomcat instances that ADSS Server runs on. Note the default values are 1024 MB for Core and Console instances and 2048 MB for Service instance. For more details on memory requirements see the section System Requirements.

3.4 Database Configurations

Review the database user credentials and connection configuration information. Ensure database connectivity is established. If possible, test the credentials as provided by the database vendor. For detailed information on database setup, refer to the technical note titled ***'What are the minimum set of database privileges required to install ADSS Server'*** in the Ascertia Community Portal knowledge base.

3.4.1 Connection Pooling

ADSS Server uses Hibernate technology for database communication and C3P0 to handle the connection pooling. The default sizes for the connection pools are:

- ADSS Server Console: minimum 20 and maximum 50
- ADSS Server Core: minimum 30 and maximum 100
- ADSS Server Service: minimum 40 and maximum 1000

For detailed information on how to adjust the database connection pool size after installing ADSS Server, refer to the technical note titled ***'How to change the default database connection pool size limits in ADSS Server'*** available in the Ascertia Community Portal knowledge base.

3.4.2 Sizing

ADSS Server is a modular based system and disk space is dependent upon the services you have purchased. Click here for more details on space required for each service (Signing, Verification, Certification, OCSP, TSA and LTANS) and for logs.

3.4.3 Permissions

Ascertia recommends that two database users are used for ADSS Server. First, a user with extended privileges for the deployment and second, a user for normal operations. For detailed information about the permissions required to install ADSS Server, refer to the technical note titled ***'What are the minimum set of database privileges required to install ADSS Server'*** in the Ascertia Community Portal knowledge base. Operations of ADSS Server mainly requires permissions to CREATE, DROP, INDEX, INSERT, SELECT, EXECUTE, UPDATE and DELETE data.

3.4.4 Real-Time Certificate Status Checking

If the ADSS Server is required to use real time certificate status database checking, then follow the Quick-Guide-for-ADSS-Realtime-Revocation-Database.pdf located at location [ADSS Server Package]/docs. This guide describes the ADSS Server real-time database schema and how it can be populated with the real-time revocation information. Note that this real-time database must be created separately by the appropriate database administrator respectively, it is not part of the ADSS Server installation.

3.4.5 Configuring SQL Server Database

The following must be considered when installing ADSS Server with SQL Server:

1) When installing ADSS Server with SQL Server, remember to select English as the default database user language, the system will not work if another database language option is selected.
2) These configurations should be applied on SQL Server 2012 before installing ADSS Server:
    - Open SQL Server Configuration Manager
    - Click **SQL Server Services** on the left pane
    - Right-click on your SQL Server instance name on the right pane -> Default: SQL Server (MSSQLSERVER)
    - Click **Properties**
    - Click **Start-up Parameters**
    - On the **specify a start-up parameter** textbox type **-T272**
    - Click **Add**
    - Confirm the changes

    Source: http://www.big.info/2013/01/how-to-solve-sql-server-2012-identity.html

3.4.6 Configuring PostgreSQL Database

The following must be considered when installing ADSS Server with PostgreSQL:

1) If PostgreSQL is deployed on a Linux machine, then make sure to assign the language **plpgsql** after creating the database by executing the following SQL query:

```
$ create language 'plpgsql'
```

2) ADSS Server establishes a database connection pool on start-up and this connection pool has configurable limits for initial pool size and maximum pool size. For console operations and service requests, ADSS Server uses database connections from the established connection pool and acquires more connections as required until the connection pool size reaches the maximum allowed limit. For guidance on configuring database connection limits, refer to the technical note titled ***'How to change the default database connection pool size limits in ADSS Server'*** available in the Ascertia Community Portal knowledge base.

   The default connection pool size for a PostgreSQL database is 100 connections and this is generally insufficient for a production deployment of ADSS Server. Follow these instructions to increase the maximum connections limit in PostgreSQL:

   - Edit file postgresql.conf file at location **[PostgreSQL Installation Dir.]/data**
   - Set the value of the **max_connections = 1150**
   - Restart PostgreSQL server
   - Restart ADSS Server to have these changes take effect.

3.4.7 Configuring MySQL Database

The following must be considered when installing ADSS Server with MySQL:

1) If the MySQL database is running with TLS authentication, refer to the technical note titled ***'How to disable the ECC cipher in Tomcat's Catalina'*** in the Ascertia Community Portal knowledge base for detailed instructions on disabling ECC ciphers in Tomcat.

2) Although ADSS supports both case sensitive and case insensitive table names, but if MySQL database is being installed on a Linux/UNIX machine, it is recommended to configure MySQL database server with table names in case insensitive mode. Follow the instructions below:

   - Open **/etc/my.cnf** in edit mode
   - Under the **mysqld** section set the parameter **lower_case_table_names=1**

   It's worthwhile reading the details of lower-case table by following the below link:
   https://dev.mysql.com/doc/refman/8.0/en/identifier-case-sensitivity.html

Modify the configuration file before starting the freshly installed MySQL Server v8.0.x otherwise you have to re-install the MySQL Server again.

If ADSS Server Installation wizard was already launched before performing the above-mentioned steps, then follow these instructions:

1. *Cancel the installation wizard and remove the current ADSS Server directory.*
2. *Extract the ADSS Server again from the Zip File*
3. *Perform the above-mentioned instructions.*
4. *Launch the Installation wizard again so that installer can pick MySQL driver.*

3) MySQL may prevent ADSS Server inserting records larger than a particular size. If this problem occurs, then you need to configure MySQL database server to overcome this problem by setting **max_allowed_packet** parameter. Click here to read how to update the MySQL configuration file.

   When using large TEXT or BLOB, the combined size on log files has to be at least 10 times the size of largest such row. e.g. for a 10MB CRL set:
   - **innodb_log_file_size = 100M**
   - **innodb_log_files_in_group = 100**

   It's worthwhile reading the documentation on how this is changed:
   https://dev.mysql.com/doc/refman/8.0/en/innodb-parameters.html#sysvar_innodb_log_file_size

4) If MySQL Percona XtraDB Cluster is hosted on Linux, then MySQL database service has to be manually started every time when hosted Linux machine is restarted. Use following command to start/stop/restart or get the status accordingly e.g.

```
$ systemctl start mysql@bootstrap.service
```

3.5 Hardening ADSS Server Installation

Prior to the deployment of ADSS Server, it is imperative to ensure that any host is physically and network secure.

3.5.1 General Considerations

1) It is recommended that the ADSS Server and HSM modules communicate on a dedicated network link, either virtual or physical.
2) ADSS Server should be deployed on a dedicated host, either virtual or physical. It is recommended that no other software other than monitoring, is deployed on the same host as ADSS Server.
3) Block all unnecessary ports and services on ADSS Server host e.g. FTP, Telnet and NetBIOS etc.
4) Once deployed all administration and configuration of ADSS Server should be done via the admin console, which is accessible via a web browser and uses mutual TLS for authentication.
5) Use two Active Directory/LDAP accounts to manage the ADSS Server installation i.e. an **administrator / root** and a **service** account.
6) The **administrator / root** account should have the following rights:
   - On the ADSS Server database (*only when the database server is SQL Server with Windows Authentication*) to allow the installer creating and updating the schema.
   - To extract the ADSS Server setup.
   - To register/unregister the ADSS Server services/daemons on Windows and Linux.
   - Restarting the ADSS Server services/daemons.
   - Reviewing logs and configuration files.
   - Only this admin/root user account should be allowed to login the ADSS Server host machine for installation and housekeeping activities.

3.5.2 Hardening ADSS Server on Windows

1) If you are using the SQL Server as a database server, then during installation it is recommended to use the SQL Server with **Windows Authentication** (see section Database Connection Parameters for details). SQL Server with Windows Authentication doesn't require database password to be stored by the ADSS Server. In this case, the database password is managed by the Windows

operating system. To change the database owner, open the SQL Server Management Studio, login to the database server, right click the ADSS Server database and select **Properties** and follow the steps in the image below:

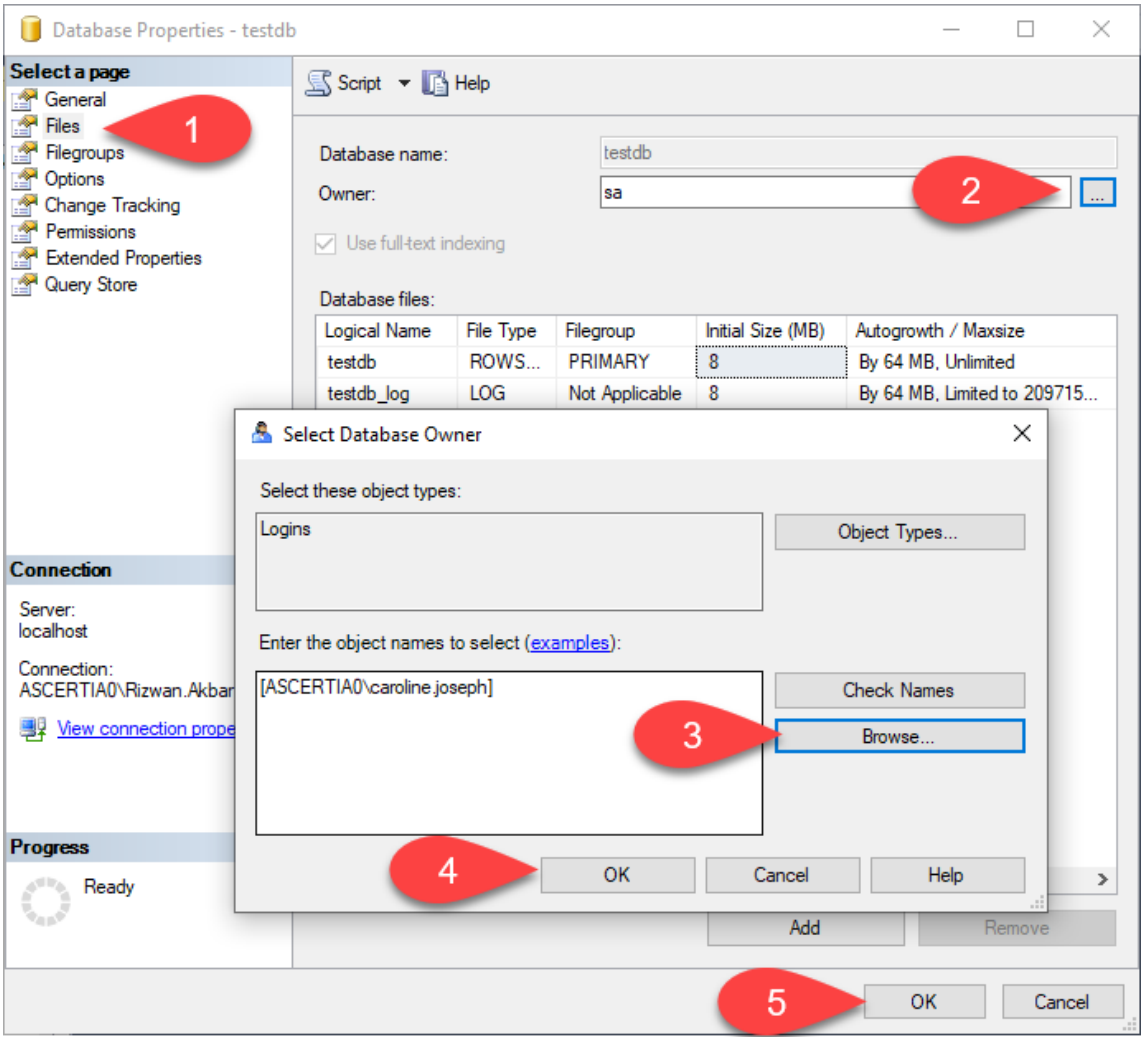


Figure 3 - Changing the Database Owner

2) Once the ADSS Server installation/upgrade has been completed, this account must no longer have the access to the ADSS Server database.

> *In future, when there is a need to upgrade the ADSS Server to the latest version, this administrator user must be given rights to the ADSS Server database on the SQL Server.*
>
> *This is only required when the database server is SQL Server.*

3) Once the ADSS Server has been installed, the ADSS Server services (i.e. Ascertia-ADSS-Core, Ascertia-ADSS-Console, Ascertia-ADSS-Service) must be running using a less privileged active directory account (Service Account) instead of a “Local Service” account.

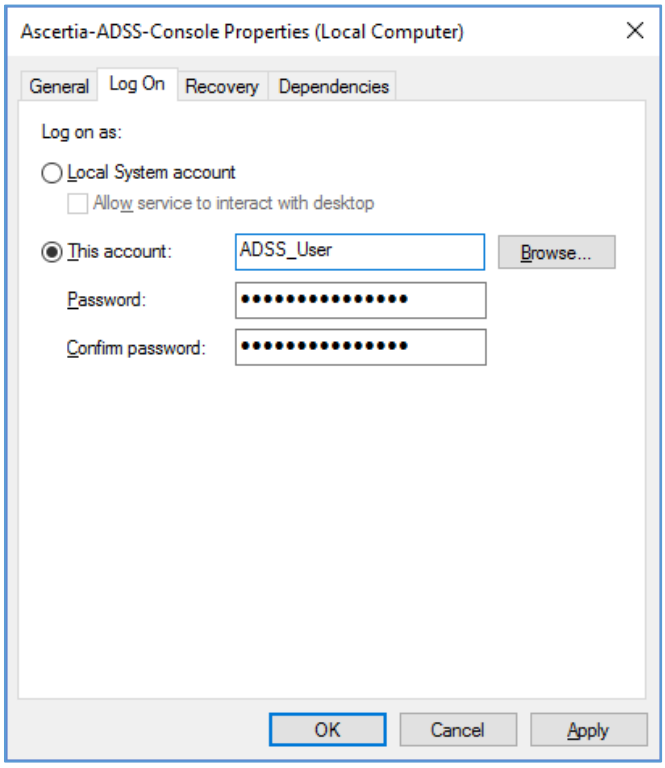


Figure 4 - Windows Service Panel ADSS Server Process Owner View

*In future when there is a need to upgrade the ADSS Server to the latest version then after upgrading you must change the **log on** user to the same "Service Account" again. The ADSS Server upgrade changes the log on user to the Local Service account.*

3.5.3 Hardening ADSS Server Installation on Linux

1) Root login should only be allowed at the Linux terminal.
2) Ensure administrators authenticate using their own login credentials and then use **su** or equivalent to obtain higher level privileges.
3) ADSS Server must be installed in **/usr/local/adss-server** directory.
4) Once the ADSS Server has been installed, the ADSS Server daemons (i.e. tomcatd-ADSS-core, tomcatd-ADSS-console, tomcatd-ADSS-service) must be running using a less privileged user instead of root user. For instructions on running ADSS Server services as a non-root user on Linux, refer to the technical note titled ***'How to run ADSS Server services as a non-root user daemon on UNIX/Linux'*** in the Ascertia Community Portal knowledge base.

3.5.4 Operational Considerations

1) **Service Account / non-root user** must not have the login rights on the ADSS Server host machine.
2) The **Service Account** should have the following rights:
   - Run the services.
   - Read/write/execute rights on the ADSS Server Installation directory only.
   - Read/write rights on the shared network path if log archives and/or CRLs are being stored on a separate network device.

- Rights on the ADSS Server database (i.e. SQL Server with Windows Authentication) to authenticate this user for the CRUD operations.

3) Login to ADSS Server Admin Console and do the following configurations:
   - Disable the default Admin account in ADSS Server and replace it with non-default named accounts. It is strongly recommended that operator TLS client certificates and associated private keys should be stored on a secure smart card/USB token thereby providing an extra layer of security for the private key plus two-factor authentication of the operator.
   - The revocation status of Ascertia ADSS Server operator TLS certificates should also be checked at the time of logon by configuring this from Global Settings > Miscellaneous. However, it is recommended that operators' accounts are also immediately updated on Ascertia ADSS Server at the time a certificate is revoked.
   - The Ascertia ADSS Server Admin Console ensures that access to system objects is strictly controlled. Users are first identified and authenticated as explained above, and once this process is complete and the user has successfully logged in, then access to system objects is controlled according to the user's role. Each role has a definition of which system objects it can access, and the type of access, e.g. read only, or edit/create/delete.
   - Replace the TLS Server authentication certificate with a non-default certificate.
   - Replace the following symmetric keys with non-default keys:
     - HMAC Key (preferably store in HSM)
     - Key Encryption Key - KEK (preferably store in HSM)
     - Data Encryption Key – DEK (DEK is always in software and encrypted with KEK)
   - Go to Trust Manager, edit the **ADSS Default Root CA**, uncheck the option **CA for verifying TLS client certificates** and **Save** the configurations.
   - Delete the following default data items:
     - **ADSS Default Root CA** from Trust Manager
     - **default_hmac_key** from Key Manager
     - **default_data_encrypting_key** from Key Manager
     - **default_key_encrypting_key** from Key Manager
     - **default_ssl_server_key** from Key Manager
4) It is recommended to Enable the Dual Control function when doing the critical configuration changes e.g. changing the Global Settings.
5) Change the ADSS Server Keystore password with a non-default secure password.
6) Disable all non-essential accounts from operating systems (Windows/Linux).
7) To ensure maximum security, it is recommended to enable strong ciphers in Apache Tomcat. For detailed guidance, refer to the technical note titled ***'Changing the SSL/TLS cipher suites used by the ADSS Server for advanced security and improved TLS ranking with SSL Labs'*** available in the Ascertia Community Portal knowledge base.
8) It is recommended to disable TLS 1.1 and TLS 1.2 in Apache Tomcat and use the stronger TLS 1.3 protocol to achieve a higher TLS ranking. For detailed instructions, refer to the technical note titled ***'Changing the SSL/TLS cipher suites used by the ADSS Server for advanced security and improved TLS ranking with SSL Labs'*** in the Ascertia Community Portal knowledge base.

3.5.5 Auditing

1) Ensure that only designated and known individuals have accounts that allow access to the ADSS Server host.

2) It is recommended to have a change management process in place for the access to the ADSS Server host machine and activities performed are monitored, logged and audited.

3.6 Separation of Data/Partitioning

It is recommended to separate aspects of ADSS Server. For Windows this means deploying ADSS Server to a separate partition from the operating system. On Linux and Solaris deployments follow good practice guidelines of deploying the binaries to a different directory structure to that of the log files ADSS Server generates during operation.

3.7 Using Default/Custom Ports

Review if it is required to use special custom ports or the default ports for ADSS Server Core, Console and Service instances. If using non-default ports note those required. The default ports are:

- 8774 for HTTPS admin console access.
- 8777 for HTTP access to services, i.e. Service instance.
- 8778 for HTTPS (TLS Server Authentication) access to services, i.e. Service instance.
- 8779 for HTTPS (TLS Client Authentication) access to services, i.e. Service instance.

If you need to change the default ports of ADSS Server Core, Console, or Service instances after deployment, refer to the technical note titled ***'How to change the default ports for ADSS Server services'*** in the Ascertia Community Portal knowledge base for detailed instructions.

For production systems, Ascertia recommends TLS Client Authentication to be used on port 8779 where possible and access to TLS Server Authentication and clear text HTTP are blocked. For example, access to OCSP Service over HTTP on port 8777 is likely to be required.

3.8 High Availability (HA) Requirements

Identify whether all services should be installed on a single machine or different machines for better performance or high availability. Note it is possible to run the Core and Console services in a high availability set-up, but it is not necessary for the Service instance.

The installer allows you to add ADSS Server instances to create a high availability environment. Click here for more details on ADSS Server High Availability notes and instructions.

3.9 Disable Anti-Virus

Disable any anti-virus, malware or equivalent protection software that may interfere with the installation process.

3.10 ADSS Server Operator Accounts & Privileges

Record who will have access to ADSS Server administration console. These individuals should be given dedicated digital IDs for ongoing accountability and traceability. It is recommended to disable the default administrator account once individual assigned ones have been put in place.

Determine whether dual authorisation control is required to administer system services. This can be applied to any of the public facing services such as Signing Service and to any of the support modules. For example, CRL Monitor or Key Manager.

3.10.1 Role Based Access & Fine Grain Access Control

ADSS Server implements role-based access for ADSS Server administrator access. The default roles of administrator, security officer and auditor are created during the installation. Please review these roles once installation is complete to ensure they match your requirements. Any number of roles can be added and configured separately to ensure the greatest flexibility.

Each role has fine grain access control based upon each module. This picture shows some of the choices available for Signing, Verification and Certification Services respectively: -

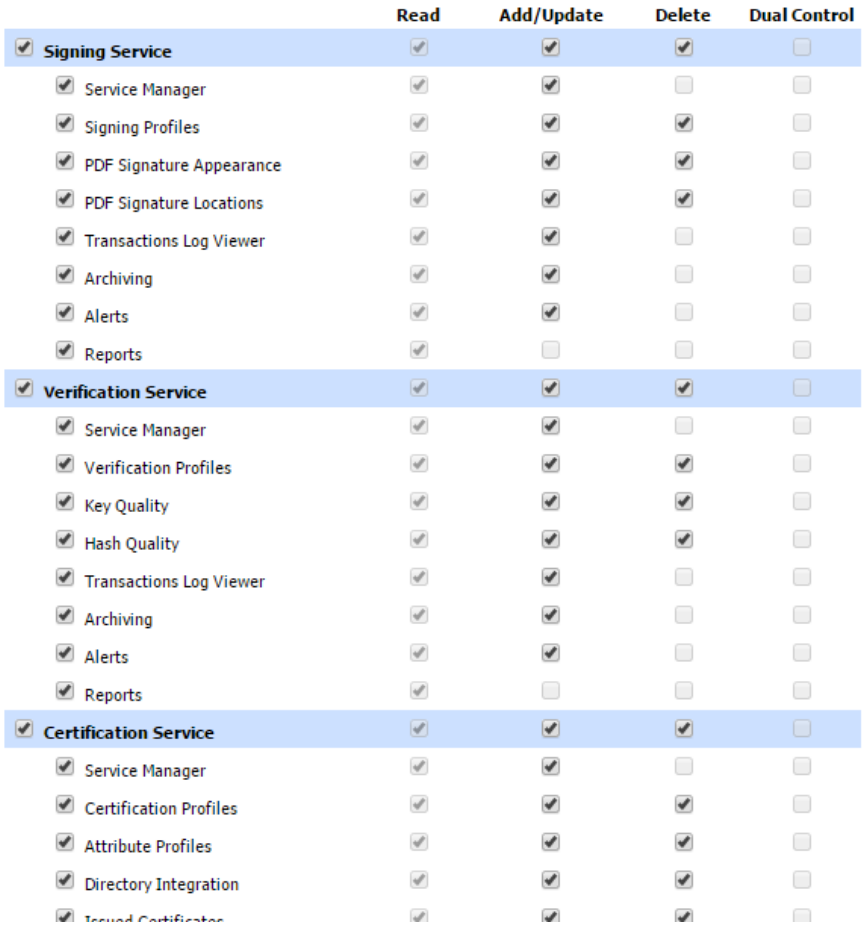


Figure 5 - ADSS Server Role Based Access Control Example

3.11 Notifications & Alerts

ADSS Server provides notification services via SMTP, SMS and SNMP. Before proceeding with the installation determine which services will be used and which administrators will receive any notifications. Note each ADSS Server Service, e.g. Signing allows configuration of notification alerts specific to the service, and which administrators should receive these.

Refer to the service provider for the configuration parameters required to access the respective services. Ascertia recommends that SMTP notification is a minimum for such, and both SMTP and SMS for high value deployments where uptime and security is key.

3.12 Hardware Security Module

For production systems Ascertia recommends the use of a Hardware Security Module. If applicable ensure the HSM and associated client software are in place. In addition, the necessary configuration has been conducted. For example, when using Thales SafeNet Luna SAs that the partition has been created and the passphrase known. In addition, gathered the required individuals if using trusted authentication mode, i.e. PED with PED Keys to protect access.

3.13 ADSS Server Profiles

Each ADSS Server Service uses a concept of profiles to distinguish configuration sets from one another. For testing the installation procedure allows the creation of sample data that will populate the licensed services with sample profiles. Ascertia recommends this option is not selected for production systems.

The appropriate profiles should be thought about prior to installation. For example, when using Signing Service what profiles are required. That is, what signature format is required, is local hashing being employed, and so forth.

3.14 PKI Based Deployments

Where the Certification Service, RA, OCSP, CRL Monitor, SCVP and associated support modules are used it is important to ensure that applicable CP/CPS and supporting data is in place prior to the installation. The documentation set will affect how the final services within ADSS Server are configured. For example, the revocation status checks of certificates within the CPS will state whether OCSP and CRL are acceptable, or if only one is allowed.

3.15 Checking Resource Limits

The server machine on which the ADSS Server is going to be deployed must be properly planned to handle concurrency needs. The recommended limit for open file descriptor in Linux OS is:

- Soft limit of at least = **4096**
- Hard limit of at least = **65536**

This need to be planned and set accordingly because for each request received on the server, it will open a file connection. You may face error "**Too many open files"** if the open file limit exhausted due to heavy workload of concurrent requests. In order to handle this, increase the open file limit after consulting with your system administrator e.g.

- Soft limit of at least = **32768**
- Hard limit of at least = **262144**

3.16 Installation of Rng-tools on Linux System

Before proceeding with the installation of the ADSS Server on a Linux-based system, it's imperative to ensure that rng-tools are installed. The rng-tools help enhance the security and performance of cryptographic operations by providing a reliable source of entropy. To guarantee a seamless installation and optimal functionality of the ADSS Server, make sure rng-tools is installed on your Linux system.

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 installer must be unzipped to a suitable directory (later referred as **[ADSS-Server-Home]**). ADSS Server 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.

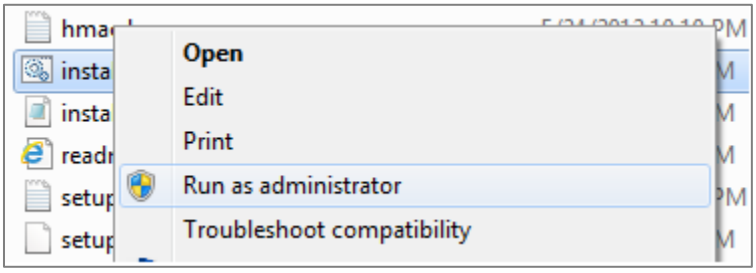


Figure 6 - Windows Example Installer Run as administrator

UNIX

To install ADSS Server on UNIX systems the installer must be launched under **root** user privileges (otherwise ADSS Server daemons will not be registered in /etc/systemd/system). After completing the installation, search the Ascertia Community Portal knowledge base for the technical note titled ***'How to run ADSS Server services as a non-root user daemon on UNIX/Linux.'*** This article provides step-by-step guidance on how to update the owner and group settings. 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.

*In case of **Ubuntu OS**, we will use the following commands:*

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

```
$ bash install.sh
```

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

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

*When upgrading an earlier version of ADSS Server, it is important to start the ADSS Server Installation wizard from the new installation directory. This **must be** different from the current installation directory for the previous release.*

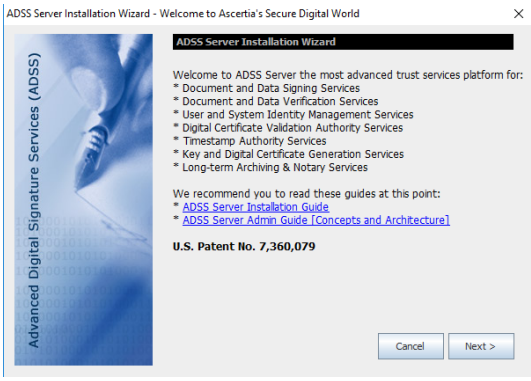
*Note current ADSS Server instances **must be** stopped before starting the upgrade process.*

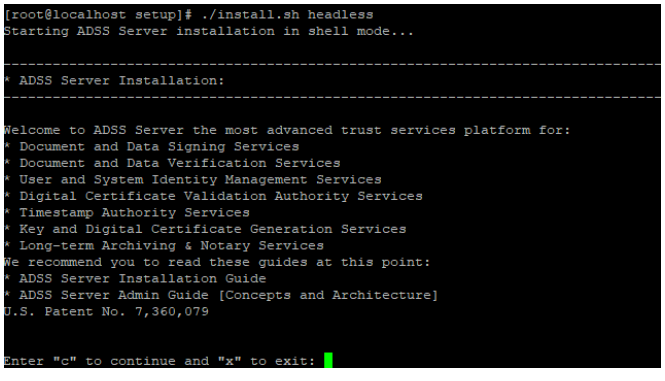
ADSS Server is not installed as a single Windows NT service or a Unix daemon. A standard installation of ADSS Server is comprised of three components:

ADSS Core, ADSS Console and ADSS Service.

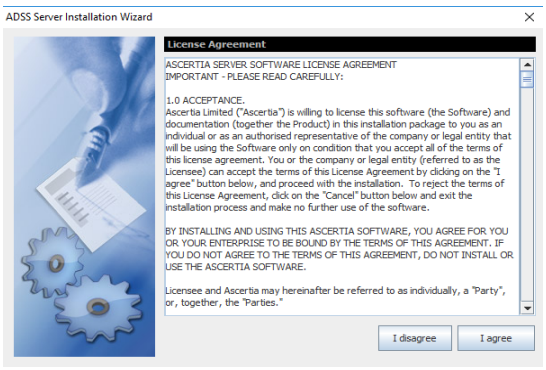
Each of these components uses a separate JVM. For a standard installation of ADSS Server all three components is installed on one machine. For a custom installation, it is possible to install the components on separate machines. It is possible to install multiple ADSS Core, and ADSS Console instances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

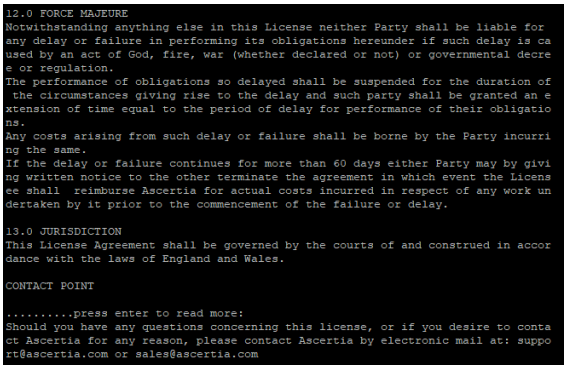
Running **install.bat/sh** shows the following screen:





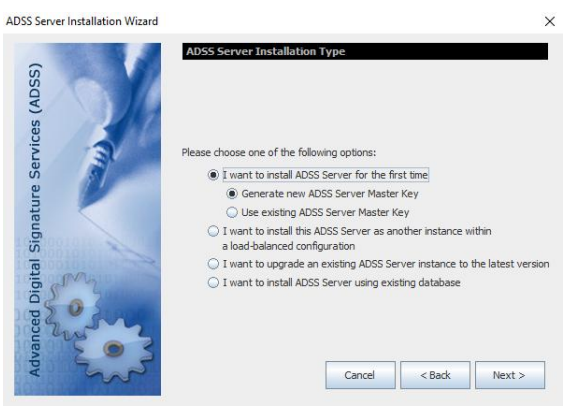
Clicking **Next >** shows the following screen:

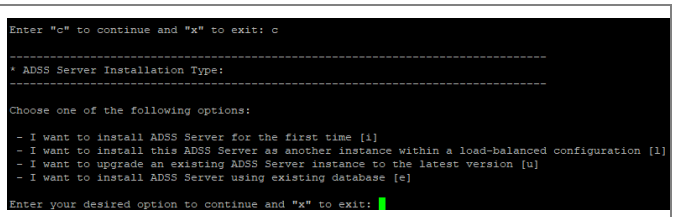




If you agree with the displayed terms and conditions, then click "**I agree**" to continue the installation process otherwise click "**I disagree"** to stop the installation process.

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





There are various installation options available in ADSS Server. These are:

- **I want to install ADSS Server for the first time** – Use this option if you want to install the latest ADSS Server using a fresh/empty database. Following options are available for this for this type of installation.
  - **Generate new ADSS Server Master Key -** This option is recommended when you are installing ADSS Server for the first time either for evaluation or production use. This option assumes that a fresh database is already created, and proper database access rights are assigned to the database user. Click here for more detail.
  - **Use Existing ADSS Server Master Key-** This option is recommended when already installed ADSS Server Master Key and configuration to be used for a new replicated ADSS Server instance. This option assumes that a fresh database is already created, proper database access rights are assigned to the database user and all the configurations of the existing ADSS Server are already exported. Click here for more detail.
- **I want to install this ADSS Server as another instance within a load-balanced configuration** – Use this option to add another ADSS Server instance to an existing ADSS Server installation. You can install all components of ADSS Server (Core, Console and/or Service) on multiple machines to better service the incoming requests. This option can also be used to achieve the high availability (fall-back mechanism) if the main instance stops responding. High availability is supported for ADSS Server Core and Console instances only. ADSS Server Service instance can always be installed in a load-balanced mode where a load-balancer (software or hardware based) manages the incoming

requests and if any of the instances fail the load balancer intelligently shifts the load to the other Service instances. Click here for more detail.

- **I want to upgrade an existing ADSS Server instance to the latest version** – Use this option if you already have an older ADSS Server version installed and want to upgrade it to the latest version. ADSS Server provides an automated way to upgrade both the application and the database from the previous versions (v3.0 and above) to the latest version without requiring any manual steps to be performed by administrators. Click here for more detail.
- **I want to install ADSS Server using existing database** – Use this option if you want to install the ADSS Server using an existing database. Click here for more detail.

These four options are further explained in the following sections:

4.1.1 Installing ADSS Server for the First Time

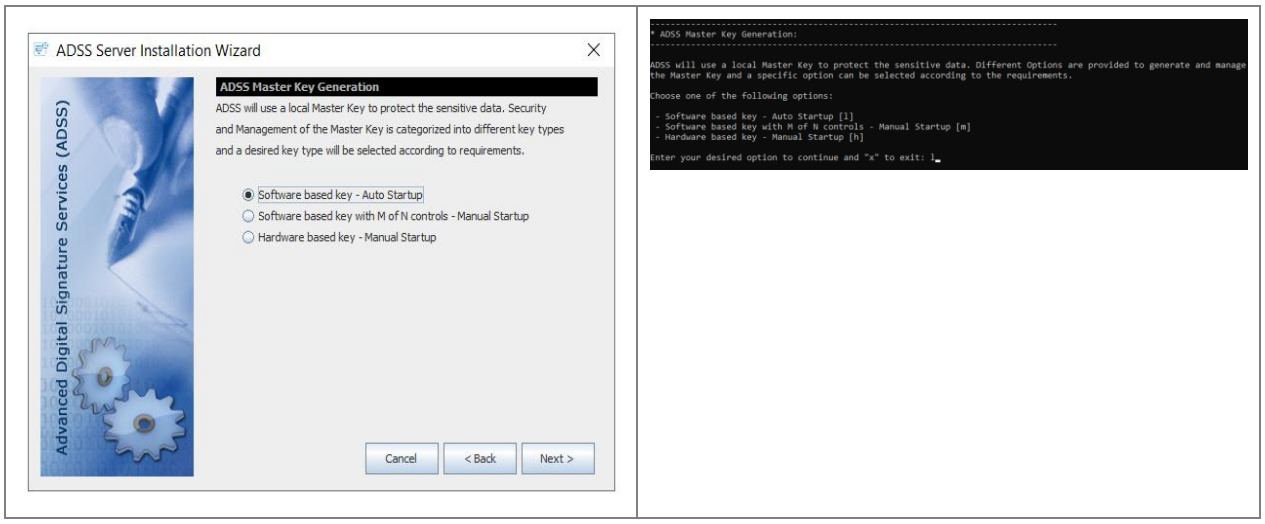
Once the required installation option is selected, it will lead us to the screen to select a specific master key type. ADSS Server uses dynamic master key to ensure protection and security of data. To generate a dynamic master key, ADSS Server provides multiple mechanisms to its users that are categorized 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

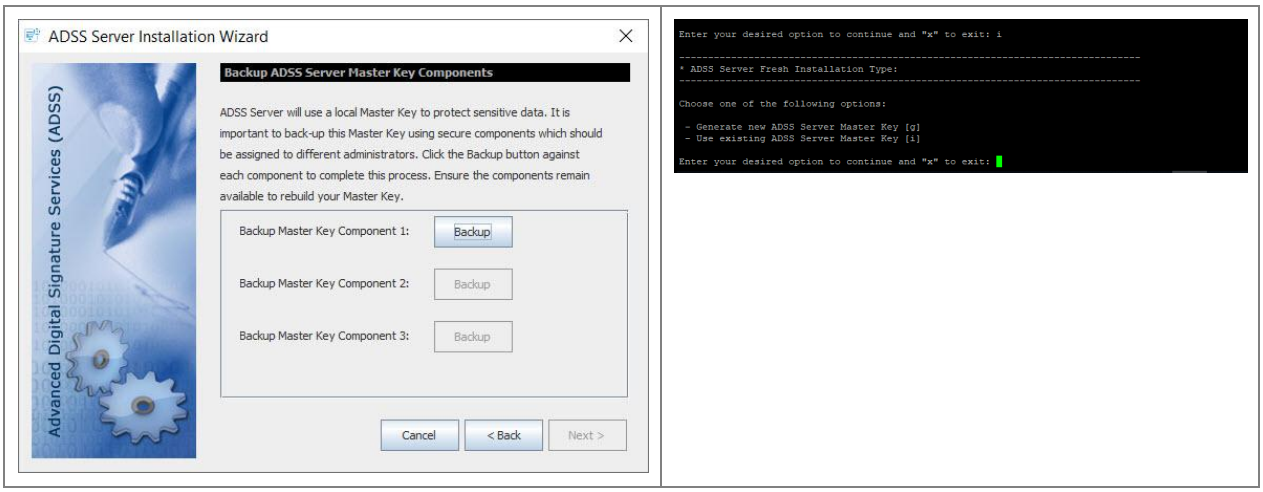
The desired key type will be selected according to the requirement. The details of each key type is explained below:

Software Based Key – Auto Startup

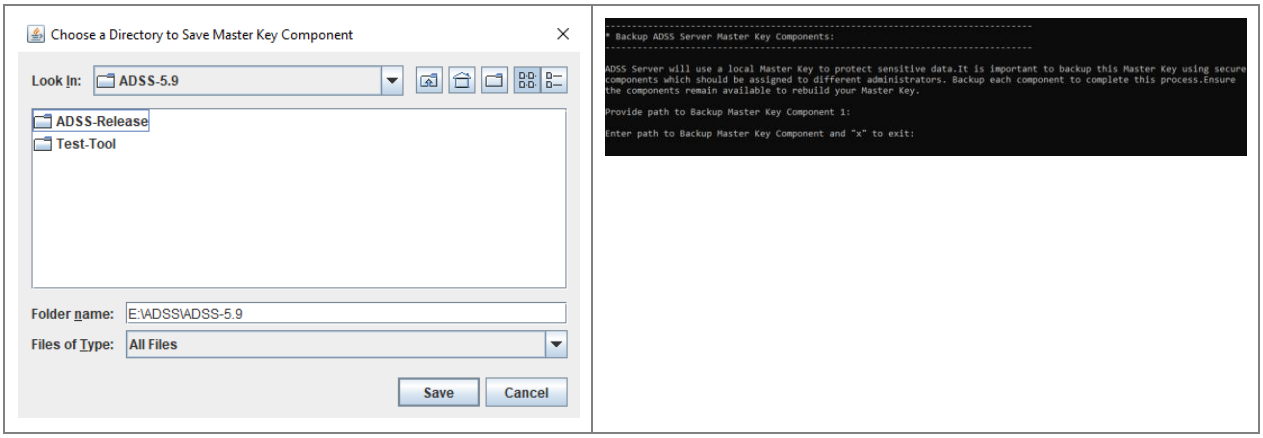
In this scheme, master key is generated using a software crypto source and protected by ADSS Server. Master key can be renewed after regular intervals in order to ensure security. Here, the master key will be protected by ADSS Server itself hence ADSS will be started without any operator's intervention.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

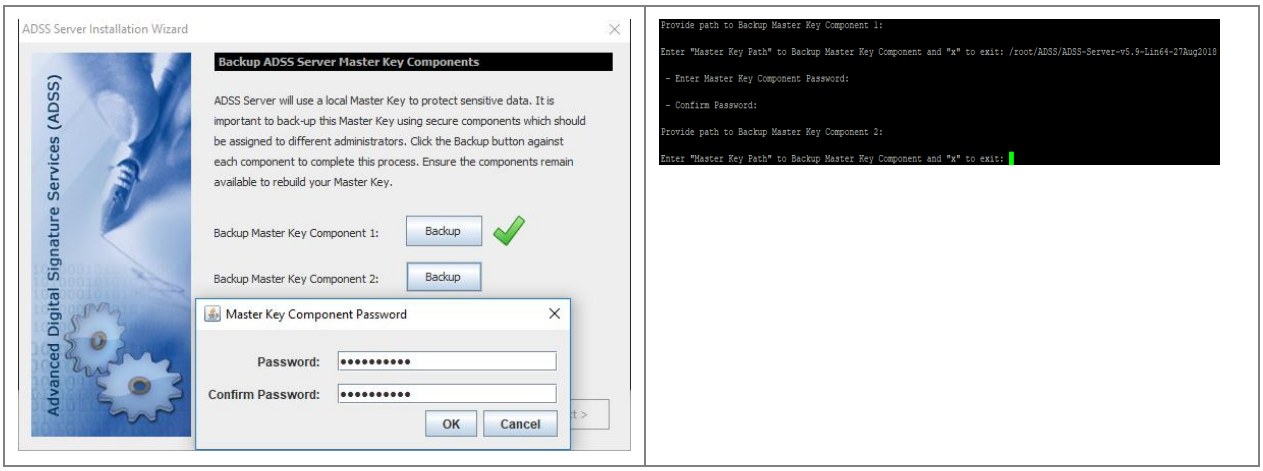


The installer will generate a Master Key and prompt to take a backup of the Master Key in the form of three components:

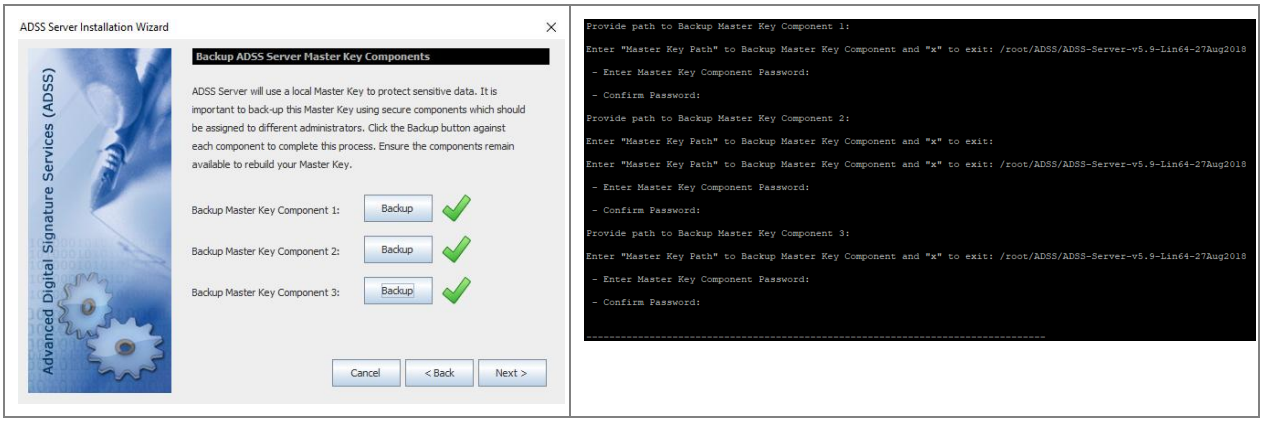


Use the **Backup** button one by one to take the backup of each Master key component, installer will prompt to provide a password for each Master Key component and encrypt it with the provided password before saving on the storage media:





It's recommended to use different password for each Master key component. After completed the Backup the following screen appeared:



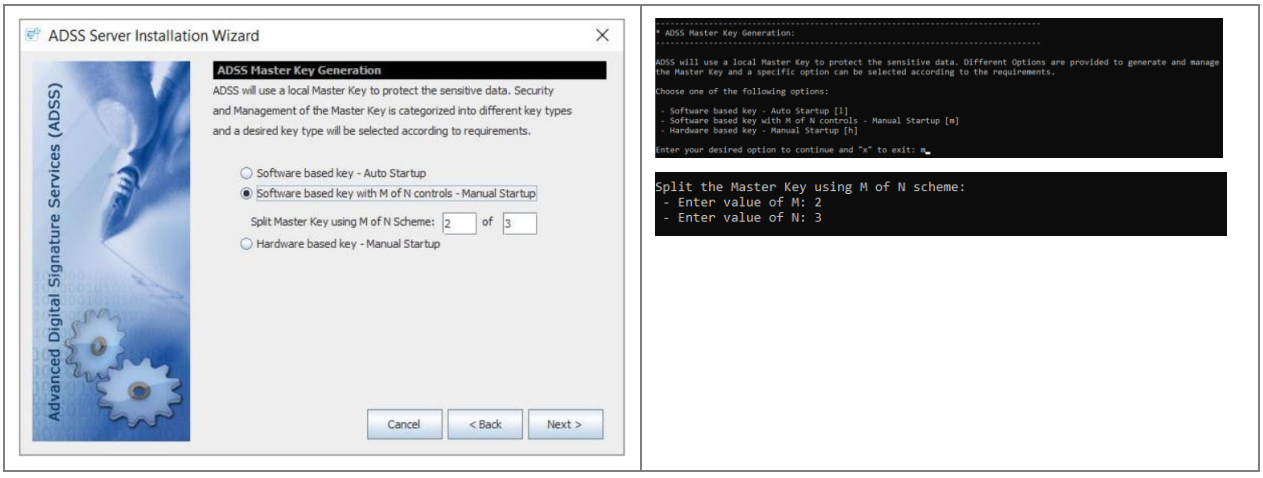
ADSS Server enforces a password strength policy for the generation/backup of Master Key components. Please follow the below guidelines. The password:

1. *Must be at least ten characters long. the installer in GUI mode:*
2. *Must contain the upper and lower case letters.*
3. *Must have digits and special characters.*

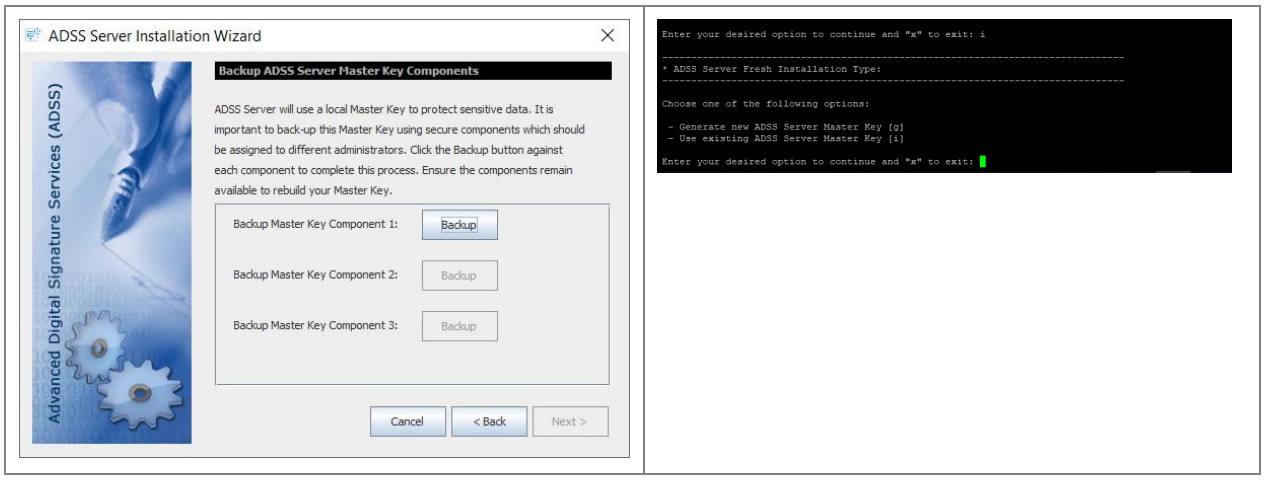
1. *It is recommended that the master key components are stored on a different media (e.g. USB drive) and they must not be stored within the ADSS Server installation folder.*
2. *Make multiple copies of these components to recover the installation in case of disaster and/or media corruption.*

Software Based Key with M of N Controls – Manual Startup

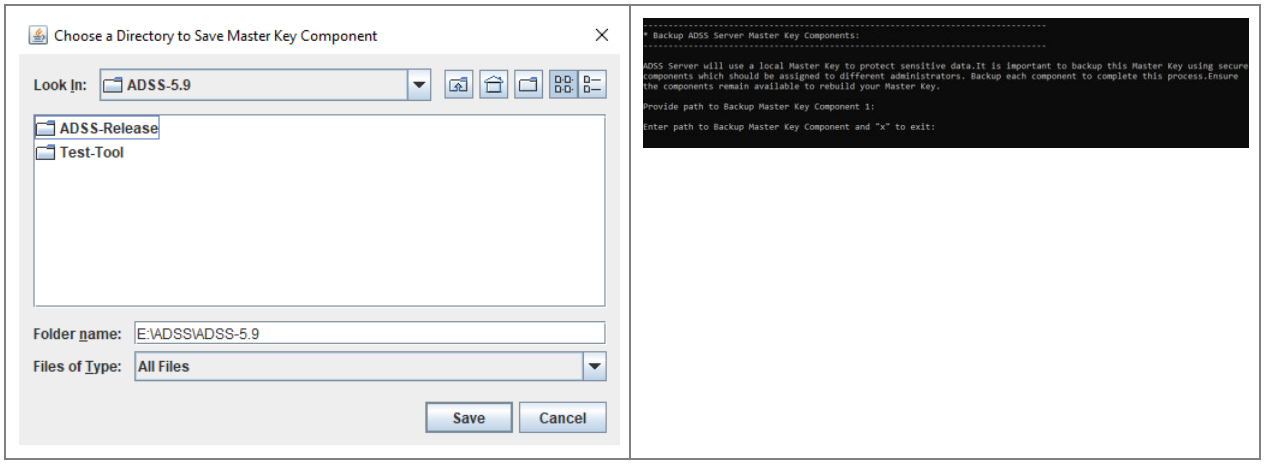
In this mode, the Master Key is generated using a software crypto provider and split according to M of N rule. The minimum value of M will be 2 and N will be 3. The maximum value of M of N will be 16. Below screen will be displayed during installation process:

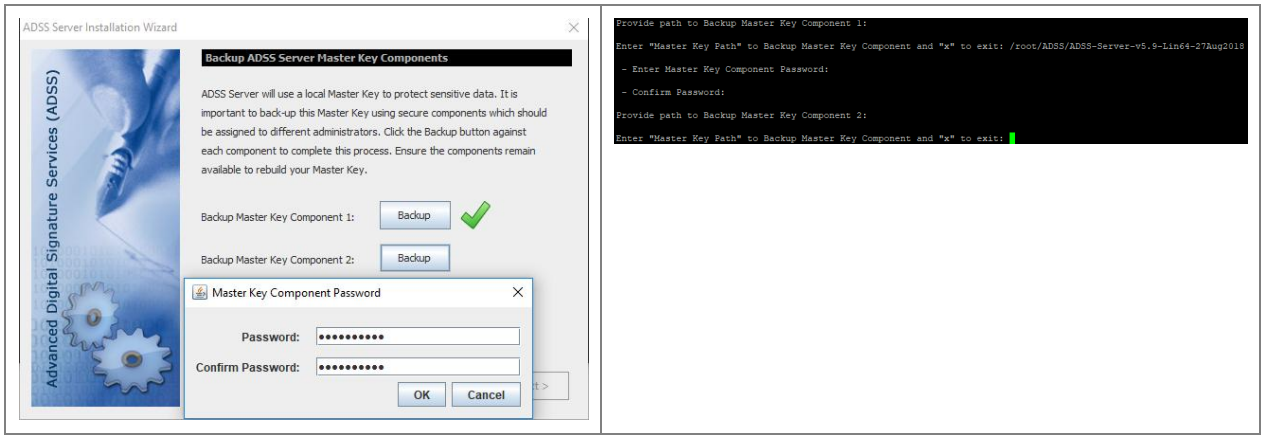


After providing M of N scheme, click on the Next button to proceed. A master key will be generated and split according to the selected M of N scheme. Since the key will be split into N parts, so on next screen, N number of backups will be taken.

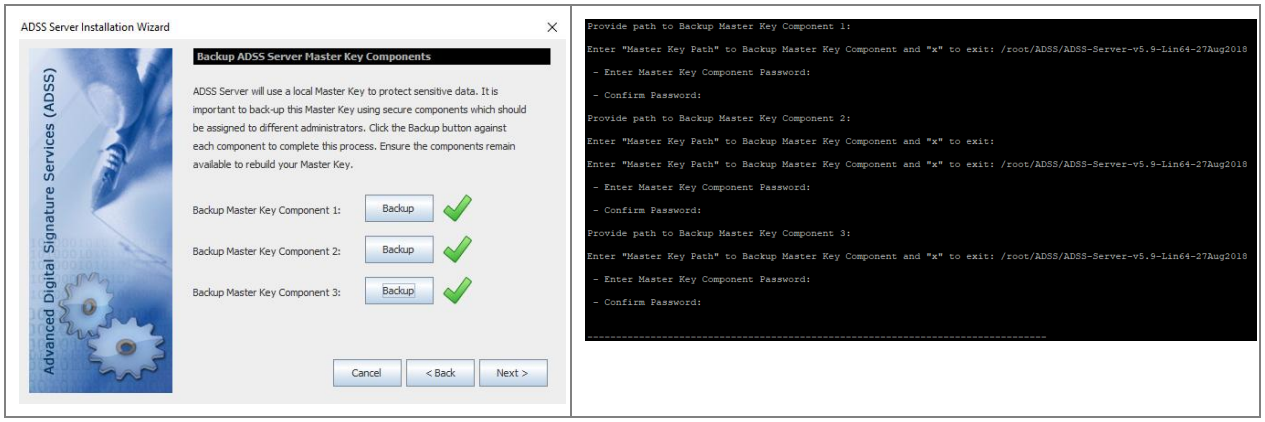


Each backup is protected by a password so a key is generated using the provided password and backup component is encrypted with this key.





It's recommended to use different password for each Master key component. After completing the Backup the following screen will display:



In this scheme, ADSS Server does not store the master key anywhere in its storage and the backup components are required at each startup of the ADSS. This is the reason this scheme requires manual startup where M number of key components will be provided to re-compose the master key and start the ADSS Server.

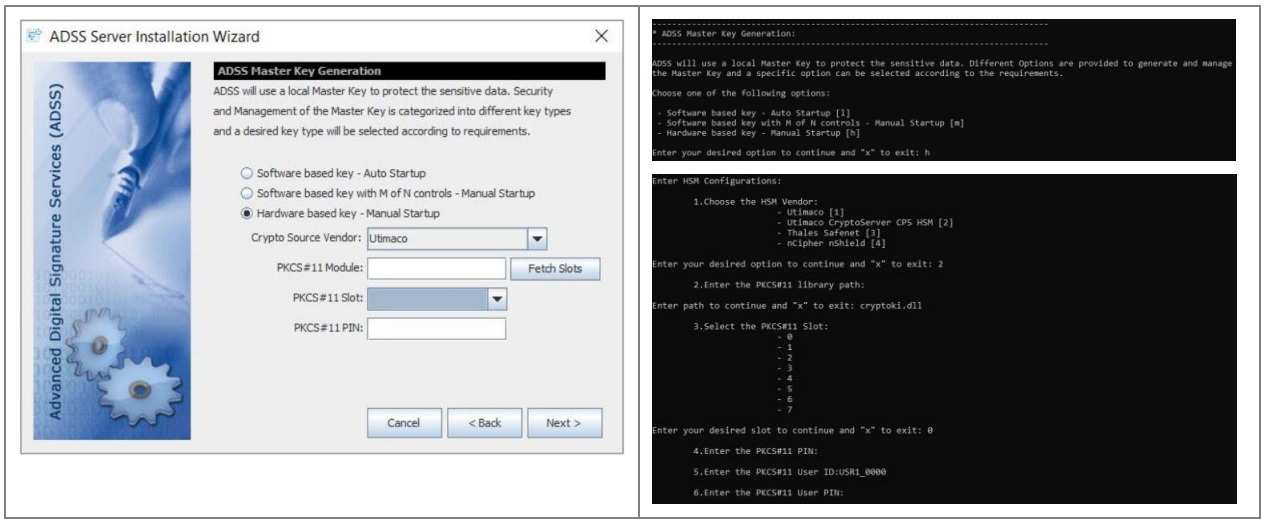
ADSS Server enforces a password strength policy for the generation/backup of Master Key components. Please follow the below guidelines. The password:

1. *Must be at least ten characters long. the installer in GUI mode:*
2. *Must contain the upper and lower case letters.*
3. *Must have digits and special characters.*

1. *It is recommended that the master key components are stored on a different media (e.g. USB drive) and they must not be stored within the ADSS Server installation folder.*
2. *If you lose these components and/or passwords:*
   a. *You cannot install the ADSS Server in a load-balanced setup.*
   b. *You cannot recover the ADSS Server installation in case of disaster.*
3. *Make multiple copies of these components to recover the installation in case of disaster and/or media corruption.*

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:



Here, the operator will define the:

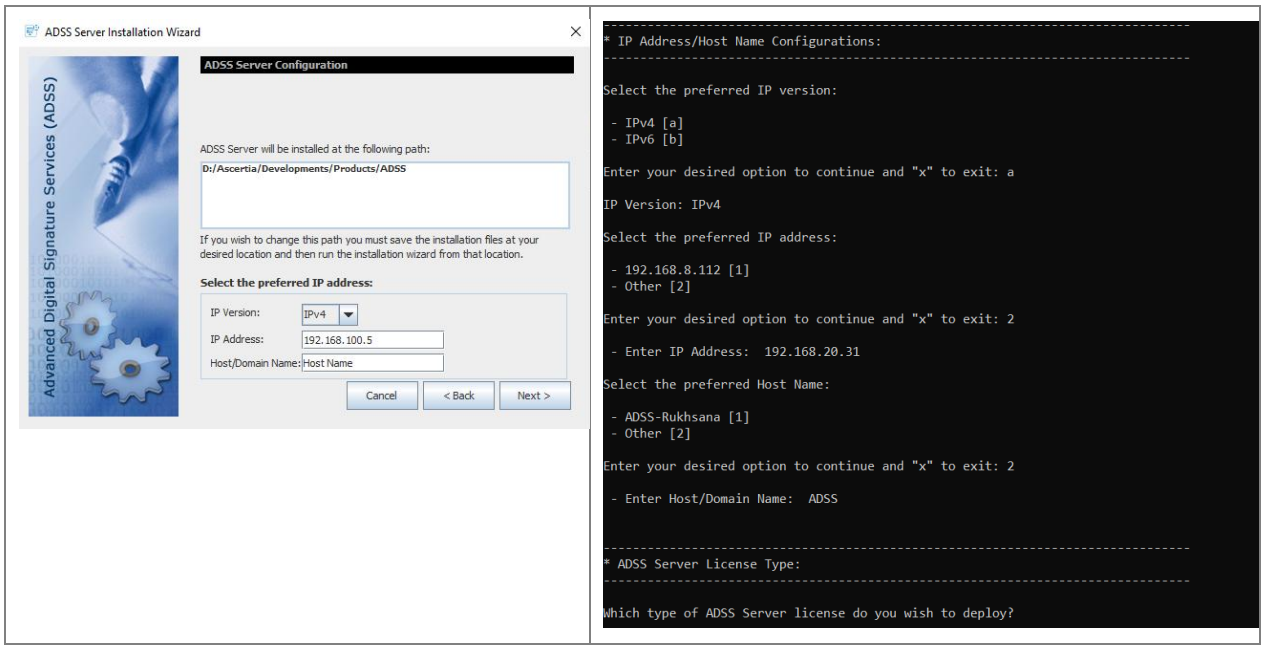
- Crypto Source Vendor form the list of available crypto source vendors in the drop-down field.
- PKCS#11 module
- PKCS#11 Slot by clicking on the Fetch Slots button
- PKCS#11 PIN
- User ID (available in case of Utimaco CryptoServer CP5 HSM)
- User PIN (available in case of Utimaco CryptoServer CP5 HSM)

In this scheme, the master key is inside the HSM, so during ADSS services startup, an operator will be required to provide the HSM PIN. So this option requires manual startup of ADSS services and should be chosen keeping in mind these requirements.

As Mater Key is being created inside an HSM, therefore no backup is required and backup screen will not be displayed in this case.

If you are using Utimaco CryptoServer CP5 HSM to generate Master Key then we use HSM PIN to generate a key to encrypt some information. If HSM PIN is changed and a new HSM PIN is provided on the start-up of ADSS Server, then startup will not be able to complete as the information encrypted with the old key cannot be decrypted. In such scenarios, we have a utility that takes old PIN, generate a key with it and decrypt the information. It then asks for the new PIN, generates a key using it and encrypts the information with new key. Now there will be no issues during startup if operator provides the new PIN. This utility can be found at [ADSS-Server-Installation-Directory]/util/bin/update_private_kak directory.

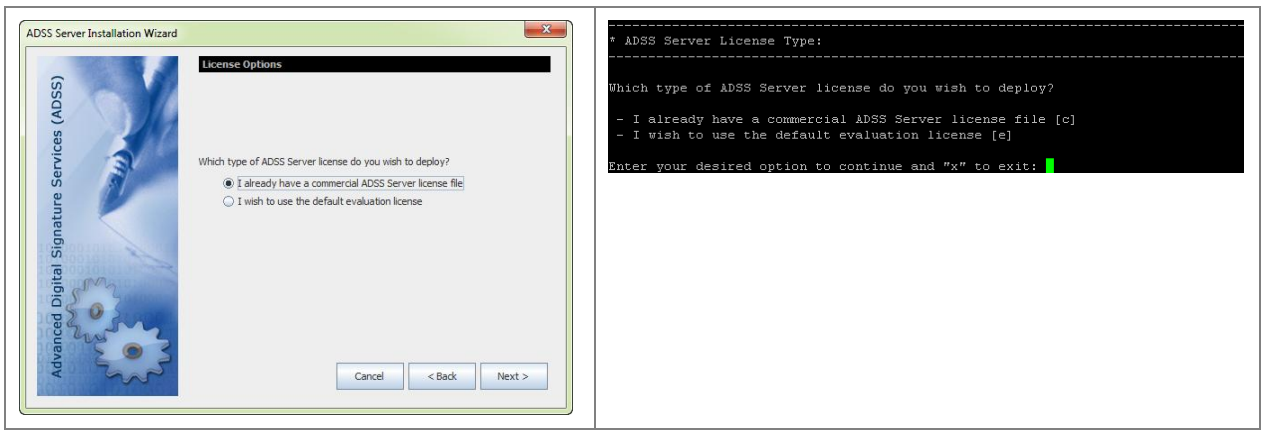
Once done, click **Next**. This screen shows the path where ADSS Server software will be installed and allows to select the IP scheme for the ADSS Server installation. The user can select the required IP Version (IPv4 or IPV6) from the drop-down and enter the required IP Address and Host/Domain Name in the given fields. Similarly in headless mode, the user can either select the system generated preferred IP Address or enter its own required IP Address and Domain Name by selecting the **Other** option.



Both IPv4 and IPv6 network addresses are supported in ADSS Server. You can select the type of network address (IPv4/IPv6) and provide the relevant address.

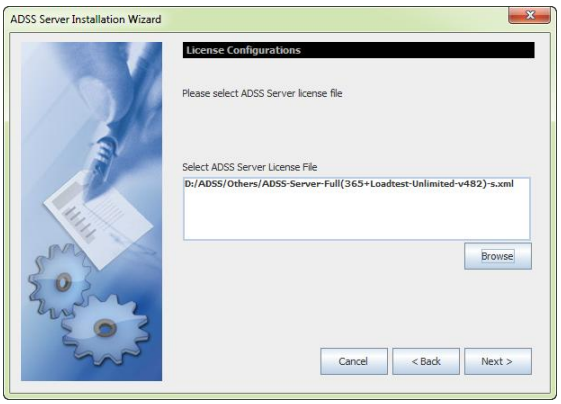
When Deploying ADSS Server to Windows Azure, ADSS Server installer will only look for the non-public IP address here, it will not look for the public address - it is not important here.

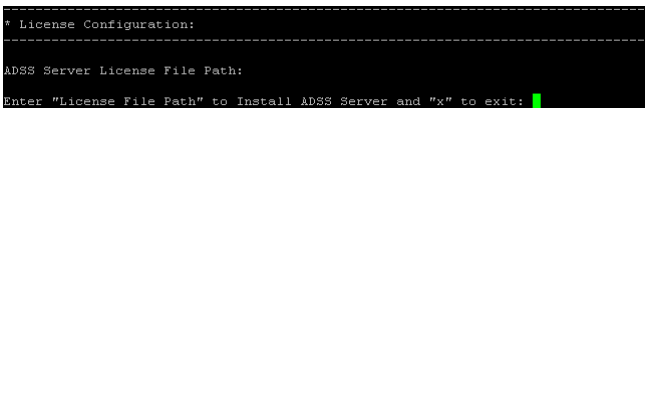
Clicking "**Next**" shows the License Options screen:



Select one of the license options on this screen: -

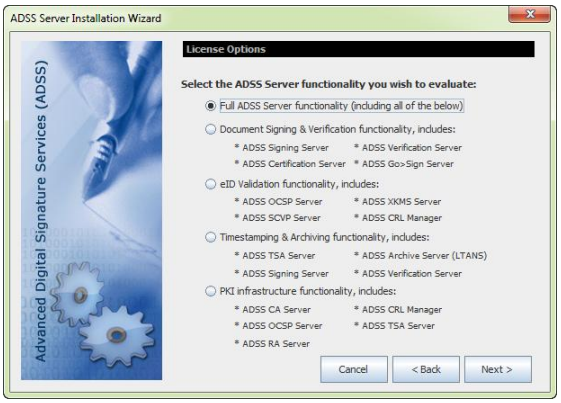
- If you wish to evaluate ADSS Server, then select option "**I wish to use the default evaluation license**". The evaluation license limit is one month from the date of first installation.
- If you have a commercial ADSS Server license, then select the option "**I already have a commercial ADSS Server license file**" and you will be prompted to browse it on the next screen:

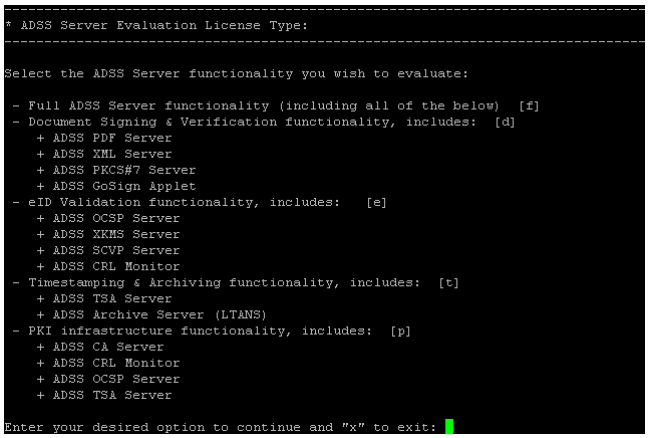




Use the **Browse** button to choose the commercial license file supplied by your software provider.

If you selected, the option "**I wish to use default evaluation license**" the following screen is shown:





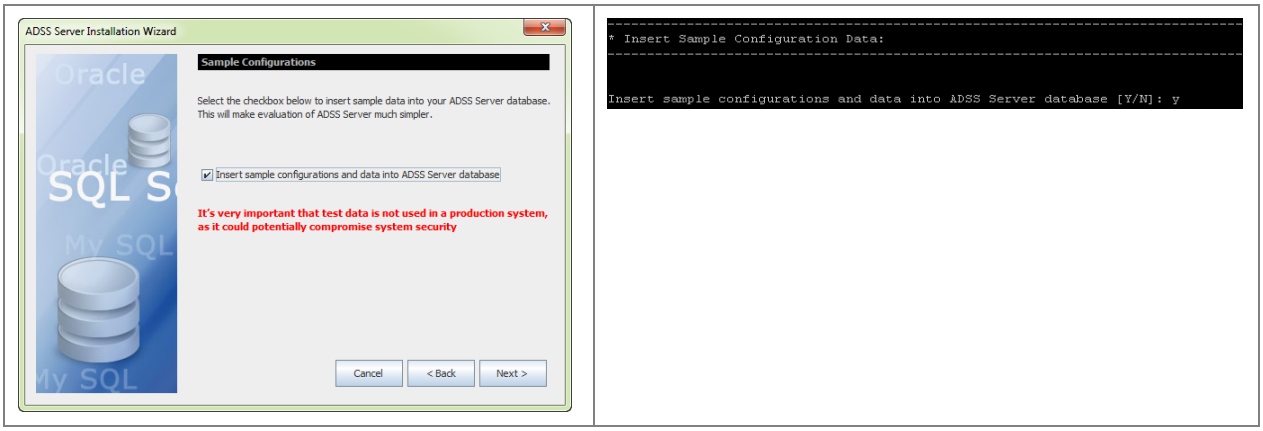
Select one of the options that best suits your business needs. When you select a license option from the list above, service features that are not licensed will not be shown. Custom evaluation licenses can be provided easily by your sales contact. Examples include a license just for PDF Signing, ADSS OCSP Server functionality or long-term archiving.

The default evaluation licenses supplied with ADSS Server allows reasonable usage within pre-set time and transaction limits.

An ADSS Server can be upgraded to use a full commercial license after evaluation just by overwriting the license file with a replacement.

New licenses take effect on the next full restart of all ADSS Server instances.

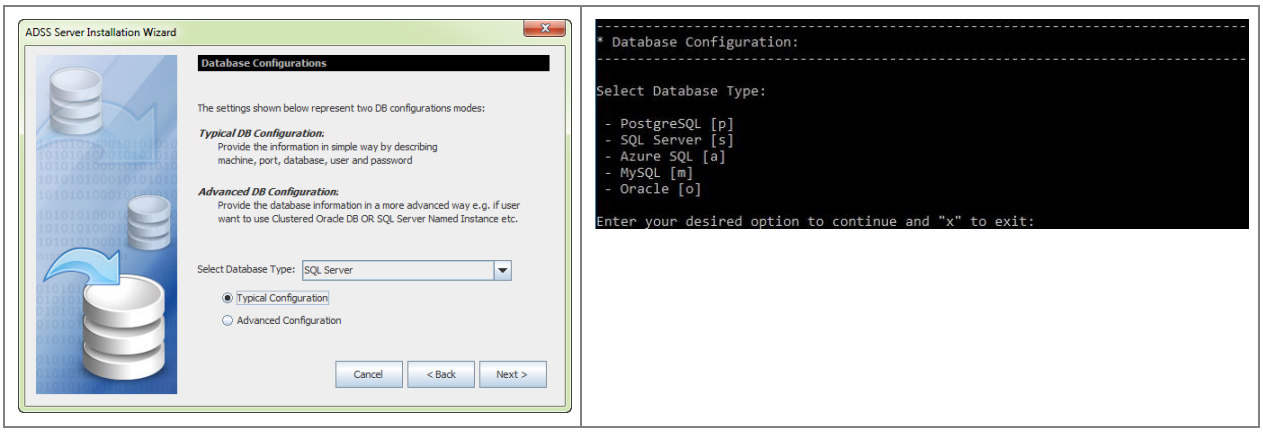
Click "**Next**" to proceed.



Selecting this option will insert the sample data in to the data to allow immediate testing/evaluating of ADSS Server.

This option is not recommended if you are installing ADSS Server in a production environment.

4.1.1.1 Database Configuration

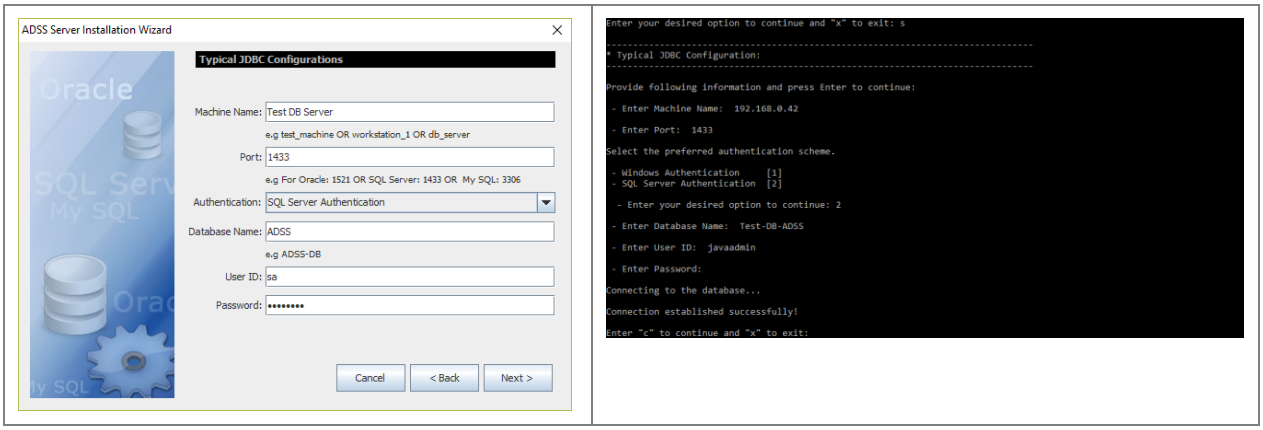


Use this screen to select database type (e.g. SQL Server) and configuration type (Typical/Advanced)

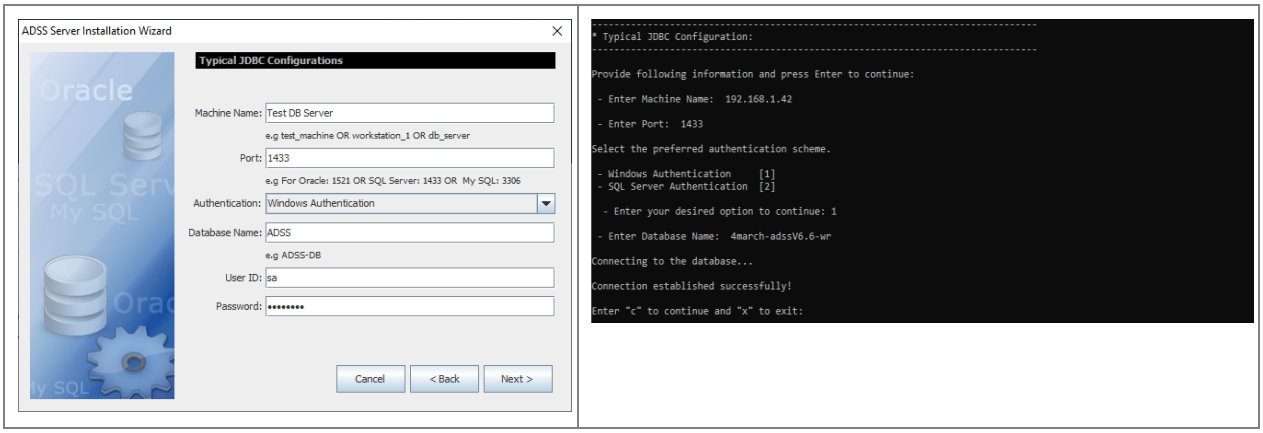
Typical Configuration allows you to specify the database server name, database name, port etc. The Advanced Configuration also allows configuration of the low-level database driver URL, JARs etc. Unless you are experienced in this area, the typical configuration is recommended.

4.1.1.2 Database Connection Parameters

Getting back to the ADSS Server installation, selecting "**Typical Configurations**" on the Database Configurations screen and clicking **Next** shows the following screen:



In case of Windows Authentication, the following screen is shown:



The configuration items are as follows:

Item	Description
Machine Name	The system name or IP address of the machine where the database server is running, e.g. asc_db_server. If you are installing ADSS Server on the same machine as the database, please enter “localhost” as the Machine Name.
Port	The port number to be used to connect to the database e.g. 1433 for SQL Server.
Authentication Scheme	In case of ADSS Server installation with SQL Server as Database, user can be authenticated by two ways i.e.: • SQL Server Authentication • Windows Authentication For SQL Server Authentication, user needs to enter the User Name and Password of SQL Server. Whereas in Windows Authentication, these fields will be disabled, and user will be authenticated by the logged-in user Windows/Domain credentials. **Note:** Under typical JDBC configurations only Kerberos authentication is supported. For NTLM based authentication use the advanced JDBC configurations.
Database Name	The name of database for ADSS Server. This can be a newly created empty database or an existing database. Make sure the database exists before clicking the Next button.
User ID	The user ID used by ADSS Server to connect to the database. Ensure that this user exists and has the appropriate privileges to create and access tables.

Item	Description
Password	The corresponding password for the User ID.

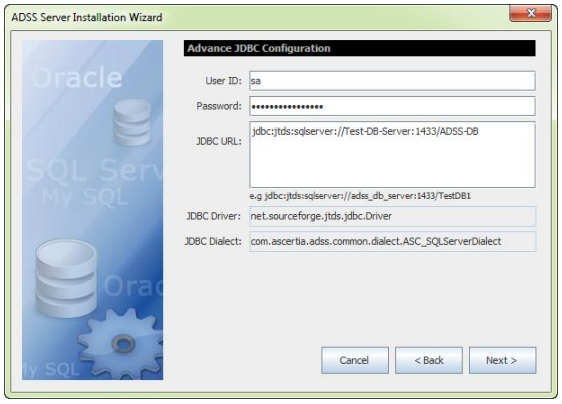
Table 2 - Database Connection Parameters - Typical

Clicking "**Next**" checks the database credentials and connectivity and moves to the next screen only if successfully validated.

Read the pre-installation checks for the special configurations required for SQL Server, Postgres and MySQL database before proceeding Installation.

4.1.1.3 Using Advanced Configurations

If you select **Advanced Configurations** in the Database Configurations screen, the following screen is shown:



Advanced Database Configurations are not supported in Headless mode. To apply such changes, you need to manually update the hibernate.cfg.xml file after the installation is complete.

The configuration items are as follows:

Item	Description
User ID	The user ID used by ADSS Server to connect to the database. Ensure that this user exists and has the appropriate privileges to create and access tables.
Password	The corresponding password for the User ID.
JDBC URL	JDBC URL is a database connection string. This is useful for configuring a connection string manually or for database connection pooling, i.e. the connection string provides details of the individual database server name, port, user ID and password running in a database pooled environment. To Install ADSS Server with SQL Server using Windows Authentication, leave the User ID and Password fields empty and use the following string: • ***Kerberos Authentication*** *jdbc:sqlserver://<DATABASE_MACHINE>;databaseName=<DATABASE_NAME>;integratedSecurity=true;authenticationScheme=JavaKerberos* *E.g.* *jdbc:sqlserver://db-machine;databaseName=adss-db;integratedSecurity=true;authenticationScheme=JavaKerberos*

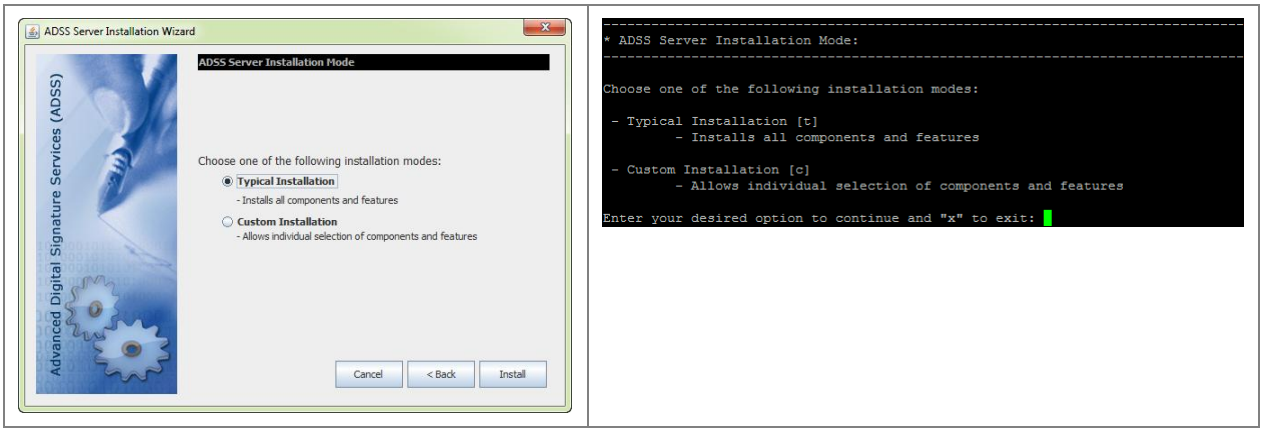
Item	Description
	• ***Windows Authentication*** *jdbc:sqlserver://<DATABASE_MACHINE>:1433;databaseName=<DATABASE_NAME>integratedSecurity=true* *E.g.* *jdbc:sqlserver://db-machine:1433;databaseName=adss-db;integratedSecurity=true* To Install ADSS Server with SQL Server using SQL Server Authentication, use the following string: • ***SQL Server Authentication*** *jdbc:sqlserver://<DATABASE_MACHINE>:1433;databaseName=<DATABASE_NAME>;trustServerCertificate=true* *E.g.* *jdbc:sqlserver://db-machine:1433;databaseName=adss-db;trustServerCertificate=true* To Install ADSS Server using TLS, perform the following steps: 1. Import TLS CA issuer certificate in the ADSS Server JDK using certificate import utility. Run utility from <ADSS-Home>/util/bin/import_cert_into_keystore.bat or import_cert_into_keystore.sh. A graphical interface will display to browse intended certificate. 2. One of the following strings must be used: • ***PostgresSQL*** *jdbc:postgresql://<Server-Name/IP>:5432/<Database_Name>?ssl=true&sslfactory=org.postgresql.ssl.NonValidatingFactory* • ***SQL Server*** *jdbc:jtds:sqlserver://<Server-Name/IP>:1433/<Database_Name>;ssl=require* If SQL named instance is used then use the following string: *jdbc:jtds:sqlserver://<Server-Name/IP>/<Database_Name>;instance=<Instance-Name>* **Note:** Note: The property *trustServerCertificate* must NOT be set to true when *ssl=require* property is set. • ***Azure SQL (database as a service)*** *jdbc:sqlserver://<Server-Name/IP>;database=<Database_Name>;ssl=require* • ***MySQL*** *jdbc:mysql://address=(protocol=tcps)(host=<Server-Name/IP>)(port=3306)/<Database_Name>* • ***Oracle*** *jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=<Server-Name/IP>)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=<SID/Service Name>)))*

Item	Description
JDBC Driver	The name of the driver used to communicate with the database.
JDBC Dialect	As ADSS Server uses Hibernate technology to communicate with the database, it provides the hibernate-package details which is used for communication. Note that Hibernate provides different JDBC dialects for different DBMS. See http://www.hibernate.org/ for more details.

Table 3 - Database Connection Parameters – Advanced

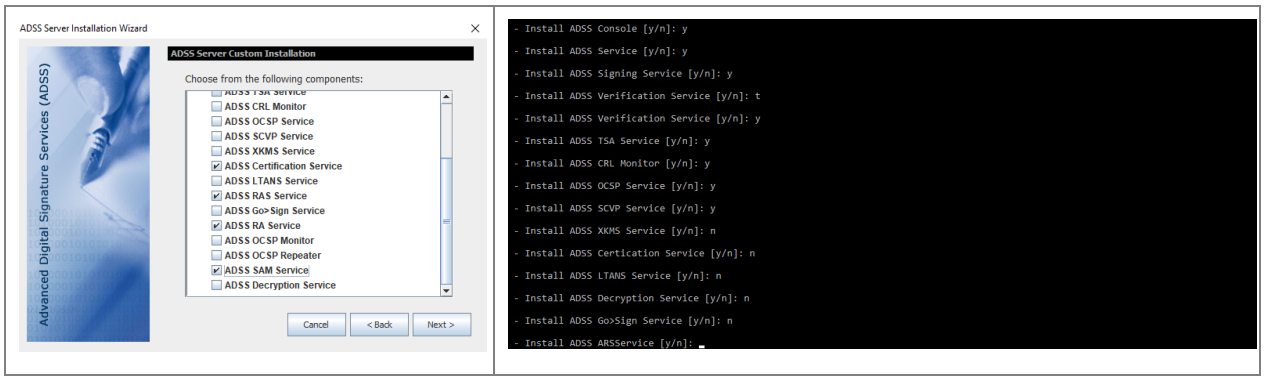
If Windows authentication is used instead of SQL Server authentication, refer to the technical note titled 'How to register all the instances of ADSS Server SigningHub Core under a domain user account' in the Ascertia Community Portal knowledge base for the required post-installation configurations.

Clicking "**Next**" on the Typical or Advanced database configuration screen shows the following screen:



Select the installation mode you wish to use:

- **Typical Installation** – Select this option if you want to install all components of ADSS Server with default memory parameters, i.e. Core (1024-MB), Console (1024-MB) and Service (2048-MB) on the current system.
- **Custom Installation** – Select this option if you want to install selected ADSS Server components and service modules with custom memory parameters. Selecting this option and clicking **Next** shows the following screen:

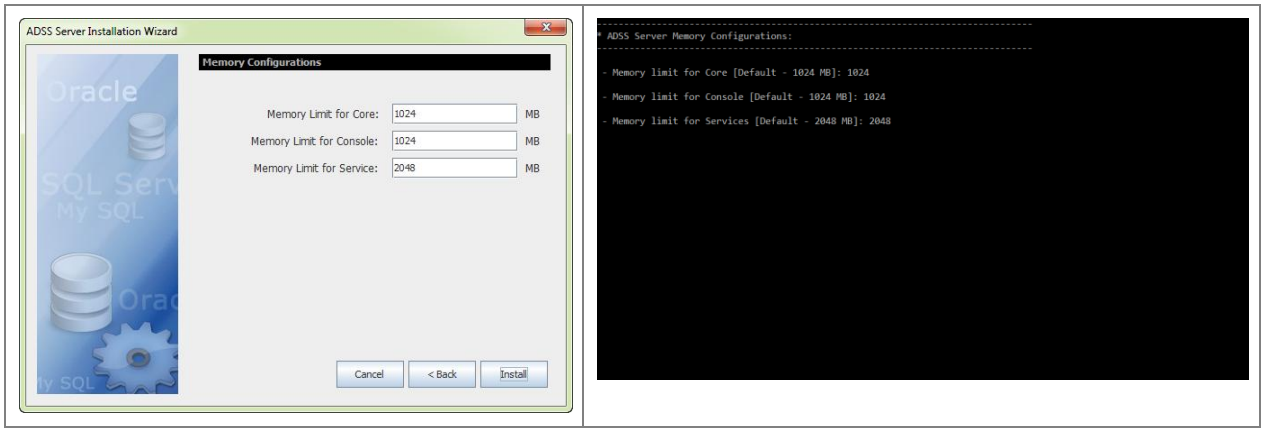


This screen allows you to select ADSS Server components to install on this system. During a fresh installation ADSS Core is selected by default and optionally you can choose to install the ADSS Console and ADSS Service components if required.

When selecting ADSS Service components, at least one component must be selected.

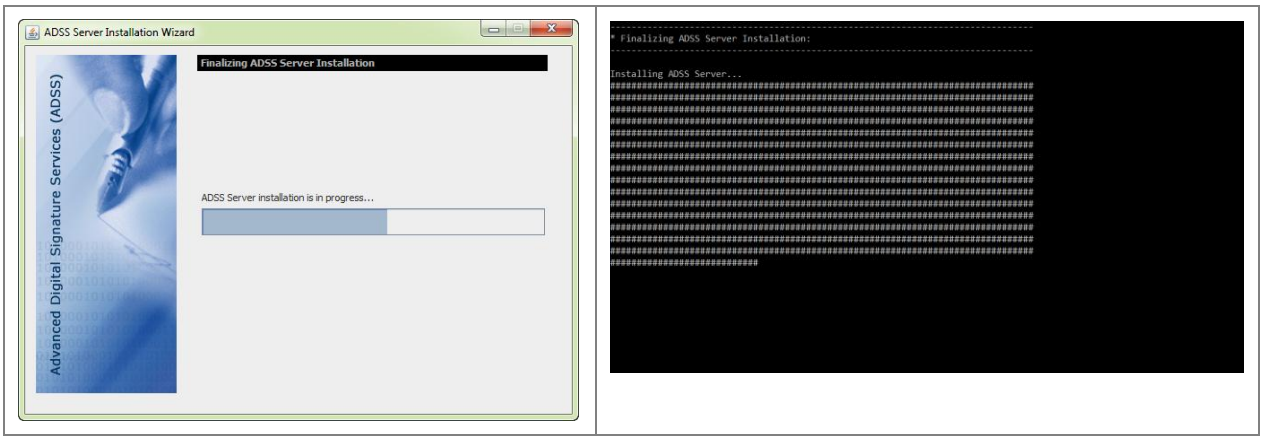
Note the above screenshot shows an installation with a full commercial license for ADSS Server. In your case, only those services are shown that are actually licensed to you.

Select the relevant components to install and click **Next** to show the following screen, which allows you to assign the maximum memory limit for each component.

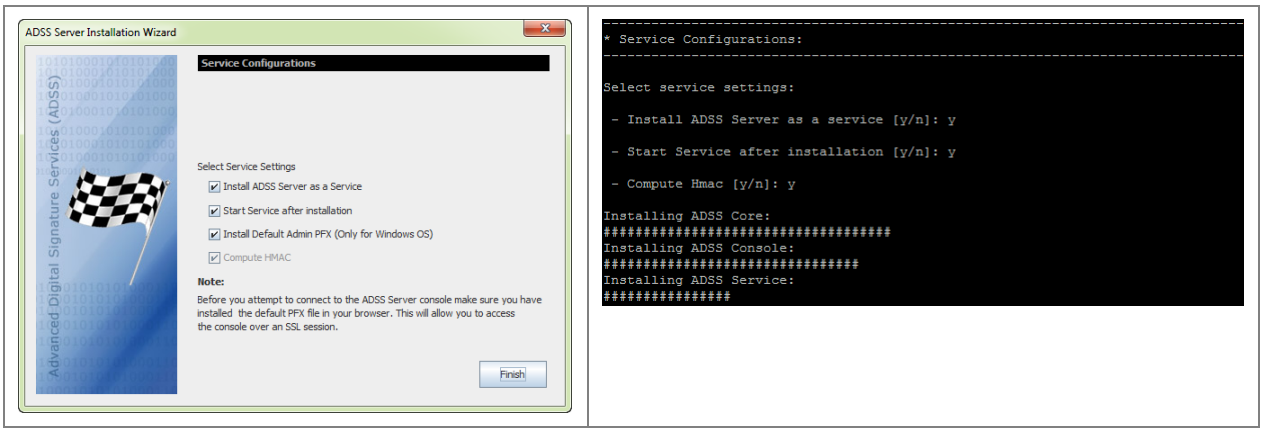


This screen allows you to define the maximum memory limit for each instance of ADSS Server. Default maximum memory limit for Core and Console is 1024 MB and for Service instance 2048 MB.

After providing the maximum memory limit clinking on **Install** starts the installation process including execution of database scripts and updating the configuration files.



Once the installation has completed, the following screen is shown:



- **Install ADSS Server as a service -** (the default recommended option) In Windows environment, the selected ADSS Server components will be registered in Windows Services Panel with the following names:
    - Ascertia-ADSS-Console
    - Ascertia-ADSS-Core
    - Ascertia-ADSS-Service

  On UNIX operating systems, the selected ADSS Server components will be registered in **/etc/systemd/system** with the following names:
    - tomcatd_console_linux.service
    - tomcatd_core_linux.service
    - tomcatd_service_linux.service
- **Start Service after installation -** This option is available only if the previous option is selected. If checked the registered services will be started automatically after installation if master key will be generated using “Software based key – Auto Startup” option. If master key was generated using other two options where manual startup is required, then services will not be started automatically after installation. In order to start the services for this case, refer to the document ‘Quick-Guide-To-Start-ADSS-Services.pdf’ located at **[ADSS Server Installation Directory]/docs/** directory.
- **Install Default Admin PFX (Only for Windows OS) –** Selecting this option will install the default client authentication certificate in MS CAPI keystore, which allows you to login to the ADSS Server

Console using Admin operator (this is only required when ADSS Server is installed very first time).

*The password of the Default Admin PFX is **password***

It is highly recommended to configure new operator from Access Control module and either inactivate or to change the client authentication certificate for default Admin operator with a certificate certified by your PKI.

It could be a security risk if you continue to use the Admin operator with default certificate in production environment.

This option is disabled on UNIX and **adss_default_admin.pfx** must be imported in Firefox or other web browser, manually from: **[ADSS-Server-Home]/setup/certs/** directory in order to login to the ADSS Server Console. Note the web browser does not need to be on the same server as ADSS Server. The administration console is accessed over HTTPS.

- **Compute HMAC –** HMAC is a cryptographic checksum computed by ADSS Server on every record within the ADSS Server database to detect unauthorized changes in the database. It is mandatory to compute HMAC for a fresh installation, so the option is checked and greyed out.

Clicking **Finish**, will complete the ADSS Server installation wizard and a success message is shown:

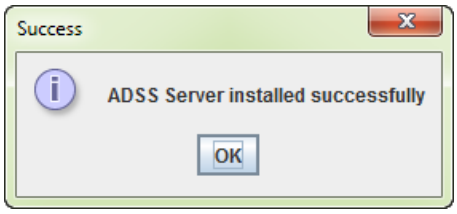
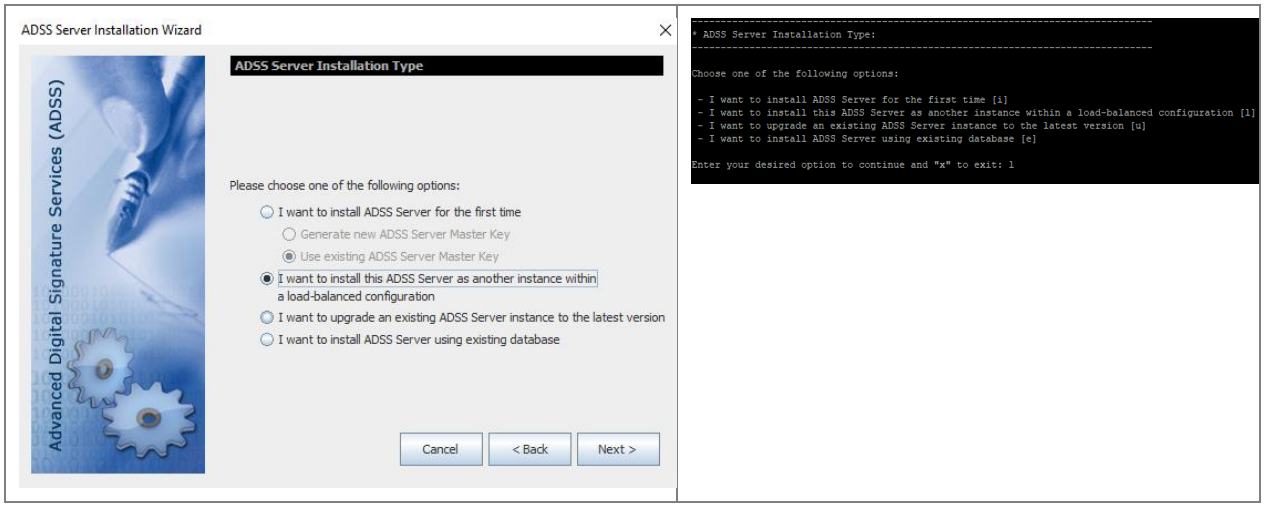


Figure 7 - ADSS Server Installation Wizard Success Screen

Once the installation is completed, refer to the ADSS Server Admin Guide to configure the required services.

4.1.2 Installing ADSS Server in a Load-Balanced Configuration

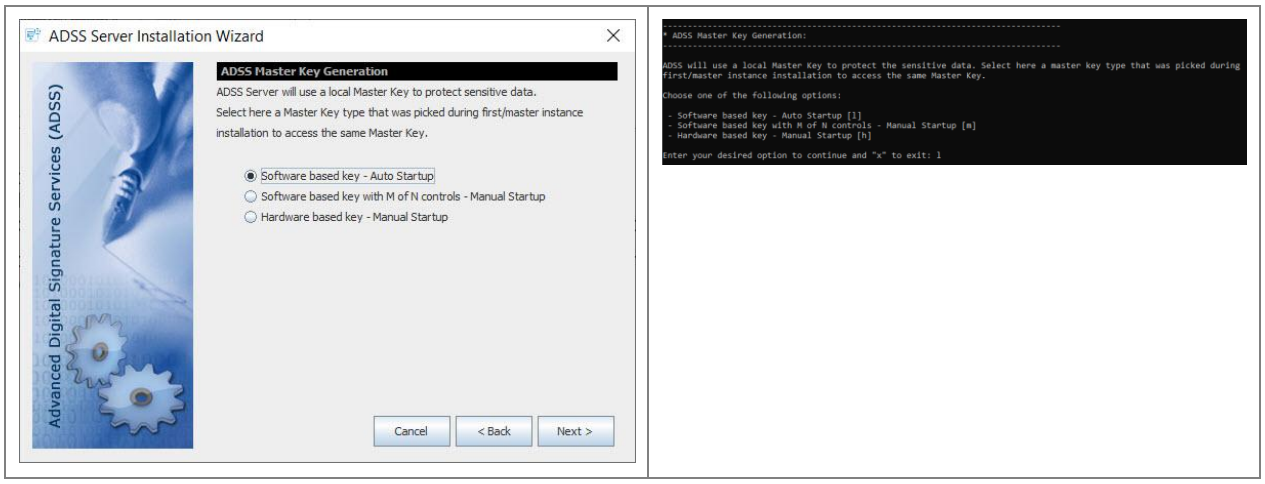
Select the option **I want to install this ADSS Server as another instance within a load-balanced configuration** on the ADSS Server Installation Type screen:



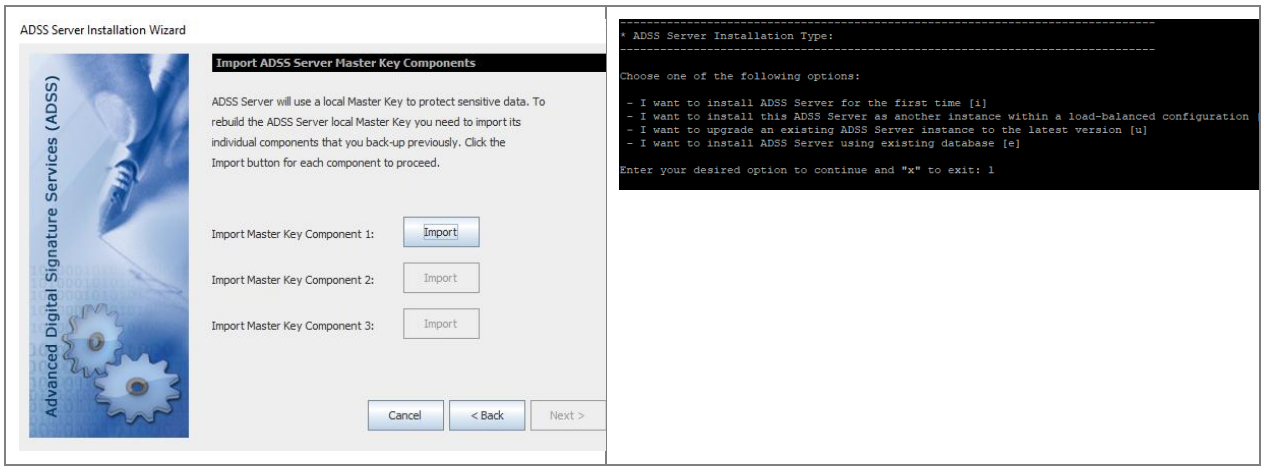
Click on the **Next** button, it will lead you to the Master Key options screen:

Software Based Key – Auto Startup

If you had installed the ADSS using this option, then you will have to select the same option during load balanced installation. The following screen will be displayed to select the master key type:

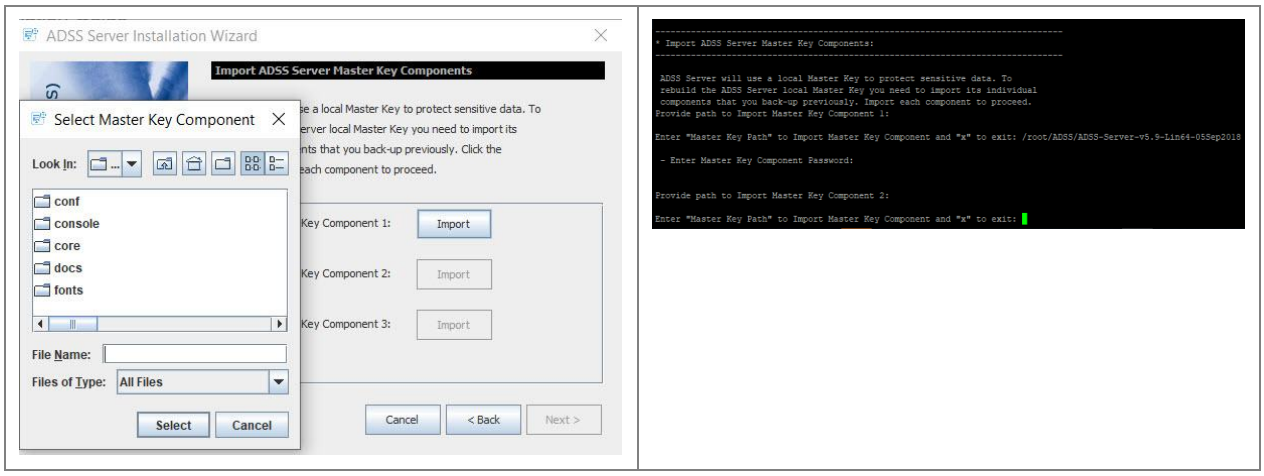


Click Next will show following screen:



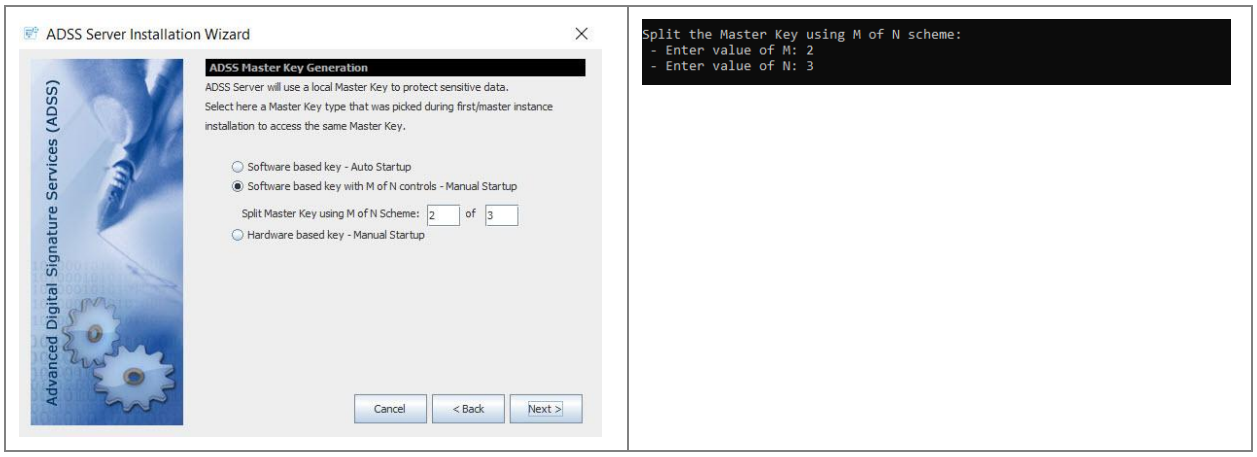
Use the **Import** button one by one to restore the backup of each Master key component (generated during the ADSS Server Primary instance installation) so that installer will restore the Master Key, installer will prompt to provide a password for each Master Key component and decrypt it with the provided password.

The Master Key backup should be imported from the file system in the same sequence and passwords when it was backup during first installation.

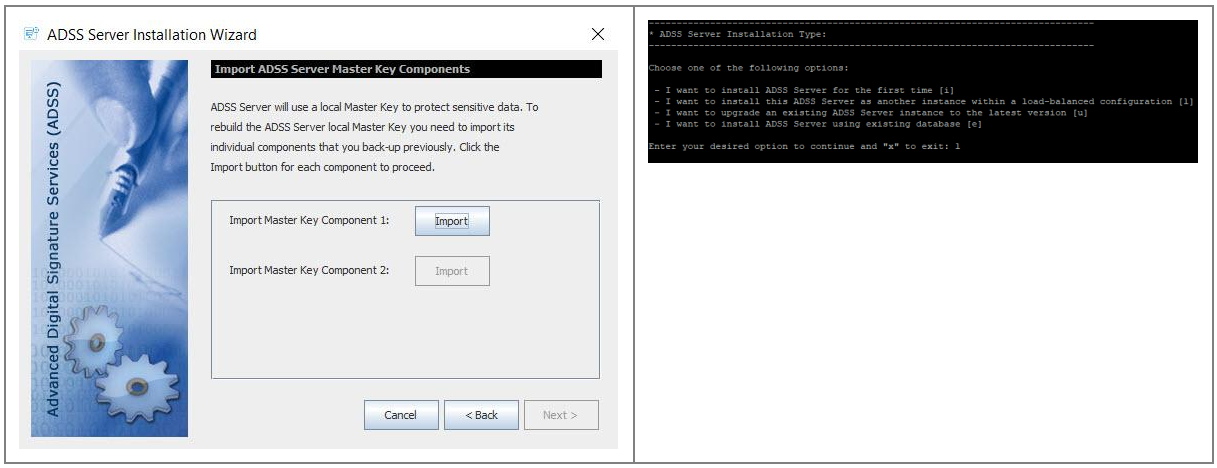


Software Based Key with M of N controls – Manual Startup

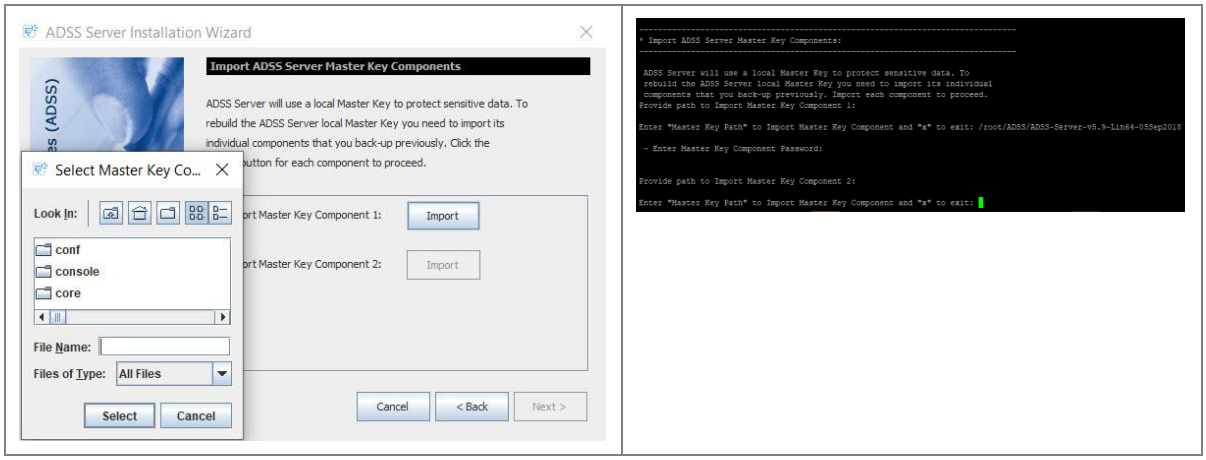
If you had installed the ADSS using this option, then you will have to select the same option during load balanced installation. Installing ADSS Server in load-balanced mode with key type M of N controls will display the following screen:



Click Next will show following screen:



Here, the number of Master Keys to be imported depends upon the number of M defined during first installation. Use the **Import** button one by one to restore the backup of each Master key component (generated during the ADSS Server Primary instance installation) so that installer will restore the Master Key, installer will prompt to provide a password for each Master Key component and decrypt it with the provided password.

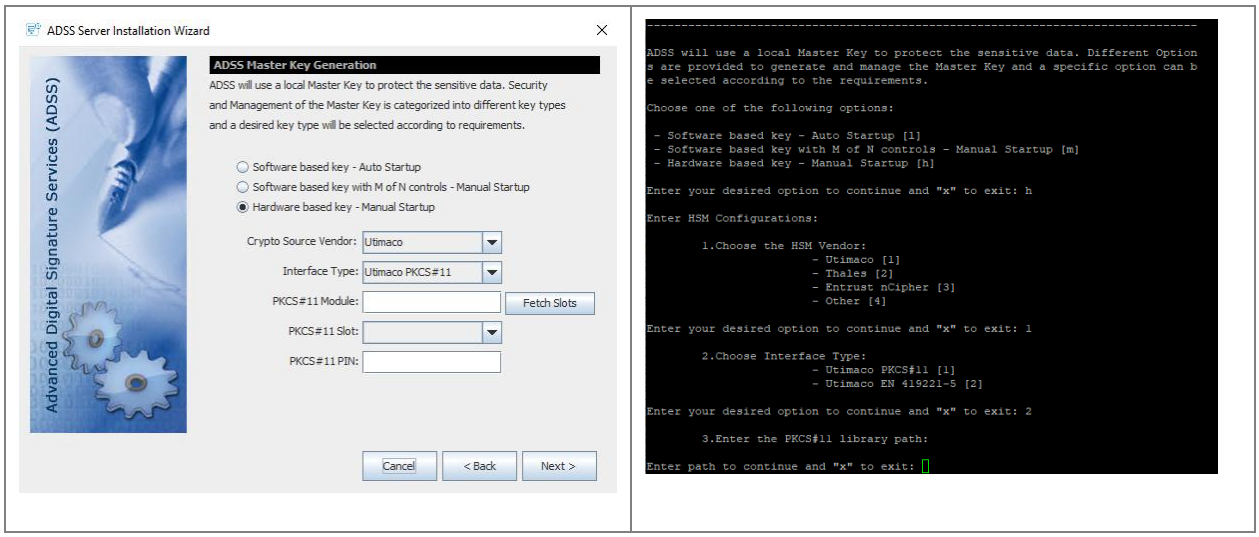


Hardware Based Key – Manual Startup

Before performing load balanced installations of ADSS with a master key inside the HSM, some pre-requisites must be met. These are given below:

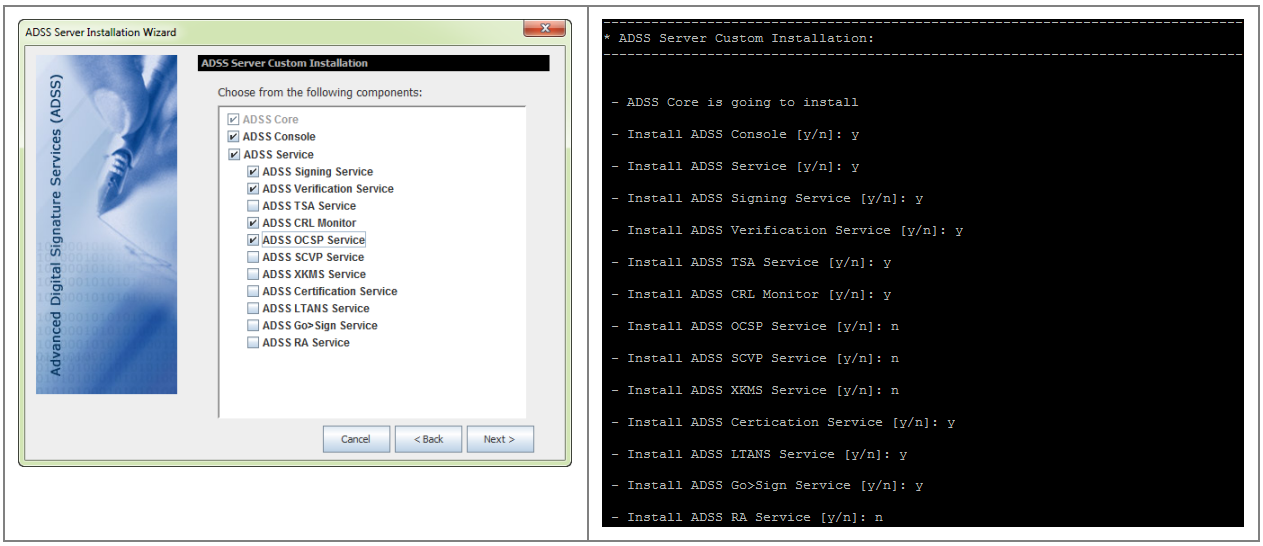
- If a network HSM is being used to store the master key, then you will have to make sure that ADSS instance has the access to that HSM so that it can access the same master key during installation.
- If PCIe HSM was used during the installation of first instance, then the master key will reside in that PCIe HSM. If you are installing ADSS with existing database on same machine then you will have access to the same master key inside the PCIe HSM. Nothing to do in this case. However, If its a different machine with its own PCIe HSM, then you will have to replicate the same master key from first instance HSM to this instance. For that, take backup of the master key from PCIe HSM of first instance and restore it to other instance using any utilities or tools provided by HSM vendor.

The following screen will be displayed and you need to select the “Hardware based key – Manual Startup” option:

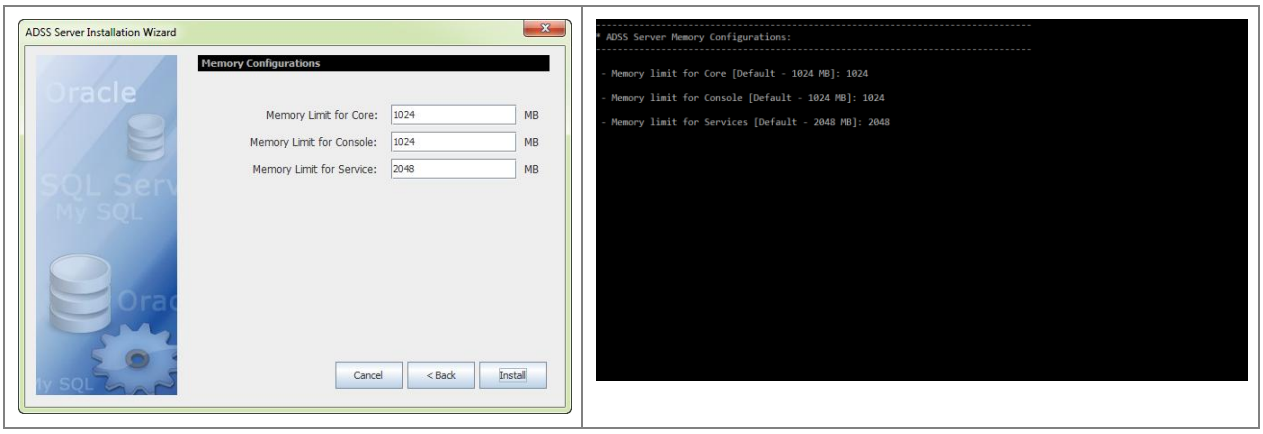


To proceed further, an adss_startup.properties file containing the configurations to connect to the HSM and other information will be imported by clicking on the Import Configuration button. The configuration file will be located at **[ADSS Server Installation Directory]/conf/adss_startup.properties**. These configurations will be used to connect to HSM and access the master key. However, HSM PIN is still required to be entered.

Proceed through the Installation wizard as before until ADSS Server custom installation screen is shown. When installing ADSS Server in a load-balanced configuration the option **Typical ADSS Server Installation** is not available.



On this screen, select ADSS Server modules you want to use, click "**Next**" button and set the maximum memory limit for the selected instances:



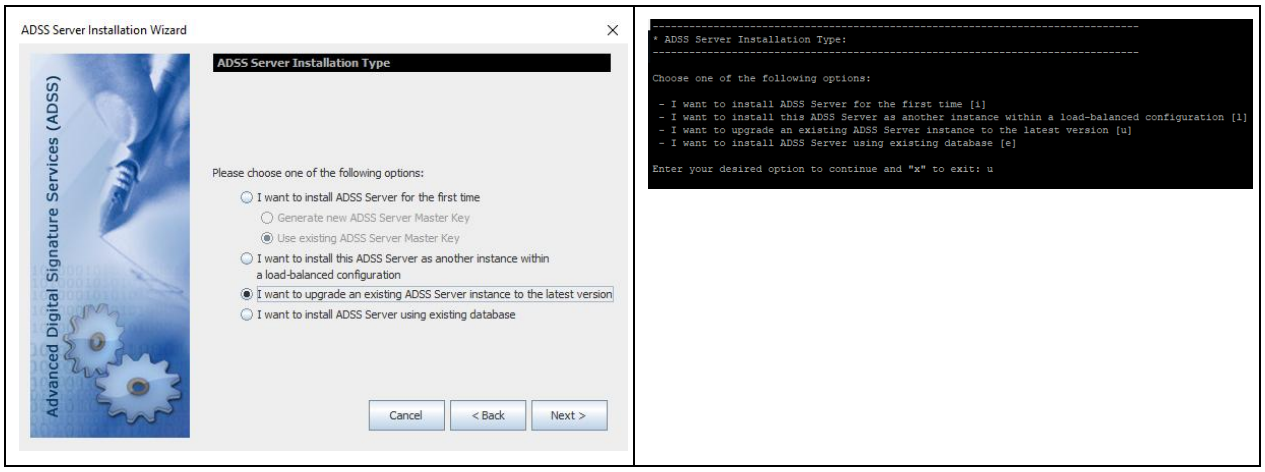
This screen allows you to define the maximum memory limit for each instance of ADSS Server. Default maximum memory limit for Core and Console is 1024 MB and for Service instance 2048 MB.

Provide the maximum memory limit and click **Install** and follow rest of the installation wizard to complete the installation.

4.1.3 Upgrading an Existing ADSS Server Instance

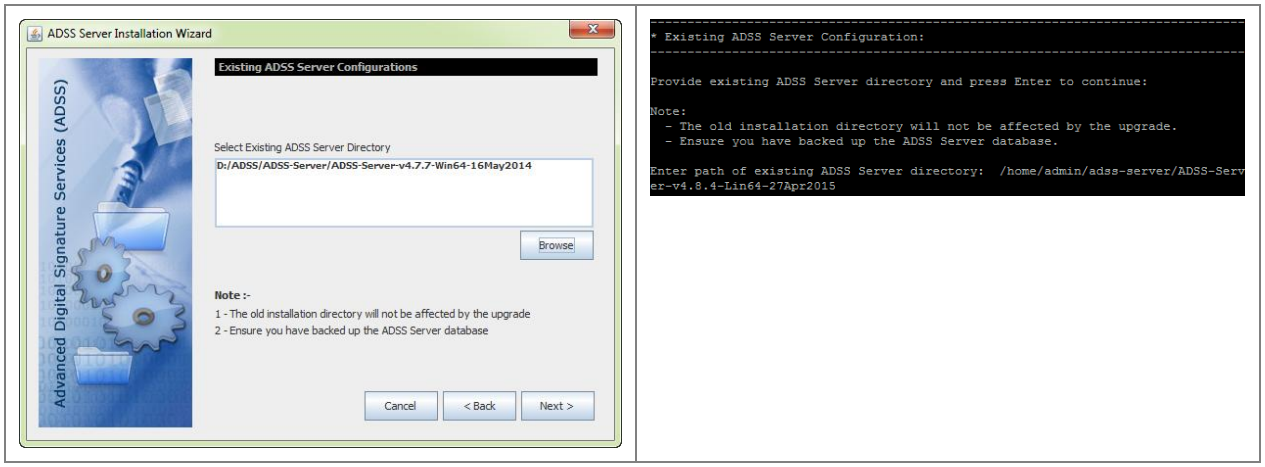
On the current system, download and extract the latest ADSS Server software to a new location that is **different** from the current ADSS Server installation folder or directory.

Now run the ADSS Server installation wizard from the new package, and on the ADSS Server Installation Type screen select the option **I want to upgrade an existing ADSS Server instance to the latest version**:



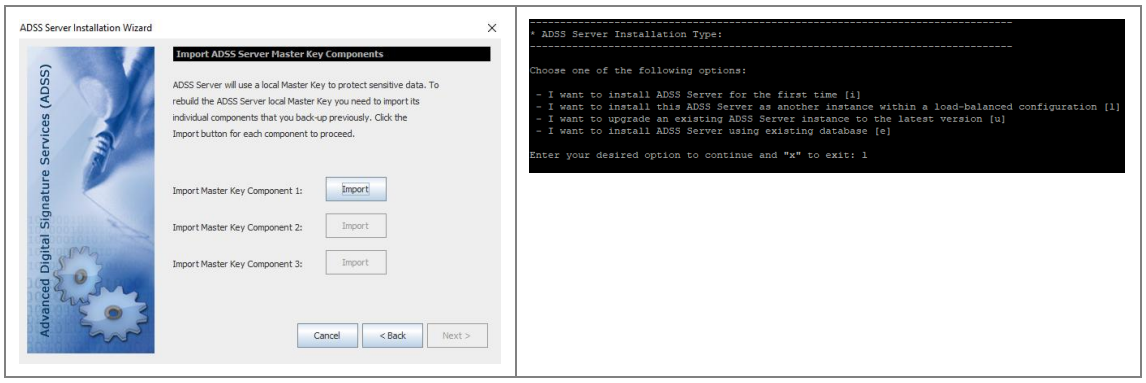
Clicking "**Next**" shows the following screen:

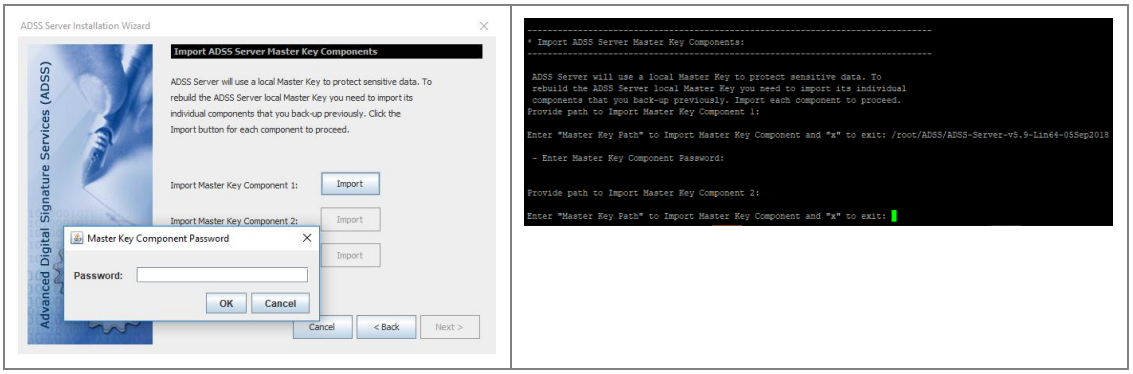
This will show the existing ADSS Directory path in the Wizard as given in below screen:



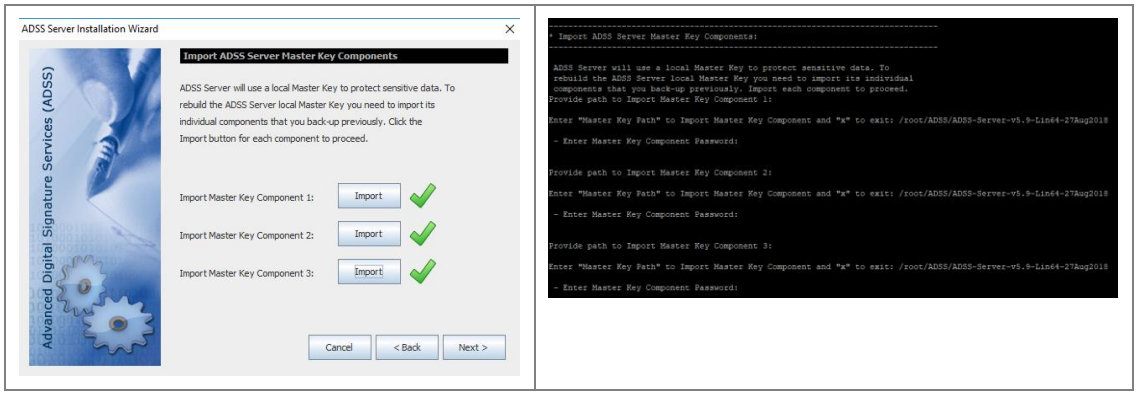
Click **Browse** to select the **existing** ADSS Server installation directory and click "**Next**" to proceed:

- Upgrading ADSS Server from version v5.9 or above doesn't require importing master key components.
- Upgrading ADSS Server from version v5.8 or below require importing master key components
    - For upgrading ADSS Server Primary Core or Standalone instance, the installer will generate a Master Key to encrypt the data in database and prompt to take a backup of the Master Key in the form of three components at the end of the installation wizard.
    - For upgrading ADSS Server Secondary Instance, the installer will prompt to **Import** each Master key component (generated during the Primary ADSS Server installation) one by one along with password in order to decrypt and import it.
      The Master Key backup should be imported in the same sequence and passwords when it was backup during first installation.



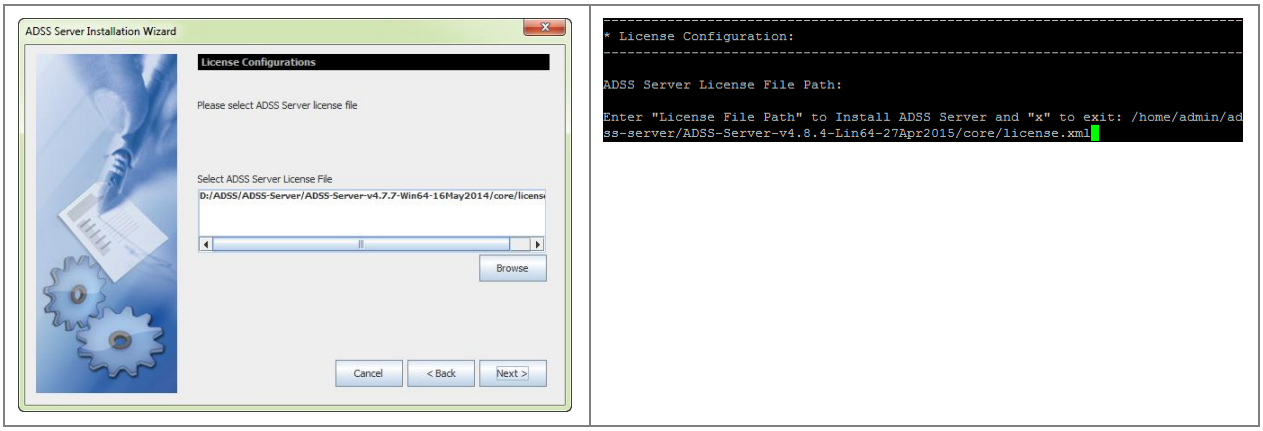


The following screen of Import Backup keys will appear.



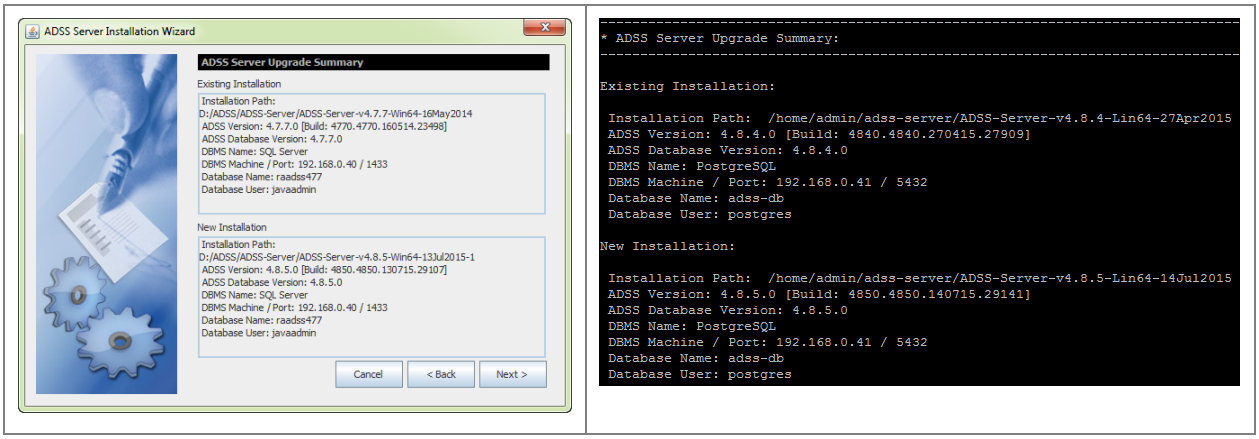
1. *It is recommended that the master key components are stored on a different media (e.g. USB drive) and they must not be stored within the ADSS Server installation folder.*
2. *If you lose these components and/or passwords:*
   a. *You cannot install the ADSS Server in a load-balanced setup.*
   b. *You cannot recover the ADSS Server installation in case of disaster.*
3. *Make multiple copies of these components to recover the installation in case of disaster and/or media corruption.*

Once done, click **Next.**



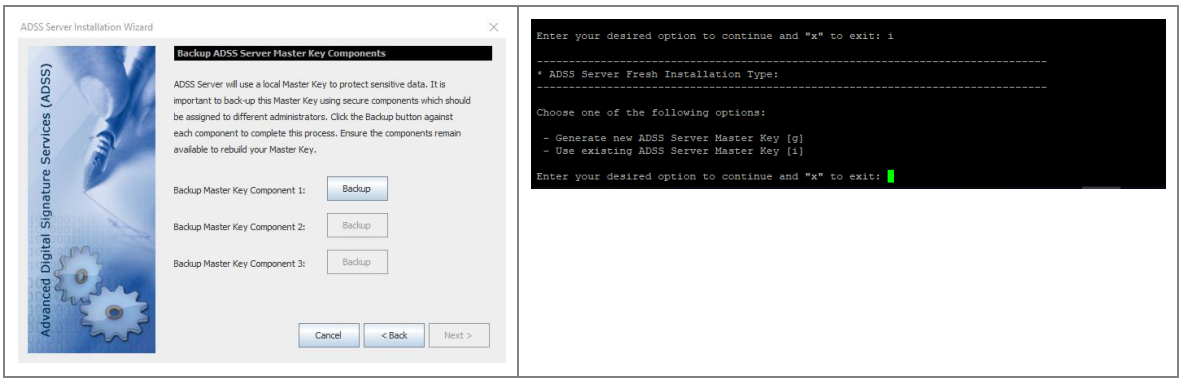
Note: *The above screen will not be shown while upgrading ADSS Server to v6.9 and onwards as the license will be stored in database and ADSS Server will automatically pick up the license from the database.*

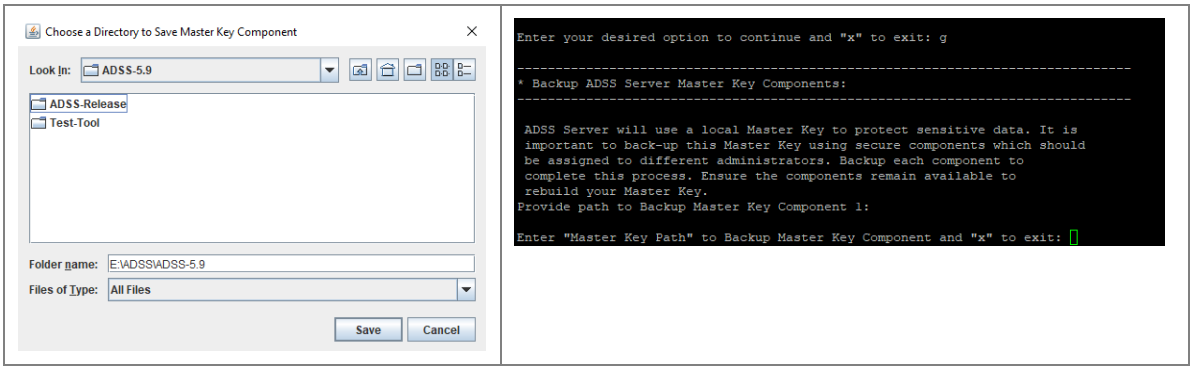
Continue through the installation wizard to reach the screen showing a summary of the old and new installation directories:

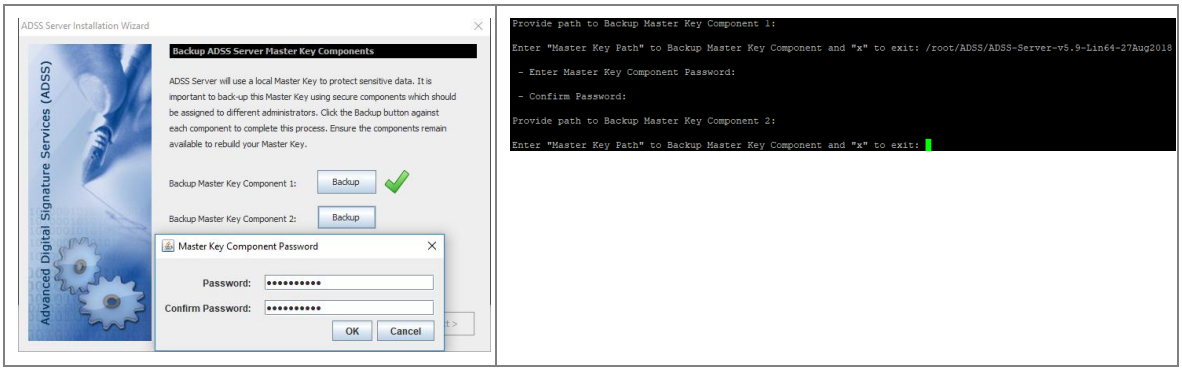


Review the details of both the new and existing installations of ADSS Server. If the summary details are correct, then click **Next >** to continue the upgrade.

- If it is Primary Core or standalone instance upgrade, then the installer will generate a Master Key to encrypt the data in database and prompt to take a backup of the Master Key in the form of three components. Use the **Backup** button one by one to take the backup of each Master key component, installer will prompt to provide a password for each Master Key component and encrypt it with the provided password before saving on the storage media:



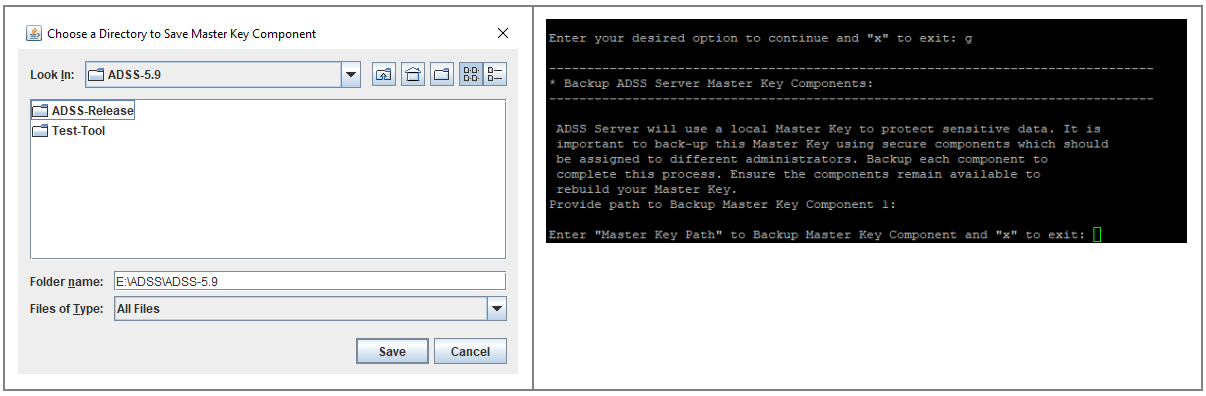


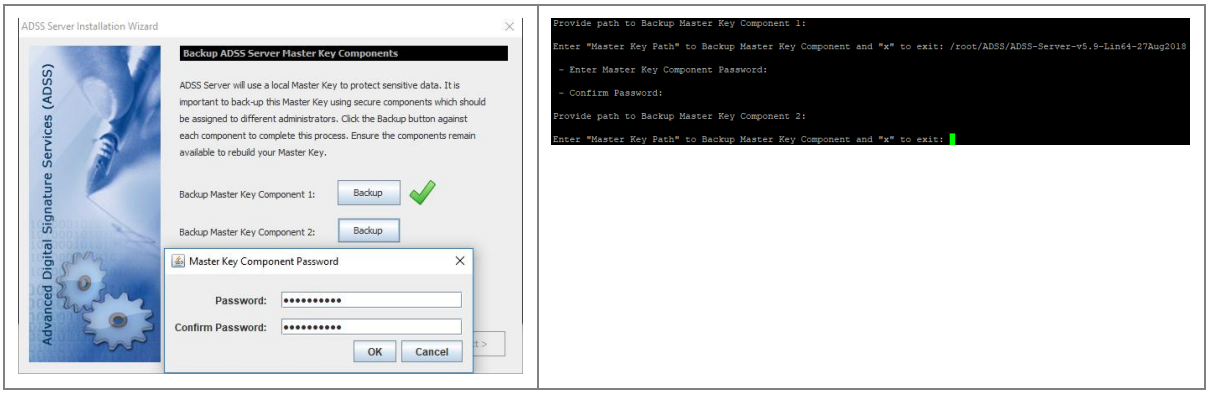


ADSS Server enforces a password strength policy for the generation/backup of Master Key components. Please follow the below guidelines. The password:

1. *Must be at least ten characters long. the installer in GUI mode:*
2. *Must contain the upper and lower case letters.*
3. *Must have digits and special characters.*

It's recommended to use different password for each Master key component. After completed the Backup the following screen appeared:

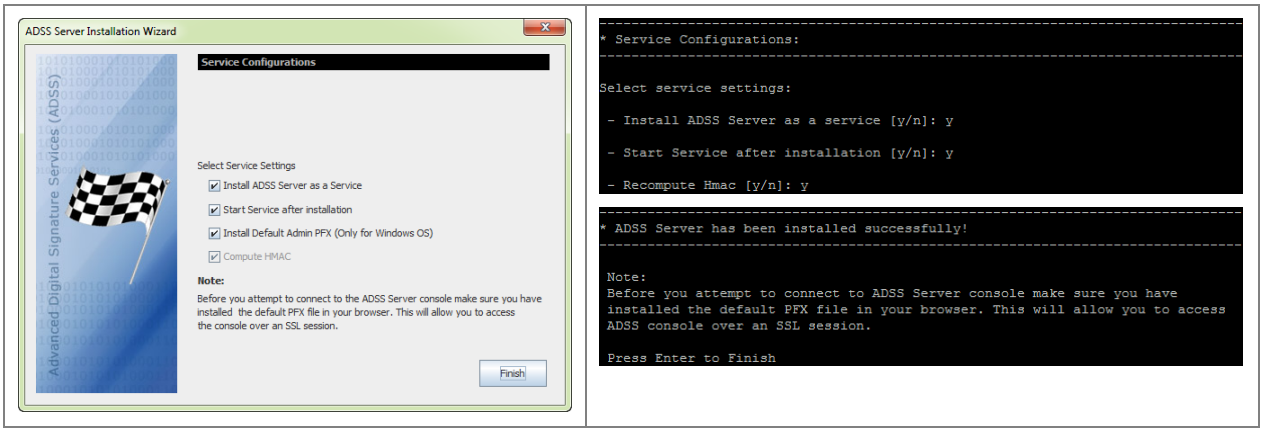




It's recommended to use different password for each Master key component.

1. *It is recommended that the master key components are stored on a different media (e.g. USB drive) and they must not be stored within the ADSS Server installation folder.*
2. *If you lose these components and/or passwords:*
   a. *You cannot install the ADSS Server in a load-balanced setup.*
   b. *You cannot recover the ADSS Server installation in case of disaster.*
3. *Make multiple copies of these components to recover the installation in case of disaster and/or media corruption.*

After the upgrade, it is strongly recommended that the HMAC values are re-computed. On **Finish** screen an option is provided for this:



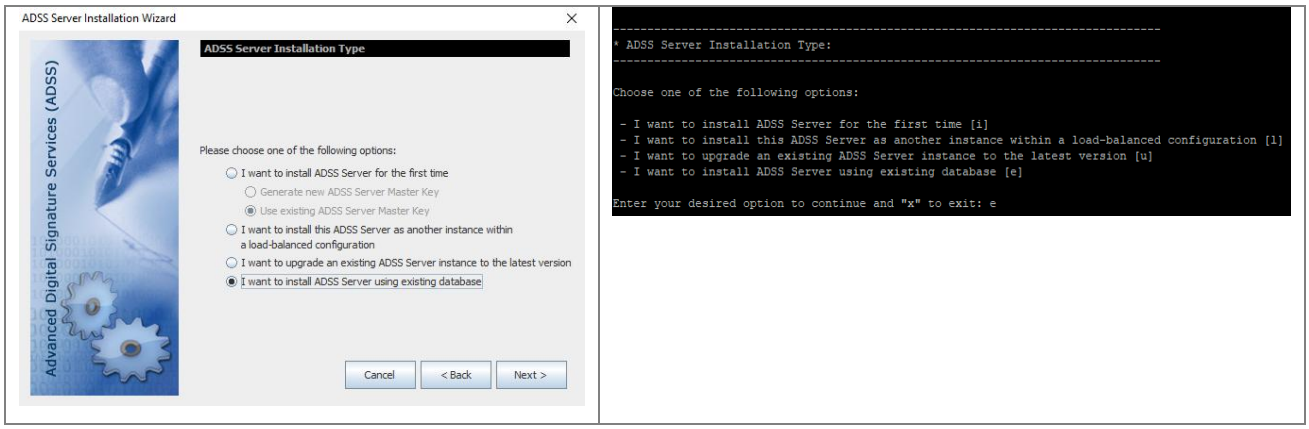
Compute HMAC – HMAC is shown corrupted for database records for which new columns are added/deleted when upgraded to new version. By checking this option, HMAC will be recomputed on the tables that are alerted to meet new functionality requirements otherwise HMAC values will no longer be valid.

HMAC re-computation can take a substantial period of time depending on the database size (i.e. measured in hours). To minimize this time, it is recommended to complete the installation without re-computing HMAC during the upgrade process to allow ADSS Server operations to be started without delay as described next.

The HMAC values on existing records can be recomputed later using a separate utility application. This utility allows ADSS Server to continue running at the same time while the utility performs its HMAC re-computation task in background. It also allows HMAC re-computation to take place overnight. For instructions on re-computing HMAC after installing or upgrading ADSS Server, refer to the technical note titled ***'How to use HMAC recomputation utility manually after installing/upgrading the ADSS Server'*** in the Ascertia Community Portal knowledge base.

4.1.4 Installing ADSS Server Using an Existing Database

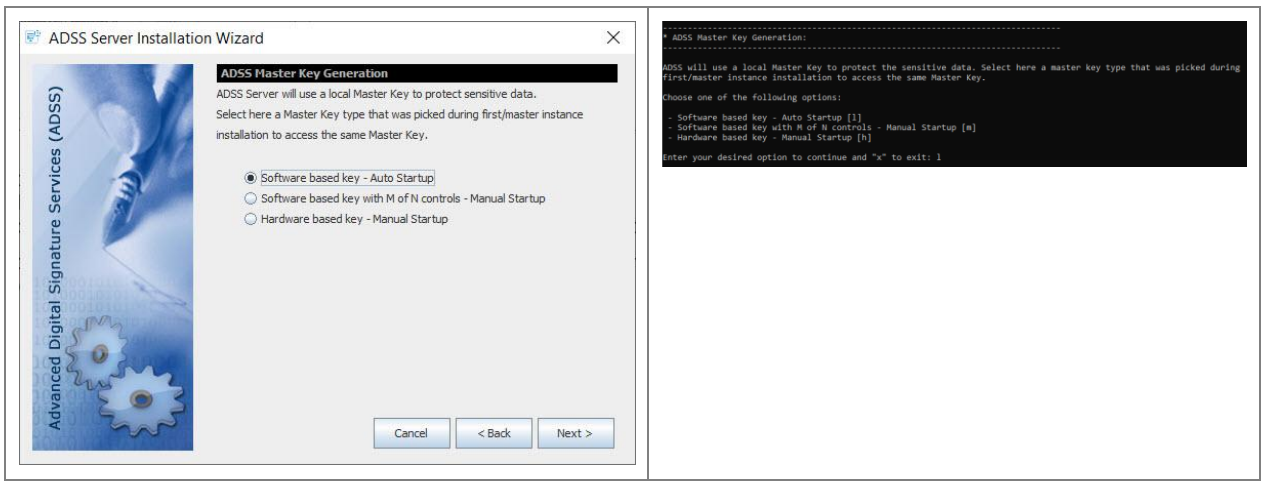
Select option **I want to install ADSS Server using existing database** on ADSS Server Installation Type screen:



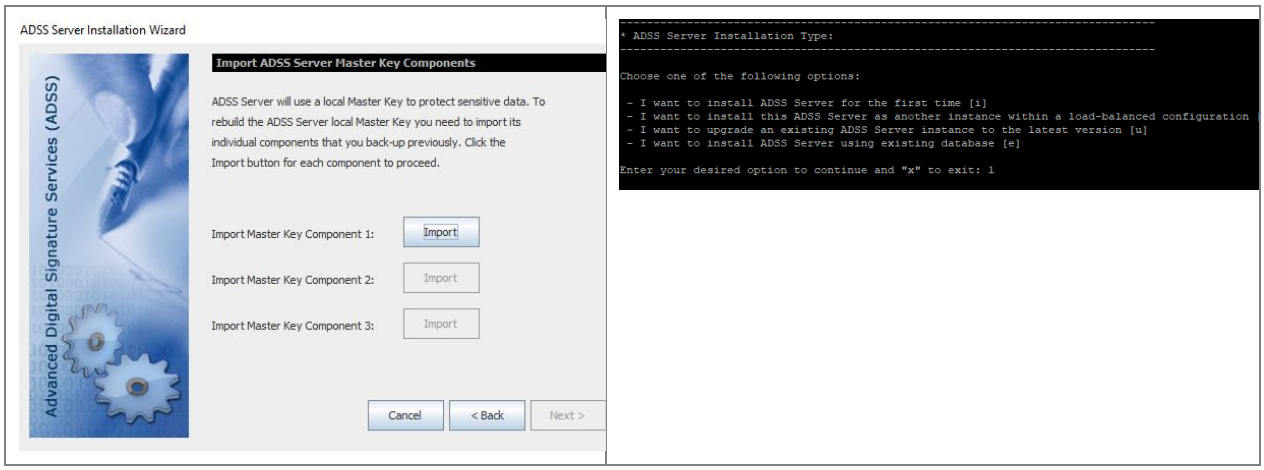
Click on the **Next** button, it will lead you to the Master Key options screen:

Software Based Key – Auto Startup

If you have installed the first instance using this option, then you need to select the same option while installing ADSS Server using an existing database. Following screen will be displayed:

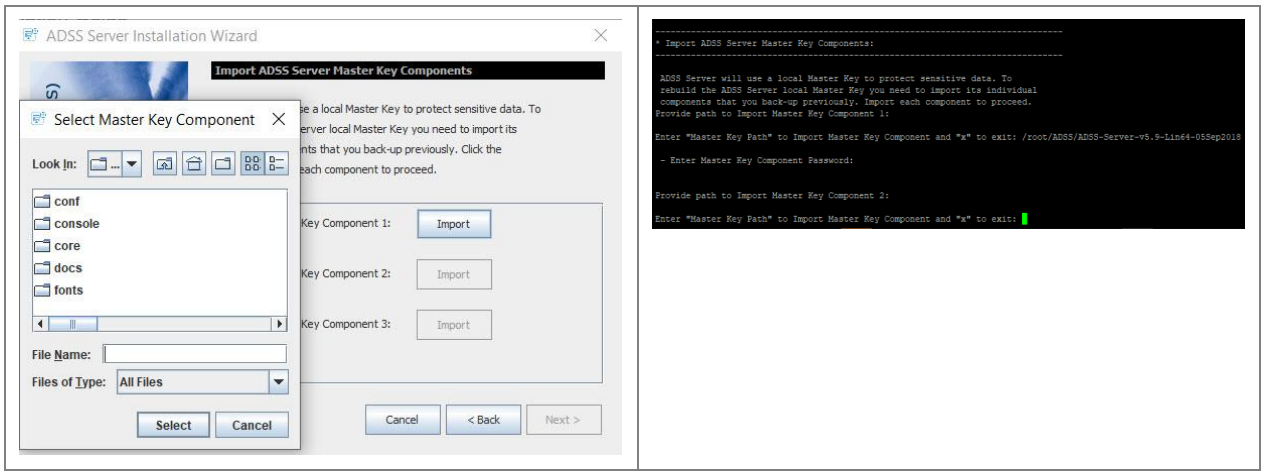


Click Next will show following screen:



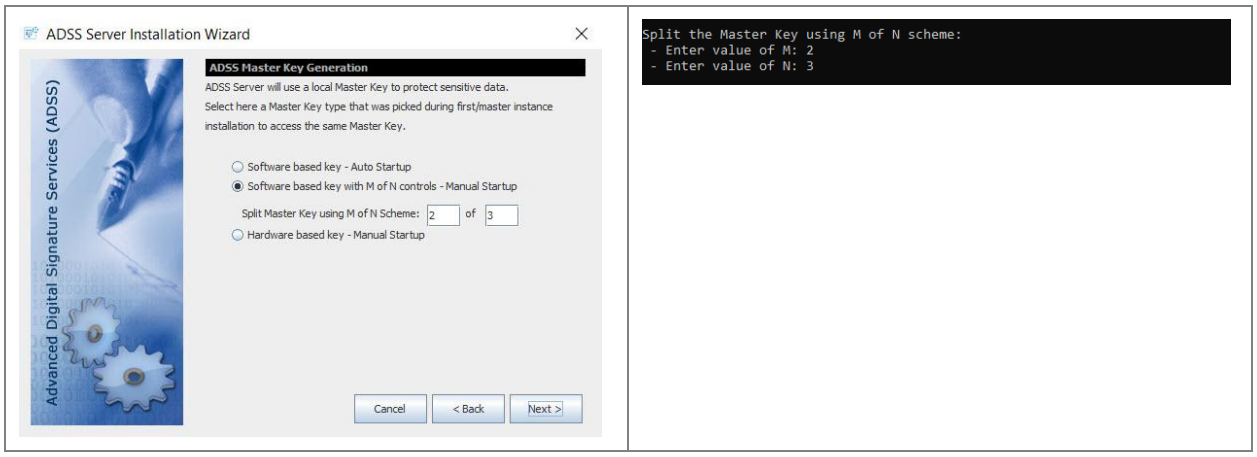
Use the **Import** button one by one to restore the backup of each Master key component (generated during the ADSS Server Primary instance installation) so that installer will restore the Master Key, installer will prompt to provide a password for each Master Key component and decrypt it with the provided password.

The Master Key backup should be imported from the file system in the same sequence and passwords when it was backup during first installation.

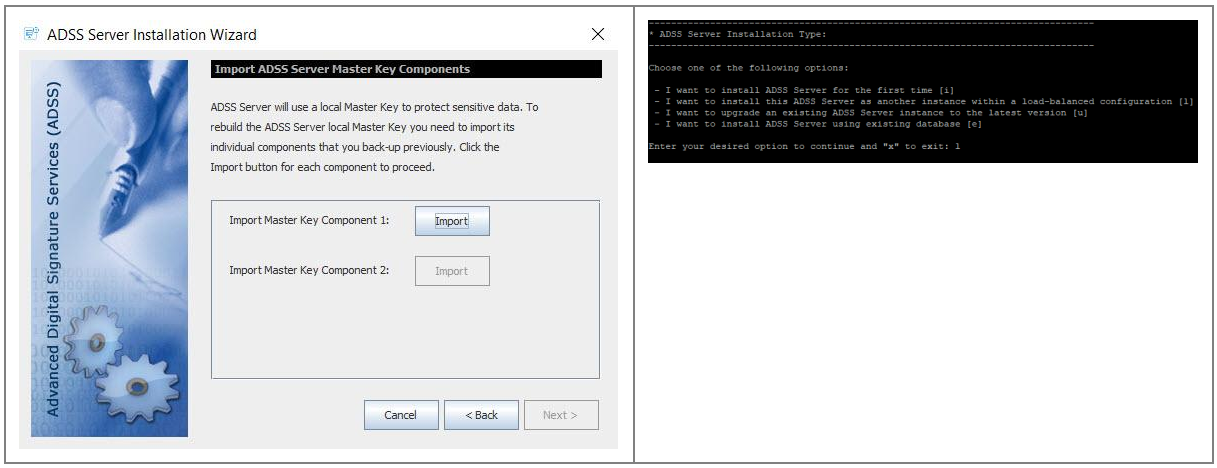


Software Based Key with M of N controls – Manual Startup

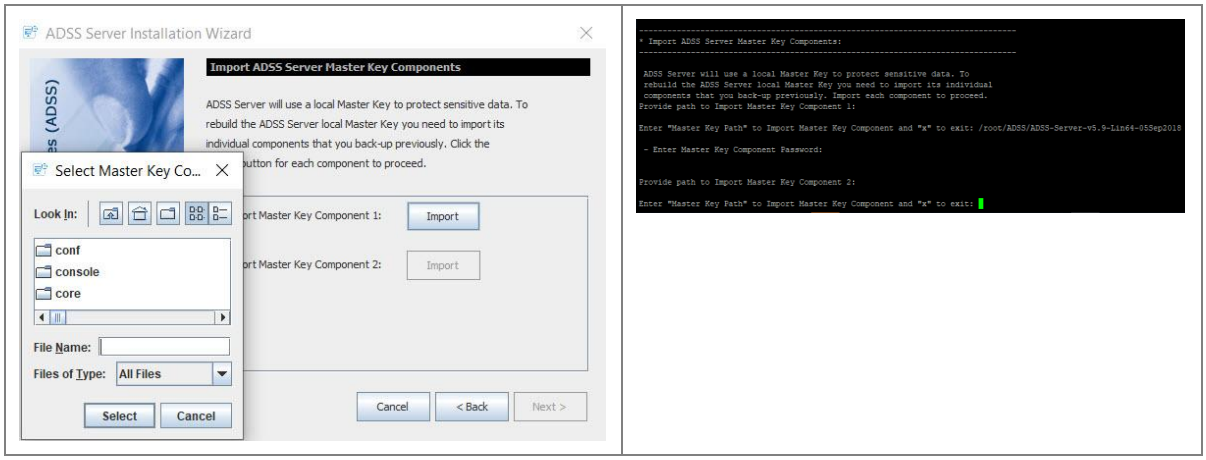
If you have installed the first instance using this option, then you need to select the same option with same M of N scheme while installing ADSS Server using an existing database. Following screen will be displayed:



Click Next will show following screen:



Here, the number of Master Keys to be imported depends upon the number of M defined during Installation. Use the **Import** button one by one to restore the backup of each Master key component (generated during the ADSS Server Primary instance installation) so that installer will restore the Master Key, installer will prompt to provide a password for each Master Key component and decrypt it with the provided password.

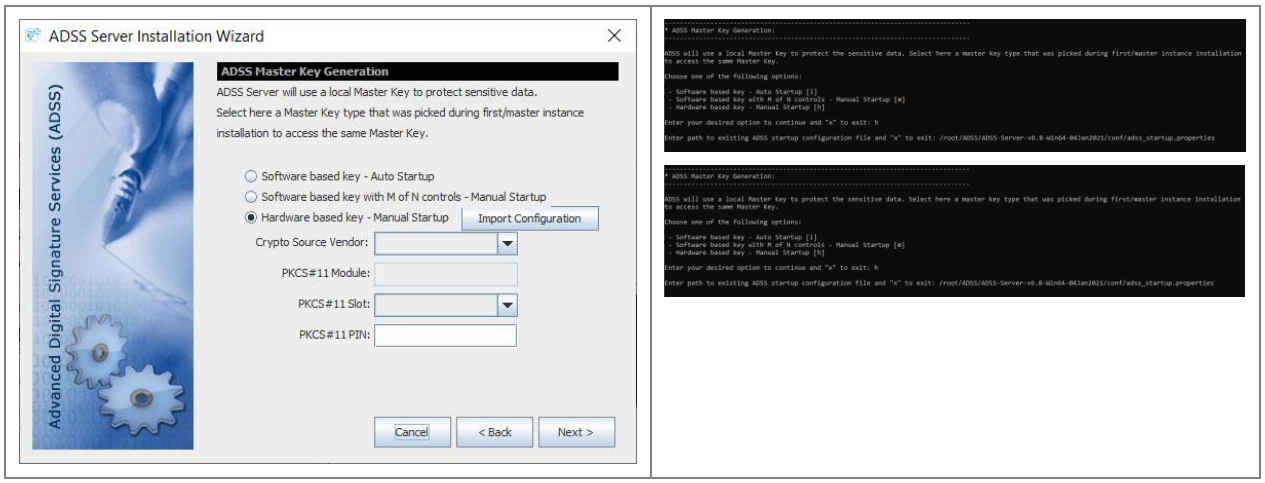


Hardware Based Key – Manual Startup

If you have installed the first instance using this option, then before installing ADSS Server with an existing database, there are some pre-requisites that must be met. These are given below:

- If a network HSM is being used to store the master key, then you will have to make sure that ADSS instance has the access to that HSM so that it can access the same master key during installation.
- If PCIe HSM was used during the installation of first instance, then the master key will reside in that PCIe HSM. If you are installing ADSS with existing database on same machine then you will have access to the same master key inside the PCIe HSM. Nothing to do in this case. However, If its a different machine with its own PCIe HSM, then you will have to replicate the same master key from first instance HSM to this instance. For that, take backup of the master key from PCIe HSM of first instance and restore it to other instance using any utilities or tools provided by HSM vendor.

The following screen will be displayed and you need to select the "Hardware based key – Manual Startup" option:



To proceed further, an adss_startup.properties file containing the configurations to connect to the HSM and other information will be imported by clicking on the Import Configuration button. The configuration file will be located at **[ADSS Server Installation Directory]/conf/adss_startup.properties.** These configurations will be used to connect to HSM and access the master key. However, HSM PIN is still required to be entered.

Once done, proceed through ADSS Server Installation wizard as before. On the **Typical DB Configurations** or **Advanced DB Configurations** screen, provide the details for the existing ADSS Server database and continue through the Installation wizard as before.

Note: After Installing ADSS Server using an Existing Database, we need to copy some files from existing ADSS Server Installation Directory to new ADSS Server. Path to the files are mentioned below:

1. *[ADSS Server installation directory]\conf\adss.keystore*
2. *[ADSS Server installation directory]\conf\pkcs11.properties*
3. *[ADSS Server installation directory]\jdk\jre\lib\security\jssecacerts*

ADSS Server contains a sophisticated Export and Import feature within its Global Settings module. This allows all or selected records to be exported and imported to a new system.

This can be a valuable way of copying ADSS Server configuration data from a pre-production system to a production system.

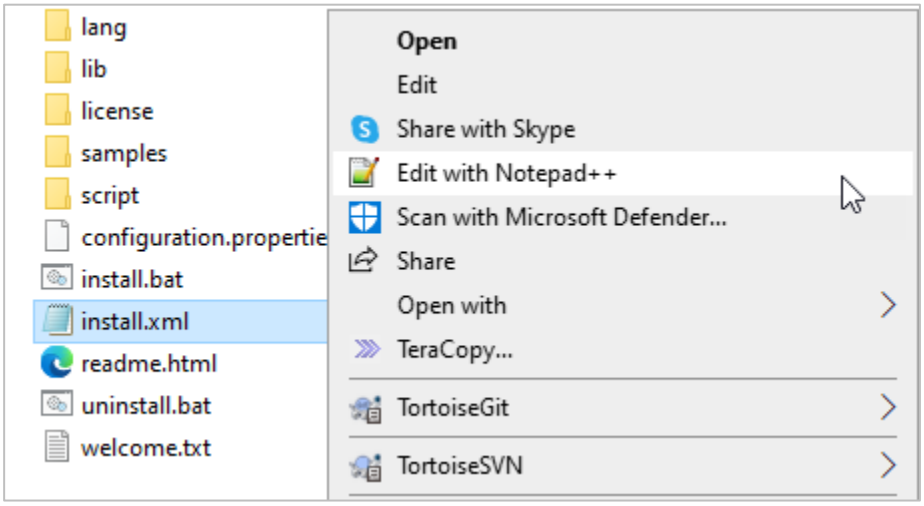
Import & Export MUST only be performed between the same versions of ADSS Server using the same Master Key.

4.2 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:

install.xml

```
<?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, 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_CP5), 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_CP5 Interface Type-->
       <Pkcs11UserPin></Pkcs11UserPin>           <!-- Specify the PIN for above Crypto User, only required for UTIMACO_CP5 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 -->
        <!-- <IpAddress></IpAddress> -->
    </PreferredIpVersion>
    <!-- upgrading screen-->
    <ExistingInstallation visible="true">
        <!--<Path>EXISTING_ADSS_INSTALLATION_DIR</Path>-->
    </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>
        <!-- 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>
        <RAS>FALSE</RAS>
        <Decryption>FALSE</Decryption>
        <CRLMonitor>FALSE</CRLMonitor>
        <OCSPRepeater>FALSE</OCSPRepeater>
        <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 and EXISTING_DATABASE.

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.
   Please note that when upgrading from the current version to the next version of ADSS Server in silent mode, only the tags shown in the screen below should be used:

```
<Wizard>
  <Welcome visible="false"/>
  <LicenseAgreement visible="false">
    <Agree>TRUE</Agree>
    <!--  possible values are TRUE/FALSE  -->
  </LicenseAgreement>
  <InstallationMode visible="false">
    <Mode>UPGRADE</Mode>
    <!--  possible values are FIRST_TIME/LOAD_BALANCE/UPGRADE/EXISTING_DATABASE, Possible values for masterKey are generate,import  -->
  </InstallationMode>
  <!--  upgrading screen -->
  <ExistingInstallation visible="false">
    <Path>D:\ADSS_Installations\ADSS_v7.1.1\ADSS-Server-v7.1-Win64-16Jun2022-temp</Path>
  </ExistingInstallation>
  <MemoryConfiguration visible="false">
    <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="false"/>
  <ServiceConfiguration visible="false">
    <InstallAsService>FALSE</InstallAsService>
    <StartService>FALSE</StartService>
    <InstallDefaultAdminPFX>FALSE</InstallDefaultAdminPFX>
    <ComputeHMAC>TRUE</ComputeHMAC>
  </ServiceConfiguration>
  <UpgradeSummary visible="false"/>
</Wizard>
```

   It is recommended to remove any tags not specified in the displayed screen to ensure a smooth upgrade process.

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.

4.3 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 can update it by generating a new certificate through the ADSS Server Admin Console. For detailed instructions, refer to the technical note titled ***'How to replace the default ADSS SSL Server Authentication Certificate with a production certificate'*** available in the Ascertia Community Portal knowledge base. Additionally, Ascertia recommends creating at least two operators for improved security and reliability.

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.

4.4 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.

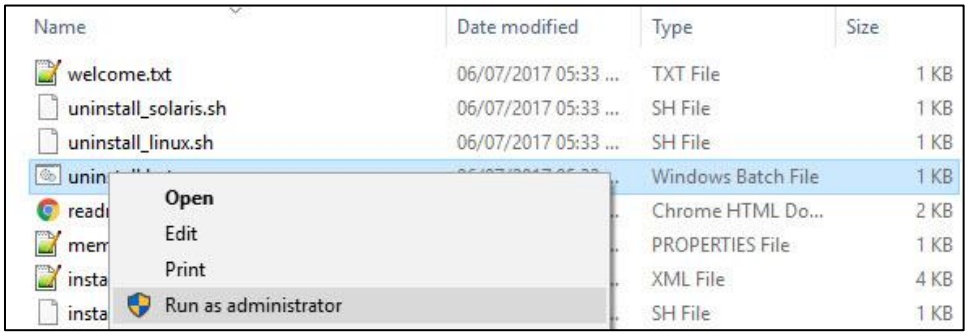


Figure 8 - Windows Example Uninstall Run as administrator

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.

4.5 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.

4.6 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.

Detailed information about known database, service, and console etc. issues can be found here:

http://manuals.ascertia.com/ADSS-Admin-Guide/troubleshooting.html

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.

5 Post-Installation Notes

Review this check list after the successful installation of ADSS Server.

5.1 Secure Inter-Service Communication of ADSS Server

After the Installation of the ADSS Server, the operator must create a new TLS Server Certificate in Key Manager where the certificate must contain either DNS Name or IP Address in SAN extension. If we are planning to install ADSS Server in load balance mode, then we need to make sure that the TLS Server Certificate must contain IP Addresses or DNS Names for all those instances. Once the certificate is created, we need to configure it in Global Settings → System Certificate. This is done to ensure that secure communication is established between all the ADSS Server Instances i.e. Core, Console, and Service.

5.2 Secure Deployment of ADSS Server

Make ADSS Server deployment secured by following these guidelines:

- Mark the value of property **INVALIDATE_SSL_ON_LOGOUT = TRUE** under Global Settings > Advanced Settings > Console
- Set the lower value of **Console Session Timeout** configured in Global Settings > Miscellaneous
- Configure strong ciphers in the server.xml file for both Console and Service instances. For detailed instructions on updating cipher settings, refer to the technical note titled ***'Changing the SSL/TLS cipher suites used by the ADSS Server for advanced security and improved TLS ranking with SSL Labs'*** in the Ascertia Community Portal knowledge base.

5.3 ADSS Server Operators

Each operator should be assigned their own unique certificate, click here for more details. Once operator created assign appropriate role and privileges as described in above link.

Note ADSS Server supports third party hardware such as USB Dongles to hold operator credentials. This is encouraged where security is paramount.

If required ADSS Server has an Approval Manager module that ensures only Security Officers can approve operations for ADSS Server administration.

5.4 HSM Configuration

ADSS Server requires HSM partition passphrase if relevant (note the partition/slots will be found automatically by the Crypto Manager of ADSS Server and Azure Key Vault does not require a passphrase but does require an Azure Key Vault account). Ensure, if relevant, that the ADSS Server host has the correct HSM set-up and configuration, and communication is established between the two entities.

The HSM is configured via the Key Manager > Crypto Source menu. Ascertia provide configuration instructions for Utimaco CP5, Thales SafeNet and nCipher nShield in Appendix section at the end of this document.

5.5 Local CA Configuration

When configuring a Local CA ensure:

- The complete certificate information of CA and end entity certificates is entered, and subsequently validated. For example, CDP, and AIA information that will be added to end entity certificates must

be accessible by relying parties. Revocation information must be available to external parties that may rely on the information.

- If publishing to LDAP, then suitable user credentials and connection configuration information is required to publish certificates to the directory.

5.6 External Trust Services

If external OCSP, CRLs, and timestamp services are required then make sure that they are available to ADSS Server. A test utility is provided for each service when configuring the respective element to ensure connectivity and possibly authentication is successful.

It is important to check that the appropriate CRL monitoring is functioning correctly if relying on third party CAs. Without a functioning CRL (based upon polling) retrieval ADSS Server will not be able to successfully validate certificates.

For external CAs the appropriate account information is required. Each respective Managed PKI service provides different account details to allow certificate requests from ADSS Server.

5.7 NTP Configuration

If using the TSA service of ADSS Server, then Ascertia recommend a suitable time source is configured and the local system clock is not used. ADSS Server allows configuration of multiple NTP time sources. The NTP client can maintain a check on the accuracy of the information provided and shut down the service if outside a validity window.

5.8 Database Log Archiving Frequency

Configure the archiving configuration settings for log files. This entails the period of archiving, and associated settings. Note each ADSS Server service has the option to configure archiving individually.

Archive settings are dependent upon usage. For high volumes Ascertia recommends an archiving schedule of 30 days maximum. However, this number may well have to be less.

5.9 Alert Configurations

Configure the alert configuration settings required. This means the method (note ADSS Server supports email, SMS, and SNMP) and which administrators should receive notifications for which services.

5.10 Licensing

For a production installation, ensure you are using a non-evaluation license of ADSS Server.

5.11 Prepare the Backup Strategy

Prepare a disk and database backup procedure to ensure the full service can be restored in the event of failure with the least amount of disruption. ADSS Server is not responsible for backing up its own database. The IT team must implement operate the backup using appropriate database vendor or third-party tools. ADSS Server provides an admin interface option that allows all the configuration settings to be exported as discussed here.

Once the system is configured it is recommended that all settings are exported. This is also the recommended way of moving a proven configuration from pre-production to production.

5.12 Trace Log Sizing Guide

ADSS Server produces low level information and debug logs on the hard disk. Retention period of these logs can be decided based on your needs. Detailed information about the trace logs and its sizing can be found here:

- Trace Logs (ascertia.com)
- Managing ADSS Server Logs (ascertia.com)

5.13 ADSS Server Clients

Once ADSS Server is deployed the appropriate clients must be created, given access to the system and required services. This means configuring their authentication and authorisation details in ADSS Server, Client Manager module allows you to do this.

When creating clients Ascertia recommends that the least privilege approach is used and clients configured to allow access to the minimum required services, and in the case of Signing Service, appropriate keys;.

5.14 Advance Settings Configuration

After the installation of ADSS Server, the operator need to replace the existing parameter URL values of Classic Console and Unity Console in Advance Settings under Global Settings. For details, see the below instructions:

1. Launch the ADSS Server Console
2. Navigate to Global Settings → Advance Settings → General
3. Replace the default value of **CLASSIC_CONSOLE_URL** from **localhost** to machine name or IP address (in local network) or by domain name (over the internet).
4. Replace the default value of **UNITY_CONSOLE_URL** from **localhost** to machine name or IP address (in local network) or by domain name (over the internet).
5. Save the updated settings by clicking on the Save button.
6. Restart Service/Daemon for all ADSS Server instances for changes to take effect.

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