Atostek ID Active Directory Registration Service 4.5 Installation Guide

for Windows

v1.0

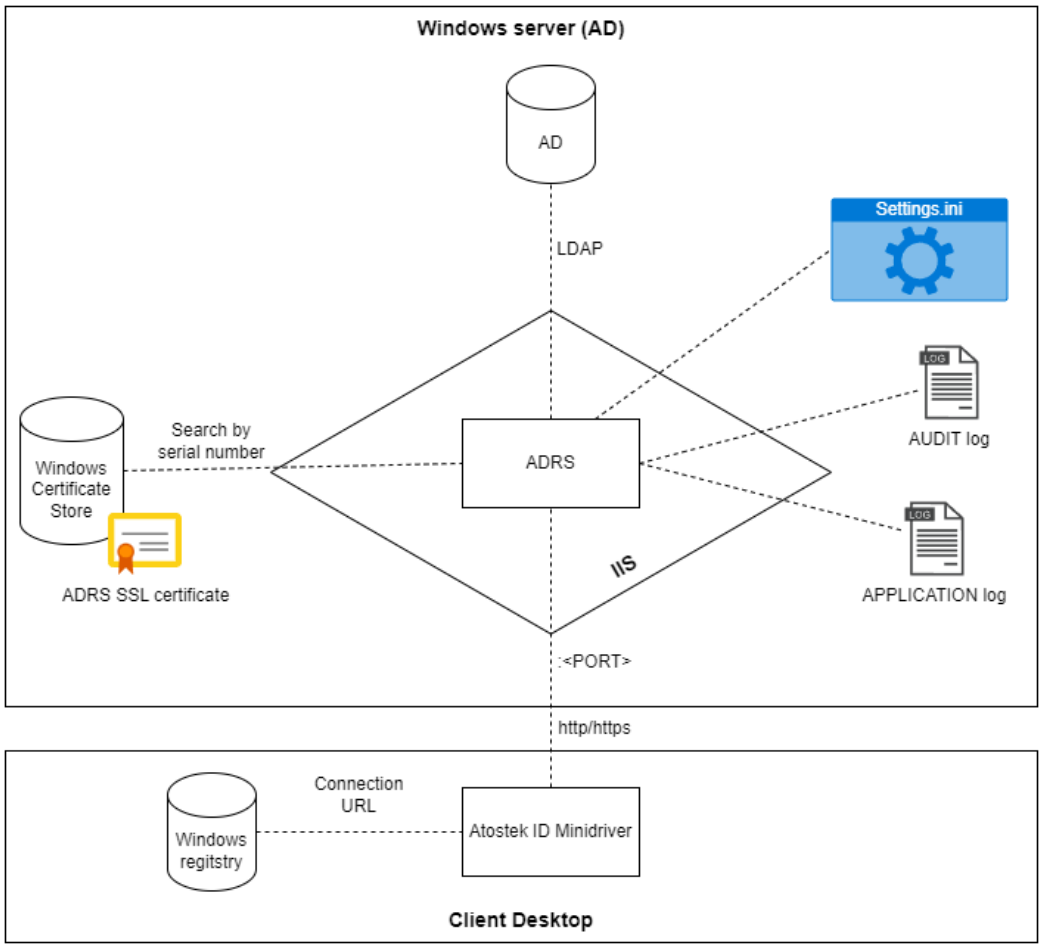
Atostek

Table of contents

1. What is Atostek ID ADRS?

Atostek ID ADRS is a separate active directory registration service (ADRS) that configures users' smart card to match AD user's identity. Microsoft has made changes to the certificate-based authentication and ADRS's responsibility is to match these new requirements (https://support.microsoft.com/en-us/topic/kb5014754-certificate-based-authentication-changes-on-windows-domain-controllers-ad2c23b0-15d8-4340-a468-4d4f3b188f16). ADRS is a service that is installed to the organization's own environment and connected to the selected AD. Service configures each user's altSecurityIdentity attribute to match new requirements. For each user X509IssuerSerialNumber mapping is added when the smart card is connected to the client device. ADRS finds the matching user from the AD and adds the corresponding X509IssuerSerialNumber mapping for the user. After that the user can log in to the device by using the smart card. Atostek ID also supports multiple installations of ADRS connected to same AD to enable higher availability. See more about the configuration of Atostek ID client from Atostek ID installation guide.

Atostek ID ADRS software runs on the Microsoft Windows operating system. The following diagram illustrates the ADRS system.



2. Installing the software using the installer

2.1. Before installing

Make sure that the firewall accepts TCP communication to the port the Active Directory is listening to and that the firewall accepts communication to the ADRS from client users. ADRS requires AD users to have the privileges to read and write data to AD user's “altSecurityIdentities” -attribute.

Example of giving right permissions:

1. Open the program "Active Directory Users and Computers".
2. If you want to create a new user for editing, please do so.
3. *View → Advanced Features* on.
4. Click domain root with the second mouse button and select *Delegate Control*.
5. Press *Add* and add the user you want to be able to edit altSecurityIdentities.
6. Select "Create a custom task to delegate".
7. Select "Only the following objects in the folder" and "User objects".
8. Permissions: In the Permissions dialog, select "Property-specific" and find and select "Read/Write altSecurityIdentities".

Install the TLS certificate that you will be using with the service to the system certificate store in the “Personal” -section. Install both private and public certificate keys. After the installation copy the certificate serial number that will be used at the ADRS installation to identify the correct certificate from the system store.

2.2. Installation

ADRS can be installed using the installation wizard or from the command line. For command line installation, you need to start a command prompt with administrator rights. If the installation package is in the current folder, it can be installed with e.g. the following command:

> msiexec /i ADRegistrationService_Setup.msi /qn ADRS_LDAP_URL="aid.atostek.com:5592" ADRS_LDAP_USERNAME="adrs" INSTALL_DESKTOP_SHORTCUT="true"

All parameters (see additional database parameters from chapter 4.2):

> msiexec /i ADRegistrationService_Setup.msi /qn CERTIFICATE_SERIALNUMBER="" ADRS_TRUSTED_CERTIFICATES="tc1,tc2" ADRS_LDAP_BASE_OBJECT="CN=Users" ADRS_LDAP_RECURSIVE_SEARCH=”false” ADRS_LDAP_URL="aid.atostek.com:5592" ADRS_LDAP_USERNAME="adrs " ADRS_LDAP_PASSWORD="" INSTALL_DESKTOP_SHORTCUT="true" INSTALL_MENU_SHORTCUT="true" PORT=”9000”

A complete list of command line parameters can be found in the following table:

Parameter	Value type	Explanation
ADRS_REQUIRE_HTTPS	Boolean (“true”/”false”)	Determines whether ADRS should handle only HTTPS requests. If this parameter is “true”, ADRS rejects all HTTP requests. HTTPS can be used even when this parameter is set to false, if ADRS can find a HTTPS certificate and use it. In that case, HTTP requests are redirected to HTTPS. HSTS is always enabled with HTTPS.
CERTIFICATE_SERIALNUMBER	string	HTTPS certificate serial number which will be used for ADRS service
ADRS_TRUSTED_CERTIFICATES	Pipe separated list. e.g. “CN=Example,O=Organisation CA,C=FI|CN=Another Example,O=Another Organisation,C=FI”	List of trusted smart card issuers. Leave empty to allow all certificate issuers.
ADRS_LDAP_BASE_OBJECT	LDAP path strings separated by pipe character “|”, e.g. “OU=Accounts” or “OU=TestAccounts,OU=OnPrem” or “CN=Users|OU=TestAccounts,OU=Accounts”. An empty string “” can be used to denote the root of the directory. (Note: do not include DC components)	LDAP paths to the base object where ADRS operates with the user accounts (default value is CN=Users)
ADRS_LDAP_RECURSIVE_SEARCH	Boolean (“true”/”false”)	Whether or not ADRS should do a recursive search over all descendants of the base object or only for direct descendants (default value is false).
ADRS_LDAP_URL	string	Address of AD to which ADRS connects with LDAP protocol.
ADRS_LDAP_USERNAME	string	AD username which ADRS uses for attribute modification
ADRS_LDAP_PASSWORD	string	User password (encrypted with ADRS password encrypter)

ADRS_REQUIRE_LDAPS	Boolean (“true”/”false”)	Reject LDAP connections without TLS.
ADRS_REQUIRE_LDAP_SIGNING	Boolean (“true”/”false”)	Reject LDAP connections without LDAP signing. Do not use with LDAPS.
ADRS_REQUIRE_LDAP_SEALING	Boolean (“true”/”false”)	Reject LDAP connections without LDAP sealing. Do not use with LDAPS.
INSTALL_DESKTOP_SHORTCUT	Boolean (“true”/”false”)	If ADRS start icon is added to the desktop
INSTALL_MENU_SHORTCUT	Boolean (“true”/”false”)	If ADRS start icon is added to the menu
PORT	int	Port number which the service starts listening
USEAD	Boolean (“true”/”false”)	This setting can be used to toggle on/off all communication between ADRS and AD, including pairing. The default value is true.

The program can be uninstalled from the Control Panel (Control Panel → Programs → Programs and Features) or from the command line (defaults as above):

> msiexec /x ADRegistrationService_Setup.msi /qn

2.3. After installation

After starting the ADRS service at the host computer you can access the administration view by opening the browser navigating the URL and port that you have defined for the service and adding ”/administration” to the URL. For example:

https://localhost:9000/administration

From the admin view, you can see the service health check and the version and encrypt the password. Password encryptor is used for encrypting all passwords configured for ADRS service. For AD connection, you must encrypt the AD user password that you have defined to be used with the ADRS. Add the password to the plain-text field and click the ”encrypt” -button. After that, you will have the encrypted password which you can copy to the settings.ini file. Settings file can be found from the same folder where you installed the service. After updating the AD-user password start the ADRS again to achieve connection to the AD.

Password encrypter

Write a plain-text password and press the "Encrypt" button to encrypt it. This encrypted password can be copied and pasted in the configuration file (settings.ini) of AD Registration Service.

Plain-text password: ldap-password

Encrypt

08C5B327DDA9E46DF6E7969AEDC645D8

3. Service maintenance and usage

When the TLS certificate is updated, you must update the new certificate serial number to the settings.ini file and after that restart the service. After that, the new certificate is in use. By default, service logs two types of logs: Audit log and Application log. By default, both logs are written to the C:/logs -path. The audit log is written to file "C:\\logs\\ADRegistrationServiceLog_AUDIT_*.txt". The audit log contains the log of each configuration made to the AD. Application log is written to file "C:\\logs\\ADRegistrationServiceLog_APPLICATION_*.txt" and it contains more logs of the running service. A new log file is created for each day.

4. Using Atostek ID ADRS

4.1. Running ADRS as Windows Service

Atostek ID ADRS can also be run as a Windows Service. All the necessary files are located in a zip file "ADRegistrationService-WindowsService.zip" in the installation folder. After extracting the files to a convenient location, open an admin Powershell in the extracted folder and run the script "install_service.ps1" therein. This script installs Atostek ID ADRS as a Windows Service under the name "ADRegistrationService". The service should display as "Atostek ID AD Registration Service" among Windows Services after successfully running the script.

To uninstall, simply run the script "uninstall_service.ps1" in the extracted folder. The script stops the service and then deletes it.

Note that the configuration settings file "settings.ini" must be copied to the extracted folder for Atostek ID ADRS Windows Service to use it.

4.2. Configuring a separate database for ADRS

You can configure ADRS to save pairing information in a separate database during installation. Creating the database and its users is the responsibility of the server administrator. See Table 1 for a more detailed division of responsibilities between the administrator and ADRS. Please note that creating, deleting, and safely storing the database on the server is entirely the responsibility of the administrator. Table 2 describes in more detail the settings that must be set in the settings.ini file to configure the database.

When a database is used, ADRS logs the pairings into one table, where the user's altSecurityIdentities, i.e. a combination of the certificate's issuer and serial number, is used as the primary key. In addition, the database collects the user's user principal name (UPN), the pairing timestamp, user details (first name, last name, registration number, certificate details, email). If a pairing row with the given altSecurityIdentities already exists, its other columns are updated to match the new pairing. Otherwise, a new row is added to the database.

Note that if you only want to save the pairing information to ADRS's database and not add the pairing to AD itself, you can set the setting "*UseDB*" to "*true*" and "*UseAD*" to "*false*".

Table 1. Partition of responsibilities between a database administrator and Atostek ID ADRS

Task	Database Admin	Atostek ID ADRS
Install PostgreSQL	x	
Create a database	x	
Create a database user and password for ADRS	x	
Encrypt the database password with ADRS password encrypter	x	
Add database configurations to settings.ini	x	
Validate configuration parameters		x
Validate database existence and correctness		x
Create a table inside a given schema in the database		x
Validate database columns		x

Table 2. ADRS settings for database (settings.ini)

Setting	Type	Default	Info
DBEncryptedPassword	string	<empty string>	The ADRS database user's password, encrypted with the ADRS administrator view's encryptor.
DBHost	string	localhost	The address of the database server.
DBName	string	adrs	Name of the database.
DBPairingTableName	string	user_certificate_mappings	Name of the database table.
DBPort	int	5432	The network port used to connect to the database.
DBSchema	string	public	
DBUserName	string	adrs	Username for the ADRS database user.
ExpiredCardRetentionDays	nullable int	<null>	Extra feature to define how many days after expiration the pairing rows are deleted from the database. By default, no deletion is done.
UseDB	bool	false	Whether ADRS uses database at all.
DBExcludedFields	string	<empty string>	Comma-separated list of field names that should be excluded when saving pairings to the database or retrieving them via the HTTP API. Field names are treated as case-sensitive.

You can only exclude non-mandatory fields using DBExcludedFields. They are "firstname", "lastname", "regnum", "issuer", "emailSan", "emailRfc", "certificate", "validFrom" and "validTo". When a field is excluded, its value is not saved to the database when pairing and its value is not included in the responses for HTTP interface queries.

You can query database information through the ADRS HTTPS interface. Please note that it is the administrator's responsibility to set up the server so that only allowed parties can call these commands and get information about the pairings. Information can be requested either with a GET /pairings request, which returns all pairings from the database, or with a POST /pairings request, where filters are given in the request body. With a POST request, it's possible to filter the results by the UPN, emailSan, and emailRfc columns, as well as by the registration time. Table 3 describes the request formats in more detail. Table 4 describes the error codes. The response to the requests is given in JSON format, and it looks like this:


```
{
    "errorCode": 0,
    "data": [
        {
            "upn": "example@teonet.org",
            "altSecurityIdentities": "X509:<I>C=FI,O=Digi- ja vaestotietovirasto TEST,OU=Testivarmenteet,CN=DVV TEST Certificates - G5R<SR>81AD2706",
            "registered": "2025-08-22T10:56:40.438489Z",
            "firstname": "Test",
            "lastname": "Example",
            "regNum": "99905322A",
            "issuer": "CN=DVV TEST Certificates - G5R, OU=Testivarmenteet, O=Digi- ja vaestotietovirasto TEST, C=FI",
            "emailSan": "example@teonet.org",
            "emailRfc": "example@teonet.org",
            "certificate": "…",
            "validFrom": "2024-10-25T11:53:27Z",
            "validTo": "2029-10-23T20:59:59Z"
        },
        …
    ]
}
```


Table 3. Requests for querying database information

Request	Body
GET https://ADRS_URL:ADRS_PORT/pairings	-

POST
https://ADRS_URL:ADRS_PORT/pairings
	{
	"upn": "example@teonet.org",
	"emailSan": "%@teonet.org",
	"emailRfc": "example@%.org",
	"registeredBefore": "2025-08-31",
	"registeredAfter": "2025-08-25"
	}

Table 4. Error codes

Code	Info
-1	Default value

0	No error, request OK
1	Database is not used (UseDB=false)
2	Generic error
3	Filter is invalid

4.2.1. Database migrations

1) 4.4.1.0 to 4.5.0.0

This migration changes the primary key of the database from "*upn*" to "*alt_security_identities*".

ADRS 4.5.0.0 installation creates a directory called "*migrations*" into the installation directory. That directory contains an SQL script called "*ADRS_4.4.1.0_to_4.5.0.0.sql*" that can be used to migrate ADRS database from earlier versions before 4.5.0.0. If you have not used a database for ADRS before 4.5.0.0, you do not need to apply this migration.

If you are not using the default values for the table name or schema, be sure to update them into the script. The script contains more information about replacing the default values with custom values. If you have not set any name for the table, ADRS has used the default value, and you do not need to make changes to the script before running it.

You can run the script in PostgreSQL's administration tool pgAdmin 4 by right clicking the ADRS database from the Object Explorer on the left side of the window and selecting "*Query Tool*". Then copy-paste or drag-and-drop the migration script to the query tool. At this point the script can be modified to adapt to custom configurations, such as a non-default table name. When you are ready to run the script, press the play button above the Query text box or press F5 on the keyboard to run the script. After running the script successfully, there should be the following kind of text in the "*Messages*" field below the Query box:

"ALTER TABLE
Query returned successfully in 60msec."

The time taken for the migration may vary depending on the number of rows in the database. If the message is different and shows some kind of error, make sure to check that you are operating in the correct database and that the database schema and table name are correct.

4.3. LDAP hardening

LDAP hardening can be set up between ADRS and AD to improve security. ADRS negotiates secure authentication type with AD depending on the available mechanisms.

ADRS has settings that can be used to enforce some LDAP hardening mechanisms from the client side, but ultimately the hardening must also be configured into AD. You can choose to use LDAPS or LDAP signing and optionally LDAP sealing. LDAP signing and sealing are not supported when using LDAPS, because TLS already ensures data integrity and confidentiality and by that already fills LDAP signing requirements.

To enforce using TLS/LDAPS, set the "*RequireLdaps"* setting to "*true"*.

To enforce using LDAP signing without TLS, set the "RequireLdapSigning" setting to "true".

To enforce using LDAP sealing without TLS, set the "RequireLdapSealing" setting to "true".

To use LDAP channel binding, AD must be configured to use channel binding and ADRS must be configured to use LDAPS and run on a machine that supports channel binding.

Note! By default, ADRS supports LDAP signing even when the corresponding setting "*RequireLdapSigning"* is set to "*false"*. This setting can be used to enforce the feature to be on but setting it to false does not guarantee it is not used. If AD requests using this feature, ADRS will comply even when it is not required by the setting.