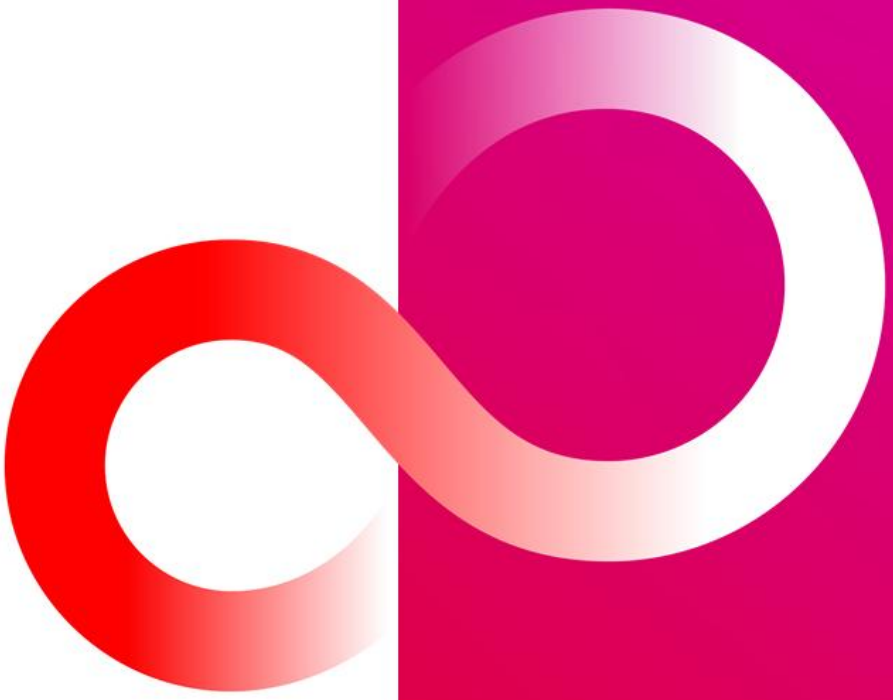
Fujitsu mPollux DigiSign Client



AD Registration Service

User's Guide

Contents

1. Introduction

Microsoft announced that "CVE-2022-34691, CVE-2022-26931 and CVE-2022-26923 address an elevation of privilege vulnerability that can occur when the Kerberos Distribution Center (KDC) is servicing a certificate-based authentication request. Before the May 10, 2022 security update, certificate-based authentication would not account for a dollar sign ($) at the end of a machine name. This allowed related certificates to be emulated (spoofed) in various ways. Additionally, conflicts between User Principal Names (UPN) and sAMAccountName introduced other emulation (spoofing) vulnerabilities that we also address with this security update."

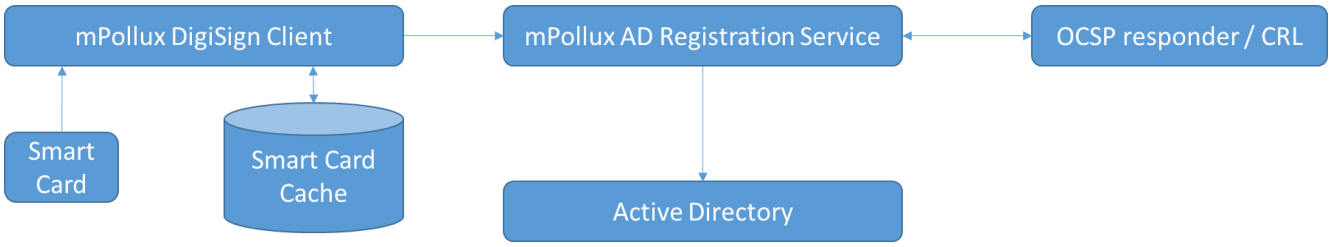
This change causes practical problems to existing domain installations that uses smart card for domain logon. To resolve this issue, Fujitsu has prepared AD registration service that works seamlessly with mPollux DigiSign Client Smart Card Middleware and domain AD.

1.1 Architecture overview

Registration service contains client and server-side parts. Registration is initiated by client side DigiSign Client. On server side, AD registration service receives registration request, validates certificate and user credentials. If validation succeeds, certificate details are written to the AD via LDAP protocol.

Certificate registration feature is supported by default DigiSign Client installation package starting from version 4.2.4.

Following picture illustrates components and information flow of certificate registration:



1.2 AD Registration Service

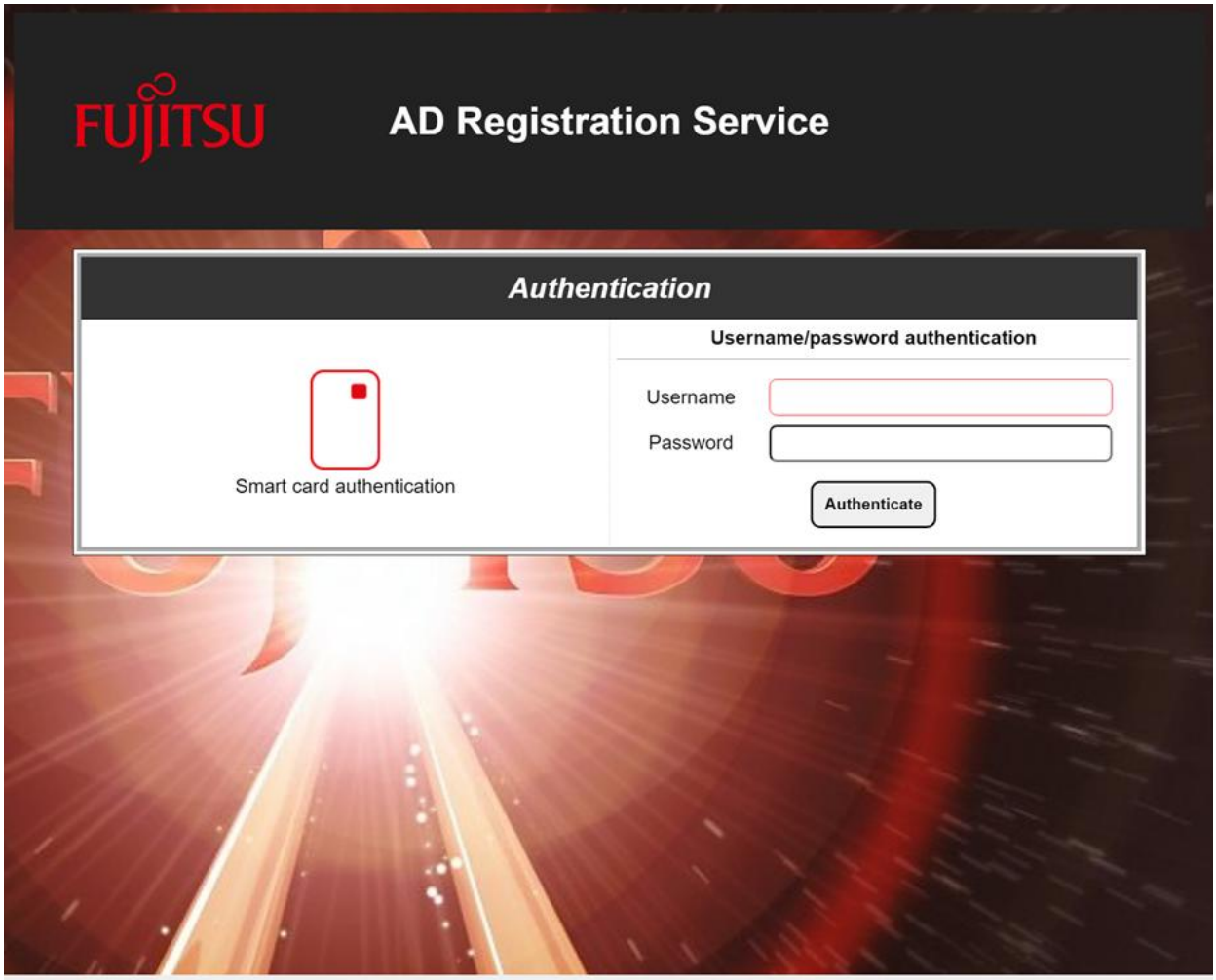
AD Registration service is implemented by using Fujitsu Finland's "mPollux WebServer" software. It contains two functional parts;

- DataServer
- WebServer

In this case, DataServer is configured to use SQLite database. AD Registration Service stores all registered certificates and application logons into database.

WebServer can be configured to run different applications. In this case, WebServer is delivered together with "AD Registration Service" application and this documentation concentrates only certificate registration functionality.

WebServer can be accessed via normal web browser. User can use AD credentials or smart card to logon into portal. Portal itself is quite simple. Administrators can see all registered certificates and audit log. Normal user can see only own registered certificate.



2. Setting up AD Registration Service

Both client and server side requires some configuration. This chapter introduces all components that needs to be configured in general. Chapter 3 introduces more details about server side configuration.

2.1 DigiSign Client Configuration

Client side registration requires only one additional register setting under "Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\DigiSign Client";

(REG_SZ) cidpRegisterUserCert

Value contains registration server's URL, for example http://register.dev.local:4000/ or https://register.dev.local:4001/ for TLS protected communication.

NOTE: If you want to allow registrations only for certificates with specific domain, add the allowed domain name to the (REG_SZ) logonDomain registry setting. Domain name must be entered with '@', like for example "@domain.com". Please see mPollux DigiSign Client Technical References for more details.

NOTE2: Information about latest registered certificates is stored into "registeredCertificates.gen" file that is located under "C:\ProgramData\Fujitsu\DS_Client" folder. New registration is not sent if previous registration has been successful. However, there might be cases where registration needs to be resent to the server. To enable re-registration, just remove this file and insert smart card to the card reader.

2.2 AD Registration Service Configuration

Service configuration is done via two configuration files that are located under installation folder;

- DataServer.conf
- WebServer.conf

WebServer configuration requires proper configuration settings to work properly. It is important that following settings are made correctly before starting to use AD Registration Service:

1) Firewall opening
2) TLS configuration
3) AD configuration

2.3 Firewall openings

DataServer is configured to use ports 5000 for http and 5001 for https connection. It is highly recommended to close these ports from outside connections.

WebServer uses ports 4000 and 4001 to communicate web page and web service requests. It is highly recommended to configure TLS certificate to registration service to protect especially logon page communication when username and password is used.

2.4 Operating modes

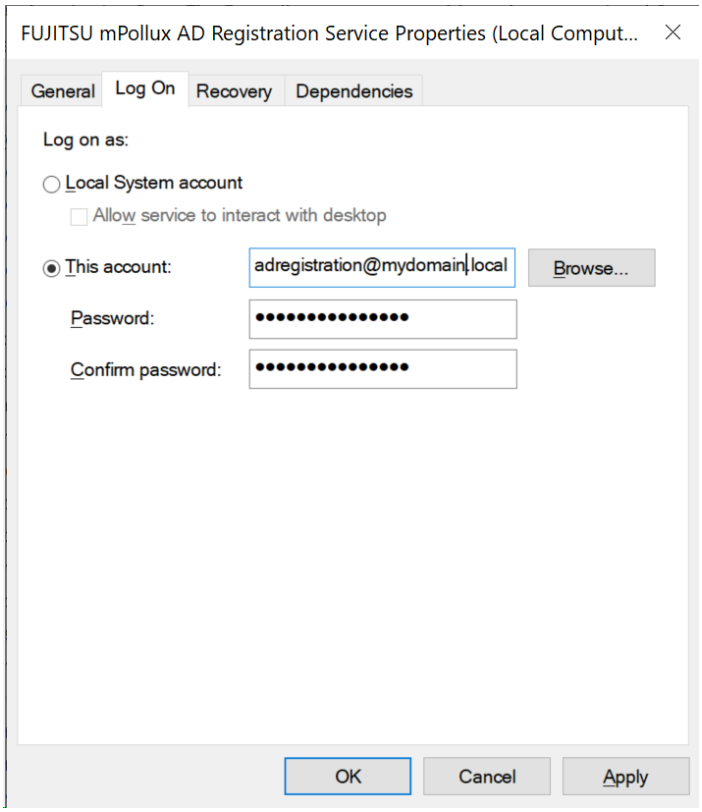
Service can be used in two different operating modes;

1) Store altSecurityIdentities to local database only
2) Store altSecurityIdentities to local database and update it into AD

When running in mode 1), only AD search credentials are required. When using second option, service needs to have proper credentials to be able to update required attribute value to AD.

IMPORTANT

If service is used to update altSecurityIdentities to AD,
DO NOT STORE AD DOMAIN USERNAME AND PASSWORD TO CONFIGURATION FILE
Instead, RUN SERVICE WITH PROPER DOMAIN USER ACCOUNT.
Please see chapter 3 for more details how to configure Kerberos authentication



3. How to configure AD Registration Service

Service configuration is done by editing WebServer.conf file with text editor. This chapter introduces all essential configuration options.

3.1 WebServer.conf details

WebServer.conf is located under installation folder. New settings takes effect only when WebServer's service is restarted.

Essential settings from AD Registration Service's point of view. Other settings are commented in the configuration file.

Defition	Explanation
serverAddress	Used when pages are loaded. If not set, FQDN us used
serverName	Used when generating CORS requests. If not set, FQDN us used.
adminGroup	AD group that elevates user to admin. Pre-configurated value is "Domain Admins"
headers	List of fixed headers that are sent to client. Use "|" to separate headers.
sessionLifeTime	Maximum session life time in minutes
maxConnections	Max. number of parallel connections
traceToFile	Application log that can be used for problem solving
requireHttps	If set to "true", WebServer doesn't accept http-requests
serverP12 serverP12password	Path and filename of TLS server certificate and keypair. Password of protected P12 package. Password can be "plain" or "encrypted". Please see chapter 3.2 how to encrypt passwords.
serverKey serverCertificate serverCertificateChain	Optional TLS credential configuration. If PKCS#12 package is not available. PEM-files can be configured via these parameters. Value should point to the existing file.
registeredCertificates	Path to the text file that will be appended when new certificates are registered.
trustedCertificates	List of trusted CA certificates. Please remove all untrusted/unused certificates from the list.
altSecurityIdentities	Attribute value generation method: 0 = Subject key identifier 1 = Certificate serial number and issuer
registrationType	Registration mode: 0 = Register certificate to database only 1 = Register certificate to database and ldap/AD

certificateValidation	In some cases, online certificate validation is not possible. If certificateValidation is et to "false", server validates only certificate issuer. Default value is "true" and certificate validity check is executed.
ldapBaseObject	Base object to search users. For example "DC=dev,DC=local"
ldapUrl	Url to access AD via ldap-protocol. For example "ldaps://dev-dc.local"
ldapUsername	Username that has search and "altSecurityIdentities" update credentials
ldapPassword	Ldap user's password. Value can be plain text or encrypted password. Please see chapter 3.2. for more details how to encrypt passwords.
userDomain	If specified, only allow the registration of certificates that include specific domain in their UPN. The user domain is defined as the part written after the '@' symbol. For example, to allow UPN registrations such as 'user@domain.com', set 'domain.com' in the userDomain configuration. Please refer to the 'userDomain' definition in the chapter 'DigiSign Client Configuration' for client side user domain configuration.

IMPORTANT

It is highly recommended to use Kerberos authentication to access AD.

To enable Kerberos authentication, leave *ldapUsername* and *ldapPassword* empty and define service to run with proper AD credentials.

IMPORTANT

It is highly recommended to configure TLS credentials to enable TLS communication.

This is done either by configuring
PKCS#12 (.p12 or .pfx) file and password
OR
Key and certificate PEM files
to WebServer.conf file.

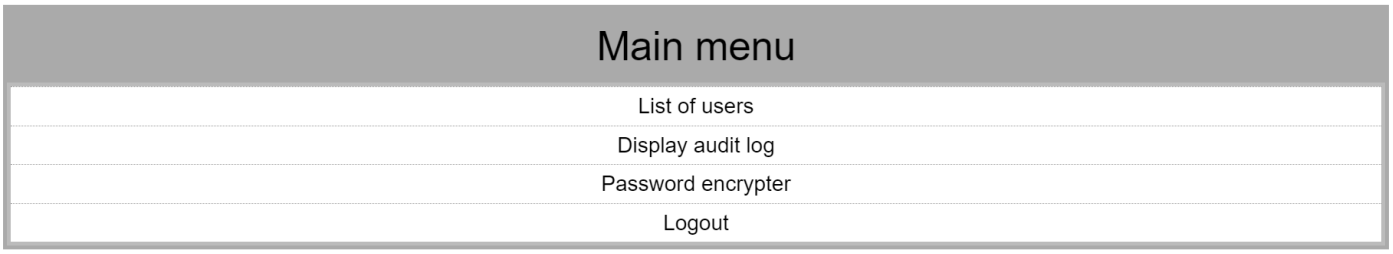
IMPORTANT

After everything works as expected, it is suggested to turn off application logging by commenting out "traceToFile" setting from WeServer.conf file.

Use tracing only when there is a need for problem solving.

3.2 How to encrypt passwords to WebServer.conf

When logging into AD Registration Service as Administrator, password encryption feature is displayed on the main page as follows:



When clicking "Password encrypter", following form is displayed:

Write password to "Plaintext password" field and click "Encrypt..."

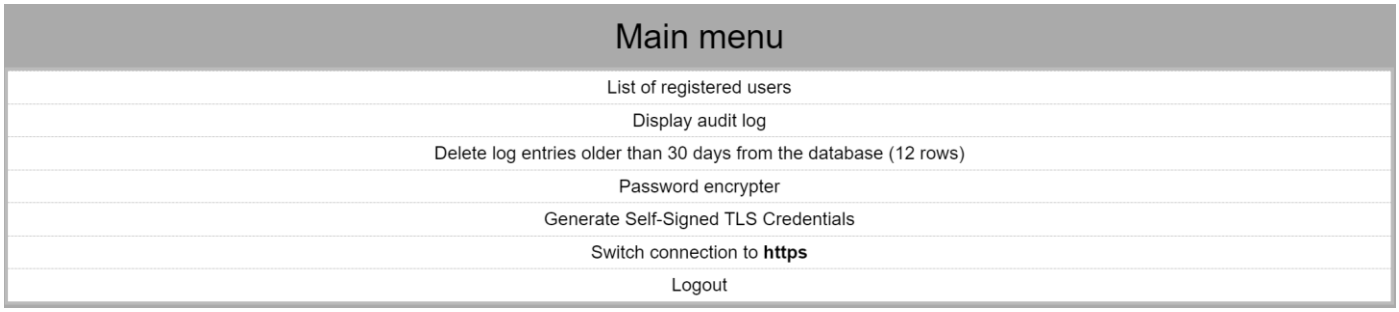
Encrypted password appears on the screen and now it can copy-pasted into configuration file.

Encryption is one way operation and only registration service can open it. If you need organization specific encryption keys, please inquiry organization specific license file from Fujitsu Finland.

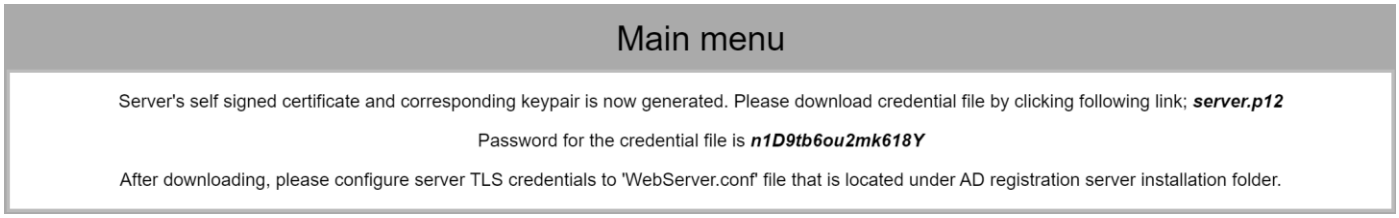
3.3 Generate self-signed TLS certificate

NOTE: Please use your domain CA to generate proper TLS-server credentials (keypair and corresponding certificate). Use this feature should be used only for test purposes.

If the TLS connection is not enabled, "Generate Self-Signed TLS Credentials" option will appear in the main menu.



When clicking “Generate Self-Signed TLS Credentials”, server will generate self-signed root certificate and “Root-CA” signed TLS server certificate. Root certificate’s keypair is not stored.



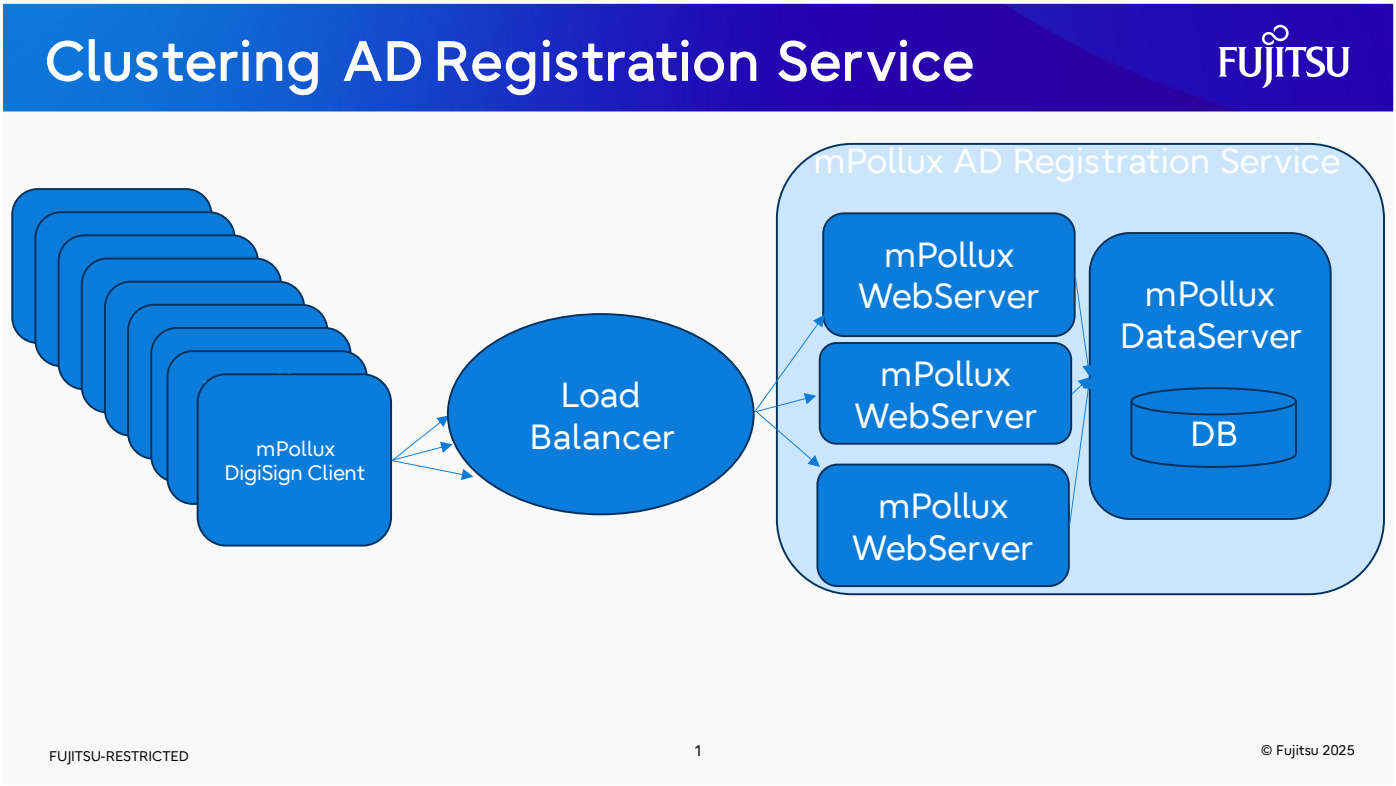
The credential file can be downloaded by clicking “server.p12” link. After downloading, copy the credential file into proper location and introduce it as TLS credentials in WebServer.conf file.

Please note that the server’s certificate chain is not trusted and your browser will complain that connection is not secure.

3.4 Clustering and High Availability for AD Registration Service

To ensure continuous operation, scalability, and fault tolerance, the AD Registration Service is designed to be deployed in a clustered configuration. This involves running multiple, stateless mPollux WebServers. A dedicated Load Balancer instance is crucial for this setup, serving as the single entry point for all client requests. The Load Balancer efficiently distributes traffic across the active WebServer instances based on configured rules and health checks. All WebServers then connect to and share a single, highly available mPollux DataServer (SQL Database) for all data persistence, ensuring consistency and resilience across the cluster.

The process throughput time is highly dependent on server configuration, especially regarding certificate checks and writing to Active Directory. Enabling all options increases processing time. Deploying to an entire group at once creates significant initial load, which stabilizes after registrations. Server capacity varies, but roughly one server can handle 3,000-6,000 users if registrations are staggered.



4. How to use Registration Service

4.1 Logon to the service

Main page offers two options to logon into registration service;

- SCS V1.1 based smart card authentication
- Username / password authentication

In both cases user credentials are validated against AD. If user belongs to admin-group, administrative features are displayed in the index page.

If user doesn't belong into admin-group, only personal registration information is displayed.

4.2 Maintenance

After successful installation and configuration AD Registration Service requires little maintenance. It is recommended to verify periodically that the service

- Is up and running
- It is not targeted as brute force attacks
- Certificate registration works as expected
    - Certificate validation works
    - New certificate registration requests are written to
        - database, registeredCertificates-file and optionally to AD
- Clean up transaction log from database when it becomes large
    - When there are older than 30 days log rows in database, log cleaning option will be displayed in the main menu.

4.3 Granting Permission to Update altSecurityIdentities

This section provides a high-level overview of how to configure Active Directory permissions to allow a designated security group to modify the altSecurityIdentities attribute on a user object. This is essential for services that interact with Active Directory Registration and need to update this specific attribute.

General Steps:

1. **Access ADSI Edit:** Launch ADSI Edit and connect to your domain's default naming context.
2. **Locate the Target User:** Navigate to the specific user account within Active Directory whose altSecurityIdentities attribute needs to be modifiable.
3. **Open Security Properties:** Access the security properties of that user account.
4. **Add the Delegated Group:** Incorporate the security group responsible for the update operations into the user's security permissions.
5. **Configure Advanced Permissions:** Utilize the advanced security settings to finely tune the permissions.
6. **Grant Specific Write Access:** Grant explicit "Write" permission *solely* for the altSecurityIdentities attribute to the previously added security group, ensuring all other write permissions remain ungranted.
7. **Apply and Save:** Confirm and save all permission changes to the user object.

APPENDIX A: Certificate registration request and response details

Registration Request

Registration request is a simple web service POST request. Message is json encoded and it contains following information:

```
{ "mode": "registrationRequest", "serialNumber": "18924600018401071979", "cert":
"MIIGPTCCBCWgAwIBAgIEEeNFATA..." }
```

Payload	Description
mode	Currently only supported mode is “registrationRequet”
serialNumber	Smart card serial number
cert	Base64 encoded certificate to be registered

Registration Response

Following JSON message will be returned after successful registration:

```
{ "cert_0": "<base64 encoded certificate>" }
```

Following JSON message will be returned if registration fails:

```
{ "error": "Detailed explanation of failure" }
```

