One Identity Manager 9.1.1

Password Capture Agent Administration Guide

Contents

The One Identity Manager Password Capture Agent

The Password Capture Agent allows you to synchronize user passwords between Active Directory domains managed by One Identity Manager and other connected target systems. The Password Capture Agent tracks changes to user passwords in the source Active Directory domain and provides that information to the web service, which in turn synchronizes the changes with connected target systems by using the password templates you specified. To synchronize passwords, you must install the Password Capture Agent on each domain controller in the Active Directory domain you want to use, as a source for the password synchronization operations.

The following diagram shows how the password synchronization feature of One Identity Manager works.

Figure 1: How the password synchronization feature works

Automated password synchronization

If your enterprise environment has multiple target systems, each with its own password policy and dedicated user-authentication mechanism, you may face one or more of the following issues:

- Because users must remember multiple passwords, they may have difficulty managing them. Some users may even write down their passwords. As a result, passwords can be easily compromised.

- Each time users forget one or several of their numerous access passwords, they must ask administrators for password resets. This increases operational costs and causes a loss of productivity.
- There is no way to implement a single password policy for all target systems. This impacts productivity, as users log on to each target system separately to change their passwords.

With One Identity Manager, you can eliminate these issues and significantly simplify password management in an enterprise environment that includes multiple target systems.

One Identity Manager provides a cost-effective and efficient way to synchronize user passwords from an Active Directory domain to other target systems used in your organization. As a result, users can access other target systems using their Active Directory domain password. Whenever a user password is changed in the source Active Directory domain, this change is immediately and automatically propagated to other target systems, so each user password remains in sync within the data systems at all times.

You must connect One Identity Manager to the target systems in which you want to synchronize passwords.

Related topics

Steps to automate password synchronization

> NOTE: The web service must be installed. For more information, see the *One Identity Manager Installation Guide* and the *One Identity Manager Configuration Guide*.

To automatically synchronize passwords from a Active Directory domain to another target system

1. Connect One Identity Manager to the Active Directory domain where you want to install the Password Capture Agent.
2. Connect One Identity Manager to the target system where you want to synchronize user account passwords with those in the source Active Directory domain.
3. Ensure that user accounts in the source Active Directory domain and the connected target system are properly mapped to employees in One Identity Manager.

   For general information on how to assign employees to user accounts, see the *One Identity Manager Target System Base Module Administration Guide*. For more information on how to assign employees to Active Directory user accounts, see the *One Identity Manager Administration Guide for Connecting to Active Directory*.

4. Install the Password Capture Agent on each domain controller in the Active Directory domain you want to have as a source for password synchronization operations.

   The Password Capture Agent tracks changes to user passwords in the source Active Directory domain and provides this information to the web service, which in turn synchronizes passwords in the connected target system you specify.

After you have completed the above steps, the Password Capture Agent starts to automatically track user password changes in the source Active Directory domain and One Identity Manager synchronizes passwords in the connected target system.

If necessary, you can fine-tune password synchronization settings by completing these optional tasks:

- Modify the default Password Capture Agent settings before installation.
- Modify the default web service settings related to password synchronization.
- Specify a custom certificate for encrypting the password synchronization traffic between the Password Capture Agent and the web service. By default, password synchronization traffic between the Password Capture Agent and the web service will be secured by transport layer security only.

Related topics

- Managing the Password Capture Agent on page 8
- Configuring Password Capture Agent on page 12
- Specifying a custom certificate for encrypting password synchronization traffic on page 25

Managing the Password Capture Agent

The Password Capture Agent is required to track changes to user passwords in the Active Directory domain that you want to be the authoritative source for password synchronization operations. To synchronize passwords, you must install the One Identity Manager Password Capture Agent on each domain controller in the source Active Directory domain.

Whenever a password changes in the source Active Directory domain, the Password Capture Agent captures that change and sends the changed password securely to One Identity Manager. In turn, One Identity Manager uses the provided information to synchronize passwords in the connected target systems according to your settings.

Detailed information about this topic

- System requirements for Password Capture Agent on page 8
- Installing the Password Capture Agent on page 9
- Using Windows PowerShell to install the Password Capture Agent on page 9
- Uninstalling the Password Capture Agent on page 10
- Using Windows PowerShell to uninstall the Password Capture Agent on page 11

System requirements for Password Capture Agent

The following are the minimum system requirements for installing and operating the Password Capture Agent.

- Windows operating system

  Following versions are supported:

- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2
- Windows Server 2012
- Microsoft .NET Framework Version 4.8 or later

Installing the Password Capture Agent

You can use this method to manually deploy the Password Capture Agent on each domain controller in the source Active Directory domain.

To manually install the Password Capture Agent

1. On a 64-bit domain controller, run the `One Identity Manager Password Capture Agent.msi` file.

   You can find the `One Identity Manager Password Capture Agent.msi` file on the One Identity Manager installation medium in `Modules\ADS\dvd\AddOn\PasswordCaptureAgent`.

2. Use the wizard to complete the Password Capture Agent installation.

Using Windows PowerShell to install the Password Capture Agent

The Password Capture Agent provides a Windows PowerShell module for remote and automated installation, configuration, and uninstall. You can use this method to automatically deploy the Password Capture Agent on each domain controller in the source Active Directory domain.

For installing the Password Capture Agent remotely you should have prepared:

- The thumbprint of the certificate for password encryption, for example:

  `1800b62e8cf19d1c4bcdcd2b6e435c3c85e04188`

- The URL of the web service, for example:

  `https://<servername.domain.com>/AppServer/`

Use the following commands in an elevated Windows PowerShell.

`Import-Module OneIM-PasswordCaptureAgentMgmt`

`$ConfigProfile = New-PCAConfigProfile`

`$ConfigProfile['WebClient.WebServiceURL'].ConfigValue = '<Your URL>'`

```
$ConfigProfile['WebClient.WebServiceType'].ConfigValue = 'REST'

$ConfigProfile['WebClient.AuthenticationType'].ConfigValue = 'WindowsIntegrated'

$ConfigProfile['Backend.AuthenticationModule'].ConfigValue = 'DialogUser'

$ConfigProfile['Backend.Credential'].ConfigValue = Get-Credential viCaptureAgent

$ConfigProfile['Backend.CertificateThumbprint'].ConfigValue = '<Your Thumbprint>'

Install-PasswordCaptureAgent`

-ComputerName <Computer name>`

–LogFile <Full UNC path to the log file on the remote server>`

-LogVerbose`

-Setup <UNC path for One Identity ManagerPassword Capture Agent MSI>`

-ConfigurationProfile $ConfigProfile
```

> NOTE: To check that the Password Capture Agent is properly installed and working, you can examine the event viewer on the deployed server. The Password Capture Agent has its own log in the event viewer. The Password Capture Agent logs its summary status to this log after every system start and other such notable events during runtime.

Related topics

- The Password Capture Agent Windows PowerShell module on page 29
- Event log for the Password Capture Agent on page 37

Uninstalling the Password Capture Agent

To remove the Password Capture Agent open the list of installed programs on the computer on which the Password Capture Agent is installed.

To remove Password Capture Agent using the control panel

1. Select **Programs and Features** in the Control Panel.
2. Double-click **One Identity Manager Password Capture Agent** in the list of installed programs.
3. Follow the on-screen instructions to uninstall the Password Capture Agent.

Using Windows PowerShell to uninstall the Password Capture Agent

The Password Capture Agent provides a Windows PowerShell module for remote and automated installation, configuration, and uninstall. You can use this method to automatically uninstall the Password Capture Agent on each domain controller in the source Active Directory domain.

For uninstalling the Password Capture Agent remotely, use the following command in an elevated Windows PowerShell.

```
Import-Module OneIM-PasswordCaptureAgentMgmt

Uninstall-PasswordCaptureAgent`

-ComputerName <Computer name>`

-LogFile <UNC path to log file>`

–LogVerbose
```

Related topics

- The Password Capture Agent Windows PowerShell module on page 29

Fine-tuning automated password synchronization

This section provides information about the optional tasks related to configuring automated password synchronization from an Active Directory domain to connected target systems.

Detailed information about this topic

Configuring Password Capture Agent

The Password Capture Agent has several settings you can modify. After you install the Password Capture Agent, each of its parameters is assigned a default value.

NOTE: If you do not configure the thumbprint for the Password Capture Agent, the password is secured by transport layer security only (HTTPS).

Detailed information about this topic

Registry configuration parameters

Some of the configuration parameters for the Password Capture Agent can be changed using the Windows Registry Editor. The parameters are split up into those used by the Password Capture Agent service and those used by the Password Capture Agent driver.

Registry configuration parameters for the Password Capture Agent service

The base path for the parameters of the Password Capture Agent service is:

`HKLM\SOFTWARE\One Identity\One Identity Manager\Password Capture Agent\Service\`

`WebService_URL`

> This setting determines the location - Uniform Resource Locator (URL) - of the web service to which the Password Capture Agent provides information about changed user passwords.
>
> Syntax: https://<serverfqdn>/AppServer/
>
> Type: REG_SZ
>
> Values: URL of the web service
>
> Default: (empty)

`CertificateThumbprint`

> This setting specifies a certificate used to encrypt the data transfer channel between the Password Capture Agent and the web service. The certificate must be accessible both for the Password Capture Agent and the web service.
>
> Type: REG_SZ
>
> Values: Certificate used to encrypt the password befor submiting to the web service.
>
> Default: (empty)
>
> NOTE: If you disable this setting or do not configure it, the password will be secured by transport layer security only (HTTPS).

`EncryptedPasswordTransmission`

> This setting specifies whether the password is encrypted when being sent to the web service. Requires the `CertificateThumbprint` parameter to be set.
>
> Type: DWORD
>
> Values: 0 | 1 - Disables or enables encrypted password transmission.
>
> Default: 1

`EncryptedPasswordTransmissionSigning`

> This setting specifies whether the password is signed after encryption, when being sent to the web service. Requires the `CertificateThumbprint` parameter to be set to a certificate with private key and the `EncryptedPasswordTransmission` parameter to be enabled.

Type: DWORD

Values: 0 | 1 - Disables or enables signed and encrypted password transmission.

Default: 1

Registry configuration parameters for the Password Capture Agent driver

The base path for the parameters of the Password Capture Agent driver is:

`HKLM\SOFTWARE\One Identity\One Identity Manager\Password Capture Agent\Driver\`

| NOTE: No reboot is required to take effect.

`DeactivateOnStart`

Disables the Password Capture Agent without uninstalling. If the value is set to **1**, the Password Capture Agent is disabled after the next reboot. The only action after reboot is a single hint, logged to the Password Capture Agent event log - named One Identity Manager Password Capture Agent - in the Windows Event Viewer.

Type: REG_DWORD

Values: 0 | 1

Default: 0

`Diagnostic`

Enables some diagnostic behavior if this parameter is set to **1**.

- Verbose logging to log file if it is specified (`LogFile` parameter). Every operation and its result is logged.

- All logs are also sent as an operating system debug message for appropriate live viewers (for example, `DebugView` from Windows Sysinternals).

- The `LogFile` parameter is enabled.

Type: REG_DWORD

Values: 0 | 1

Default: 0

`FaultToleranceWaitTimeBeforeRetryInSeconds`

If an error occurs, the value specified is the wait time in seconds before retrying. If the value is **0**, a retry is run immediately.

Type: REG_DWORD

Values: Time in seconds

Default: 120

`Logfile`

Specifies a name for a log file that must be created. If no value is specified, no log file is created. Only the file name, without a path, needs to be specified, so the file will reside in the `%ProgramData%\One Identity\One Identity Manager\Password Capture`

`Agent\Driver` installation folder.

The log file logs all activities, and more details if the `Diagnostic` parameter is enabled. The log file is read-only but can be accessed from any text viewer. It is always recreated on reboot and does not yet contain any history. The time format of the logged time stamps depends on the local language of the operating system and not on the user.

Type: REG_SZ

Values: File name (without a path)

Default: (empty)

`LoggingSuccessfulOperations`

Enable to force the One Identity Manager to log successful transmissions to the web service to the event log.

Type: REG_DWORD

Values: 0 | 1

Default: 0

`RequiredServices`

Services that the Password Capture Agent driver is waiting for, before starting the Password Capture Agent service.

Type: REG_MULTI_SZ

Values: List of services

Default: RpcSs EventSystem COMSysApp

`PendingCapturesArchiveDepthInDays`

Specifies the number of days for undelivered password changes to be saved for retrying. Undelivered password changes can arise if errors have occurred: for example, if the associated web service is not available due to network errors, timeouts, and so on. Every password change that cannot be delivered is also logged to the Password Capture Agent event log in Windows Event Viewer. If **0** is specified, no undelivered password changes are saved; they will be lost.

Type: REG_DWORD

Value: Number of days

Default: 7

`Synchronous`

If this parameter is set with a value of **1**, every password change is handled sequentially. As a result, the initialization process is blocked until all other components in the beyond-processing chain are complete. All password change events that occur in parallel are also blocked until the current password change is complete. This setting also means that a user who only changes their password in the password-change-dialog must wait until the entire processing is complete. This setting is only for test purposes.

Type: REG_DWORD

Values: 0 | 1

Default: 0

`Ignoring\PasswordResetOperations`

Enable to force One Identity Manager to ignore password resets and only transmit password changes to the One Identity Manager Service.

Type: REG_DWORD

Values: 0 | 1

Default: 0

`Ignoring\UserNames`

Specifies a list of names of accounts that are to be ignored and whose password changes are irrelevant and are not to be tracked. It can be built-in accounts, such as machine accounts and guest accounts, or other operating system-related accounts, such as virtual machine accounts. Every account in this list is specified as a regular expression. The default is the machine account (**^.*$$**), which is to be ignored.

Type: REG_MULTI_SZ

Values: List of account names as regular expressions

Default: ^.*$$

`Ignoring\UserRids`

Specifies a list of User-RIDs (relative part of a user SID number) that are to be ignored and whose password changes are irrelevant and are not to be tracked. These are built-in accounts, such as machine accounts and guest accounts. Every account in this list is specified as a User-RID. RIDs of built-in accounts are the same on every machine. The default for this parameter is the RID of the built-in administrator account (**500**), the RID of the built-in guest account (**501**), and the RID of the built-in Kerberos ticket-granting ticket account (**502**).

Type: REG_MULTI_SZ

Values: List of numbers

Default: 500 501 502

Secured configuration parameters

The configuration parameters in this section are secured using the Microsoft Cryptography API and are not directly accessible. If you want to change or review these parameters after installing the Password Capture Agent installation, use either the `Set-ServiceConfig.exe` command line or the Password Capture Agent Windows PowerShell module.

The command line is supplied with the Password Capture Agent and is located in the Password Capture Agent installation folder `...\Service`.

> **Example: local**
>
> ```
> "%ProgramFiles%\One Identity\One Identity Manager\Password Capture
> Agent\Service\Set-ServiceConfig.exe" WebServiceClientSkipHttpsValidation:0
> ```

NOTE: Retrieving secured configuration parameters requires a privileged user account. The process used to query for secured configuration parameters must be elevated to retrieve parameter values.

Secured configuration parameters for Password Capture Agent

WebServiceType

> Specifies whether the web service should be accessed using the One Identity Manager application server (REST) or the One Identity ManagerSOAP Web Service (Soap).
>
> It is strongly recommended you use the One Identity Manager application server. The One Identity ManagerSOAP Web Service support is only included for backward compatibility to One Identity Manager version 6.x and should not be used anymore.
>
> Values: REST | Soap
>
> Default: REST

WebServiceClientSkipHttpsValidation

> If **1** (enabled), HTTPS connections are established without validation.
>
> This is potentially unsecured and should never be used in production.
>
> Values: 0 | 1
>
> Default: 0

WebServiceClientCredentialType

> Specifies if the authentication against the Internet Information Services (IIS) should use Windows integrated authentication or certificate based authentication.
>
> Values: WindowsIntegrated | Certificate
>
> Default: WindowsIntegrated

WebServiceClientCredentialCertificateFindByType

> Specifies how to search for the authentication certificate. Used in combination with WebServiceClientCredentialType=Certificate.
>
> Values: All values of the X509FindType-enumeration are allowed.
>
> Default: FindByThumbprint

WebServiceClientCredentialCertificate

> Finds the certificate based on the find type defined in the WebServiceClientCredentialCertificateFindByType parameter. Used in combination with WebServiceClientCredentialType=Certificate.

BackendClientCredentialType

Specifies how to authenticate against One Identity Manager. **WebADS** and **ADSAccount** reuse the Windows credentials used for authentication against IIS.

- ADSAccount = One Identity Manager 7.x or later
- WebADS = One Identity Manager 6.1.x

Values: DialogUser | WebADS | ADSAccount

Default: DialogUser

BackendClientCredentialUserName

Specifies a system user for the authentication against One Identity Manager. Used in combination with `BackendClientCredentialType=DialogUser`.

Default: viCaptureAgent

BackendClientCredentialUserPwd

Specifies the password of the system user used for authentication against One Identity Manager. Used in combination with `BackendClientCredentialType=DialogUser`.

NOTE: `BackendClientCredentialUserPwd` is a write-only parameter. The currently configured value cannot be retrieved using `Set-ServiceConfig`.

BackendClientCredentialUserPwd_AcceptEmpty

Required if your system user uses a blank password. This is potentially unsecured and should never be used in production. Used in combination with `BackendClientCredentialType=DialogUser`.

Values: 0 | 1

Default: 1

Example: Retrieve information about a secured configuration parameter

"%ProgramFiles%\One Identity\One Identity Manager\Password Capture Agent\Service\Set-ServiceConfig.exe" Describe:WebServiceClientCredentialType

Configuration parameter 'BackendClientCredentialType':

Name: BackendClientCredentialType

Possible values: DialogUser;WebADS;ADSAccount

Default value: DialogUser

Corresponding installer property: PROP_BACKEND_CLIENT_CREDENTIAL_TYPE

Description: Specify one of the credential types for authentication against the One Identity Manager

Present in installer GUI: Yes

Write only (read out not allowed): No

```
Read only (setting not allowed): No

Public in registry: No

Hint:

Comment:
```

Example: Retrieving a secured configuration parameter

```
"%ProgramFiles%\One Identity\One Identity Manager\Password Capture
Agent\Service\Set-ServiceConfig.exe" Get:WebServiceClientCredentialType

WebServiceClientCredentialType=Certificate

Value was written to stderr.

Get configuration parameter - operation done.
```

Related topics

- Authentication against the web service on page 19
- Authentication against One Identity Manager on page 20
- The Password Capture Agent Windows PowerShell module on page 29
- Certificate lookup options on page 49

Authentication options

The One Identity Manager Password Capture Agent supports several authentication options that can be configured separately for authentication against the IIS hosting the web service and for the authentication against the One Identity Manager database.

Detailed information about this topic

- Authentication against the web service on page 19
- Authentication against One Identity Manager on page 20

Authentication against the web service

Authentication against the web service can be configured with the secured WebServiceClientCredentialType parameter.

Permitted values are:

- **WindowsIntegrated**: Uses the credentials of the user running the Password Capture Agent service to authenticate against the IIS hosting the web service. By default, this is the **Local System** user that uses the machine account to authenticate over the network. You can change the user of the Password Capture Agent service. The user requires administrative privileges to access the configuration parameters.

- **Certificate**: Uses a certificate to authenticate against the IIS hosting the web service. The certificates are searched in `Cert:\CurrentUser\My\` and, if not found there, are searched in `Cert:\LocalMachine\My\`. Ensure that the user running the Password Capture Agent service has permissions to access the private key of the certificate.

Related topics

- Secured configuration parameters on page 16
- Certificate lookup options on page 49

Authentication against One Identity Manager

Authentication against the One Identity Manager database can be configured with the secured `BackendClientCredentialType` parameter.

Permitted values are:

- **DialogUser**: The One Identity Manager Service uses the credentials stored in the `BackendClientCredentialUserName` parameter and the `BackendClientCredentialPwd` parameter to log in as a One Identity Manager system user.

  You can test your configuration by running the Object Browser with the system user login.

- **ADSAccount**: This option uses the credentials of the user running the Password Capture Agent service to authenticate against the One Identity Manager database. This option works for One Identity Manager version 7.x or later.

  NOTE: The user account must be synchronized by the One Identity Manager database and needs to be linked to an employee whose system user property is set accordingly. A machine account will not be able to authenticate against the One Identity Manager database.

  You can test your configuration by running the Object Browser with the same credentials as the Password Capture Agent service and using the Active Directory user account login.

- **WebADS**: This option behaves the same as **ADSAccount** but also works for One Identity Manager version 6.1.x.

Example: Windows authentication and One Identity Manager system user login

The Password Capture Agent service uses Windows authentication to authenticate against the IIS with the web service running. To authenticate against One Identity Manager, the system user **viCaptureAgent** is used.

- Prerequisites

  Configure the IIS site to only use Windows authentication for the web service.

- Testing

  You should be able to access the web service with a browser and the given WindowsActive Directory user account. Start a Windows PowerShell and try to access the web service using the given user account.

  ```
  Invoke-WebRequest -Uri https://<servername.domain.com>/AppServer/ -
  Credential $(Get-Credential <AD domain>\<AD user account>)
  ```

  You should be able to log into the Object Browser using the system user login and the credentials provided.

- Password Capture Agent configuration settings
  - WebServiceClientCredentialType = WindowsIntegrated
  - BackendClientCredentialType = DialogUser
  - BackendClientCredentialUserName = viCaptureAgent
  - BackendClientCredentialUserPwd = viCaptureAgentPasswordHere

---

Example: Windows authentication and Active Directory login

The Password Capture Agent service uses Windows authentication to authenticate against the IIS with the web service running. The Windows user account used to authenticate against the IIS will be reused to authentication against One Identity Manager.

- Prerequisites
  - Configure the IIS site to only use Windows authentication for the web service.
  - Configure IIS site to allow given users to access the web service (authorization).
  - The Password Capture Agent service is not allowed to run as **Local System** and requires an administrative user account to run with.

- Given user accounts must be known to the One Identity Manager database and must be linked to an employee who has a system user configured to use for this type of authentication.

- Testing

  You should be able to access the web service with a browser and the given Active Directory user account. Start a Windows PowerShell and try to access the web service using the given user account.

  ```
  Invoke-WebRequest -Uri https://<servername.domain.com>/AppServer/ -
  Credential $(Get-Credential <ADDomain>\<ADUser>)
  ```

  You can test your configuration by running the Object Browser as the given user account and using the Active Directory user account login.

- Password Capture Agent configuration settings
  - WebServiceClientCredentialType = WindowsIntegrated
  - BackendClientCredentialType = ADSAccount

Example: Certificate authentication and One Identity Manager system user login

This scenario allows you to connect from a host outside of your Active Directory domain. Stored credentials will be used to authenticate against One Identity Manager as system user.

- Prerequisites
  - Configure the IIS site to use HTTPS and Client Certificate Mapping. If you are not using Active Directory Certificate Services, you need to map the certificate to an Active Directory user account within IIS.
  - Client certificate with private key installed on the domain controller.

- Testing

  You should be able to access the web service with a browser using the given certificate. Start a Windows PowerShell as the user with the assigned certificate and try to access the web service.

  ```
  Invoke-WebRequest -Uri https://<servername.domain.com>/AppServer/ -
  CertificateThumbprint <ThumbprintOfGivenCertificate>
  ```

  You should be able to log into the Object Browser using the system user login and credentials.

- Password Capture Agent configuration settings

- WebServiceClientCredentialType = Certificate
- WebServiceClientCredentialCertificateFindByType = FindByThumbprint
- WebServiceClientCredentialCertificate = 0123456789ABCED0123456789ABCED0123456789
- BackendClientCredentialType = DialogUser
- BackendClientCredentialUserName = viCaptureAgent
- BackendClientCredentialUserPwd = viCaptureAgentPasswordHere

Related topics

-

Password

To change the password used to authenticate against One Identity Manager, use either the `Set-ServiceConfig.exe` command line or the Password Capture Agent Windows PowerShell module.

The command line is supplied with the Password Capture Agent and is located in the Password Capture Agent installation folder `...\Service`.

NOTE: The Password Capture Agent must be configured to use the `BackendClientCredentialType` parameter with the `DialogUser` value.

Example: local

```
"%ProgramFiles%\One Identity\One Identity Manager\Password Capture
Agent\Service\Set-ServiceConfig.exe" BackendClientCredentialUserPwd:<new
password>
```

The command line can also be used to set the password on a remote server on which the Password Capture Agent is installed. Use the optional `Servername` parameter to specify the name or the IP address of the remote server. In this case, COM+ Network Access must be enabled on the remote server in the application server role. If it is not enabled, see the Microsoft documentation to enable it.

> **Example: remote**
>
> ```
> "%ProgramFiles%\One Identity\One Identity Manager\Password Capture
> Agent\Service\Set-ServiceConfig.exe" BackendClientCredentialUserPwd:<new
> password> Servername: <Server name or IP address>.
> ```

NOTE: It is not required to restart the Password Capture Agent service. The new password takes effect immediately.

Related topics

- The Password Capture Agent Windows PowerShell module on page 29

Delete processes

The Password Capture Agent manages a queue with the password change processes that are sent to One Identity Manager. If you need to delete some of these processes from the internal queue, use the `Set-ServiceConfig` command line.

> **Example: local**
>
> ```
> "%ProgramFiles%\One Identity\One Identity Manager\Password Capture
> Agent\Service\Set-ServiceConfig.exe" DeleteJob:<Job-ID>::=<YYYY.MM.DD
> HH.MM.SS.mmm>|*
> ```
>
> Sample for a certain Job-ID: '2014.10.03 16:45:07.647'
>
> ```
>     Set-ServiceConfig.exe DeleteJob:"2014.10.03 16:45:07.647"
> ```
>
> To delete all processes use ***** as the Job-ID.
>
> ```
>     Set-ServiceConfig.exe" DeleteJob:*
> ```

Logging with NLog

Starting with version 2.0, the Password Capture Agent uses NLog for logging. NLog allows logging to be configured with an XML file.

By default, an `nlog.config` in the Password Capture Agent installation folder is provided, which uses the same event log as previous versions.

This `nlog.config` also provides additional examples of how to configure NLog to log directly to a file or other tools, such as chainsaw. You can enable these by uncommenting the matching rules in the rules section of the `nlog.config`.

More detailed examples of how to configure NLog can be found here: https://nlog-project.org/.

| NOTE: A faulty `nlog.config` will cause the Password Capture Agent to stop logging.

Configuring the web service

You can modify the default values of the following configuration parameters related to password synchronization in the Designer.

Table 1: Configuration parameters and default values

Configuration parameter	Description
QER | Person | UseCentralPassword | PasswordCaptureAgent | Certificate	Specifies if a certificate is used to encrypt the password synchronization traffic between the Password Capture Agent and the web service. Default value: enabled.
QER | Person | UseCentralPassword | PasswordCaptureAgent | Certificate | SignAndEncrypt	Specifies if a certificate is used to sign the encrypted password synchronization traffic between the Password Capture Agent and the web service. Default value: enabled.

| IMPORTANT: Passwords for user accounts marked as privileged in One Identity Manager are not synchronized with other connected target systems.

| TIP: If you have configured more than one Active Directory domain or have employees with more than one user account, use the Password Capture Agent to check your password policy for employee's central password. To avoid circular password resets, the password history value should be **1** or greater.

Specifying a custom certificate for encrypting password synchronization traffic

By default, the password synchronization traffic between the Password Capture Agent and the web service is secured by transport layer security only. Therefore, it is strongly recommended that you specify a custom certificate.

| IMPORTANT: You need a certificate file including the private key to encrypt password synchronization traffic.

Detailed information about this topic

Step 1: Import certificate into certificates store

In this step, import the certificate to the `Personal\Certificates` machine certificate store by using the Certificates snap-in. You must complete this step on each domain controller running the Password Capture Agent and on each computer running the web service that will participate in password synchronization.

To import the certificate

1. Open the Certificates - Local Computers snap-in.
2. In the console tree, click **Personal | Certificates**.
3. On the **Action** menu, point to **All Tasks** and click **Import**.
4. Use the wizard.

   a. On the **File to Import** page, in **File name**, enter the file name containing the certificate to be imported, or click **Browse** to locate and select the file. When finished, click **Next**.

   b. On the **Password** page, enter the password used to encrypt the private key, and click **Next**.

   c. On the **Certificate Store** page, ensure that **Place all certificates in the following store** is selected and that **Certificate store** displays **Personal**. Then click **Next**.

   d. On the **Completion** page, revise the specified settings and click **Finish**.

To add read permissions to the certificate for the web service

1. Open the Certificates - Local Computers snap-in.
2. In the console tree, click **Personal | Certificates**.
3. Select your imported certificate from the list.
4. On the **Action** menu, point to **All Tasks** and click **Manage Private Keys**.
5. Add **Read Permissions** for the **Network Service** security principal and click **OK**.

Related topics

Step 2: Copy certificate's thumbprint

Copy the thumbprint of your custom certificate. (In the next step, you will need to provide the thumbprint to the Password Capture Agent.)

To copy the thumbprint of your custom certificate

1. Open the Certificates - Local Computer snap-in.

2. In the console tree, click **Personal**

3. Click **Certificates**.

4. In the details pane, double-click the certificate.

5. In the **Certificate** dialog, click **Details**, and scroll through the list of fields to select **Thumbprint**.

6. Copy the hexadecimal value of thumbprint to the clipboard.

NOTE: You will need the copied thumbprint value to configure the Password Capture Agent.

Related topics

Step 3: Provide certificate's thumbprint to the Password Capture Agent

This step assumes that the Password Capture Agent Windows PowerShell module for the Password Capture Agent is installed on your workstation and all other requirements are met.

To provide the thumbprint to the Password Capture Agent

1. Sign on to the workstation installed with Password Capture Agent Windows PowerShell module as member of the **Domain Admins** group.

2. Open an elevated command line.

3. Run the following command to modify the configuration profile with the new thumbprint:

    ```
    REG ADD "\\<COMPUTERNAME>\HKLM\Software\One Identity\One Identity
    Manager\Password Capture Agent\Service" /v "CertificateThumbprint" /t REG_SZ
    /d "1800b62e8cf19d1c4bcdcd2b6e435c3c85e04188"
    ```

4. Run the following commands to restart the Password Capture Agent service:

    ```
    sc \\COMPUTERNAME stop "Password Capture Agent"
    ```

```
sc \\COMPUTERNAME start "Password Capture Agent"
```

Related topics

- Step 1: Import certificate into certificates store on page 26
- Step 2: Copy certificate's thumbprint on page 27
- The Password Capture Agent Windows PowerShell module on page 29

The Password Capture Agent Windows PowerShell module

The Password Capture Agent Windows PowerShell module was designed to simplify the setup and management of the Password Capture Agent on domain controllers. This module requires Windows PowerShell remoting to be configured and enabled on the domain controllers to establish a connection and run the commands.

This Windows PowerShell module is intended for use on a Windows workstation with Windows PowerShell version 3.0 or later installed and while logged on with a domain account that is in the **Domain Admins** built-in group. The Password Capture Agent installer must be placed on a shared network or copied manually to all domain controllers.

To allow administrators to better check for configuration errors, we integrated some validations into the functions that will display warnings on any possible misconfiguration: for example, if the password encryption certificate is not installed.

NOTE: This module does not install the web service. This module also does not generate and install the certificate required to encrypt passwords sent to the web service.

Prerequisites

The Password Capture Agent Windows PowerShell module has different requirements for the workstation or server the module is running on and for the domain controllers where the Password Capture Agent is installed.

Related topics

- Running the Password Capture Agent Windows PowerShell module on page 30
- Configuration targets on page 30

Running the Password Capture Agent Windows PowerShell module

The Password Capture Agent Windows PowerShell module requires Windows PowerShell version 3.0 or later. It is recommended to use Windows PowerShell version 4.0.

The execution policy for Windows PowerShell must allow the execution of signed scripts. For more information, see the Microsoft documentation (About Execution Policies).

Configuration targets

To use the Password Capture Agent Windows PowerShell module to remotely configure the Password Capture Agent on the domain controllers, these servers must have Windows PowerShell remoting configured and enabled.

For more information, see the Microsoft documentation (About Remote Troubleshooting).

Installing the Password Capture Agent Windows PowerShell module

To install the Password Capture Agent Windows PowerShell module

- Copy the `OneIM-PasswordCaptureAgentMgmt` folder, including content, to `C:\Windows\System32\WindowsPowerShell\v1.0\Modules\` - the systems global Password Capture Agent Windows PowerShell module path.

  - OR -

- Copy the `OneIM-PasswordCaptureAgentMgmt` folder to any path on your host, and add this path to the environment variable `PSModulePath`.

Before installing Password Capture Agent on a domain controller:

- Ensure that the web service is installed and configured.
- Ensure that the certificate to decrypt passwords with is installed with a private key in the `LocalMachine\My\` certificate store on the server hosting the web service.
- Ensure that the certificate to encrypt passwords with is installed with a private key in the `LocalMachine\My\` certificate store on all domain controllers.

You should have prepared:

- The thumbprint of the certificate for password encryption, for example:

  `1800b62e8cf19d1c4bcdcd2b6e435c3c85e04188`

- The URL to the web service, for example:

  https://<servername.domain.com>/Appserver/

Using the Password Capture Agent Windows PowerShell module

Using the Password Capture Agent Windows PowerShell module to install Password Capture Agent on a specific domain controller

1. Sign on to the workstation where the Password Capture Agent Windows PowerShell module is installed as a member of the **Domain Admins** group.

2. Copy `One Identity Manager Password Capture Agent.msi` to a network share that can be accessed by you on all domain controllers, for example, `\\StorageServer\SHARE\One Identity Manager Password Capture Agent.msi`.

3. Open an elevated Windows PowerShell prompt.

4. Run the following command:

   `Import-Module OneIM-PasswordCaptureAgentMgmt`

5. Run the following commands to define your configuration profile:

   `$ConfigProfile = New-PCAConfigProfile`

   `$ConfigProfile['WebClient.WebServiceURL'].ConfigValue = 'https://<server.domain.com>/AppServer/'`

   `$ConfigProfile['WebClient.WebServiceType'].ConfigValue = 'REST'`

   `$ConfigProfile['Backend.Credential'].ConfigValue = Get-Credential viCaptureAgent`

   `$ConfigProfile['Backend.CertificateThumbprint'].ConfigValue = '1800b62e8cf19d1c4bcdcd2b6e435c3c85e04188'`

6. Run the following command:

   `Install-PasswordCaptureAgent` `

   `-ComputerName "DC01.DEMOCORP.COM"` `

   `-Setup "\\StorageServer\SHARE\One Identity Manager Password Capture Agent.msi"` `

   `-ConfigurationProfile $ConfigProfile`

   By running this command, you install the Password Capture Agent on `DC01.DEMOCORP.COM`. The installation runs off a network location, and the `WebServiceURL` parameter and the `CertificateThumbprint` parameter are passed to the setup.

   Because the `-Restart` switch is not specified, the domain controllers do not automatically reboot after successful installation.

Using the Password Capture Agent Windows PowerShell module to install
Password Capture Agent on all domain controllers

1. Sign on to workstation where the Password Capture Agent Windows PowerShell
    module is installed as a member of the **Domain Admins** group.

2. Copy `One Identity Manager Password Capture Agent.msi` to a network share that
    can be accessed by you on all domain controllers, for example,
    `\\StorageServer\SHARE\One Identity Manager Password Capture Agent.msi`.

3. Open an elevated Windows PowerShell prompt.

4. Run the following command:

    `Import-Module OneIM-PasswordCaptureAgentMgmt`

5. Run the following commands to define your configuration profile:

    `$ConfigProfile = New-PCAConfigProfile`

    `$ConfigProfile['WebClient.WebServiceURL'].ConfigValue =`
    `'https://<server.domain.com>/AppServer/'`

    `$ConfigProfile['WebClient.WebServiceType'].ConfigValue = 'REST'`

    `$ConfigProfile['Backend.Credential'].ConfigValue = Get-Credential`
    `viCaptureAgent`

    `$ConfigProfile['Backend.CertificateThumbprint'].ConfigValue =`
    `'1800b62e8cf19d1c4bcdcd2b6e435c3c85e04188'`

6. Run the following command:

    `Get-DomainController | Install-PasswordCaptureAgent\``

    `-Setup \\StorageServer\SHARE\One Identity Manager Password Capture Agent.msi\``

    `-ConfigurationProfile $ConfigProfile`

    `-Restart`

    By running this command, you receive a list of domain controllers and
    sequentially start the install on each one. The install runs off a network location,
    and the `WebServiceURL` parameter and the `CertificateThumbprint` parameter are
    passed to the setup.

    Because the `-Restart` switch is specified, the domain controllers automatically reboot
    after successful installation.

Working with configuration profiles

The Password Capture Agent Windows PowerShell module includes functions to create,
show, get, set, import, and export a Password Capture Agent configuration profile.

NOTE: The `Show-PCAConfigProfile` function may also be used to get an overview of all parameters and read their descriptions or destinations.

Getting and setting the configuration profile is only possible if the Password Capture Agent is installed and running. It is not possible to access the secured configuration parameters without it.

Example: Create new profile and edit it

```
Import-Module OneIM-PasswordCaptureAgentMgmt

$ConfigProfile = New-PCAConfigProfile

$ConfigProfile['WebClient.WebServiceURL'].ConfigValue =
'https://<server.domain.com>/AppServer/'

$ConfigProfile['WebClient.AuthenticationType'].ConfigValue =
'WindowsIntegrated'

$ConfigProfile['Backend.AuthenticationModule'].ConfigValue = 'DialogUser'

$ConfigProfile['Backend.Credential'].ConfigValue = Get-Credential
viCaptureAgent

$ConfigProfile['Backend.CertificateThumbprint'].ConfigValue =
'0123456789ABCED0123456789ABCED0123456789'
```

Example: Read current profile and show it using GUI

```
Import-Module OneIM-PasswordCaptureAgentMgmt

$ConfigProfile = Get-PCAConfigProfile

Show-PCAConfigProfile -ConfigurationProfile $ConfigProfile
```

Example: Read current profile and export it to XML

```
Import-Module OneIM-PasswordCaptureAgentMgmt

$ConfigProfile = Get-PCAConfigProfile

Export-PCAConfigProfile -ConfigurationProfile $ConfigProfile -FilePath
C:\tmp\CurrentPCAConfig.xml
```

Example: Import profile, edit, and set it

```
Import-Module OneIM-PasswordCaptureAgentMgmt

$ConfigProfile = Import-PCAConfigProfile -Filepath
C:\tmp\CurrentPCAConfig.xml

$ConfigProfile['Backend.CertificateThumbprint'].ConfigValue =
'0123456789ABCED0123456789ABCED0123456780'

Set-PCAConfigProfile -ConfigurationProfile $ConfigProfile
```

Example: Import profile and install Password Capture Agent

```
Import-Module OneIM-PasswordCaptureAgentMgmt

$ConfigProfile = Import-PCAConfigProfile -Filepath C:\CurrentPCAConfig.xml

Install-PasswordCaptureAgent`

–LogFile <Full UNC path to the log file on the remote server>`

-Setup <UNC path for Password Capture Agent MSI>`

-ConfigurationProfile $ConfigProfile
```

Example: Change parts of the configuration

```
Import-Module OneIM-PasswordCaptureAgentMgmt

$ConfigProfile = Get-PCAConfigProfile

$ConfigProfile['Backend.Credential'].ConfigValue = Get-Credential
viCaptureAgent

Set-PCAConfigProfile -ConfigurationProfile $ConfigProfile
```

Example: Change parts of the configuration on all domain controllers

```
Get-DomainController | Foreach-Object {
    $ConfigurationProfile = Get-PCAConfigProfile -ComputerName $_
```

```
        $ConfigurationProfile['Backend.CertificateThumbprint'].ConfigValue =
        '1800b62e8cf19d1c4bcdcd2b6e435c3c85e04188'

        Set-PCAConfigProfile -ComputerName $_ -ConfigurationProfile
        $ConfigurationProfile -RestartService

}
```

Troubleshooting

I am unable to import the Password Capture Agent Windows PowerShell module.

Windows PowerShell has an execution policy to restrict what may run. For more information about troubleshooting, see the Microsoft documentation (About Remote Troubleshooting).

- Is the `OneIM-PasswordCaptureAgentMgmt` folder in any folder listed in `$env:PSModulePath`?

I am unable to establish a connection to the domain controllers.

The connection to the domain controllers requires Windows PowerShell remoting to be configured and enabled. The firewall may also block this connection by default. For more information about troubleshooting, see the Microsoft documentation (About Remote Troubleshooting).

I am experiencing problems installing the Password Capture Agent. Is there a way to get a log file?

Yes. Both `Install-PasswordCaptureAgent` and `Uninstall-PasswordCaptureAgent` have parameters that allow you to specify a log file and if logging should be verbose. The log file will be used by `msiexec.exe`.

Example:

```
Uninstall-PasswordCaptureAgent`

-ComputerName "DC01.DEMOCORP.COM"

-LogFile \\StorageServer\SHARE\DC01.uninstall.log`

-LogVerbose
```

> **Example:**
>
> ```
> Install-PasswordCaptureAgent`
>
> -ComputerName "DC01.DEMOCORP.COM"`
>
> -LogFile \\StorageServer\SHARE\DC01.install.log`
>
> -LogVerbose`
>
> -Setup "\\StorageServer\SHARE\One Identity Manager Password Capture
> Agent.msi"
> ```

Is it possible to automatically reboot the domain controllers after installing/uninstalling Password Capture Agent?

Yes. Both `Install-PasswordCaptureAgent` and `Uninstall-PasswordCaptureAgent` have a switch called restart that will do exactly this. It is **$False** by default.

> **Example:**
>
> ```
> Uninstall-PasswordCaptureAgent -ComputerName "DC01.DEMOCORP.COM" -Reboot
> ```

> **Example:**
>
> ```
> Uninstall-PasswordCaptureAgent -ComputerName "DC01.DEMOCORP.COM" -
> Reboot:$True
> ```

Advanced scenarios and more examples

With the Password Capture Agent Windows PowerShell module, there are many ways to install Password Capture Agent on your domain controllers. Use the built-in Windows PowerShell help to find more examples of usage:

```
Get-Help Get-PasswordCaptureAgentServiceConfig -Full

Get-Help Set-PasswordCaptureAgentServiceConfig -Full

Get-Help Install-PasswordCaptureAgent -Full

Get-Help Uninstall-PasswordCaptureAgent -Full
```

Event log for the Password Capture Agent

You can read the Password Capture Agent log in the event viewer, in the `Applications and Services Logs` folder. It shows you details of hints, warnings, and errors if they occur.

- Level

- Date and time

- Source

- Event ID

- Track category

In addition, you will find information about the configuration summary on every startup process.

> **Example:**
>
> Configuration summary:
>
> - This DLL: "C:\WINDOWS\system32\PCA_Driver.DLL"
>
> - File Version: "1.0.1.9"
>
> - DLL File Version: "1.0.1.9"
>
> - Used log in event log: "One Identity Manager Password Capture Agent", with source name: Driver
>
> - Configuration key: "HKEY_LOCAL_Machine\SOFTWARE\One Identity\One Identity Manager\Password Capture Agent\Driver"
>
> - Diagnostic mode: No
>
> - Deactivate on start: No
>
> - Retry on error after seconds: 120
>
> - Storage time of pending captures in days: 7

- Log file: "<no log file specified>"
- Domain name for accounts: "democorp"
- Companion service: "One Identity Manager Password Capture Agent" has successfully initialized
- Number of unfinished captures in queue: 0
- Driver initialization completed.

Customizing security for the Password Capture Agent service

You can limit the scope of users and groups that are permitted to configure the Password Capture Agent service using built-in Windows techniques.

Use the COM+ Management Console to specify permissions for the `SetConfigParameter` task under `Component Services\Computers\My Computer\COM+ Applications\One Identity Manager Password Capture Agent\Components\PCA.Com_Class\Interfaces\COM_ Interface\Methods`.

Achieving high availability for the web service with Windows Network Load Balancing

This appendix describes how to achieve high availability for the web service using the Network Load Balancing service.

The Network Load Balancing cluster requires a dedicated IP address and fully qualified domain name. This should be set up before installing the cluster. The fully qualified domain name will be used later to access the web service. This means that every host needs a certificate that is valid for the chosen fully qualified domain name and is trusted by each domain controller.

Hosts in a Network Load Balancing cluster require at least two network interface cards. The first network interface card should be for general communication and maintenance and the second network interface card should be dedicated to Network Load Balancing traffic.

To allow high availability in a Network Load Balancing cluster, you need multiple hosts installed and configured with the web service. These hosts should be dedicated to that task. Installing Network Load Balancing on domain controllers is not supported.

Example: Settings in this lab with network interface card (NIC) and fully qualified domain name (FQDN)

Host1

      Web01.democorp.com (Windows Server 2012 R2)

          NIC1: 192.168.0.20

          NIC2: 192.168.0.200 (STATIC)

Host2

      Web02.democorp.com (Windows Server 2012 R2)

          NIC1: 192.168.0.21

          NIC2: 192.168.0.201 (STATIC)

Network Load Balancing Cluster:

FQDN: ServiceCluster.democorp.com

IP: 192.168.0.50

Detailed information about this topic

Step 1: Install the Windows Network Load Balancing service

This step shows you how to install the required Windows feature to allow the configuration of Network Load Balancing. You should complete this task on all hosts that are to be part of this cluster before continuing with the next step.

To install the required Windows feature (manually)

1. Start the Server Manager.
2. Click **Add roles and Features**.
3. Skip the first page of the wizard.
4. Select **Role-based or feature-based installation**.
5. Select the server on which you want to install the Network Load Balancing feature.
6. On the **Server roles** page, click **next**.
7. On the **Features** page, check **Network Load Balancing**.
8. Click **Add-Feature**.
9. On the **Features** page, click **next**.
10. On the confirmation page, click **install**.

To install the required Windows feature (with Windows PowerShell)

1. Start a Windows PowerShell as administrator.
2. Enter **Install-Windows Feature NLB**.

Step 2: Configure Windows Network Load Balancing

This step shows you how to configure the Network Load Balancing process. This task runs on one of the hosts that should be clustered for Network Load Balancing. These settings require you to have administrative privileges on the selected hosts.

To configure Network Load Balancing (manually)

1. Start Network Load Balancing Manager.

2. In the **Cluster** menu, click **New**.

3. In the **New Cluster: Connect** window, perform the following tasks:

   a. Connect to your first host, for example: web01.democorp.com, and click **Connect**.

   b. In the list of network interfaces, select **Ethernet 2** with the IP that is dedicated to Network Load Balancing and set to **static**.

   c. Click **Next**.

4. In the **New Cluster: Host Parameters** window, click **Next**.

5. In the **New Cluster: Cluster IP Addresses** window, perform the following tasks:

   a. Click **Add** and enter the Cluster IP, for example: 192.168.0.50 with matching subnet mask.

   b. Click **Next**.

6. In the **New Cluster: Cluster Parameters** window, perform the following tasks:

   a. Enter the Full Internet Name, for example: ServiceCluster.democorp.com.

   b. Click **Next**.

7. in the **New Cluster: Port Rules** window, perform the following tasks:

   a. Select the existing rule and click **Remove**.

   b. Click **Add**.

8. In the **Add/Edit Port Rule** window, perform the following tasks:

   a. Set the **Port range** to **From 443 to 443**.

   b. Select **TCP** as protocol.

   c. Set the **Filtering Mode** to **Multiple Host**.

   d. Set the **Affinity** to match your requirements or leave it at **Single (*)**.

   e. Click **OK**.

   f. Click **Finish**.

(*) The affinity is used to determine to which back-end server a client is connected. The web service uses a stateless architecture, so any affinity will work.

To add additional hosts to the Network Load Balancing cluster

1. Start Network Load Balancing Manager.

2. In the **Cluster** menu, click **Connect to existing**.

3. In the **Connect to Existing: Connect** window, enter the Cluster IP / FQDN, and click **Connect**.

4. In the Clusters list, select the cluster, and click **Finish**.

5. In the tree view, select the cluster.

6. In the **Cluster** menu, click **Add Host**.

7. In the **Add Host to Cluster: Connect** window, perform the following tasks:

   a. Connect to your next host, for example: web02.democorp.com, and click **Connect**.

   b. In the list of network interfaces, select **Ethernet 2** with the IP that is dedicated to Network Load Balancing and set to **static**.

   c. Click **Next**.

8. In the **Add Host to Cluster: Host Parameters** window, click **Next**.

9. In the **Add Host to Cluster: Port Rules** window, click **Finish**.

Step 3: Configuration validation

Before changing the configuration of the Password Capture Agent, you must validate the configuration. After the previous steps, you should be able to access https://ServiceCluster.democorp.com and see the IIS welcome screen.

Step 4: Applying Password Capture Agent web service URL on the Password Capture Agent

To set the Password Capture Agent web service URL

1. Start an elevated command line.

2. Run the following command to modify the web service URL at the Password Capture Agent.

   REG ADD "\\<COMPUTERNAME>\HKLM\Software\One Identity\One Identity Manager\Password Capture Agent" /v "WebService_URL" /t REG_SZ /d "https://ServiceCluster.democorp.com/AppServer/"

3. Run the following commands to restart the Password Capture Agent service.

```
sc \\<COMPUTERNAME> stop "Password Capture Agent"

sc \\<COMPUTERNAME> start "Password Capture Agent"
```

Troubleshooting

When accessing https://ServiceCluster.democorp.com, I receive an invalid certificate error in my browser.

Because you are not accessing each host by its real host name, you must ensure that the SSL certificate is issued to the common name matching the cluster's fully qualified domain name, and that the fully qualified domain name is set in the **Subject Alternative Names** (SAN) field.

When accessing https://ServiceCluster.democorp.com, Kerberos authentication fails.

Because you are accessing all servers in this cluster with the same fully qualified domain name, Kerberos authentication will fail. If you have NT Lan Manager disabled as fallback, authentication will not work.

ONE IDENTITY
by Quest

One Identity Manager 9.1.1 Password Capture Agent Administration
Guide

Appendix: Achieving high availability for the web service with
Windows Network Load Balancing

44

Installing the Password Capture Agent with MSIEXEC

The Password Capture Agent setup can be automated using MSIEXEC parameters.

NOTE: MSIEXEC does not recognize **0** to clear check boxes; instead, for example, use **PROP_FINAL_FUNCTION_TEST=""**.

Parameters for MSIEXEC

PROP_WEBSERVICE

Values: URL of the web service

Configuration after setup: Registry value `Service\WebService_URL`

PROP_CERTIFICATE

Values: One Identity Manager password encryption certificate

Configuration after setup: Registry value `Service\CertificateThumbprint`

PROP_ENCRYPTED_PASSWORD_TRANSMISSION

Values: 0 | 1

Default: 1

Configuration after setup: Registry value `Service\EncryptedPasswordTransmission`

PROP_ENCRYPTED_PASSWORD_TRANSMISSION_SIGNING

Values: 0 | 1

Default: 1

Configuration after setup: Registry value `Service\EncryptedPasswordTransmissionSigning`

PROP_WEB_SERVICE_TYPE

It is strongly recommended you use the One Identity Manager application server (REST). The One Identity ManagerSOAP Web Service support (Soap) is only included for backward compatibility to One Identity Manager version 6.x and should not be used anymore.

Values: REST | Soap

Configuration after setup: `Set-ServiceConfig.exe WebServiceType`

PROP_LOGGING_SUCCESSFUL_OPERATIONS

  Values: 0 | 1

  Default: 0

  Configuration after setup: Registry value `Driver\LoggingSuccessfulOperations`

PROP_IGNORE_PASSWORD_RESET_OPERATIONS

  Values: 0 | 1

  Default: 0

  Configuration after setup: Registry value `Driver\Ignoring\PasswordResetOperations`

PROP_BACKEND_CLIENT_CREDENTIAL_TYPE

  Values: DialogUser | WebADS | ADSAccount

  Default: DialogUser

  Configuration after setup: `Set-ServiceConfig.exe BackendClientCredentialType`

PROP_BACKEND_CLIENT_CREDENTIAL_USER_NAME

  Default: viCaptureAgent

  Configuration after setup: `Set-ServiceConfig.exe BackendClientCredentialUserName`

PROP_BACKEND_CLIENT_CREDENTIAL_USER_PWD

  Configuration after setup: `Set-ServiceConfig.exe BackendClientCredentialUserPwd`

PROP_BACKEND_CLIENT_CREDENTIAL_USER_PWD_ACCEPT_EMPTY

  Values: 0 | 1

  Default: 0

  Configuration after setup: `Set-ServiceConfig.exe BackendClientCredentialUserPwd_AcceptEmpty`

PROP_WEB_SERVICE_CLIENT_SKIP_HTTPS_VALIDATION

  Values: 0 | 1

  Default: 0

  Configuration after setup: `Set-ServiceConfig.exe WebServiceClientSkipHttpsValidation`

PROP_WEB_SERVICE_CLIENT_CREDENTIAL_TYPE

  Values: WindowsIntegrated | Certificate

  Default: WindowsIntegrated

  Configuration after setup: `Set-ServiceConfig.exe WebServiceClientCredentialType`

PROP_WEB_SERVICE_CLIENT_CREDENTIAL_CERTIFICATE_FIND_BY_TYPE

  Values: All values of the `X509FindType`-enumeration are allowed.

  Default: FindByThumbprint

Configuration after setup: `Set-ServiceConfig.Exe`
`WebServiceClientCredentialCertificateFindByType`

PROP_WEB_SERVICE_CLIENT_CREDENTIAL_CERTIFICATE

Configuration after setup: `Set-ServiceConfig.Exe`
`WebServiceClientCredentialCertificate`

PROP_FINAL_FUNCTION_TEST

Only used by setup to determine whether final function test should be run. Failure will cause setup to fail.

Values: 0 | 1

Default: 1

Configuration after setup: Only used by setup.

Example 1: Silent install with default settings

```
msiexec.exe /i "<SETUP_MSI_FILE>" /quiet /norestart /L "<LOGFILE>"
```

Example 2: Silent install with parameters

```
msiexec.exe /i "<SETUP_MSI_FILE>" /quiet /norestart PROP_
WEBSERVICE="<WEBSERVICE_URL>" PROP_WEB_SERVICE_TYPE="<WEBSERVICE_TYPE>"
PROP_CERTIFICATE="<CERTIFICATE_THUMBPRINT>" PROP_ENCRYPTED_PASSWORD_
TRANSMISSION="1" PROP_ENCRYPTED_PASSWORD_TRANSMISSION_SIGNING="1" PROP_
BACKEND_CLIENT_CREDENTIAL_USER_NAME="<One Identity Manager system user>"
PROP_BACKEND_CLIENT_CREDENTIAL_USER_PWD="<System user password>" PROP_FINAL_
FUNCTION_TEST="1" PROP_IGNORE_PASSWORD_RESET_OPERATIONS="" /L "<LOGFILE>"
```

Example 3: Interactive installation

```
msiexec.exe /i "<SETUP_MSI_FILE>" /norestart PROP_WEBSERVICE="<WEBSERVICE_
URL>" PROP_WEB_SERVICE_TYPE="<WEBSERVICE_TYPE>" PROP_
CERTIFICATE="<CERTIFICATE_THUMBPRINT>" PROP_ENCRYPTED_PASSWORD_
TRANSMISSION="1" PROP_ENCRYPTED_PASSWORD_TRANSMISSION_SIGNING="1" PROP_
BACKEND_CLIENT_CREDENTIAL_TYPE="DialogUser" PROP_BACKEND_CLIENT_CREDENTIAL_
USER_NAME="<One Identity Manager system user>" PROP_BACKEND_CLIENT_
CREDENTIAL_USER_PWD="<System user password>" PROP_FINAL_FUNCTION_TEST="1"
PROP_IGNORE_PASSWORD_RESET_OPERATIONS="" /L "<LOGFILE>"
```

Example 4: Uninstall

```
msiexec.exe /X{E7D3E2C0-0BD9-4EBB-A70C-E835D575611B} /quiet /norestart /L
"<LOGFILE>"
```

Related topics

- Registry configuration parameters on page 13
- Secured configuration parameters on page 16

Certificate lookup options

Because certificates have a limited lifetime and therefore need to be renewed or updated, Password Capture Agent service has the option to configure the search for valid certificates. Note that not all configurable `FindByTypes` may be suitable for your needs.

Example: Use certificate from local trusted root certificate authority (Active Directory Certificate Services)

All certificates issued by **DEMOCORP DEMO ROOT CA** are valid for this purpose. Automatic enrollment is used to distribute the certificates, and new certificates will automatically be generated before expiration.

- WebServiceClientCredentialCertificateFindByType = FindByIssuerName
- WebServiceClientCredentialCertificate = "DEMOCORP DEMO ROOT CA"

- OR-

- WebServiceClientCredentialCertificateFindByType = FindByIssuerDistinguishedName
- WebServiceClientCredentialCertificate = "CN=DEMOCORP DEMO ROOT CA, DC=Democorp, DC=com"

Example: Use certificate based on subject

All certificates with the subject **demoadmn** are valid for this purpose.

- WebServiceClientCredentialCertificateFindByType = FindBySubjectName
- WebServiceClientCredentialCertificate = "demoadmn"

- OR-

- WebServiceClientCredentialCertificateFindByType = FindBySubjectDistinguishedName
- WebServiceClientCredentialCertificate = "CN=demoadmn, CN=Users, DC=Democorp, DC=com"

Example: Use static certificate by thumbprint and change manually when new certificate is available

- WebServiceClientCredentialCertificateFindByType = FindByIssuerName
- WebServiceClientCredentialCertificate = 0123456789ABCED0123456789ABCED0123456789

Known error codes

There are several known error codes that the `VI_CaptureAgent_SetPassword` script can use to reject a password change. The script is stored in the One Identity Manager database. If that script does not suit your needs, you can overwrite it.

Following is the list of possible errors and appropriate actions that are returned by the `VI_CaptureAgent_SetPassword` script.

Table 2: Errors and appropriate actions

Error code	Error message	Action	Adminstration action
0	No Error. Change went through.	OK	
1	Password cycle detected.	Skip	Check manual for password cycles.
2	ADS Account is marked as privileged and will not be handled.	Skip	
1212	ADS Account has no domain.	Skip	
1317	ADS Account is not known by One Identity Manager.	Skip	Check if your Active Directory domain has been configured to be synchronized regularly within One Identity Manager.
1332	ADS Account exists but is not mapped to a Person in One Identity Manager.	Skip	Check One Identity Manager configuration; you should not have Active Directory user accounts without mapped employees.
1355	ADS Domain is not known by One Identity Manager.	Skip	Check if your Active Directory domain has been configured to be synchronized within One Identity Manager.
9901	More than one ADS Account	Skip	Check for duplicate entries in table

Error code	Error message	Action	Adminstration action
	found in One Identity Manager database matching DOMAIN\SAMAccountName.		`ADSAccount` within One Identity Manager.
9902	Failed to load Person mapped to ADS Account from One Identity Manager database.	Skip	Check One Identity Manager for problems; try loading that employee within the Object Browser.
8205	Password encryption does not match the configuration in One Identity Manager.	Skip	Compare configuration of One Identity Manager and Password Capture Agent.

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales and other inquiries, such as licensing, support, and renewals, visit https://www.oneidentity.com/company/contact-us.aspx.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product