

Univention Active Directory Connector

Topic:	Installation and configuration of Univention Active Directory Connector	
Date:	December 9, 2010	
Pages:	16	
Version number:	7397	
Autoren:	Univention GmbH	feedback@univention.de

Contents

1	Introduction to the Univention Active Directory connector	3
2	Setup of the Univention AD connector	3
2.1	Installation of the connector via the Univention Management Console	3
2.2	Basic configuration of the connector	3
2.3	Importing the SSL certificate of the Active Directory	5
2.4	Setting up the password service on the AD system	5
2.5	Starting/Stopping the Active Directory connector	6
3	Details on preconfigured synchronisation	6
3.1	Containers and organisational units	7
3.2	Groups	7
3.3	Users	8
4	Configuring mapping settings	9
5	Univention Configuration Registry variables	11
5.1	Basic configuration	11
5.2	Mapping definitions	12
5.3	Synchronisation with Windows 2000	13
6	Advanced configuration settings	13
6.1	Setting up with a different Active Directory user	13
6.2	Installing the password service in a different path	15
6.3	Synchronisation of several server AD domains with one UCS directory service	15
7	Tools	16
7.1	univention-adsearch	16
7.2	univention-connector-list-rejected	16

1 Introduction to the Univention Active Directory connector

The Univention Active Directory Connector (AD Connector for short) makes it possible to synchronise directory service objects between a Windows 2000/2003/2008 server under Active Directory (AD) and a Univention Corporate Server.

The synchronisation settings can be defined individually, making it possible for the administrator to control synchronisation very accurately by just synchronising certain objects and attributes.

In the default setting, containers, organisational units, users, and groups are synchronised. Users have an exceptional position since the password cannot be queried via the LDAP protocol in Active Directory. A special service is installed on the Windows server for this purpose, which enables password synchronisation. The client accounts are not synchronised in the default configuration since Windows clients can only be a member in one domain, so they cannot be simply adopted from an Active Directory environment into a Windows NT domain mapped via UCS.

Users can make access to services of both environments in a transparent way; this is due to the possibility of having the same user settings in both domains. After logging into a UCS domain, a subsequent connection to a file share or to an Exchange server with Active Directory is possible without a renewed password request. Users and administrators will find users and groups of the same name on the resources of the other domain, and can thus work with their familiar permission structures.

2 Setup of the Univention AD connector

2.1 Installation of the connector via the Univention Management Console

The installation is performed either during the installation of UCS or subsequently by installing the ***univention-ad-connector*** package.

The Univention AD connector can only be installed on a DC Master or DC Backup system as this is the only place the complete data in LDAP are available.

Despite intensive tests it is not possible to rule out that the results of the synchronisation may affect the operation of a productive domain. The connector should therefore be tested for the respective requirements in a separate environment in advance.

2.2 Basic configuration of the connector

The Univention Active Directory connector can be configured using a module of the Univention Management Console. If the UMC module is not to be used, a part of the configuration can also be performed manually using the corresponding Univention Configuration Registry variables, which are referred to in the text.

The top part of the **Active Directory Connector status** menu contains three entries: whether the AD connector has been configured; whether the certificate for encrypted communication with the Active Directory has been installed and the runtime status of the AD connector service.

The configuration of the AD connector can be begun by clicking on **Configure Active Directory Connector**.

The fully qualified host name of the Active Directory server must be given in the **Hostname of Active Directory Server** field. If the host name of the AD system is not resolvable for the UCS system, it may be necessary to create a **DNS Host Record** object for the AD system in Univention Directory Manager. (The setting is saved in the Univention Configuration Registry variable `connector/ad/ldap/host`.)

The **BaseDN of Active Directory** can either be entered directly or be read out of a LDAP request automatically by clicking on [**Determine BaseDN**].

The LDAP DN of the user used for access to the Active Directory is configured in the **DN of replication user** field. (The setting is saved in the Univention Configuration Registry variable `connector/ad/ldap/binddn`). When using the function for the automatic transfer of the base DN of the Active Directory, the Administrator account is entered by default for the base DN. Further information on the operation of the AD connector with a different user can be found in Chapter 6.1.1. The password used for access is entered in the **Password of replication user** field and saved in a text file. The file name is stored in the Univention Configuration Registry variable `connector/ad/ldap/bindpw`.

Some internals of the Active Directory directory service differentiate between Windows 2000 and Windows 2003/2008. The version in use is configured in the **Version of Windows server** entry field. (Univention Configuration Registry variable `connector/ad/windows_version`)

Some group names are saved differently in Active Directory depending on the installation language of the server. The localisation used can be selected under **Active Directory Connector group mapping language**. Further information can be found in Chapter 3.2. (Univention Configuration Registry variable `connector/ad/mapping/group/language`)

The AD connector can be operated in different modes, which can be selected in **Active Directory Connector sync mode**. Alongside bidirectional synchronisation, unidirectional replication towards the AD or UCS directory service is also possible. (Univention Configuration Registry variable `connector/ad/mapping/syncmode`)

In **Poll Interval (seconds)** you can specify how long the system should wait after a run showing no changes before sending a new request. (Univention Configuration Registry variable `connector/ad/poll/sleep`)

Retry interval for rejected objects specifies after how many synchronisation intervals retained changes are imported subsequently. (Univention Configuration Registry variable `connector/ad/retryrejected`)

The **Debug level of Active Directory connector** configures how much debugging information is written to the `/var/log/univention/connector.log` file. The presetting (1) documents only errors and warnings. **Add debug output for functions** can be used to specify

additionally whether further debug output is added for function calls. (Univention Configuration Registry variable `connector/ad/level` und Univention Configuration Registry variable `connector/ad/function`)

Clicking on **Save changes** adopts the configuration in Univention Configuration Registry. Changes are only adopted following a restart of the Univention AD connector, see Chapter 2.5.

2.3 Importing the SSL certificate of the Active Directory

An SSL certificate must be created on the Active Directory system and exported to allow encrypted communication. The certificate is created by the Active Directory's certificate service.

The certificate service can be installed subsequently if necessary: Start -> Properties -> System settings -> Software -> Windows components, choose Certificate Services -> Next select Enterprise root CA -> Next, Enter domain name -> Next -> Next.

This certificate must be exported and copied onto the UCS system: Root CA -> AD domain -> Properties -> Show certificate -> Details -> Copy to file -> DER binary encoded X.509.

The SSL AD certificate should now be imported into the UCS system. This is done by clicking on **Upload Active Directory certificate**. This opens a window in which a file can be selected using **Browse** and confirmed with **Upload file**. Clicking on **Save** makes the uploaded certificate available to the AD connector. The exact storage place is specified Univention Configuration Registry variable `connector/ad/ldap/certificate`.

2.4 Setting up the password service on the AD system

Active Directory prohibits the request of passwords via the LDAP protocol, which requires the installation of a package on the Windows server.

Selecting **Download ucs-ad-connector.msi and UCS certificate** opens a new browser window in which five files are available to download:

- [ucs-ad-connector.msi \(for32bitWindows\)](#)
- [ucs-ad-connector-64bit.msi \(for64bitWindows\)](#)
- [MicrosoftVisualC++2010RedistributablePackage \(x86\)](#)
- [private.key](#)
- [cert.pem](#).

On 64-bit Windows versions, the **Microsoft Visual C++ 2010 Redistributable Package (x86)** must be installed before the installation of the AD connector.

The MSI files are the installation files for the password service and can be started by double clicking on it.

The package is installed in the `C:\Windows\UCS-AD-Connector` directory automatically. Additionally, the password service is integrated into the Windows environment as a system service, which means the service can be started automatically or manually.

The ***private.key*** and ***cert.pem*** files contain the SSL certificates created in UCS for secure communication. They must also be copied into the installation directory of the password service.

The password service can then be started via **Start -> Settings -> Services**.

During a standard installation in Windows 2008 the Windows firewall blocks the access to the AD connector. This must either be deactivated in **System settings** or Port 6670/TCP authorised.

As standard, no LAN manager hash values (also known as NTLM hashes) are saved in Windows 2008. As these are required for operation with the UCS AD connector, the function must be reactivated using a policy. To do this, set **Network security: Do not store LAN Manager hash value on next password change** to **Disabled** in the group policy management editor: **Computer Configuration -> Policies -> Windows Settings -> Security Settings -> Local Policies -> Security Options**.

2.5 Starting/Stopping the Active Directory connector

The connector can be started using **Start Active Directory connector** and stopped using **Stop Active Directory connector**. Alternatively, the starting/stopping can also be performed with the `/etc/init.d/univention-ad-connector` init script.

3 Details on preconfigured synchronisation

All containers which are ignored due to corresponding filters are exempted from synchronisation as standard. That refers to the following subcontainers of the LDAP base:

On the UCS side:

```
cn=univention
cn=policies
cn=shares
cn=printers
cn=networks
cn=kerberos
cn=dhcp
cn=dns
cn=computers
```

On the AD side:

```
cn=System
cn=Builtin
cn=ForeignSecurityPrincipals
ou=Domain Controllers
cn=Program Data
```

3.1 Containers and organisational units

Containers and organisational units are synchronised together with their description. In addition, the **cn=mail** and **cn=kerberos** containers are ignored on both sides. Some particularities must be noted for containers on the AD side. In the **User manager** Active Directory offers no possibility to create containers, but displays them only in the advanced mode (**View → Advanced settings**).

Particularities

- Active Directory cannot create organisational units below containers; OUs created in UCS in this way and any subordinate objects will not be synchronised.
- Containers or organisational units deleted in AD are deleted recursively in UCS, which means that any non-synchronised subordinate objects, which are not visible in AD, are also deleted.

3.2 Groups

Groups are synchronised using the group name, whereby a user's primary group is taken into account (which is only stored for the user in LDAP in AD).

Group members with no opposite in the other system, e.g., due to ignore filters, are ignored (thus remain members of the group).

The description of the group is also synchronised.

Particularities

- The **Pre-Windows 2000 name** (LDAP attribute **SamAccountName**) is used in AD, which means that a group in Active Directory can appear under a different name from in UCS.
- The connector ignores groups, which have been configured as a **Well-Known Group** under **Samba group type** in Univention Directory Manager. There is no synchronisation of the SID or the RID.
- Groups which were configured as **Local Group** under **Samba group type** in Univention Directory Manager are synchronised as a **global group** in the Active Directory by the connector.

- Newly created or moved groups are always saved in the same subcontainer on the opposite side. If several groups with the same name are present in different containers during initialisation, the members are synchronised, but not the position in LDAP. If one of these groups is migrated on one side, the target container on the other side is identical, so that the DN's of the groups can no longer be differentiated from this point onwards.
- Certain group names are converted using a mapping table so that, for example in a German language setup, the UCS group **Domain Users** is synchronised with the AD group **Domänen-Benutzer**. When used in anglophone AD domains, this mapping can result in germanophone groups' being created and should thus be deactivated in this case. This can be done using Univention Configuration Registry variable `connector/ad/mapping/group/language` (see also Chapter 5.2).

The complete table is:

UCS group	AD group
Domain Users	Domänen-Benutzer
Domain Admins	Domänen-Admins
Windows Hosts	Domänencomputer

- Nested groups are represented differently in AD and UCS. In UCS, if groups are members of groups, these objects can not always be synchronised on the AD side and appear in the list of rejected objects. Due to the existing limitations in Active Directory, nested groups should only be assigned there.
- If a global group A is accepted as a member of another global group B in Univention Directory Manager, this membership does not appear in Active Directory because of the internal AD limitations in Windows 2000/2003. If group A's name is then changed, the group membership to group B will be lost. As of Windows 2008 this limitation no longer exists and thus global groups can also be nested in Active Directory.
- Active Directory limits search results lists from queries to a maximum of 1000 objects. During the actual synchronisation the connector bypasses this limit by requesting data in chunk. Nevertheless, in some situations this limit can still be reached. If a group has been assigned as the primary group for more than 1000 users, this limit results in the group's being rejected. This is often the case for the **Domain Users** group in large Active Directory environments. The **MaxPageSize** should therefore be increased in Active Directory (see <http://support.microsoft.com/kb/315071>).

3.3 Users

Users are synchronised like groups using the user name or using the AD pre-Windows 2000 name. The **First name**, **Last name**, **Primary group** (in so far as present on the other side), **Organisation**, **Description**, **Street**, **City**, **Postal code**, **Windows home path**, **Windows login script**, **Disabled** and **Account expiry date** attributes are transferred. Indirectly **Password**, **Password expiry date** and **Change password on next login** are also synchronised. **Primary e-mail address** and **Telephone number** are prepared but commented out due to differing syntax in the mapping configuration.

The **root** and **Administrator** users are exempted.

Particularities

- Users are also identified using the name, so that for users created before the first synchronisation on both sides, the same process applies as for groups as regards the position in LDAP.
- The synchronisation of the password expiry date and the ***Change password on next login*** user option occurs on the UCS side on the Samba level alone. If a password change is initiated by Univention Directory Manager, but the password changed in Active Directory, the expiration details for the Kerberos and Posix passwords are not changed, so that the user must change his password again if he, for example, logs on to a thin client.
- During the initial synchronisation the ***Administrator*** user is removed from the ***Domain Admins*** group, as the AD Administrator is not usually in this group. As a result, the ***Administrator*** no longer has write permissions in the UCS directory service. This can be avoided by adding an additional user in the ***Domain Admins*** group prior to the synchronisation, which can later be used for further administrative tasks.
- In some cases, a user to be created under AD, for which the password has been rejected, is deleted from AD immediately after creation. The reasoning behind this is that AD created this user firstly and then deletes it immediately once the password is rejected. If these operations are transmitted to UCS, they are transmitted back to AD. If the user is re-entered on the AD side before the operation is transmitted back, it is deleted after the transmission. The occurrence of this process is dependent on the polling interval set for the connector.
- AD and UCS create new users in a specific primary group (usually ***Domain Users*** or ***Domänen-Benutzer***) depending on the presetting. During the first synchronisation from UCS to AD the users are therefore always a member in this group.

4 Configuring mapping settings

The definition of the objects and attributes to be synchronised can be found in the </etc/univention/connector/ad/mapping.py> file. The file is written directly in the script language Python, which means that very flexible definitions are possible. This file is managed via Univention Configuration Registry, so that changes should always be made on the respective template (</etc/univention/connector/ad/mapping>). It evaluates Univention Configuration Registry variables for some standard options, so that an edit of the file is not always necessary. The existing Univention Configuration Registry variables are described in Chapter 5.2.

The `global_ignore_subtree` variable is used to specify a list of areas to be ignored. In doing so, the Univention Configuration Registry variables are used according to the template mechanism, for example:

```
global_ignore_subtree=[ 'cn=univention,@\%%ldap/base@\%%',  
                       'cn=System,@\%%connector/ad/ldap/base@\%%' ]
```

These details ensure that the `cn=univention` container on the UCS side and `cn=System` container on the Active Directory side are ignored with all subordinate objects.

All further mapping options are defined in the **ad_mapping** directory. A clear name is always given as a key, e.g., **user**. An **univention.connector.property** object is transferred to this key as a value. This object has the following characteristics:

ucs_module The Univention Directory Manager module used to edit the object. A list of all possible modules can be output using the `univention-directory-manager modules` command.

sync_mode The synchronisation mode to be used. Possible values are **read**, **write**, **sync** and **none**. With **read** the objects are replicated from Active Directory to UCS. With **write** objects are replicated from Active Directory to UCS, with **sync** the data are synchronised bidirectionally and with **none** no changes are performed. The global sync mode from the Univention Configuration Registry variable `connector/ad/mapping/syncmode` is used in the predefined mapping.

scope This specifies the search depth for the LDAP searches. Possible values are **sub**, **one**, **base**.

con_search_filter The LDAP search filter used to identify the objects in Active Directory.

match_filter The LDAP search filter used to identify the objects in UCS.

ignore_filter A filter can be entered here for objects which should be ignored.

ignore_subtree This container list is ignored along with the subordinate objects.

con_create_objectclass A list of object classes used on the Active Directory side.

dn_mapping_function A list of functions called when converting the DN from Active Directory to UCS and vice versa. This setting is required, e.g., with users, as the **uid** attribute is used on the UCS side and the **cn** attribute is used on the Active Directory side.

attributes A dictionary of **univention.connector.attribute** objects which can be processed directly when creating or making changes.

ucs_create_functions A list of functions which are run in UCS after an object has been created.

post_con_modify_function A list of functions run in Active Directory after an object is modified.

post_ucs_modify_functions A list of functions run in the UCS directory service after an object is modified.

post_attributes A dictionary of attributes which cannot be changed when an object is created, but rather only in the second step.

mapping_table A dictionary with keys corresponding to the attributes. A list of string tuples is assigned to the key, which contain UCS and AD denominators respectively. If a UCS object from this list is found during synchronisation, the corresponding AD attribute is entered and vice versa. Used, for example, when mapping group names.

position_mapping Assigns a container in AD to a container in UCS. Should be used cautiously in case identically named containers exist on the other side. For example, groups of the UCS containers **cn=groups** should not be mapped to the standard AD container **cn=users** as this container already exists in UCS. Groups saved in UCS in **cn=users** are then not placed within this reference and can cause errors.

5 Univention Configuration Registry variables

A number of Univention Configuration Registry variables are available for configuration on the UCS side. These are evaluated when the connector is started/restarted.

5.1 Basic configuration

- `connector/ad/ldap/base`
The LDAP base DN of the Active Directory Server, e.g.,
`dc=ad,dc=univention,dc=de`.
- `connector/ad/ldap/binddn`
The Univention AD connector uses this user to make changes in the LDAP of the Active Directory, e.g.,
`cn=Administrator,cn=users,dc=ad,dc=univention,dc=de`
- `connector/ad/ldap/bindpw`
The file, which contains the password of the user saved in the Univention Configuration Registry variable `connector/ad/ldap/binddn`, e.g.,
`/etc/univention/ad.secret`. This file should contain exactly one line.

- `connector/ad/ldap/certificate`
File name of a certificate exported by Active Directory with its full path (for encrypted transmission of passwords). The certificate is saved in PEM format.
- `connector/ad/ldap/host`
This variable contains the fully qualified host name of the Active Directory server, e.g., **w2k3.ad.univention.de**.
- `connector/ad/ldap/port`
The port of the LDAP server on the Active Directory server, preset as 389.
- `connector/ad/ldap/ssl`
If the configuration option is set to **no**, there is no SSL encryption for the access to the Active Directory. This can be necessary when no certificate service can be installed on the Active Directory service.
- `connector/ad/listener/dir`
Directory in which the objects transferred from UCS to Active Directory are stored, preset as `/var/lib/univention-connector/ad`. The corresponding listener module saves the changes in this path; it should thus not be altered.
- `connector/ad/poll/sleep`
Time in seconds which is waited after a run without changes until the next request is made. Only new files are searched for locally in the directory named above; a LDAP request is made on the Active Directory side. The shorter this time, the higher the replication speed and the unnecessary system load. Preset as five seconds.
- `connector/ad/retryrejected`
Number of requests without new changes after which an attempt is made to import retained changes subsequently. Preset as 10. This procedure can be followed in the `/var/log/univention/connector-status.log` file.
- `connector/debug/level`
Specifies the amount of debug information to be entered in `/var/log/univention/connector.log`. Preset as 1, so that warnings and errors are documented. Can be increased to 4.
- `connector/debug/function`
Preset as 0. When set to 1 the function calls are also documented as additional debug information

5.2 Mapping definitions

- `connector/ad/mapping/syncmode`
Defines the synchronisation mode; **read** (reading only from Active Directory to UCS), **write** (writing only from UCS to Active Directory) and **sync** (bidirectional synchronisation) are supported. Preset as **sync**.
- `connector/ad/mapping/user/primarymail`
Defines whether the primary e-mail address on user objects in UCS should be synchronised with the **mail** attribute in Active Directory. As **mail** is a multivalued attribute, this can cause problems during the synchronisation. Thus preset as **false**. During the installation of the **univention-ad-connector-exchange** package the value is set to **true**.

- `connector/ad/mapping/group/primarymail`
Defines whether the primary e-mail address on group objects in UCS should be synchronised with the **mail** attribute in Active Directory. As **mail** is a multivalued attribute, this can cause problems during the synchronisation. Thus preset as **false**. Active Directory may require the Exchange expansion for this option. During the installation of the **univention-ad-connector-exchange** package the value is set to **true**.
- `connector/ad/mapping/group/language`
Defines which form of standard group names should be used between UCS (group names are always English) and Active Directory. The mapping to a Active Directory service in German language is preset using the value **de**.
- `connector/password/service/encoding`
The password service in Windows requires the user name to be in iso8859 format for changing the password. The encoding can be set with this variable. Deviations from the preset (iso8859-15) should only be necessary in corner cases.

5.3 Synchronisation with Windows 2000

- `connector/ad/windows_version`
The type of LDAP database accesses is different between Windows 2000 and Windows 2003/2008. This Univention Configuration Registry variable must be set to **win2000** to synchronise against an Active Directory from Windows 2000.
- `connector/ad/mapping/user/win2000/description`
Due to limitations in operation with Windows 2000 servers the connector cannot set any object description as blank in Active Directory. As a result the synchronisation of descriptions in Windows 2000 mode is deactivated. If this Univention Configuration Registry variable is set to **true**, descriptions are nevertheless synchronised for users in so far as this is possible.

6 Advanced configuration settings

This chapter illustrates some advanced configuration steps, e.g., the operation of the Univention Active Directory Connector with a non-Administrator user.

6.1 Setting up with a different Active Directory user

Alongside the by default sufficiently privileged **Administrator** user, other user accounts in Active Directory can also be used for the password service or LDAP access. To do this, sufficient permissions must be assigned to these users.

The configuration in this document is performed using the **Administrator** user account as an example, as this usually holds the requisite permissions for the application of the AD connector. However, if the permissions are too broad because, for example, it is only possible to perform

a unidirectional synchronization, different accounts can be created as "replication users" for the access to the LDAP directory.

6.1.1 User account for LDAP replication

In the standard configuration of an Active Directory directory based on Windows 2003 each authorized user is assigned sufficient read permissions for the use of the AD Connector: Container and organisational units as well as all users and groups can be read. Write permission, which is required for matching UCS accounts to AD, is only assigned to members of the **Administrators** group.

In the advanced view of the MMC plug-in **Active Directory Users and Computers** the permissions for a replication slave can be assigned via the properties of organisational units. If the permission for an organisational unit is revoked from an account used for replication, the subordinate users and groups will no longer be synchronised to UCS.

If such limited read permissions are defined and conditions for user and groups for the synchronisation cannot be fully satisfied, "rejects" can occur. A typical cause is a primary group in a container which can no longer be read by the connector. If, for example, the read permission for the standard group container is suppressed, AD users can no longer be stored in UCS since the **Domain users** primary group is found there. A further primary group must then be assigned in AD or the group must be moved.

To avoid unnecessary "rejects", appropriate settings must be defined in the mapping file so that the connector does not attempt to synchronise directory contents to AD for which it does not possess write access. If, for example, a read-only access to AD is intended, a read access can be configured by setting the `sync_mode` to `read`. (see Chapter 4).

6.1.2 User account for the password service

The password service that the AD Connector accesses requires considerably more privileges than a standard user for a read access to the SAM database. If a different user with limited rights is used for the LDAP access on the AD, this can not similarly be used for the operation of a password service.

Following installation, the password service is started as a local system service and is initially not assigned to any user account. If a specific user is to be assigned to it, the user account saved for this purpose must be added in the **Administrators** group in order to retain all the necessary privileges. In addition, it requires a read access to the installation directory `C:\Windows\UCS-AD-Connector` and a write access to the `copypwd.txt` and `copypwd.in.txt` files therein as well as the log file.

The service can be started manually for a different user if it has the permission for local login and the debugging of programs in the local security policies. Unfortunately these permissions are not applied in a sufficient form if the program is started automatically as a service.

6.2 Installing the password service in a different path

If the password service should be installed in a different directory from the installation default, it can be copied into different directory subsequently. Firstly the service should be stopped and then removed. This can be done by entering the following commands in the Windows prompt:

```
C:\Windows\UCS-AD-Connector\ucs-ad-connector.exe -stop
C:\Windows\UCS-AD-Connector\ucs-ad-connector.exe -remove
```

The UCS-AD-Connector directory can then be moved and the service reinstalled.

```
C:\AD\UCS-AD-Connector\ucs-ad-connector.exe -install
C:\AD\UCS-AD-Connector\ucs-ad-connector.exe -start
```

6.3 Synchronisation of several server AD domains with one UCS directory service

There is the possibility of synchronising several separate active directory domains in one common UCS domain. This can be done to synchronise several domains of a forest, for example. One OU (organisational unit) can be defined in LDAP for each AD domain, under which the objects of the respective domains are synchronised.

Several connector instances are started parallel to each other for the multiple replication. Each connector instance is operated with a self-contained configuration base. The `prepare-new-instance` script is used to create a new instance, e.g.:

```
/usr/share/univention-ad-connector/scripts/prepare-new-instance \
-a create -c connector2
```

This script creates a further init script for the second connector instance (`/etc/init.d/univention-ad-connector2`), a configuration directory `/etc/univention/connector2` with a copy of the mapping settings of the main connector instance (this can be adapted if necessary) and an array of internal runtime directories.

The additional connector instances are registered in the Univention Configuration Registry variable `connector/listener/additionalbasenames`.

If a synchronisation is performed from Univention Corporate Server towards the Active Directory, the replication of the listener module must be restarted after a further connector instance is created. This is done with the command

```
univention-directory-listener-ctrl resync ad-connector
```

This alteration may take some time for large directory services and should be undertaken in a maintenance window where possible.

The command line tools provided with to the Univention AD Connector such as `univention-adsearch` support the entering of a connector instance with the parameter `-c`.

The configuration of further connector instances is not covered by the Univention Management Console module.

7 Tools

The AD connector installs the following tools for diagnosis:

7.1 univention-adsearch

Facilitates a simple LDAP search in Active Directory. The values for accessing the AD server specified via Univention Configuration Registry are used. Objects deleted in AD are always shown (they are still kept in an LDAP subtree in AD). As the first parameter the script awaits an LDAP filter; the second parameter can be a list of LDAP attributes to be displayed.

Example:

```
univention-adsearch cn=administrator cn,givenName
```

AD limits the numbers of results to a maximum of 1000 results (Size limit). If the search delivers more results, an error message is displayed (Sizelimit exceeded).

7.2 univention-connector-list-rejected

This tool lists the DNs of non-synchronised objects. In addition, in so far as temporarily stored, the corresponding DN in the respective other LDAP directory will be displayed. In conclusion **lastUSN** shows the ID of the last change synchronised by AD.

This script may display an error message or an incomplete output if the AD connector is in operation.

For troubleshooting when experiencing synchronisation problems, corresponding messages can be found in the following files:

```
/var/log/univention/connector.log  
/var/log/univention/connector-status.log  
/var/log/univention/connector-tracebacks.log
```