Software Defined Infrastructure

Daniel Hiller & Micha Huhn

July 29, 2021

Contents

1 DNS

1.1 Queriyng DNS data

Due to the absence of `dig`, this was installed with the following command:

```
$ apt install dnsutils
```

1.1.1 Queriyng www.hdm-stuttgart.de

MX:

```
$ dig +nocmd hdm-stuttgart.de mx +noall +answer:
  hdm-stuttgart.de. 2752    IN  MX  10 mx2.hdm-stuttgart.de.
  hdm-stuttgart.de. 2752    IN  MX  10 mx4.hdm-stuttgart.de.
  hdm-stuttgart.de. 2752    IN  MX  10 mx3.hdm-stuttgart.de.
  hdm-stuttgart.de. 2752    IN  MX  10 mx1.hdm-stuttgart.de.
```

```
$ dig +noall +answer 10 mx2.hdm-stuttgart.de.:
  mx2.hdm-stuttgart.de. 3197    IN  A   141.62.1.23
```

```
$ dig +nocmd +noall +answer -x 141.62.1.23:
  23.1.62.141.in-addr.arpa. 3142    IN  PTR mx2.hdm-stuttgart.de.
```

NS:

```
$ dig +nocmd hdm-stuttgart.de ns +noall +answer:
  hdm-stuttgart.de. 3590    IN  NS  iz-net-4.hdm-stuttgart.de.
  hdm-stuttgart.de. 3590    IN  NS  iz-net-3.hdm-stuttgart.de
  hdm-stuttgart.de. 3590    IN  NS  dns1.belwue.de.
  hdm-stuttgart.de. 3590    IN  NS  iz-net-2.hdm-stuttgart.de.
  hdm-stuttgart.de. 3590    IN  NS  dns3.belwue.de.
```

```
$ dig +noall +answer dns1.belwue.de.:
  dns1.belwue.de.       86400   IN  A   129.143.2.10
```

```
$ dig +nocmd +noall +answer -x 129.143.2.10:
  10.2.143.129.in-addr.arpa. 86400 IN   PTR dns1.belwue.de.
```

1.2 Queriyng www.spotify.com

CNAME:

```
$ dig +noall +answer www.spotify.com:
  www.spotify.com.  230 IN  CNAME   edge-web-split-geo.dual-gslb.spotify.com.
  edge-web-split-geo.dual-gslb.spotify.com. 80 IN   A 35.186.224.25
```

```
$ dig +noall +answer -x 35.186.224.25:
  25.224.186.35.in-addr.arpa. 120   IN  PTR 25.224.186.35.bc.googleusercontent.com.
```

1.3 Installing Bind

With the following command we can install `bind9` and `bind9utils`:

```
apt install bind9 bind9utils
```

In `/etc/bind/` we need to adjust the `named.conf.options`, for that we need to know the IP-address of our domain `sdi3a.mi.hdm-stuttgart.de` to which we want to forward. For that we can use the following command:

```
$ dig +nocmd sdi3a.mi.hdm-stuttgart.de +noall +answer:
  sdi3a.mi.hdm-stuttgart.de. 86400 IN   A   141.62.75.103
```

Now we can enter the IP-address in the already mentioned file.

1.3.1 Configure the zone file

To register our zones (which we will create later) we need to adjust the file `:named.conf.local` which should look like the following:

```
//
// Do any local configuration here
//

zone "mi.hdm-stuttgart.de" {

  type master;

  file "/etc/bind/zones/db.forward";

  allow-transfer { 141.62.75.103; };

  };


zone "75.62.141.in-addr.arpa" {

  type master;

  file "/etc/bind/zones/db.reverse";

  allow-transfer { 141.62.75.103; };

  };

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
```

1.3.2 Configure the zone file

For our zones we need to enable IPv4 in the File `/etc/default/bind9` with the parameter:

```
# startup options for the server
OPTIONS="-4 -u bind"
```

1.3.3 Create cache directory

```
$ mkdir -p /var/cache/bind
```

1.3.4 Configure the created zones

In the first step we need to change our directory to

```
$ cd /etc/bind
$ mkdir zones
```

1.3.4.1 Configure forward zone We start to configure our forward lookup zone `zones/db.forward` with

```
$ vim db.forward
```

To get the host record we need to `dig` sdi3a.mi.hdm-stuttgart.de.

```
$ dig +noall +answer sdi3a.mi.hdm-stuttgart.de.:
  sdi3a.mi.hdm-stuttgart.de. 86400 IN   A   141.62.75.103
```

With this information we can adjust our file `zones/db.forward` which looks like the following:

```
; db.forward
; Forward lookup zone

$TTL 604800
@         IN     SOA     ns3.mi.hdm-stuttgart.de. kuhn.hdm-stuttgart
            01;
            28800;
            7200;
            2419200;
            86400;
 )

                       NS           ns3
ns3             IN     A            141.62.75.103
sdidoc.sdi3a    IN     A            141.62.75.103
sdi3a           IN     A            141.62.75.103
www             IN     A            141.62.75.103
manual.sdi3a    IN     A            141.62.75.103
```

```
www3-1                           IN            CNAME               www
www3-2                           IN            CNAME               www
info                             IN            CNAME               www
```

1.3.4.2 Configure reverse zone

With the information we became above from the dig command, we can configure our reverse zone:

```
; db.rev-local
; reverse lookup zone

$TTL 604800
@                    IN            SOA            ns3.mi.hdm-stuttgart.de. kuhn.hdm-stuttgar
                         01;
                         28800;
                         7200;
                         2419200;
                         86400;
 )
                                   NS            ns3.
103                  IN            PTR           sdi3a.mi.hdm-stuttgart.de.
```

1.3.4.3 Forwarders

We use the CloudFlare DNS service, as a forwarder.

Add the forwarder in the file `/etc/bind/named.conf.options`:

```
forwarders {
    1.1.1.1
};
```

1.3.4.4 Set mail exchange record

To achieve this we need to set another record in our forward zone `etc/bind/zones/db.forward`:

```
mail                             IN            MX          10      mx1.hdm-stuttgart.de.
```

Test the record via `nslookup`:

```
$ nslookup manual.sdi3a.mi.hdm-stuttgart.de 141.62.75.103
Server:     141.62.75.103
Address:    141.62.75.103#53

Name:   manual.sdi3a.mi.hdm-stuttgart.de
Address: 141.62.75.103
```

```
$ nslookup -type=ptr 141.62.75.103
Server:     127.0.0.53
Address:    127.0.0.53#53

103.75.62.141.in-addr.arpa  name = sdi3a.mi.hdm-stuttgart.de.
103.75.62.141.in-addr.arpa  name = dh102.sdi3a.mi.hdm-stuttgart.de.
```

```
103.75.62.141.in-addr.arpa  name = manual.sdi3a.mi.hdm-stuttgart.de.
```

2 Bibliography

3 LDAP

3.1 Recommended Preparations

3.1.1 What is the LDAP Protocol? What is the difference between the two protocols ldap and ldaps?

“The Lightweight Directory Access Protocol can be used for querying and modifying information from distributed directory services.”

The difference between these two protocols are the encryption, LDAPS is encrypted via SSL and running on the default port 636, LDAP is encrypted via STARTTLS or decrypted and running on default port 389. (“Editorial - LDAP”, 2021)

3.1.2 What does the acronym dc in dc=somedomain, dc=org stand for?

It stands for domain component and represents the namespaces of an object (Willeke, 2019).

3.1.3 What is the role of LDAP objectclass definitions? How do they relate to LDAP schema definitions?

The ObjectClass is a LDAP Schema element AttributeType (Willeke, 2019).

3.1.4 Describe the relationship between LDAP entries and objectClass values.

Each LDAP Entry in the Directory Information Tree has an ObjectClass attribute. The Values of this attribute can be modified but not removed (Willeke, 2019).

3.1.5 Is it possible to dynamically change an entries structure?

No, the structure must conforms the constraint defined by the LDAP Schema (Willeke, 2019).

3.1.6 What does the term “bind to an LDAP” server mean? What is an “anonymous” bind?

Bind is used to authenticate clients to the directory server.

There are three elements include in the request:

1. LDAP protocol version
2. Distinguished Name (DN)
3. Credentials for user authentication

At an anonymous bind the above points 2. and 3. are submitted as an empty string.

(Wilson, -)

3.1.7 Do LDAP servers in general support database features like transactions, ACID semantic etc.?

“Lightweight Directory Access Protocol (LDAP) Transactions is define din RFC 5805 and is defined as”Experimental”.

As with distinct update operations, each transaction has atomic, consistency, isolation, and durability properties ACID.” (Willeke, 2017)

3.1.8 Explain the term “replication” in an LDAP server context.

For distribution reasons the LDAP-database can be distributed to several servers. There exists one master, on which write-operations are allowed, at the others can only pull the changes from the master (Anonym, 2019).

3.1.9 Why do organizations sometimes prefer LDAP data repositories rather than using relational database systems?

LDAP is very suitable in cases of high read rates and low write rates (write-once-read-many-times). Furthermore relational databases like SQL requires a detailed knowledge about the data structure, which isn’t the case when it comes to LDAP. (ZyTrax, 2019)

3.1.10 How is the LDIF format being organized? Explain the practical use of LDIF data when running a LDAP service.

The format is organized with objects and attributes. The LDIF data describes the directory structure which is needed for exchange (“Editorial - LDIF”, 2021)

3.1.11 LDAP filters

3.1.11.1 How do LDAP filters work? There are several filters in LDAP, with these filters its possible to add criteria to an object search. (Föckeler, -)

3.1.11.2 What is the meaning of the term scope? The LDAP search scope indicates the set of entries at or below the BaseDN that may be considered potential matches for a SearchRequest (Willeke, 2019).

3.1.11.3 How do predicate based filters connected by logical and/or/not look like? And: (& (...K1...) (...K2...) (...K3...) (...K4...)) Or: (| (...K1...) (...K2...) (...K3...) (...K4...)) Not: (! (...K1...) (...K2...) (...K3...) (...K4...))

3.1.12 OpenLDAP server software specific questions

3.1.12.1 What does the term "database backend" refer to with respect to OpenLDAP server implementation? Backend do the actual work of storing or retrieving data in response to LDAP requests. Backend may be compiled statically into slapd, or when module support is enabled, they may be dynamically loaded (Open LDAP Foundation, 2021).

3.1.12.2 Why is LDAP replication important? The risk of a failure will be minimized and the traffic load will be reduced.

3.1.13 Bibliography

Willeke, J. (various dates). LDAP Wiki 3. May 2021, from https://ldapwiki.com/wiki

Editorial - LDAP. (2021, April 19). In Wikipedia. https://de.wikipedia.org/wiki/Lightweight_Directory_Acces

Editorial - LDIF. (2021, April 19). In Wikipedia. https://de.wikipedia.org/wiki/LDAP_Data_Interchange_Fo

Bosswell, W. (2003, October 10). ObjectClasses queried 3. May 2021, from https://www.informit.com/articles/article.aspx?p=101405&seqNum=7#:~:text=Domain%20Component%20(

Wilson, N. (No datum available). The LDAP Bind Operation queried 3. May 2021, from https://ldap.com/the-ldap-bind-operation/

Anonym (2019, September 3). LDAP Wiki 3. May 2021, from https://ldapwiki.com/wiki

ZyTrax Inc. (2019, February 19). LDAP Concepts & Overview 7. May 2021, from http://www.zytrax.com/books/ldap/ch2/

Föckeler, P. (No datum available). Das LDAP Scripting Tutorial queried 10. May 2021, from http://www.selfadsi.de/ldap-filter.htm

Open LDAP Foundation. (2021, February 26). OpenLDAP queried 10. May 2021, from https://www.openldap.org/doc/admin25/

3.2 Exercises

3.2.1 Browse an existing LDAP Server

3.2.1.1 No Authentication vs. Authentication? When you are authenticated on the LDPA-server, you can see all data which belongs to your user. When you are not authenticated you can also see all data with the exception of the `matrikelNr`.

3.2.2 Set up an OpenLdap server

First we need to install several packages on our server:

```
$ apt install slapd ldap-utils dialog
```

To reconfigure `slapd` we need to type into our console:

```
$ dpkg-reconfigure slapd
```

```
DNS-Domainname: sdi3a.mi.hdm-stuttgart.de
```

3.2.3 Populating your DIT

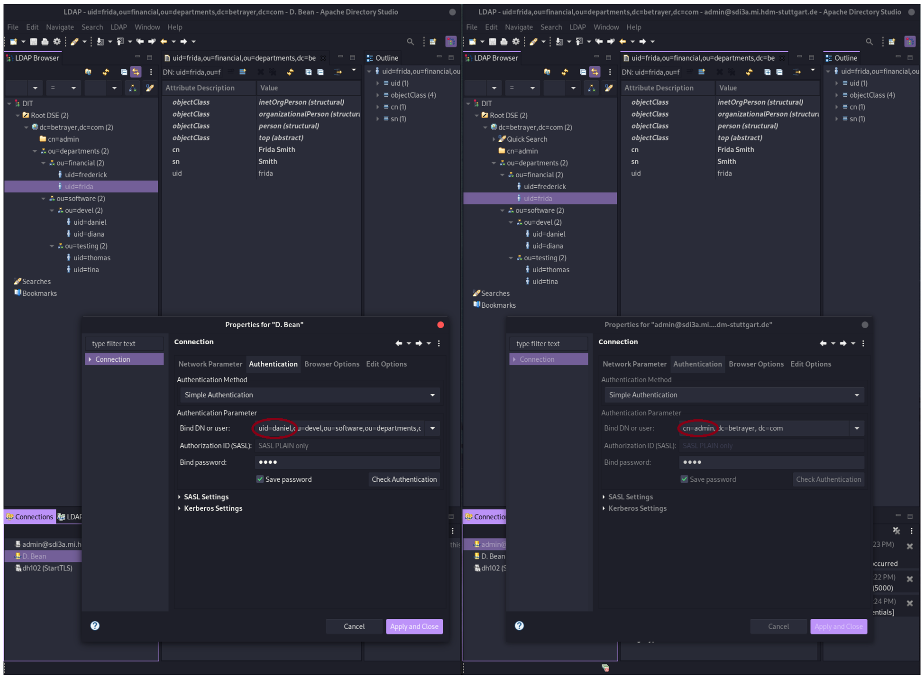
After add all entry's in our tree, it look like the following:

```
version: 1

dn: dc=betrayer,dc=com
objectClass: dcObject
objectClass: organization
objectClass: top
dc: betrayer
o: betrayer.com

dn: cn=admin,dc=betrayer,dc=com
objectClass: organizationalRole
objectClass: simpleSecurityObject
cn: admin
userPassword:: e1NTSEF9UUpzZm96RVFxVTFadEhGN3VrWE96dDNZRi9hc09LaXY=
description: LDAP administrator

dn: ou=departments,dc=betrayer,dc=com
objectClass: organizationalUnit
objectClass: top
ou: departments

dn: ou=software,ou=departments,dc=betrayer,dc=com
objectClass: organizationalUnit
objectClass: top
ou: software

dn: ou=financial,ou=departments,dc=betrayer,dc=com
objectClass: organizationalUnit
objectClass: top
ou: financial

dn: ou=devel,ou=software,ou=departments,dc=betrayer,dc=com
objectClass: organizationalUnit
```

```
objectClass: top
ou: devel

dn: ou=testing,ou=software,ou=departments,dc=betrayer,dc=com
objectClass: organizationalUnit
objectClass: top
ou: testing

dn: uid=diana,ou=devel,ou=software,ou=departments,dc=betrayer,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: Diana Smith
sn: Smith
uid: diana

dn: uid=daniel,ou=devel,ou=software,ou=departments,dc=betrayer,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: Daniel Bean
sn: Bean
uid: daniel
userPassword:: e1NNRDV9QlRqWVBrL2tuSjkrUGNIRk1SeUhBWXdCOHFLeGVMQ2I=

dn: uid=tina,ou=testing,ou=software,ou=departments,dc=betrayer,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: Tina Bean
sn: Bean
uid: tina

dn: uid=thomas,ou=testing,ou=software,ou=departments,dc=betrayer,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: Thomas Smith
sn: Smith
uid: thomas

dn: uid=frida,ou=financial,ou=departments,dc=betrayer,dc=com
```

```
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: Frida Smith
sn: Smith
uid: frida

dn: uid=frederick,ou=financial,ou=departments,dc=betrayer,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
cn: Frederick Bean
sn: Bean
uid: frederick
```

3.2.4 Testing a bind operation as non - admin user



LDAP - uid=frida,ou=financial,ou=departments,dc=betrayer,dc=com - D. Bean - Apache Directory Studio

Properties for "D. Bean"

Connection — Authentication

Authentication Method: Simple Authentication

Authentication Parameter

Bind DN or user: uid=daniel,ou=devel,ou=software,ou=departments,c

Authorization ID (SASL): SASL PLAIN only

Bind password: ●●●●

☑ Save password

LDAP - uid=frida,ou=financial,ou=departments,dc=betrayer,dc=com - admin@sdi3a.mi.hdm-stuttgart.de - Apache Directory Studio

Properties for "admin@sdi3a.mi....dm-stuttgart.de"

Connection — Authentication

Authentication Method: Simple Authentication

Authentication Parameter

Bind DN or user: cn=admin, dc=betrayer, dc=com

Authorization ID (SASL): SASL PLAIN only

Bind password: ●●●●

☑ Save password

3.2.5 Filter based search

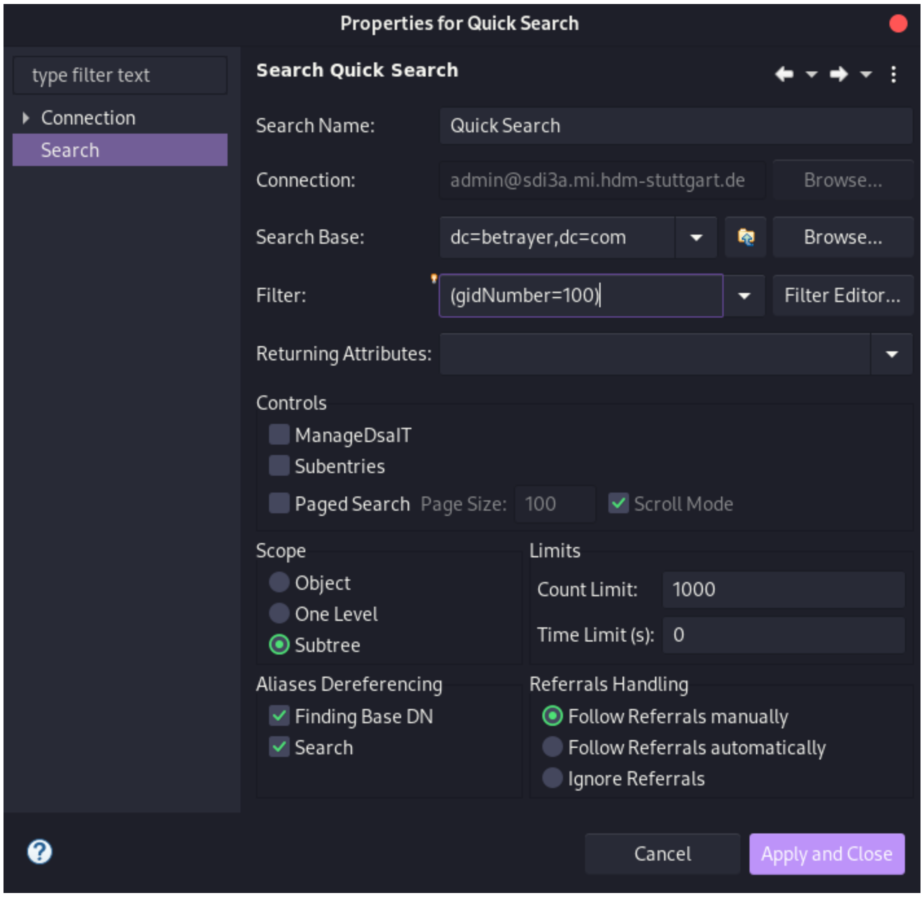
All users with an `uid` attribute value starting with the letter “b”:

```
(uid=b*)
```

All entries with either a defined `uid` attribute or a `ou` attribute starting with letter “d”:

```
(|(uid=d*)(ou=d*))
```

All users entries within the whole DIT having a gidNumber value of 100:



Properties for Quick Search

Search Quick Search

Search Name: Quick Search

Connection: admin@sdi3a.mi.hdm-stuttgart.de

Search Base: dc=betrayer,dc=com

Filter: (gidNumber=100)

Returning Attributes:

Controls

□ ManageDsaIT

□ Subentries

□ Paged Search Page Size: 100 ☑ Scroll Mode

Scope

○ Object

○ One Level

● Subtree

Limits

Count Limit: 1000

Time Limit (s): 0

Aliases Dereferencing

☑ Finding Base DN

☑ Search

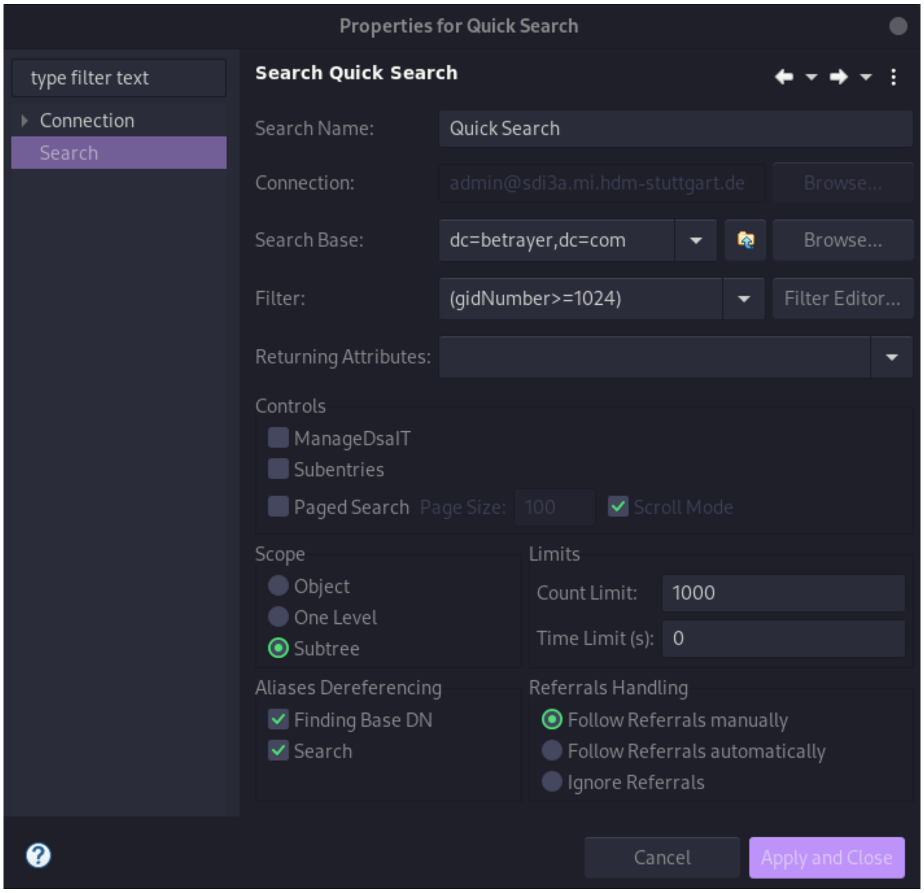
Referrals Handling

● Follow Referrals manually

○ Follow Referrals automatically

○ Ignore Referrals

All users entries within the whole DIT having a gidNumber value greater then 1023:



Properties for Quick Search

Search Quick Search

Search Name: Quick Search

Connection: admin@sdi3a.mi.hdm-stuttgart.de

Search Base: dc=betrayer,dc=com

Filter: (gidNumber>=1024)

Returning Attributes:

Controls

□ ManageDsaIT

□ Subentries

□ Paged Search Page Size: 100 ☑ Scroll Mode

Scope

○ Object

○ One Level

● Subtree

Limits

Count Limit: 1000

Time Limit (s): 0

Aliases Dereferencing

☑ Finding Base DN

☑ Search

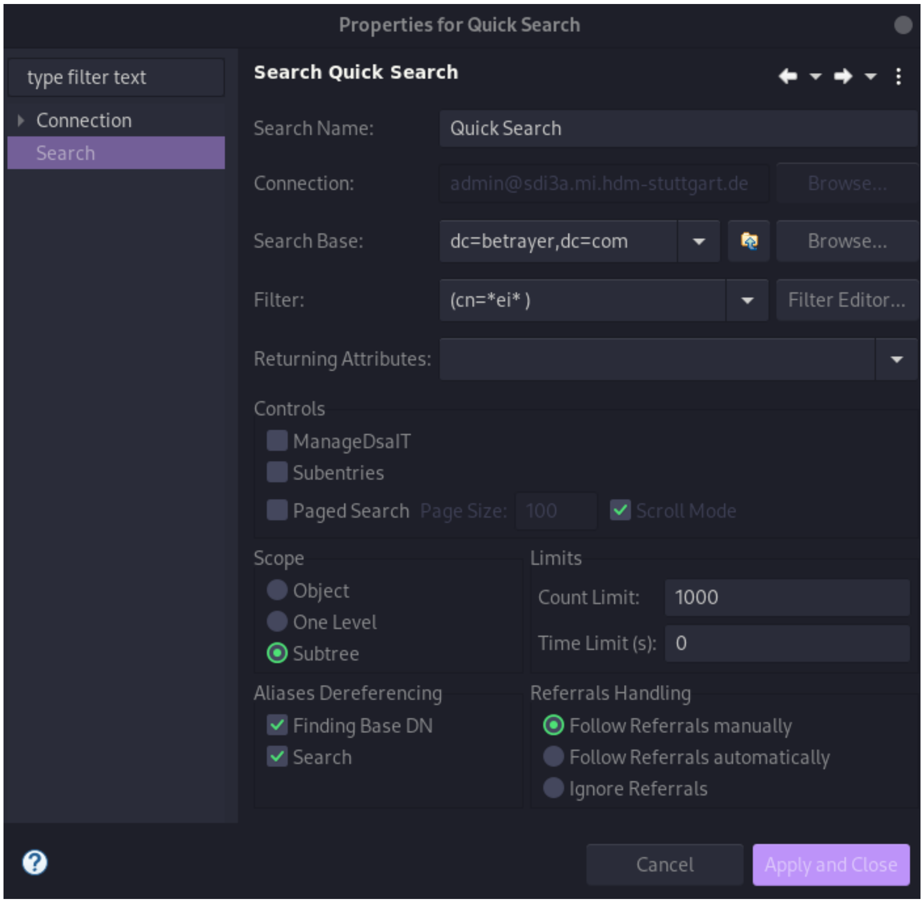
Referrals Handling

● Follow Referrals manually

○ Follow Referrals automatically

○ Ignore Referrals

All users entries within the whole DIT having the substring “ei” in their cn attribute:



Properties for Quick Search

type filter text

- Connection
- Search

Search Quick Search

Search Name: Quick Search

Connection: admin@sdi3a.mi.hdm-stuttgart.de Browse...

Search Base: dc=betrayer,dc=com Browse...

Filter: (cn=*ei*) Filter Editor...

Returning Attributes:

Controls

□ ManageDsaIT

□ Subentries

□ Paged Search Page Size: 100 ☑ Scroll Mode

Scope

○ Object

○ One Level

● Subtree

Limits

Count Limit: 1000

Time Limit (s): 0

Aliases Dereferencing

☑ Finding Base DN

☑ Search

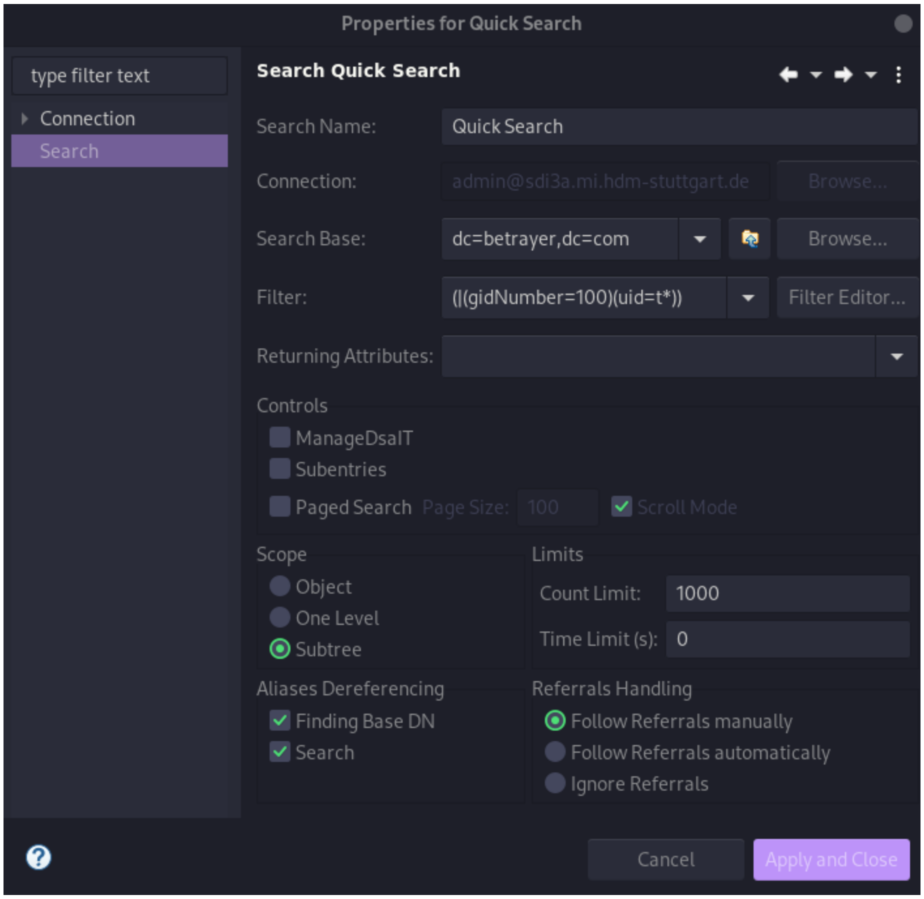
Referrals Handling

● Follow Referrals manually

○ Follow Referrals automatically

○ Ignore Referrals

All users entries within the whole DIT starting with the character “t” in their uid attribute or the gidNumber is equal to 100:



Search Name: Quick Search

Connection: admin@sdi3a.mi.hdm-stuttgart.de

Search Base: dc=betrayer,dc=com

Filter: (|(gidNumber=100)(uid=t*))

Returning Attributes:

Controls

□ ManageDsaIT

□ Subentries

□ Paged Search Page Size: 100 ☑ Scroll Mode

Scope

○ Object

○ One Level

● Subtree

Limits

Count Limit: 1000

Time Limit (s): 0

Aliases Dereferencing

☑ Finding Base DN

☑ Search

Referrals Handling

● Follow Referrals manually

○ Follow Referrals automatically

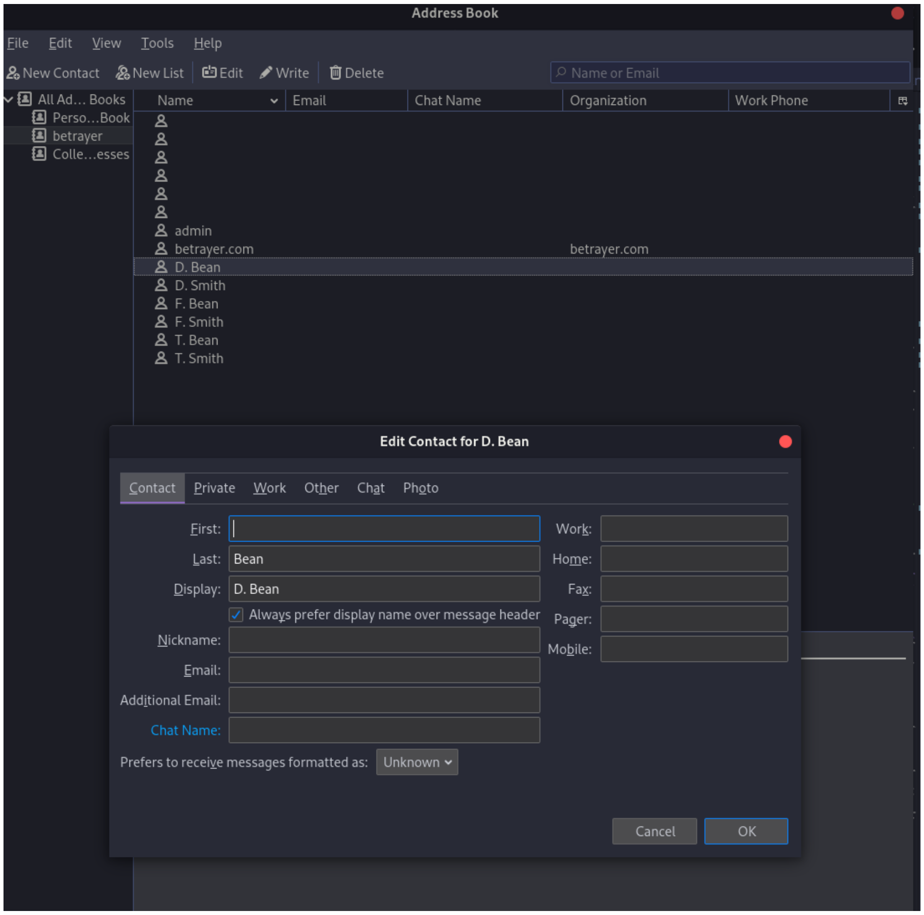
○ Ignore Referrals

3.2.6 Extending an existing entry

The entry `uid=bean,ou=devel,ou=software,ou=departments,dc=betrayer;dc=com` may be extended by the `objectclass=posixAccount`. Construct a LDIF file to add the attributes `uidNumber`, `gidNumber` and `homeDirectory` by a modify/add operation:

```
uid=bean, ou=devel, ou=software, ou=departments, dc=betrayer, dc=com
changetype: add
objectClass: posixAccount
uidNumber: 42
gidNumber: 1337
homeDirectory: /home/daniel
```

3.2.7 Accessing LDAP data by a mail client



Name	Email	Chat Name	Organization	Work Phone
admin				
betrayer.com			betrayer.com	
D. Bean				
D. Smith				
F. Bean				
F. Smith				
T. Bean				
T. Smith				

First:

Last: Bean

Display: D. Bean

☑ Always prefer display name over message header

Nickname:

Email:

Additional Email:

Chat Name:

Work:

Home:

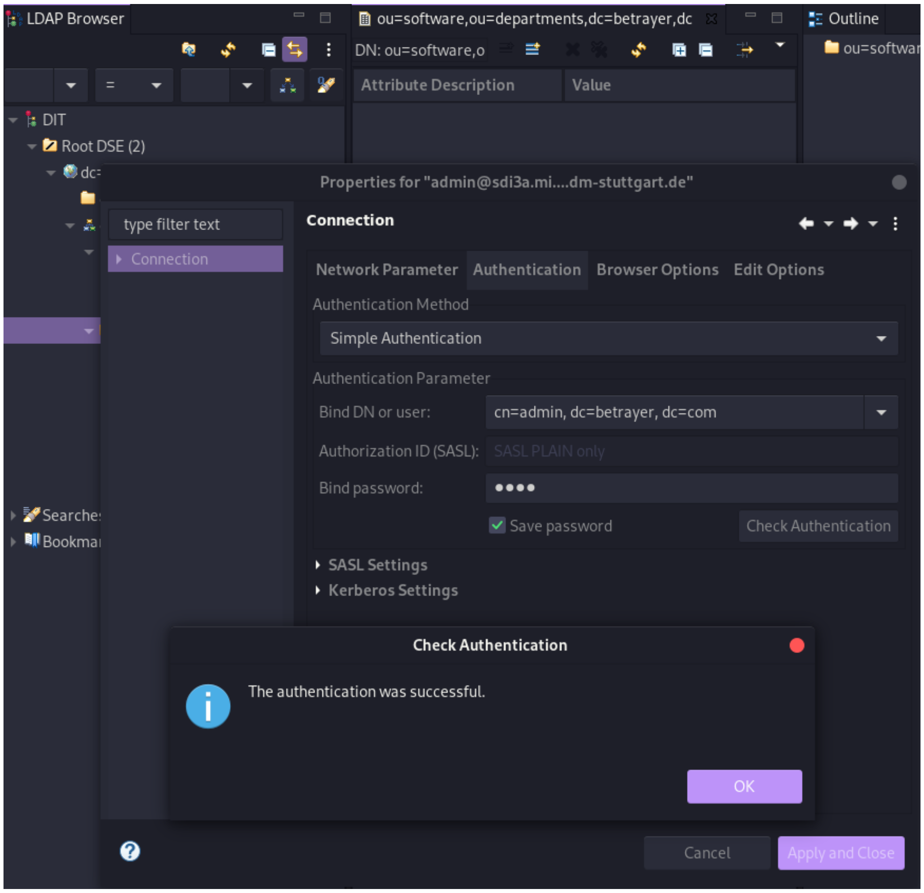
Fax:

Pager:

Mobile:

Prefers to receive messages formatted as: Unknown

3.2.8 LDAP configuration



Properties for "admin@sdi3a.mi....dm-stuttgart.de"

Connection

Authentication Method: Simple Authentication

Bind DN or user: cn=admin, dc=betrayer, dc=com

Authorization ID (SASL): SASL PLAIN only

Bind password: ●●●●

☑ Save password

Check Authentication

The authentication was successful.

3.2.9 LDAP based user login

3.2.9.1 Test connection to active directory

Use the following command:

```
$ root@sdi3b:~# telnet sdi3a.mi.hdm-stuttgart.de 389
```

Then something like this should appear:

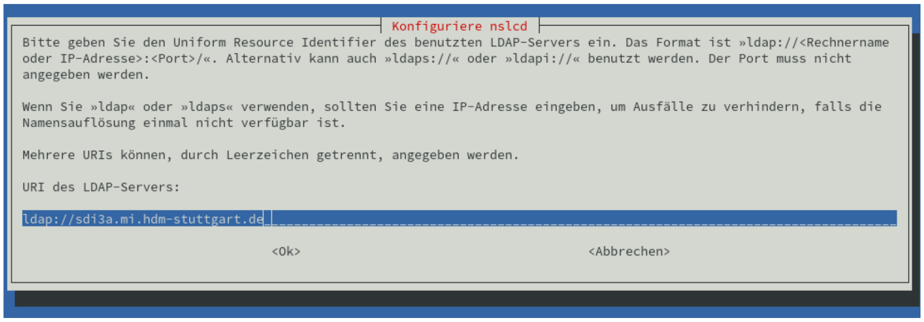
```
Trying 141.62.75.103...
Connected to sdi3a.mi.hdm-stuttgart.de.
Escape character is '^]'.
```

3.2.9.2 Install and configure libpam-ldapd

```
$ apt-get install libpam-ldapd
```

After the installation a window will open, where we can configure the package.

In the following window we need to enter the hostname to our active directories.



Konfiguriere nslcd

Bitte geben Sie den Uniform Resource Identifier des benutzten LDAP-Servers ein. Das Format ist »ldap://<Rechnername oder IP-Adresse>:<Port>/«. Alternativ kann auch »ldaps://« oder »ldapi://« benutzt werden. Der Port muss nicht angegeben werden.

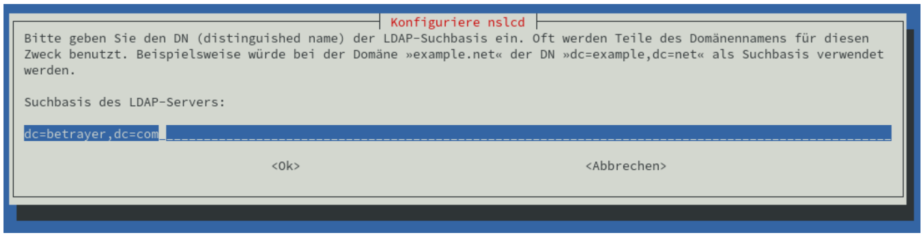
Wenn Sie »ldap« oder »ldaps« verwenden, sollten Sie eine IP-Adresse eingeben, um Ausfälle zu verhindern, falls die Namensauflösung einmal nicht verfügbar ist.

Mehrere URIs können, durch Leerzeichen getrennt, angegeben werden.

URI des LDAP-Servers:

ldap://sdi3a.mi.hdm-stuttgart.de

After that we need to enter the distinguished name.

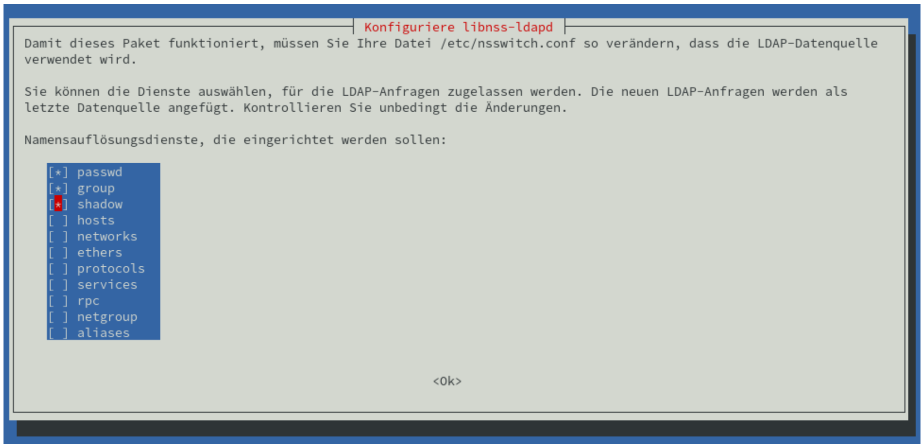


Konfiguriere nslcd

Bitte geben Sie den DN (distinguished name) der LDAP-Suchbasis ein. Oft werden Teile des Domänennamens für diesen Zweck benutzt. Beispielsweise würde bei der Domäne »example.net« der DN »dc=example,dc=net« als Suchbasis verwendet werden.

Suchbasis des LDAP-Servers:

dc=betrayer,dc=com



Konfiguriere libnss-ldapd

Damit dieses Paket funktioniert, müssen Sie Ihre Datei /etc/nsswitch.conf so verändern, dass die LDAP-Datenquelle verwendet wird.

Sie können die Dienste auswählen, für die LDAP-Anfragen zugelassen werden. Die neuen LDAP-Anfragen werden als letzte Datenquelle angefügt. Kontrollieren Sie unbedingt die Änderungen.

Namensauflösungsdienste, die eingerichtet werden sollen:

[*] passwd
[*] group
[*] shadow
[ ] hosts
[ ] networks
[ ] ethers
[ ] protocols
[ ] services
[ ] rpc
[ ] netgroup
[ ] aliases

After the configuration the installation of the package will be finished and we need to reboot our server.

After that we can run request

```
id daniel
uid=42(daniel) gid=1337 Gruppen=1337
```

In the last step we need to create a user and a group accordingly, which we need to assign to the user:

```
$ groupadd -g 1337 betrayer_software_devel
$ useradd -u 42 daniel
$ usermod -g betrayer_software_devel daniel
$ mkhomedir_helper daniel
```

3.2.10 Backup and recovery / restore

Create a backup of the OpenLDAP database configuration in a LDIF-file.

```
$ slapcat -b cn=config -l ldap-config.ldif
```

Create a backup of the OpenLDAP data.

```
$ slapcat -l ldap-data.ldif
```

Copy the data and configuration backup from the OpenLDAP provider server to the OpenLDAP consumer server.

```
$ scp {ldap-data.ldif,ldap-config.ldif} root@sdi3b.mi.hdm-stuttgart.de
```

Now we need to access our consumer server via ssh.

```
$ ssh root@sdi3b.mi.hdm-stuttgart.de
```

Restore the OpenLDAP provider Data and configs on the consumer server. Stop the LDAP service:

```
$ systemctl stop slapd
```

Ensure that the LDAP configuration and data directories are empty:

```
$ rm -rf /etc/ldap/slapd.d/*
$ rm -rf /var/lib/ldap/*
```

Restore the configuration backup:

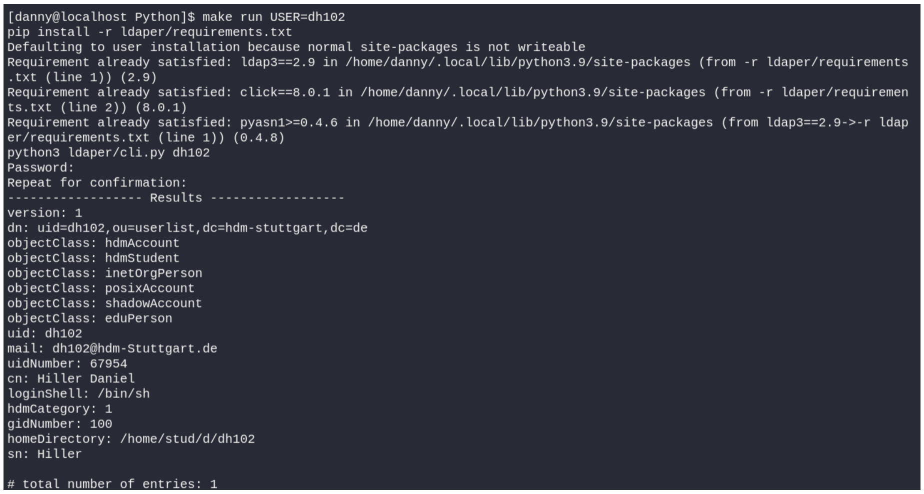
```
$ slapadd -b cn=config -l /root/ldap-config.ldif -F /etc/ldap/slapd.d/
```

Restore the LDAP data directories:

```
$ slapadd -n 1 -l /root/ldap-data.ldif -F /etc/ldap/slapd.d/
```

3.2.11 Accessing LDAP by a Pyhton application.

Please find the `application` and the associated `README.md` in the Python directory.



```
[danny@localhost Python]$ make run USER=dh102
pip install -r ldaper/requirements.txt
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: ldap3==2.9 in /home/danny/.local/lib/python3.9/site-packages (from -r ldaper/requirements
.txt (line 1)) (2.9)
Requirement already satisfied: click==8.0.1 in /home/danny/.local/lib/python3.9/site-packages (from -r ldaper/requiremen
ts.txt (line 2)) (8.0.1)
Requirement already satisfied: pyasn1>=0.4.6 in /home/danny/.local/lib/python3.9/site-packages (from ldap3==2.9->-r ldap
er/requirements.txt (line 1)) (0.4.8)
python3 ldaper/cli.py dh102
Password:
Repeat for confirmation:
----------------- Results -----------------
version: 1
dn: uid=dh102,ou=userlist,dc=hdm-stuttgart,dc=de
objectClass: hdmAccount
objectClass: hdmStudent
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
objectClass: eduPerson
uid: dh102
mail: dh102@hdm-Stuttgart.de
uidNumber: 67954
cn: Hiller Daniel
loginShell: /bin/sh
hdmCategory: 1
gidNumber: 100
homeDirectory: /home/stud/d/dh102
sn: Hiller

# total number of entries: 1
```

The following framework were used:

`https://www.python-ldap.org/en/python-ldap-3.3.0/`

`https://click.palletsprojects.com/en/8.0.x/`

4 Apache Web Server

4.1 Exercises

For the following tasks we need the package `apache2`, which we can install with the following command:

```
$ aptitude install apache2
```

4.2 First Steps

1. After we install the package apache is running per default and can in our case be queried with `http://sdi3a.mi.hdm-stuttgart.de/`.
2. When we move the index.html file out of the directory we can discover another page, for this we need to query the adress again. Now we can se an empty table and below that we find the version of our Apache Server, the domain where its hosted and the associated port.
3. In the next step we povide our own simple webpage which looks like the following:

```
<!DOCTYPE html>
<html>
  <body>
    <h1>TEST</h1>
  </body>
</html>
```

4. In the next step we install the apache2 documentation with the following command:

```
$ apt install apache2-doc
```

In our case we can find all related files from the packe `apache2-doc`:

```
$ dpkg -L apache2-doc
```

The result is a huge list of file which all belongs to the following path: `/usr/share/doc/apache2-doc/manual/`

5. In the last task we want to host our documentation on our web server. But first we need to convert our .md to valid .html, which can be done with the pandoc package:

```
$ docker run -v "${PWD}:/data:z" pandoc/latex doku.md --number-sections --toc --toc-depth=6
```

We want to store the index.html later in `home/sdidoc` so we need to create this directory:

```
$ cd /home
$ mkdir sdidoc
```

Now we can transfer our file from the local machine to our server:

```
$ scp index.html root@sdi3a.mi.hdm-stuttgart.de:/home/sdidoc/
```

Last but not least we need to adjust our config file in `/etc/apache2/sites-available/000-default.conf` with the following terms:

```
<Directory /home/sdidoc>
  Options Indexes FollowSymLinks Includes ExecCGI
  AllowOverride All
  Require all granted
  Allow from all
</Directory>
```

To make our change effective we need to restart the apache web service:

```
$ systemctl reload apache2
```

4.3 Virtual hosts

To realize virtual hosts we need to create a .con file in `/etc/apache2/sites-available`, the config in this file should look like the following:

```
  <VirtualHost *:80>
      ServerAdmin dh102@hdm-stuttgart.de
      ServerName sdi3a.mi.hdm-stuttgart.de
      ServerAlias dh102.sdi3a.mi.hdm-stuttgart.de
      DocumentRoot /home/sdidoc/
      ErrorLog ${APACHE_LOG_DIR}/error.log
      CustomLog ${APACHE_LOG_DIR}/access.log combined
  </VirtualHost>
```

Now the side must be enabled with `$ a2ensite dh102.conf` and add the follwing instructions to `/etc/apache2/apache2.conf`:

```
<Directory /home/sdidoc/>
        AllowOverride None
        Require all granted
        Options Indexes FollowSymLinks
</Directory>
```

Now it is important to grant apache2 the access to the directory where our `index.html` is placed: `$ chown -R www-data /home/sdidoc`

To access the webpage from a local machine, we need to give our local machine the relevant information to reach the page. This can be done by enter the information on our local machine with `$ sudo vim /etc/hosts`:

```
141.62.75.103 sdi3a.mi.hdm-stuttgart.de dh102.sdi3a.mi.hdm-stuttgart.de
```

To setup the `manual.sdi3a.mi.hdm-stuttgart.de` we can copy our first .conf file, enable it and register the information on localhost.

4.4 SSL / TLS Support

The first step ist that we need to create our private root key whith a bit length of 2048:

```
$ openssl genrsa -out rootCA.key 2048
```

For security reasons we should encrypt our key:

```
$ openssl genrsa -des3 -out rootCA.key 2048
```

With our `rootCA.key` we can now self-sign a certificate:

```
$ openssl req -x509 -new -nodes -key rootCA.key -sha256 -days 1024 -out rootCA.pem
```

The above command starts an interactive script, which in our case looked like the following after processing:

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:DE
```

```
State or Province Name (full name) [Some-State]:Baden Wuerttemberg
Locality Name (eg, city) []:Stuttgart
Organization Name (eg, company) [Internet Widgits Pty Ltd]:HdM
Organizational Unit Name (eg, section) []:MI
Common Name (eg, YOUR name) []:manual.sdi3a.mi.hdm-stuttgart.de
Email Address []:dh102@hdm-stuttgart.de
```

To access our created certificate we can transfer the file via scp from the server to our local machine:

```
$ scp root@sdi3a.mi.hdm-stuttgart.de:/root/ssl-cert/rootCA.pem /home/user/certificates/
```

To import the root ca on the local machine:

```
$ sudo cp /home/user/certificates/rootCA.pem /etc/pki/ca-trust/source/anchors/sdi3a
$ sudo update-ca-trust
```

In the next step we need to create a certificate for our webpage. We starting again with the key:

```
$ openssl genrsa -out device.key 2048
```

Now we can create our webpage certificate:

```
$ openssl req -new -key device.key -out device.csr
```

The interactive script starts again and we go through it pretty much the same as before.

Now that we have our CA and the device certificate we are able to sign it:

```
openssl x509 -req -in device.csr -CA rootCA.pem -CAkey rootCA.key -CAcreateserial -out devic
```

Enabling the apache SSL module:

```
$ a2enmod ssl
```

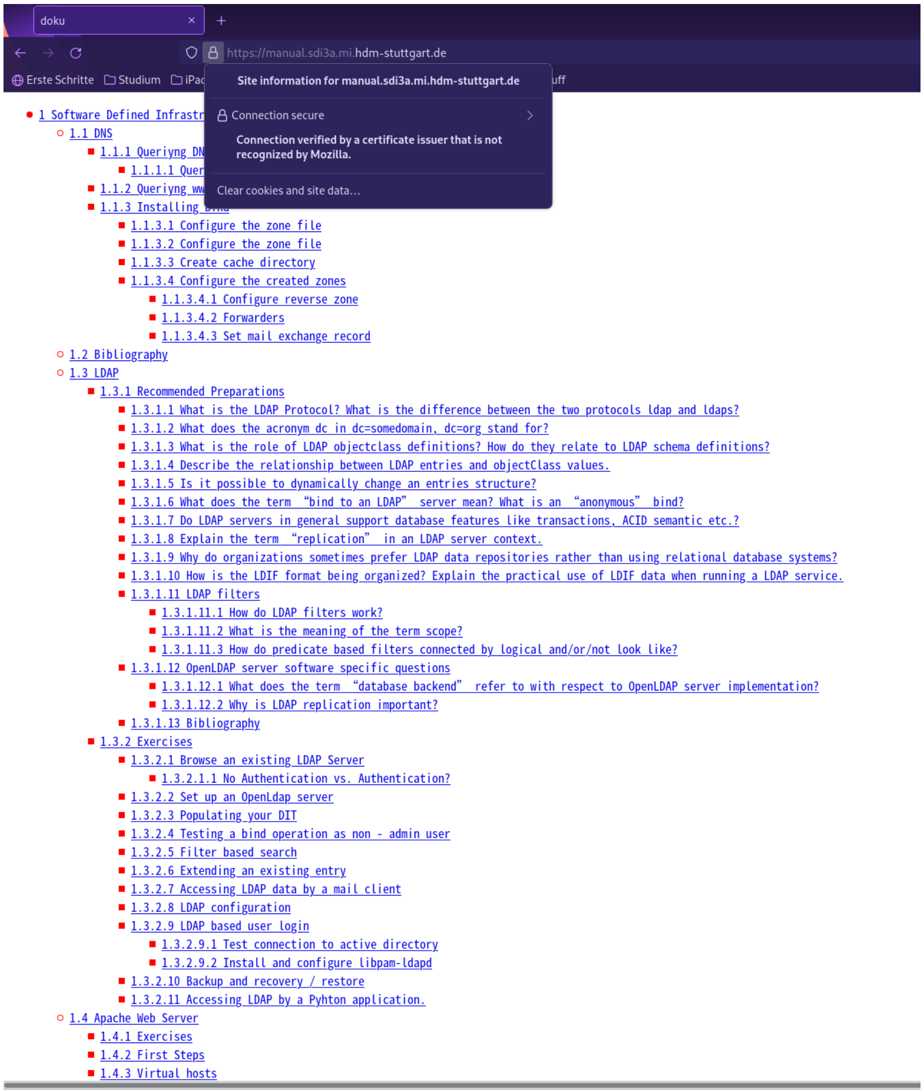
In the last step we need to adjust our configuration from the previous task `/etc/apache2/sites-available/manual.conf`:

```
<VirtualHost *:443>
    ServerAdmin dh102@hdm-stuttgart.de
    ServerName sdi3a.mi.hdm-stuttgart.de
    ServerAlias manual.sdi3a.mi.hdm-stuttgart.de
    DocumentRoot /home/sdidoc/
    SSLEngine on
    SSLCertificateFile "/root/ssl-cert/device.crt"
    SSLCertificateKeyFile "/root/ssl-cert/device.key"
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

To make the change effective we need to restart the service:

```
systemctl restart apache2.service
```

Now the Connection is secure:


doku
https://manual.sdi3a.mi.hdm-stuttgart.de
Erste Schritte
Studium
iPad
uff
Clear cookies and site data…

Site information for manual.sdi3a.mi.hdm-stuttgart.de

Connection secure

Connection verified by a certificate issuer that is not recognized by Mozilla.

- 1 Software Defined Infrastr
  - 1.1 DNS
    - 1.1.1 Queriyng DN
      - 1.1.1.1 Quer
    - 1.1.2 Queriyng ww
    - 1.1.3 Installing
      - 1.1.3.1 Configure the zone file
      - 1.1.3.2 Configure the zone file
      - 1.1.3.3 Create cache directory
      - 1.1.3.4 Configure the created zones
        - 1.1.3.4.1 Configure reverse zone
        - 1.1.3.4.2 Forwarders
        - 1.1.3.4.3 Set mail exchange record
  - 1.2 Bibliography
  - 1.3 LDAP
    - 1.3.1 Recommended Preparations
      - 1.3.1.1 What is the LDAP Protocol? What is the difference between the two protocols ldap and ldaps?
      - 1.3.1.2 What does the acronym dc in dc=somedomain, dc=org stand for?
      - 1.3.1.3 What is the role of LDAP objectclass definitions? How do they relate to LDAP schema definitions?
      - 1.3.1.4 Describe the relationship between LDAP entries and objectClass values.
      - 1.3.1.5 Is it possible to dynamically change an entries structure?
      - 1.3.1.6 What does the term “bind to an LDAP” server mean? What is an “anonymous” bind?
      - 1.3.1.7 Do LDAP servers in general support database features like transactions, ACID semantic etc.?
      - 1.3.1.8 Explain the term “replication” in an LDAP server context.
      - 1.3.1.9 Why do organizations sometimes prefer LDAP data repositories rather than using relational database systems?
      - 1.3.1.10 How is the LDIF format being organized? Explain the practical use of LDIF data when running a LDAP service.
      - 1.3.1.11 LDAP filters
        - 1.3.1.11.1 How do LDAP filters work?
        - 1.3.1.11.2 What is the meaning of the term scope?
        - 1.3.1.11.3 How do predicate based filters connected by logical and/or/not look like?
      - 1.3.1.12 OpenLDAP server software specific questions
        - 1.3.1.12.1 What does the term “database backend” refer to with respect to OpenLDAP server implementation?
        - 1.3.1.12.2 Why is LDAP replication important?
      - 1.3.1.13 Bibliography
    - 1.3.2 Exercises
      - 1.3.2.1 Browse an existing LDAP Server
        - 1.3.2.1.1 No Authentication vs. Authentication?
      - 1.3.2.2 Set up an OpenLdap server
      - 1.3.2.3 Populating your DIT
      - 1.3.2.4 Testing a bind operation as non - admin user
      - 1.3.2.5 Filter based search
      - 1.3.2.6 Extending an existing entry
      - 1.3.2.7 Accessing LDAP data by a mail client
      - 1.3.2.8 LDAP configuration
      - 1.3.2.9 LDAP based user login
        - 1.3.2.9.1 Test connection to active directory
        - 1.3.2.9.2 Install and configure libpam-ldapd
      - 1.3.2.10 Backup and recovery / restore
      - 1.3.2.11 Accessing LDAP by a Pyhton application.
  - 1.4 Apache Web Server
    - 1.4.1 Exercises
    - 1.4.2 First Steps
    - 1.4.3 Virtual hosts

4.5 LDAP authentication

For this exercises we use our user “daniel” from 2.2.9 LDAP based user login.

To use LDAP with Apache Web Server, we need to enable the module `authnz_ldap`:

```
$ a2enmod authnz_ldap
```

We can copy one of our previous .conf files and edit the config, which should look like the following:

```
<VirtualHost *:443>
    ServerAdmin dh102@hdm-stuttgart.de
    DocumentRoot /home/sdidoc/
    SSLEngine on
    SSLCertificateFile "/root/ssl-cert/device.crt"
    SSLCertificateKeyFile "/root/ssl-cert/device.key"
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
  <Directory "/home/sdidoc">
      Options Indexes FollowSymlinks
      AuthType Basic
      AuthName "Apache LDAP authentication"
      AuthBasicAuthoritative Off
      AuthBasicProvider ldap
      AuthLDAPURL "ldap://141.62.75.103/uid=daniel,ou=devel,ou=software,ou=departments,dc=be
      AuthLDAPBindDN "uid=daniel,ou=devel,ou=software,ou=departments,dc=betrayer,dc=com"
      AuthLDAPBindPassword test1
      Require valid-user
  </Directory>
</VirtualHost>
```

Enabling the site and restart apache web server.

```
$ a2ensite daniel.conf
$ systemctl restart apache2.service
```

Now it should be possible to enter `https://sdi3a.mi.hdm-stuttgart.de/test` in our browser and login.

4.6 Mysql™ database administration

To install `mysql-server` use:

```
$ apt install default-mysql-server
```

After facing a issue with LXC Container we need to adjust the config `/etc/systemd/system/mariadb.service.d/lxc.conf`:

```
[Service]
ProtectHome=false
ProtectSystem=false

# These settings turned out to not be necessary in my case, but YMMV
#PrivateTmp=false
#PrivateNetwork=false
PrivateDevices=false
```

And run the follwing commands:

```
$ systemctl daemon-reload
$ systemctl restart mariadb
```

To install `php` just enter:

```
$ apt install php
```

To install phpMyadmin we used a buster backport becuase apt didn't know any package with the name `phpmyadmin`: For this we need to create an apt source file `/etc/apt/sources.list.d/buster-backports.list` and add:

```
deb http://deb.debian.org/debian buster-backports main
```

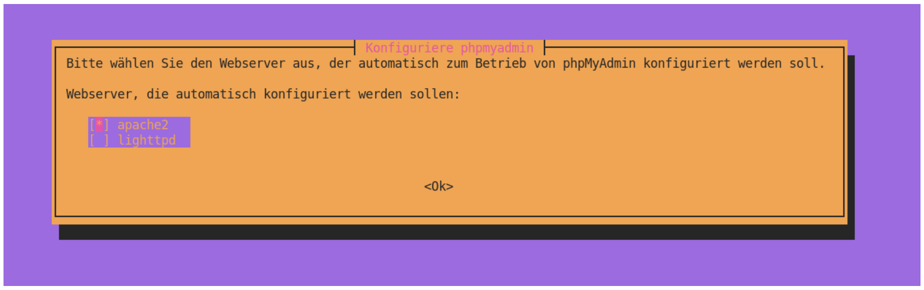
Now we need to refresh the package cache and install `php-twig`:

```
$ apt-get update
$ apt-get install -t buster-backports php-twig
```

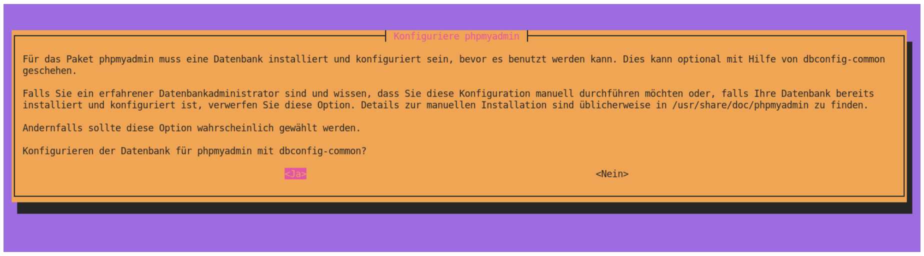
And finally we can install `phpMyAdmin`:

```
$ apt-get install -t buster-backports phpmyadmin
```

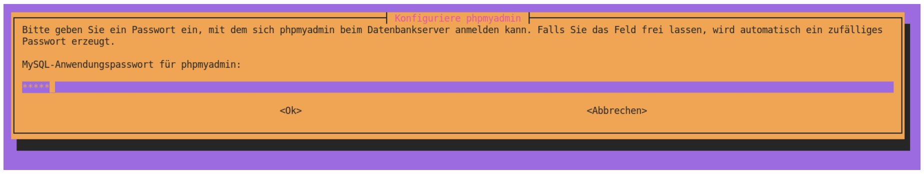
During the installation a dialog should open up:



```
Konfiguriere phpmyadmin
Bitte wählen Sie den Webserver aus, der automatisch zum Betrieb von phpMyAdmin konfiguriert werden soll.

Webserver, die automatisch konfiguriert werden sollen:

  [*] apache2
  [ ] lighttpd
```



```
Konfiguriere phpmyadmin
Für das Paket phpmyadmin muss eine Datenbank installiert und konfiguriert sein, bevor es benutzt werden kann. Dies kann optional mit Hilfe von dbconfig-common
geschehen.

Falls Sie ein erfahrener Datenbankadministrator sind und wissen, dass Sie diese Konfiguration manuell durchführen möchten oder, falls Ihre Datenbank bereits
installiert und konfiguriert ist, verwerfen Sie diese Option. Details zur manuellen Installation sind üblicherweise in /usr/share/doc/phpmyadmin zu finden.

Andernfalls sollte diese Option wahrscheinlich gewählt werden.

Konfigurieren der Datenbank für phpmyadmin mit dbconfig-common?
```



```
Konfiguriere phpmyadmin
Bitte geben Sie ein Passwort ein, mit dem sich phpmyadmin beim Datenbankserver anmelden kann. Falls Sie das Feld frei lassen, wird automatisch ein zufälliges
Passwort erzeugt.

MySQL-Anwendungspasswort für phpmyadmin:

*****
```

Now we need to create a user with which we can log in:

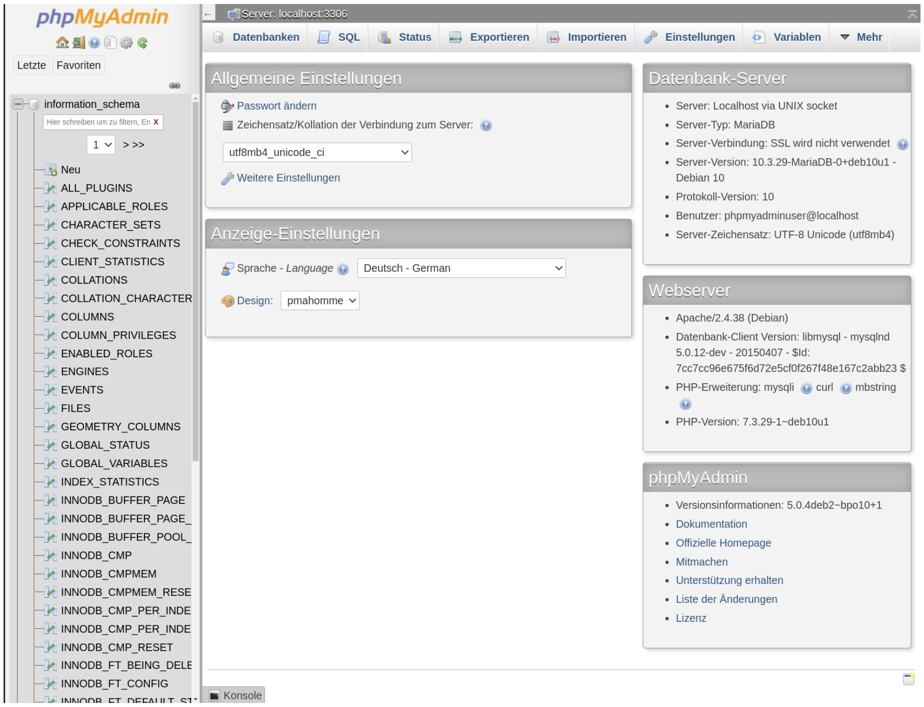
```
$ mariadb
```

```
> CREATE USER 'phpmyadminuser'@'localhost' IDENTIFIED BY 'test1';
```

And restart Apach2:

```
$ systemctl restart apache2.service
```

Last but not least we can open the following domain and login `http://sdi3a.mi.hdm-stuttgart.de/phpmyadm`



Allgemeine Einstellungen

Passwort ändern

Zeichensatz/Kollation der Verbindung zum Server: utf8mb4_unicode_ci

Weitere Einstellungen

Anzeige-Einstellungen

Sprache - *Language*: Deutsch - German

Design: pmahomme

Datenbank-Server

- Server: Localhost via UNIX socket
- Server-Typ: MariaDB
- Server-Verbindung: SSL wird nicht verwendet
- Server-Version: 10.3.29-MariaDB-0+deb10u1 - Debian 10
- Protokoll-Version: 10
- Benutzer: phpmyadminuser@localhost
- Server-Zeichensatz: UTF-8 Unicode (utf8mb4)

Webserver

- Apache/2.4.38 (Debian)
- Datenbank-Client Version: libmysql - mysqlnd 5.0.12-dev - 20150407 - $Id: 7cc7cc96e675f6d72e5cf0f267f48e167c2abb23 $
- PHP-Erweiterung: mysqli curl mbstring
- PHP-Version: 7.3.29-1~deb10u1

phpMyAdmin

- Versionsinformationen: 5.0.4deb2~bpo10+1
- Dokumentation
- Offizielle Homepage
- Mitmachen
- Unterstützung erhalten
- Liste der Änderungen
- Lizenz

4.7 Providing WEB based user management to your LDAP Servern

To install the LDAP Account Manager we need to download it and forward it to the server via `scp` because `ldap-account-manager` isn't availabel via the official `apt` repositorys:

`https://sourceforge.net/projects/lam/`

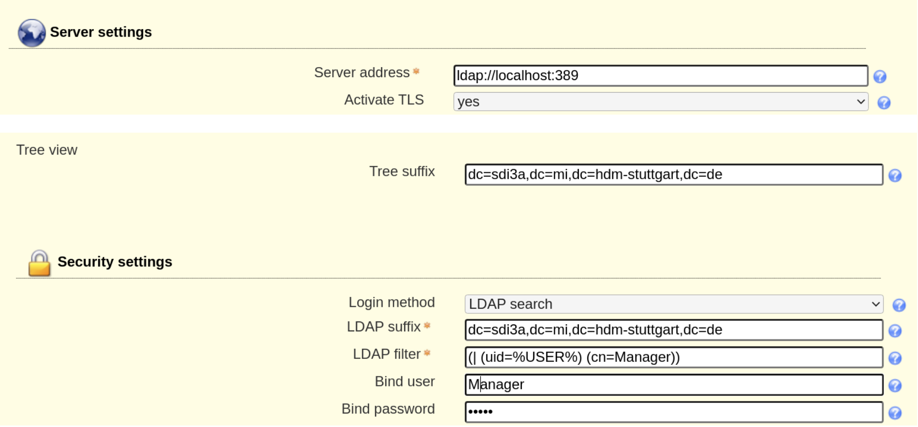
```
$ scp /home/user/Downloads/ldap-account-manager_7.6-1_all.deb root@sdi3a.mi.hdm-stuttgart.de
```

... and install it with `apt`:

```
$ apt install /home/ldap-account-manager_7.6-1_all.deb
```

Now we can configure the LDAP Account Manager `http://sdi3a.mi.hdm-stuttgart.de/lam/templates/con` The default master password for `Edit general settings` is `lam` and should be changed to something secure.

The password for `Edit server profiles` is also `lam`. Here we can can edit `TLS` and a `List of valid users`:



Server settings

Server address * ldap://localhost:389

Activate TLS yes

Tree view

Tree suffix dc=sdi3a,dc=mi,dc=hdm-stuttgart,dc=de

Security settings

Login method LDAP search

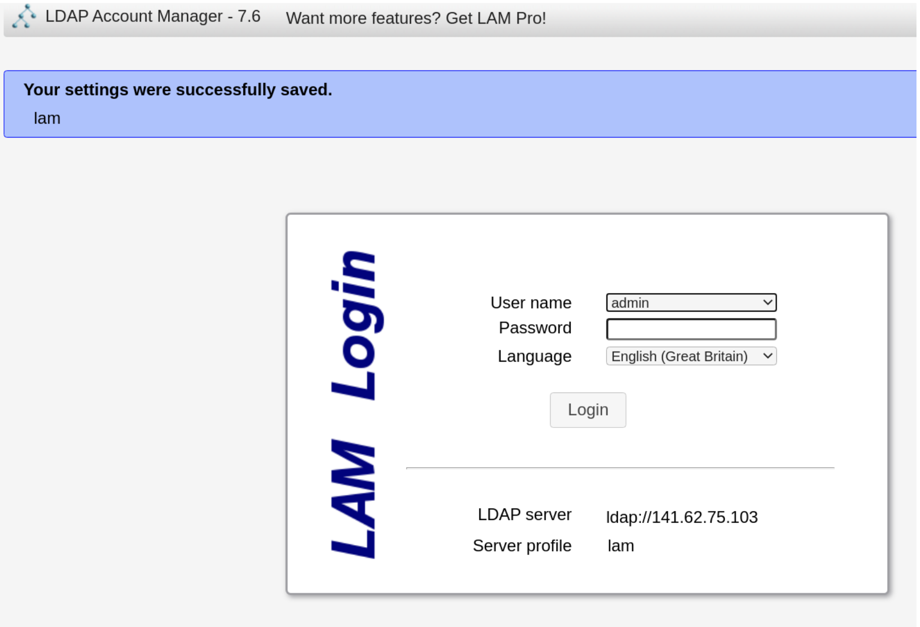
LDAP suffix * dc=sdi3a,dc=mi,dc=hdm-stuttgart,dc=de

LDAP filter * (| (uid=%USER%) (cn=Manager))

Bind user Manager

Bind password •••••

After saving this settings we are able to so the our user:



Your settings were successfully saved.

lam

User name admin

Password

Language English (Great Britain)

LDAP server ldap://141.62.75.103

Server profile lam

4.8 Publish your documentation

Our documentation is written as a .md-file, so we need to convert it with pandoc into a valid .html-file:

```
$ docker run -v "${PWD}:/data:z" pandoc/latex doku.md --number-sections --toc --toc-depth=6
```

Now we transfer the .html-file to our server, which can be done with `scp`:

```
$ scp index.html root@sdi3a.mi.hdm-stuttgart.de:/home/sdidoc/
```

We doesn't use rsync because we anyway need to convert our file with pandoc to get an actual version. But if you want to use rsync the command would be:

```
$ rsync -avz -e ssh root@sdi3a.mi.hdm-stuttgart.de:/home/sdidoc/
```

We can adjust the .conf-file `etc/apache2/apache2.conf/` and and add:

```
<Directory /home/sdidoc/>
        AllowOverride None
        Require all granted
        Options Indexes FollowSymLinks
</Directory>

Alias /doc /home/sdidoc/
```

Now we can query `http://sdi3a.mi.hdm-stuttgart.de/doc/`.

5 File Cloud

5.1 Exercises

5.1.1 Setup Nextcloud with Apache Web Server

First we need to install packages for apache, mariadb and php:

```
$ apt install vim unzip
$ apt install apache2 mariadb-server libapache2-mod-php
$ apt install php-gd php-json php-mysql php-curl
$ apt install php-intl php-mcrypt php-imagick
$ apt install php-zip php-xmlwriter php-xmlreader php-xml php-mbstring php-simplexml
```

We need another user for our nextcloud in our databse:

```
$ mariadb
> CREATE USER 'ncadmin'@'localhost' IDENTIFIED BY 'test1';
> CREATE DATABASE IF NOT EXISTS nextcloud CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
> GRANT ALL PRIVILEGES ON nextcloud.* TO 'ncadmin'@'localhost';
> FLUSH PRIVILEGES;
> quit;
```

In the next step we download nextcloud and move it to `/var/www`:

```
$ wget https://download.nextcloud.com/server/releases/latest.zip
$ unzip latest.zip
$ mv nextcloud/ /var/www
```

Add the following lines to `/etc/apache2/apache2.conf`:

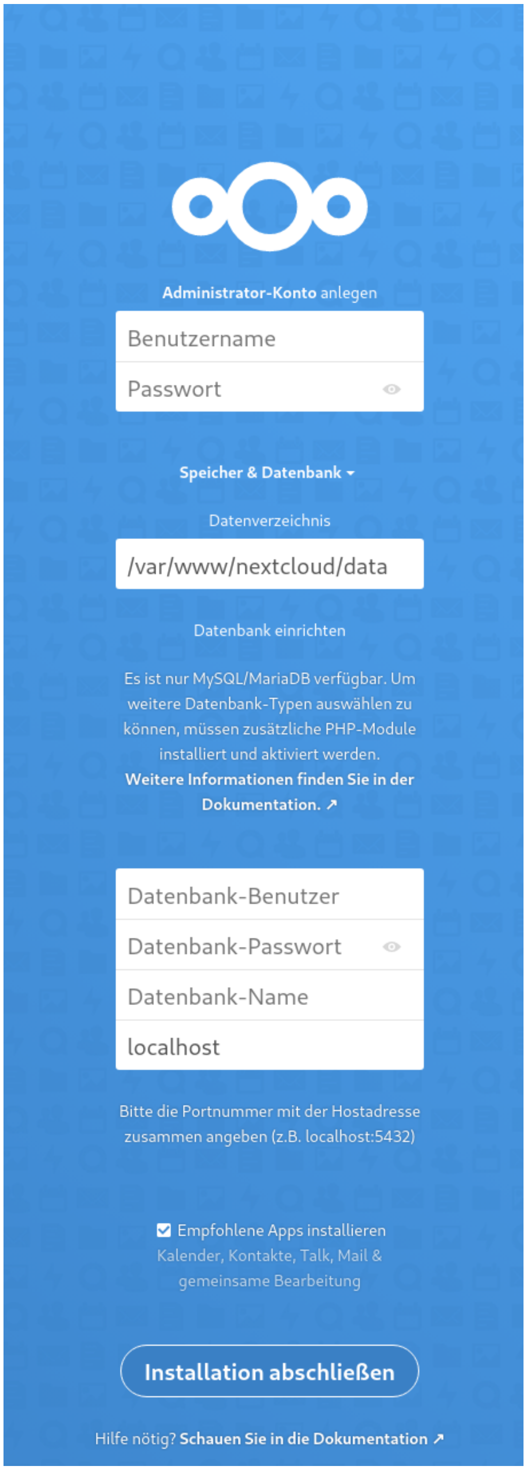
```
<Directory /var/www/nextcloud/>
Require all granted
AllowOverride All
Options FollowSymLinks MultiViews
</Directory>
Alias /nextcloud "/var/www/nextcloud/"
```

Give apache2 the permissions on the folder: `$ chown -R www-data /var/www/nextcloud/`

Enable the follwing modules and restart apache2:

```
$ a2enmod rewrite
$ a2enmod headers
$ a2enmod env
$ a2enmod dir
$ a2enmod mime

$ systemctl restart apache2.service
```

Now we can open in our browser `sdi3a.mi.hdm-stuttgart.de/nextcloud` which should look like the following:



/var/www/nextcloud/data

Es ist nur MySQL/MariaDB verfügbar. Um weitere Datenbank-Typen auswählen zu können, müssen zusätzliche PHP-Module installiert und aktiviert werden. **Weitere Informationen finden Sie in der Dokumentation. ↗**

localhost

Bitte die Portnummer mit der Hostadresse zusammen angeben (z.B. localhost:5432)

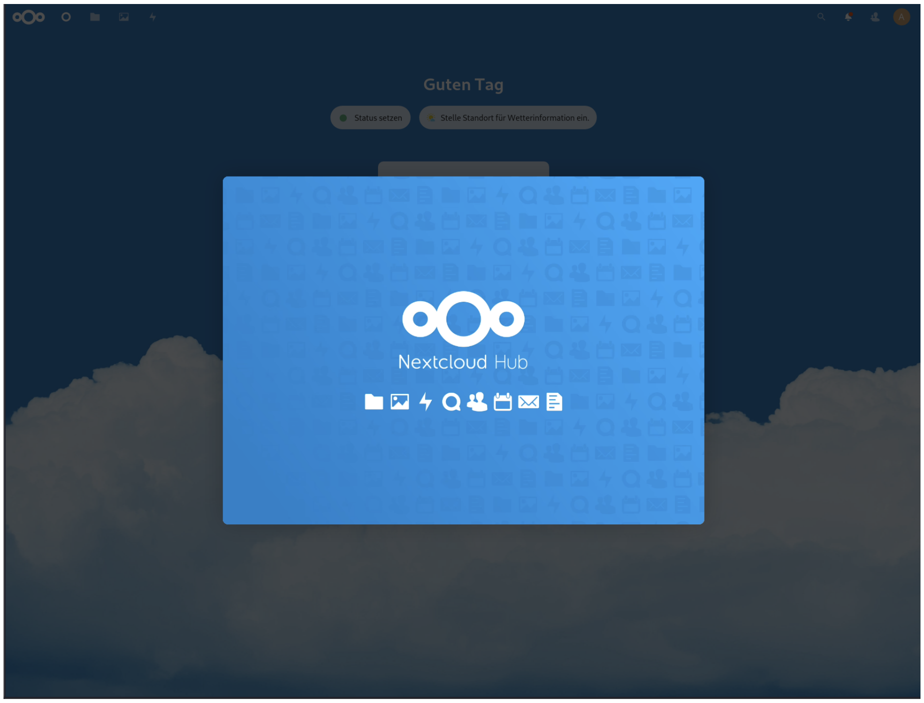
☑ Empfohlene Apps installieren
Kalender, Kontakte, Talk, Mail & gemeinsame Bearbeitung

Hilfe nötig? **Schauen Sie in die Dokumentation ↗**

To finish the installation type in the necessary data and click `Installation abschließen`.

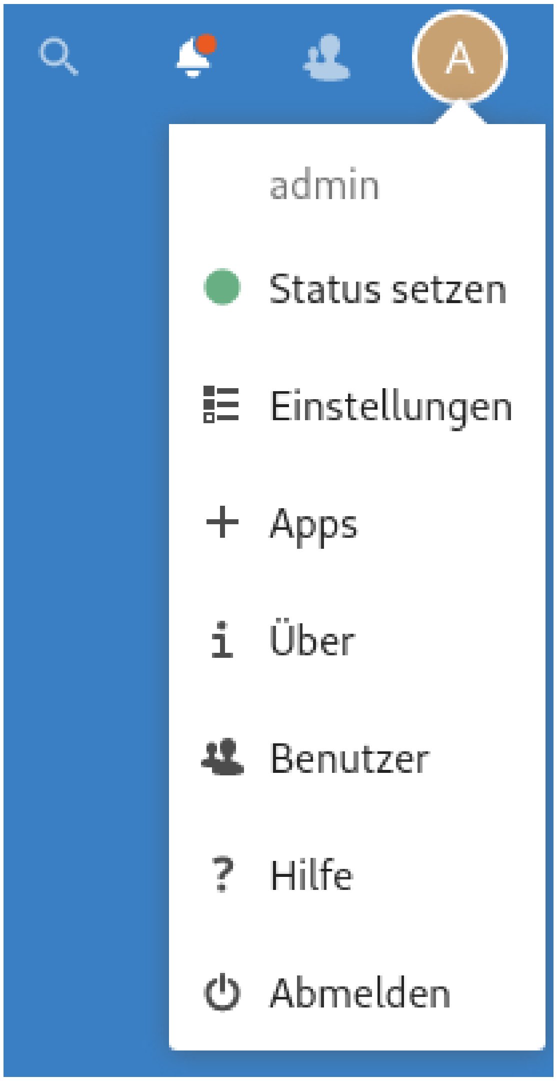
```
User = "admin"
Password = "test1"

Database-User = "ncadmin"
Database-User = "test1"
Database-Name = "nextcloud"
```

After we waiting a bit we can enter again `sdi3a.mi.hdm-stuttgart.de/nextcloud` and now it should look like the screenshot below:

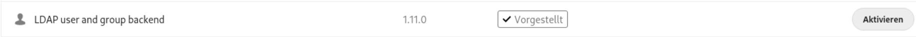


5.1.2 User authentication with LDAP

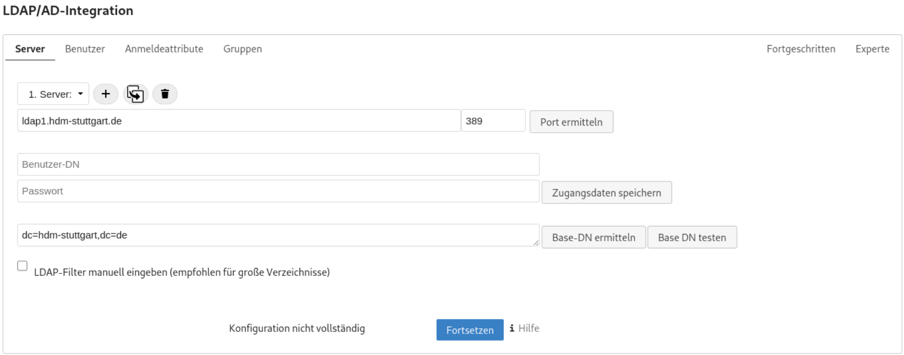
To enable ldap support click on Icon in right top corner then go to `Apps`:



Search for the module `LDAP user and group backend` and enable it:



Now we can configure ldap under settings and `LDAP/AD-Integration`:


LDAP/AD-Integration

ldap1.hdm-stuttgart.de 389

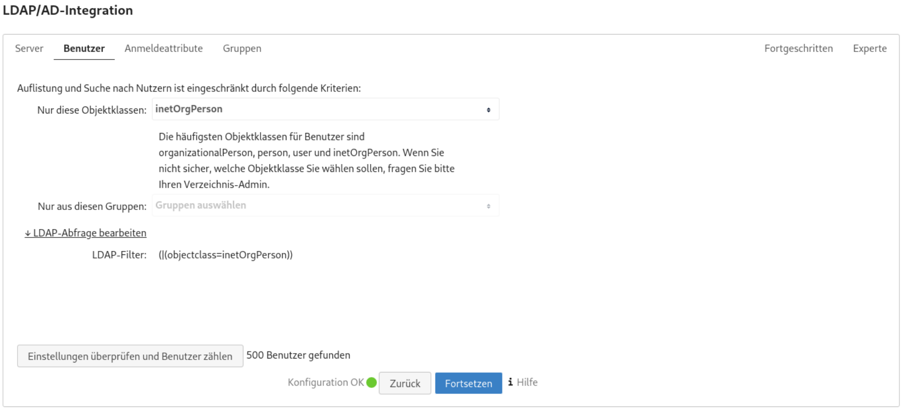
Benutzer-DN

Passwort

dc=hdm-stuttgart,dc=de

□ LDAP-Filter manuell eingeben (empfohlen für große Verzeichnisse)

Konfiguration nicht vollständig


LDAP/AD-Integration

Auflistung und Suche nach Nutzern ist eingeschränkt durch folgende Kriterien:

Nur diese Objektklassen: inetOrgPerson

Die häufigsten Objektklassen für Benutzer sind organizationalPerson, person, user und inetOrgPerson. Wenn Sie nicht sicher, welche Objektklasse Sie wählen sollen, fragen Sie bitte Ihren Verzeichnis-Admin.

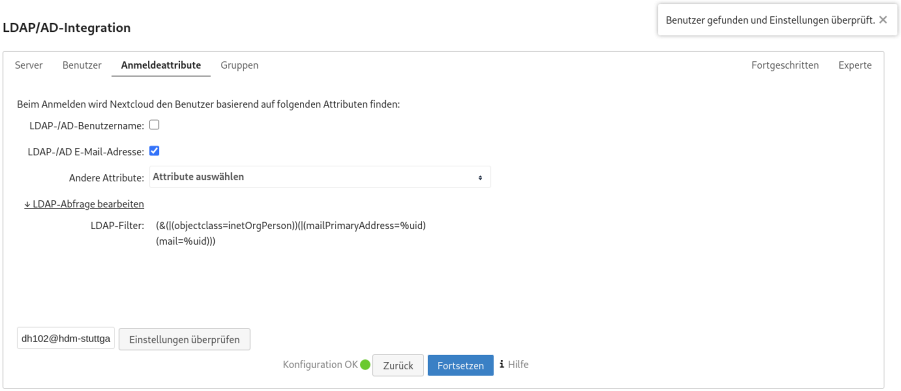
Nur aus diesen Gruppen: Gruppen auswählen

↓ LDAP-Abfrage bearbeiten

LDAP-Filter: (|(objectclass=inetOrgPerson))

500 Benutzer gefunden

Konfiguration OK ●


LDAP/AD-Integration

Benutzer gefunden und Einstellungen überprüft. ×

Beim Anmelden wird Nextcloud den Benutzer basierend auf folgenden Attributen finden:

LDAP-/AD-Benutzername: □

LDAP-/AD E-Mail-Adresse: ☑

Andere Attribute: Attribute auswählen

↓ LDAP-Abfrage bearbeiten

LDAP-Filter: (&(|(objectclass=inetOrgPerson))(|(mailPrimaryAddress=%uid)(mail=%uid)))

dh102@hdm-stuttga

Konfiguration OK ●