



DezSys-Protokoll DezSysLabor-04 "Authentifizierung & Autorisierung"

Dezentrale Systeme 5BHITT 2015/16

Thomas Stedronsky

Version 1.0

Begonnen am 08. Jänner 2016

Beendet

Note:

Betreuer: Th. Micheler

Inhaltsverzeichnis

1	Einfül	nrung	3
		Ziele	
		Voraussetzungen	
		Aufgabenstellung	
		nisse	
	_	LDAP Vorbereitungen	
		Java Implementierung	
		LDAP Search	
		LDAP Modify	
•		en	
4 Abbildungsverzeichnis			

1 Einführung

Diese Übung soll zur Vertiefung der Begriffe "Authentifizierung und Autorisierung" dienen.

1.1 Ziele

Das Ziel dieser Übung ist die Funktionsweise eines Verzeichnisdienstes zu verstehen und Erfahrungen mit der Administration auszuprobieren. Ebenso soll die Verwendung des Dienstes aus einer Anwendung heraus mit Hilfe der JNDI geübt werden.

Authentifizierung bedeutet hier, dass per Username und Passwort eine Anmeldung beim Verzeichnisdienst erfolgt. Autorisierung wird hier im Zusammenhang mit Service-Gruppen und zugeordneten Usern durchgefuehrt.

1.2 Voraussetzungen

- Grundlagen Verzeichnisdienst
- Administration eines LDAP Dienstes
- Verwendung von Commandline Werkzeugen fuer LDAP (LDAPSEARCH, LDAPMODIFY)
- Grundlagen der JNDI API f
 ür eine JAVA Implementierung
- Verwendung einer virtuellen Instanz für den Betrieb des Verzeichnisdienstes

1.3 Aufgabenstellung

Mit Hilfe der zur Verfuegung gestellten VM wird ein vorkonfiguriertes LDAP Service zur Verfuegung gestellt. Dieser Verzeichnisdienst soll um folgende Eintraege erweitert werden. Das verwendete Namensschema (eg. group.service1 oder vorname.nachname) soll fuer alle Eintraege verwendet werden.

- 5 Posix Groups (beliebe Zuweisung von UserIDs)
- 10 User Accounts

Weiters soll eine Java-Applikationen zur Authentifizierung und Autorisierung entwickelt werden. Folgende Fragestellungen stehen dabei im Mittelpunkt:

- Sind Username und Passwort korrekt? (Identifikation des Benutzers)
- Ist der User berechtigt ein bestimmtes Service zu nutzen? (Benutzer-Berechtigung)

2 Ergebnisse

2.1 LDAP Vorbereitungen

Zu aller erst mussten die Gruppen und die User für die Übung angelegt werden. In den folgenden Bildern kann man die Maske für das Gruppen bzw. User anlegen sehen.

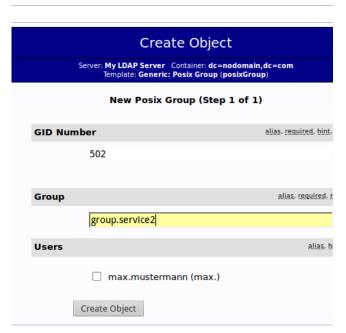


Abbildung 1 create_group



Abbildung 2 create_user

2.2 Java Implementierung

Nach dem die User angelegt worden sind, konnte mit der Java Implementierung begonnen werden. Dafür wurde ein Example von [1] verwendet.

```
String url = "ldap://192.168.17.128:389";
String user_name = "thomas.stedronsky";
String user_data = "cn=" + user_name + ",dc=nodomain,dc=com";
String user_password = "user";
```

Hier werden die Daten für die Authentifizierung festgelegt. Anschließend werden mittels put Befehl die Daten an das LDAP System gesendet. Daraufhin wird der Authentication-Vorgang durchgeführt. Sollte dies erfolgreich sein wird Authentication ok zurückgegeben, sollte dies nicht möglich sein wird Authentication nok ausgegeben.

2.3 LDAP Search

- -D ... steht hierbei für den User mit dem auf das LDAP zugegriffen wird.
- -b ... ist der konkrete Suchbefehl

```
ldapsearch -h 127.0.0.1 -p 389 -D "cn=thomas.stedronsky,dc=nodomain,dc=com" -W -b "cn=group.service1,dc=nodomain,dc=com" memberUid

user@vmxubuntu:-$ ldapsearch -h 127.0.0.1 -p 389 -D "cn=thomas.stedronsky,dc=nodomain,dc=com" -W -b "cn=group.service1,dc=nodomain,dc=com" memberUid

# cxtended LDIF

# LDAPV3

# base <cn=group.service1,dc=nodomain,dc=com> with scope subtree

# filter: (objectclass=*)

# group.service1, nodomain.com
dn: cn=group.service1,dc=nodomain,dc=com
memberUid: max.mustermann
memberUid: max.mustermann
memberUid: simon.wortha
memberUid: simon.wortha
memberUid: patrick.malik

# search result
search: 2
result: 0 Success

# numResponses: 2
```

Abbildung 3 ldap_search_1

numEntries: 1

Bei diesem Idapsearch wird gezielt nach den Membern anhand der MemberUid der group.service1 gesucht.

```
ldapsearch -h 127.0.0.1 -p 389 -D "cn=thomas.stedronsky,dc=nodomain,dc=com" -W -b "cn=thomas.stedronsky,dc=nodomain,dc=com" UidNumber user@vmxubuntu:-$ ldapsearch -h 127.0.0.1 -p 389 -D "cn=thomas.stedronsky,dc=nodomain,dc=com" -W -b "cn=thomas.stedronsky,dc=nodomain,dc=com" UidNumber Enter LDAP Password:
```

```
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <cn=thomas.stedronsky,dc=nodomain,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: UidNumber
#
# thomas.stedronsky, nodomain.com
dn: cn=thomas.stedronsky,dc=nodomain,dc=com
uidNumber: 1001
# search result
search: 2
result: 0 Success
# numResponses: 2
# numEntries: 1
```

Abbildung 4 ldap_search_2

Bei diesem Idapsearch wurde nach der User ID des User thomas.stedronsky gesucht.

Abbildung 5 ldap_search_3

```
ldapsearch -h 127.0.0.1 -p 389 -D "cn=thomas.stedronsky,dc=nodomain,dc=com" -W
-b "dc=nodomain,dc=com" "cn=group.*" memberUid
user@vmxubuntu:~$ ldapsearch -h 127.0.0.1 -p 389 -D "cn=thomas.stedronsky,dc=nodomain,dc=com" -W -b "dc=nodomain,dc=com" "cn=group.*" memberUid
Enter LDAP Password:
# extended LDIF
# LDAPv3
  base <dc=nodomain,dc=com> with scope subtree
# filter: cn=group.*
# requesting: memberUid
# group.default, nodomain.com
dn: cn=group.default,dc=nodomain,dc=com
# group.service1, nodomain.com
dn: cn=group.service1,dc=nodomain,dc=com
memberUid: matthias.ritter
memberUid: max.mustermann
memberUid: thomas.stedronsky
memberUid: simon.wortha
memberUid: patrick.malik
# group.service2, nodomain.com
dn: cn=group.service2,dc=nodomain,dc=com
memberUid: tobias.perny
# group.service3, nodomain.com
dn: cn=group.service3,dc=nodomain,dc=com
memberUid: manuel.reilaender
# group.service4, nodomain.com
dn: cn=group.service4,dc=nodomain,dc=com
memberUid: michael.weinberger
# group.service5, nodomain.com
dn: cn=group.service5,dc=nodomain,dc=com
memberUid: stefan.polydor
memberUid: thomas.taschner
# search result
search: 2
result: 0 Success
```

Bei diesem Idapsearch wird nach allen Gruppen gesucht, dies geschieht mit group.*. Die User werden dann wieder anhand der MemberUid ausgegeben.

2.4 LDAP Modify

ldapmodify -h 127.0.0.1 -p 389 -D "cn=admin,dc=nodomain,dc=com" -W

dn: cn=group.service1,dc=nodomain,dc=com

changetype: modify
replace: description
description: testtest

user@vmxubuntu:~\$ ldapmodify -h 127.0.0.1 -p 389 -D "cn=admin,dc=nodomain,dc=com" -W

Enter LDAP Password:

dn: cn=group.service1,dc=nodomain,dc=com

changetype:modify
replace: description
description: testtest

modifying entry "cn=group.service1,dc=nodomain,dc=com"

Abbildung 6 Idap_modify_1

Bei diesem Modify wurde die Description von der Gruppe Service1 verändert.

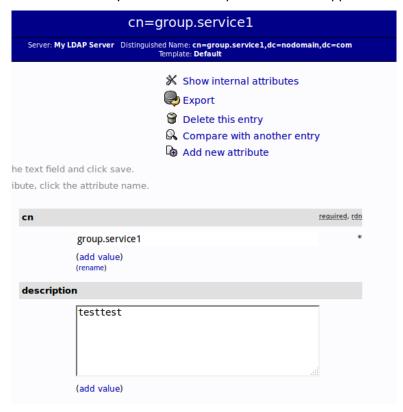


Abbildung 7 Idap_modify_result_1

Hier sieht man die veränderte Description im LDAP-Webinterface

```
ldapmodify -h 127.0.0.1 -p 389 -D "cn=admin,dc=nodomain,dc=com" -W
dn: cn=tobias.perny,dc=nodomain,dc=com
changetype: modify
replace: sn
description: Testtest
user@vmxubuntu:~$ ldapmodify -h 127.0.0.1 -p 389 -D "cn=admin,dc=nodomain,dc=com" -W
Enter LDAP Password:
dn: cn=tobias.perny,dc=nodomain,dc=com
changetype:modify
replace: sn
sn: Testtest
modifying entry "cn=tobias.perny,dc=nodomain,dc=com"
```

Abbildung 8 Idap_modify_1



Abbildung 9 Idap_modify_result_2

Der Nachname ("Second Name") von dem User tobias.perny wurde geändert.

2.5 LDAP Änderung ohne Admin Rechte

Mittels Access Control ist es möglich "normalen" Usern bestimmte Rechte zu geben. Somit können Read, Write oder Break Rechte vergeben werden. Um dies zu konfigurieren muss im Konfigurationsfile von LDAP ein gewisser Access Befehl eingefügt werden.

```
access to dn.subtree="dc=example,dc=com"
    by self write
    by dn.children="dc=example,dc=com" search
    by anonymous auth
[2]
```

2.6 Brute-Force

In einer Schulübung wurde eine Brute-Force Attacke auf einen Account des LDAP Systems durchgeführt. . Hierbei wurde ein Algorithmus entwickelt der das Passwort eines Accounts knacken soll. Es wurden 4-, 8-, 16, und 64-stellige Passwörter vergeben. Es war uns leider aus zeitlichen Gründen und des TGM-Netzes nicht möglich ein Passwort herauszufinden.

3 Quellen

[1] JNDI Tutorial, Oracle,

docs.oracle.com/javase/tutorial/displayCode.html?code=http://docs.oracle.com/javase/tutorial/jndi/ldap/examples/Simple.java, zuletzt besucht am 08.01.2016

[2] OpenLDAP Access Control Examples, http://www.openldap.org/doc/admin24/access-control.html, zuletzt besucht 08.01.2016

4 Abbildungsverzeichnis

Nbbildung 1 create_group	4
Abbildung 2 create_user	
sbbildung 3 ldap_search_1	
sbildung 4 ldap_search_2	
sbildung 5 ldap_search_3	
Abbildung 6 Idap modify 1	
sbildung 7 Idap modify result 1	
sbildung 8 Idap modify 1	
Abbildung 9 Idap modify result 2	