Effiziente Nutzerverwaltung mit LDAP

Einführung, Überblick und Anwendung

Reiner Klaproth,
Mittelschule Johannstadt-Nord Dresden
Maintainer des Arktur-Schulservers V4.0

Übersicht

1. Was ist LDAP?

- Geschichte
- Modell

2. LDAP - Strukturen

- Objekte und Objektklassen
- Baumstruktur und Adressen
- Benutzeraccounts

Übersicht (2)

3. Anwendung zur Benutzerverwaltung

- Login-Verwaltung unter Linux
- Samba 3 – Domänenserver
- Apache und Proxy-Authentifizierung

4. Rechteverwaltung

- Arten von Gruppen
- Rechteverwaltung in der Konfiguration
- Ausblick: ACI

Übersicht (3)

5. Anwendung zur Systemverwaltung

- Möglichkeiten
- Beispiel: DHCP mit LDAP

6. Sicherheit und Backup

- Verschlüsselung per SSL/TLS
- SASL und Kerberos
- Backup der Daten
- Replikation zweier LDAP-Server

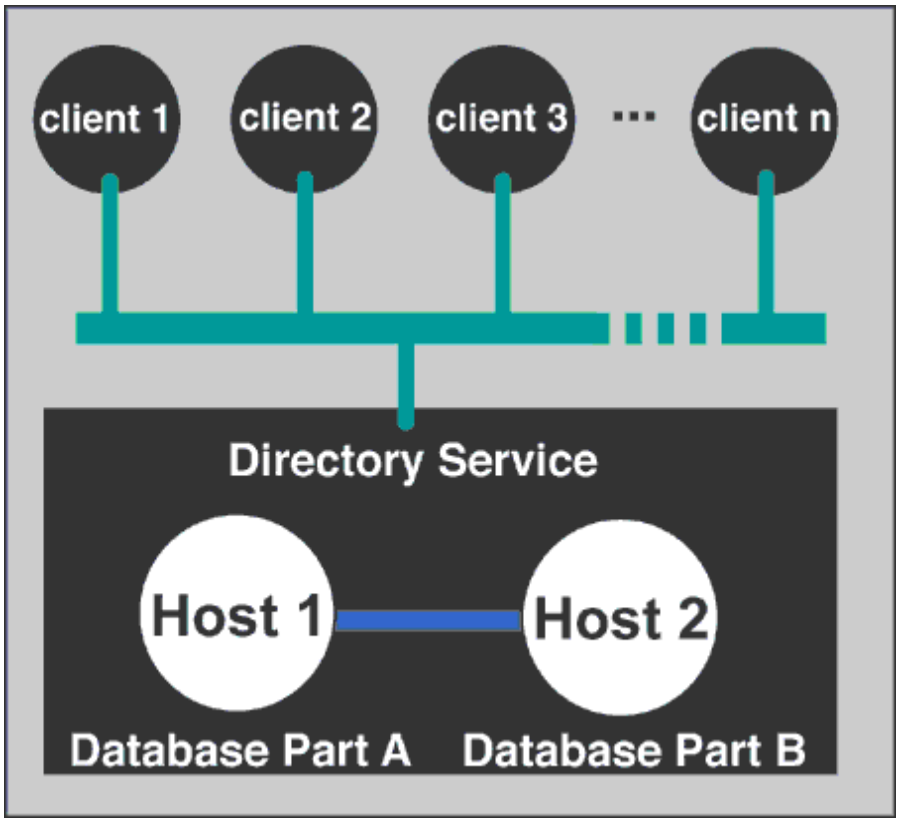
Was ist LDAP ?
Ein Blick in die Geschichte

- Das Problem der 80er Jahre: gewachsene Strukturen verschiedener Hersteller
- etwa 1985: Basis für X.500 (DAP)
  ein an OSI angepasster weltweiter Verzeichnis-dienst, von ITU 1988 standardisiert
- Juli 1993: „Tiny“-Variante: X.501
  Lightweigth Directory Access Protocol=LDAP
- ebenfalls weltweiter Verzeichnisdienst, aber an TCP/IP angepasst.

Was ist LDAP ?

- Basiert auf vier Grundprinzipien der Informatik

1. Client-Server-Prinzip



Was ist LDAP ?

2. Datenbank-Prinzip

- Optimiert auf schnelles Finden und Lesen
- Wenige Schreiboperationen
- Keine Transaktionen
- Beschränkte Anzahl von Datentypen
- Hohe Sicherheitsanforderungen
- Erweiterbarkeit

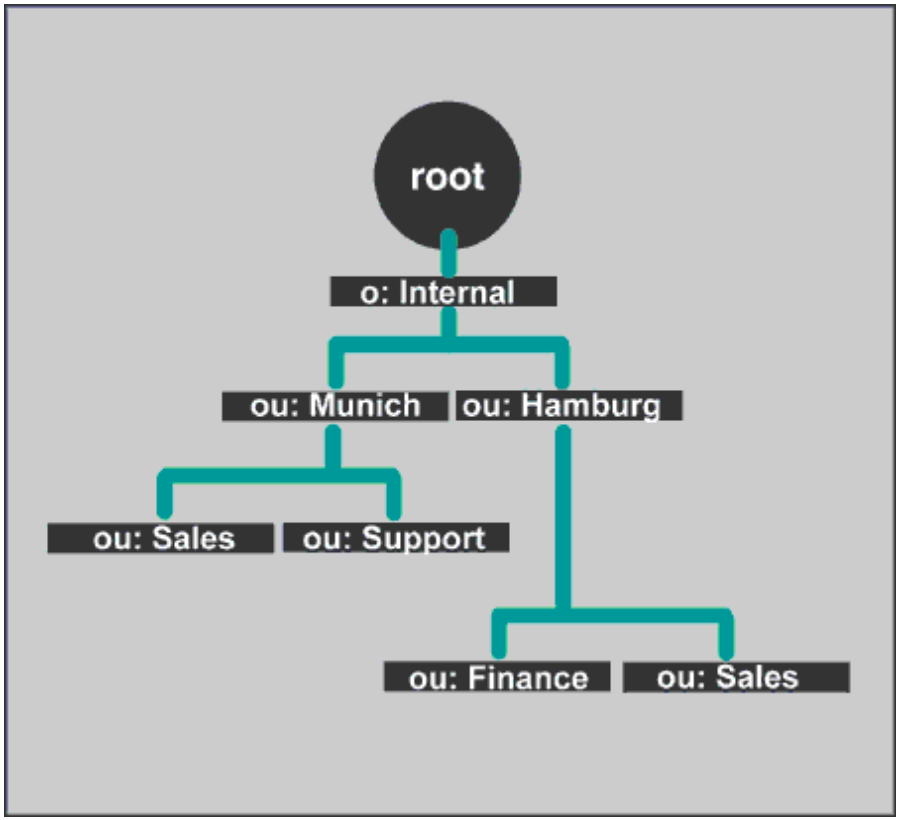
Was ist LDAP ?

3. Objektorientierung

uid=klaproth	
cn:	Reiner Klaproth
displayName:	Herr Klaproth
uidNumber:	500
gidNumber:	101
homeDirectory:	/home/Lehrer/klaproth
loginShell:	/bin/bash
gecos:	Reiner Klaproth, Lehrer
sn:	Klaproth
...	

Was ist LDAP ?

4. Baum-Struktur



Objekte

- Wichtige Objekttypen:

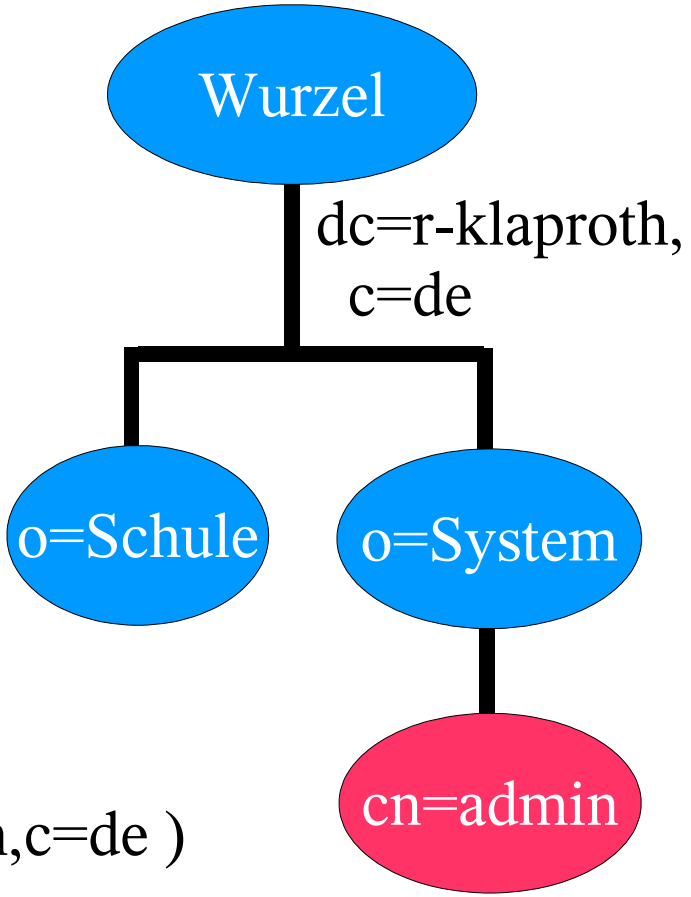
Kürzel	Typ	Beschreibung
c	Container	country; zwei-Buchstaben Landes-kürzel, z.B. c=de
o	Container	organization; also Firma bzw. über-geordnete Einheit
ou	Container	organizational unit; also Abteilung oder Bereich
cn	Blatt	common name; Name der Person oder Gruppe (des Objekts)
dc	Container	domain component; Teile der Domain. Eigentlich zur Einbindung der DNS-Domain als Baumwurzel

Adresse der Objekte

- RDN – Relative Distinguished Name: eindeutige Kennzeichnung des Objekts (Beispiel: cn=admin)

- DN – Distinguished Name: eindeutiger „Pfad“ des Objekts im Baum

(Beispiel: cn=admin,o=System,dc=r-klaproth,c=de)



Schema-Daten (Klassen)

- Definition der Attribute

```
attributetype ( 2.5.4.10 NAME ( 'o' 'organizationName' )
        DESC 'RFC2256: organization this object belongs to'
        SUP name )
```

- Definition der Klassen

```
objectclass ( 2.5.6.4 NAME 'organization'
        DESC 'RFC2256: an organization'
        SUP top STRUCTURAL
        MUST o
        MAY ( userPassword $ searchGuide $ seeAlso $ businessCategory $
                x121Address $ registeredAddress $ destinationIndicator $
                preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
                telephoneNumber $ internationaliSDNNumber $
                facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $
                postalAddress $ physicalDeliveryOfficeName $ st $ l $ description ) )
```

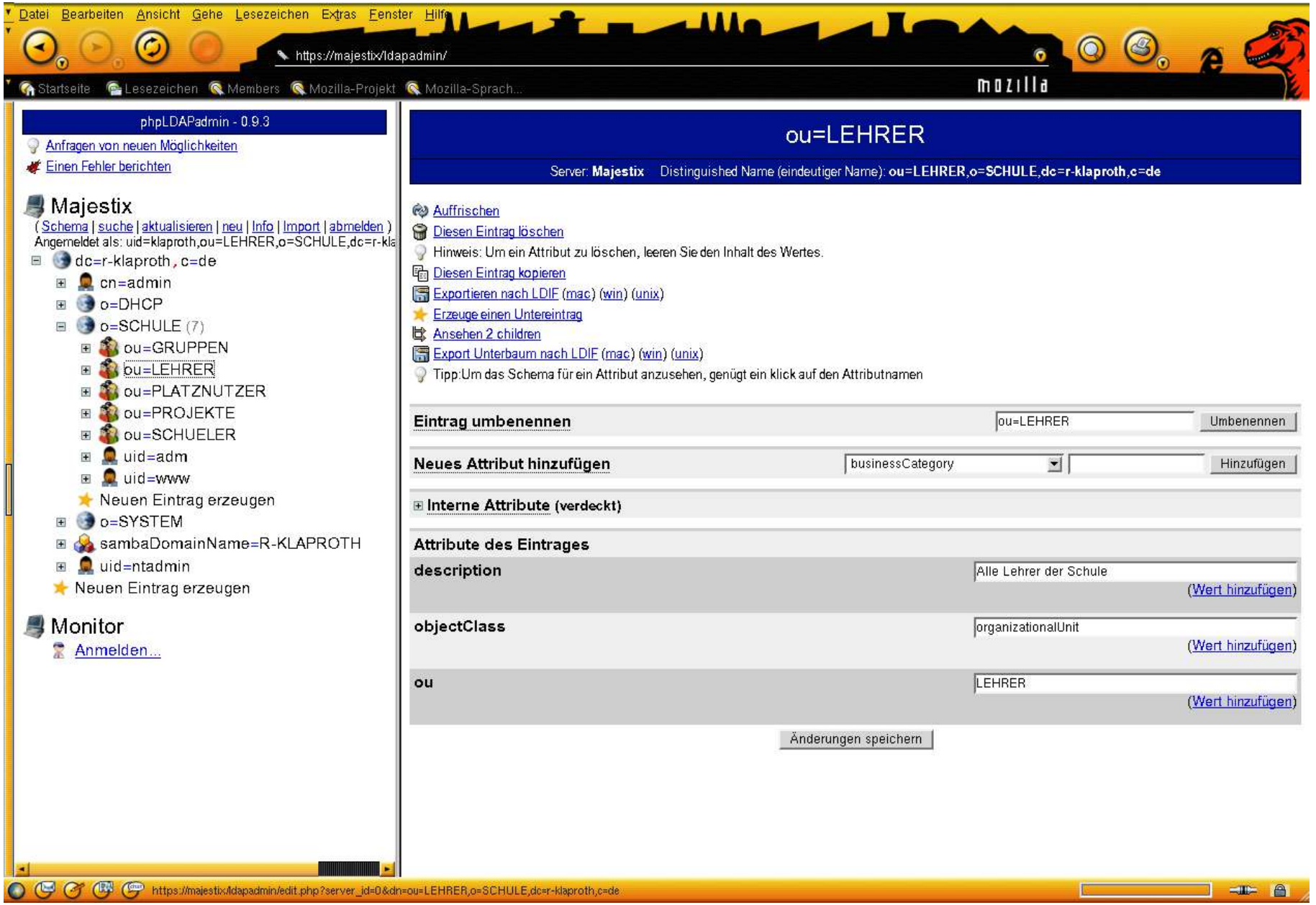
- Standard-Definitionen werden in OpenLDAP mitgeliefert
- sie müssen in der Konfiguration des slapd eingebunden werden

Objekterzeugung aus Klassen

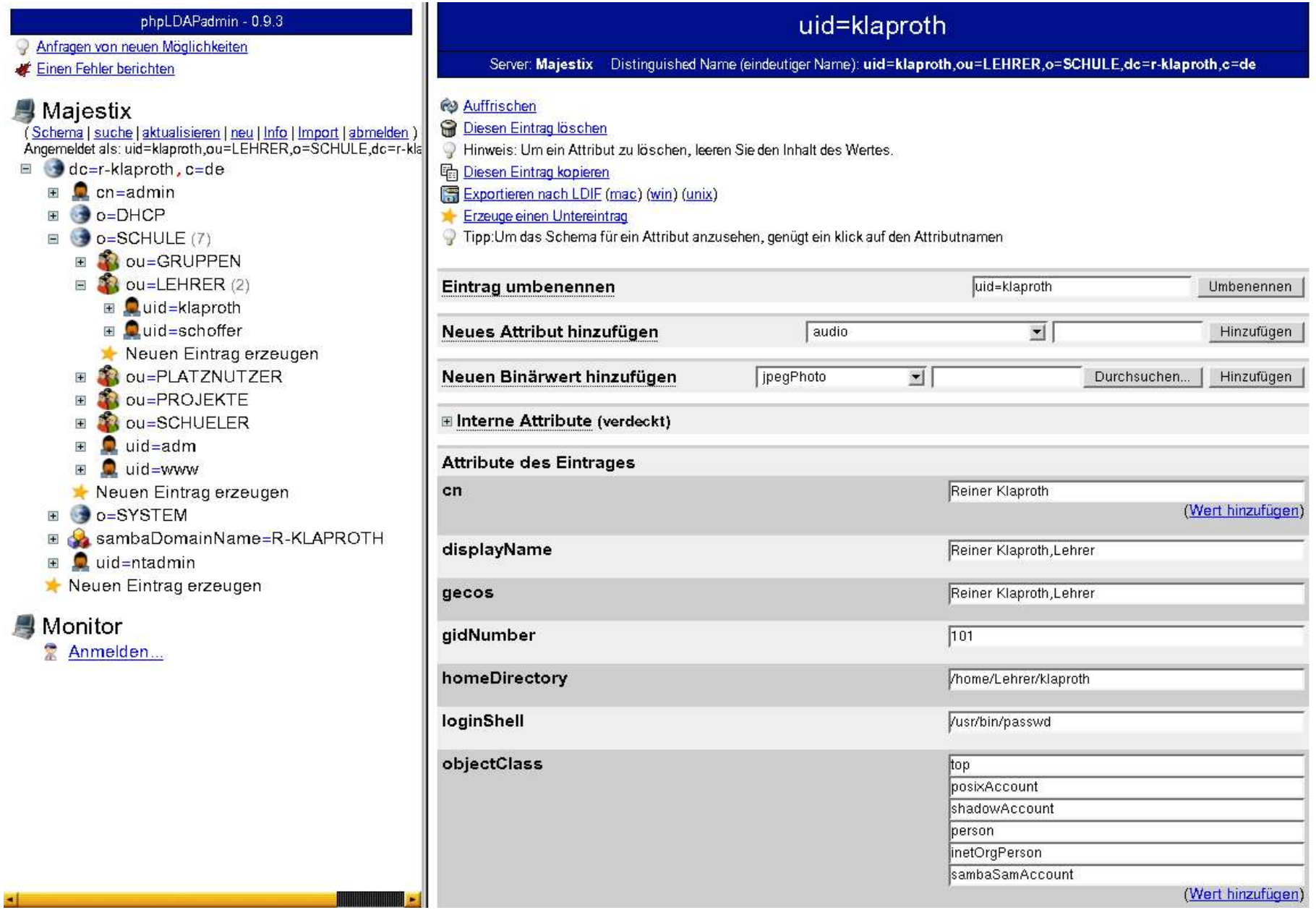
- Grundprinzip: Für jedes Objekt müssen die benötigten Klassen angegeben werden
- Einige Klassen hängen voneinander ab
  z.B. shadowAccount und posixAccount von account oder inetOrgPerson von person
- Beispiel für einen Nutzeraccount für Linux und Windows

```
uid=klaproth
objectclass: top
objectclass: account
objectclass: posixAccount
objectclass: shadowAccount
objectclass: sambaSamAccount
```

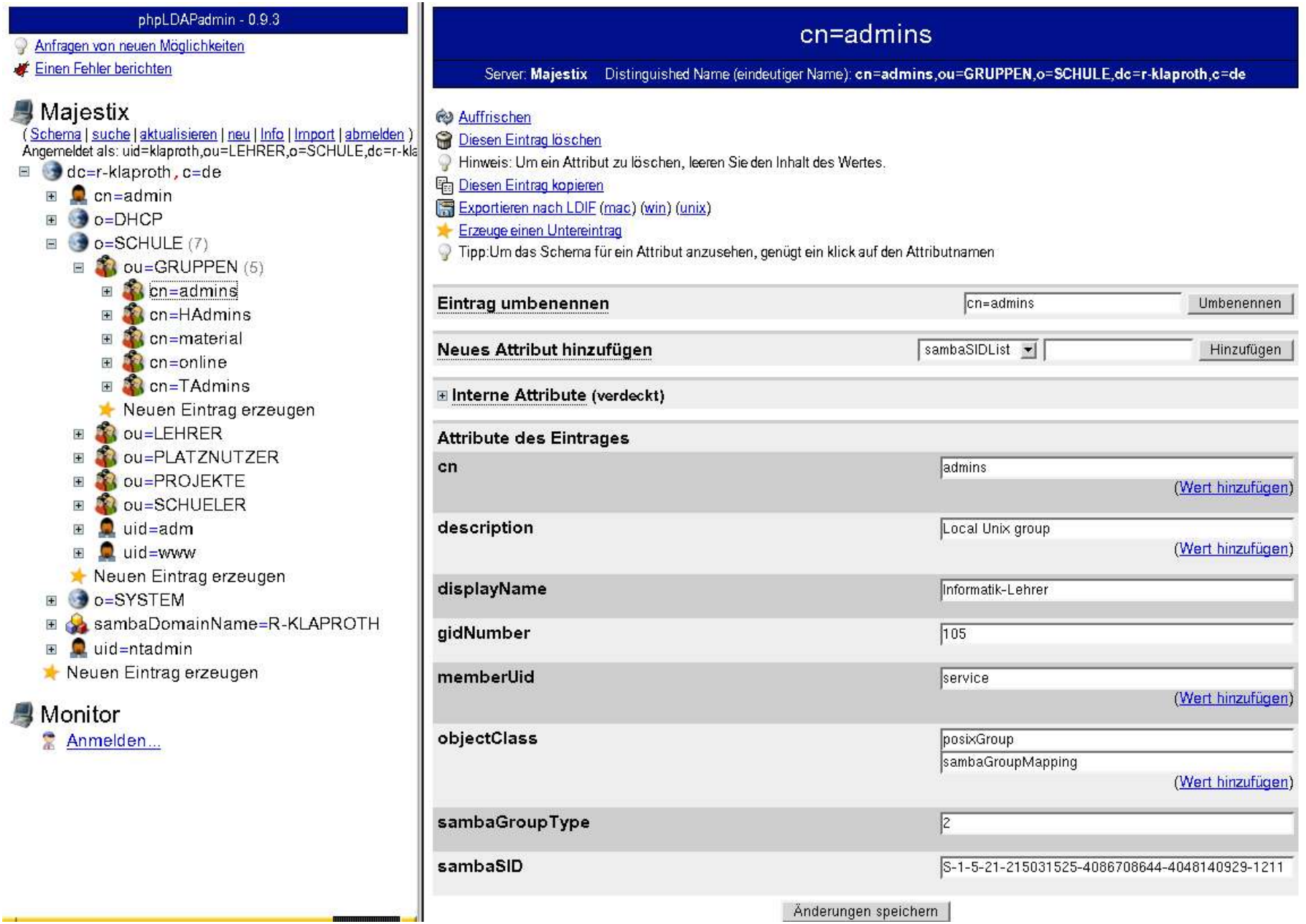
Beispiel 1: Container-Objekt



Beispiel 2: Nutzer-Objekt



Beispiel 3: Gruppen-Objekt



Konfiguration von OpenLDAP

- Konfigurationsdateien in `/etc/openldap`
- `ldap.conf`
  für Client-Programme (ldapsearch,...)
- `slapd.conf`
  für den Server-Dienst slapd

ldap.conf

```
# See ldap.conf(5) for details
# This file should be world readable but not world writable.

#BASE   dc=example, dc=com
#URI    ldap://ldap.example.com ldap://ldap-master.example.com:666

#SIZELIMIT      12
#TIMELIMIT      15
#DEREF          never
URI             ldap://127.0.0.1:389/
# host          127.0.0.1
base            dc=r-klaproth,c=de
rootbinddn      cn=admin,dc=r-klaproth,c=de

# Bei der Anmeldung per pam werden alle Objekte der Klasse
# posixAccount durchsucht
pam-filter      objectclass=posixAccount

# sowie diejenigen, die das Attribut uid enthalten
pam-login-attribute     uid
pam_password            exop

# SSL aus
ssl     no
```

LDAP-Linuxclient

- benötigt werden zwei Pakete:
  - pam_ldap (Authentifizierung) und
  - nss_ldap (Nutzerdatenbank)
- Konfiguration der LDAP-Adresse in
  `/etc[/openldap]/ldap.conf`
- Einträge in
  - `/etc/nsswitch.conf` und
  - `/etc/pam.d/<dienst>` oder besser (z.B. bei SuSE)
  - `/etc/security/pam_unix2.conf`

LDAP-Linuxclient

- /etc/nsswitch.conf

```
passwd: files ldap [NOTFOUND=return] nis
group:  files ldap [NOTFOUND=return] nis

hosts:          files dns ldap [NOTFOUND=return]

# LDAP is nominally authoritative for the following maps.
services:   files ldap [NOTFOUND=return]
networks:   files ldap [NOTFOUND=return] dns nis
protocols:  files ldap [NOTFOUND=return]
rpc:        files ldap [NOTFOUND=return]
ethers:     ldap [NOTFOUND=return] files nis
```

LDAP-Linuxclient

- /etc/pam.d/login

```
#%PAM-1.0
auth     required   pam_securetty.so
auth     required   pam_nologin.so
auth     sufficient pam_ldap.so
auth     requisite  pam_unix2.so      nullok #set_secrpc
auth     required   pam_env.so
account sufficient  pam_ldap.so
account required        pam_unix2.so
password required   pam_pwcheck.so      nullok
password required   pam_ldap.so
```

- /etc/security/pam_unix2.conf

```
auth:             use_ldap nullok
account:          use_ldap
password:         use_ldap
session:          none
```

Einrichtung mit SuSE-YaST

- Kontrollzentrum: Netzdienste

Einrichtung mit SuSE-YaST

- LDAP-Angaben eintragen – fertig.

Einbindung von Windows (Samba)

Vorteile:

- gemeinsame Verwaltung eines Accounts
- Passwort wird konsistent gehalten (Windows und Linux) und kann mit Windows-Bordmitteln geändert werden
- Nutzer und Gruppen sind wie im Domänenserver sicht- und nutzbar
- Zusätzliche Attribute wie SambaUserWorkstation nutzbar
- Persönliche Login-Scripte und Einstellungen
- IDMaps und Trusted Domains

Einbindung von Windows (Samba)

- Anpassungen in der `smb.conf`

```
passdb backend = ldapsam:ldap://localhost:389/
ldap suffix = dc=r-klaproth,c=de
ldap admin dn = cn=admin,dc=r-klaproth,c=de
ldap machine suffix = ou=ARBEITSSTATIONEN,o=SCHULE
ldap idmap suffix = ou=idmaps,o=SYSTEM
ldap passwd sync = Yes
```

- LDAP-Admin-Passwort einmalig eintragen:

```
smbpasswd -w <passwort>
```

Anbindung von Apache

- Modul mod_auth_ldap
  (http://www.muquit.com/muquit/software/mod_auth_ldap/mod_auth_ldap.html)
  Version 2.4.2 für Apache 1.3
  Version 3.x für Apache 2.0

- Anpassungen in der `httpd.conf` (V2.4.2)

```
LoadModule mm_auth_ldap_module
                        lib/apache/mod_auth_ldap.so
AddModule mod_auth_ldap.c
```

Anbindung von Apache

- Zugriffsregel

```
<Directory /usr/www/secure/admin>
Options FollowSymLinks ExecCGI
AuthType Basic
AuthName "Admin-Interface"
LDAP_Server 127.0.0.1
LDAP_Port 389
LDAP_Protocol_Version 3
UID_Attr  uid
Group_Attr memberUid
Base_DN "o=SCHULE,dc=r-klaproth,c=de"

<Limit GET POST>
require group "cn=online,ou=GRUPPEN"
</Limit>
</Directory>
```

- oder

```
require filter "(gidNumber=101)"
```

Proxy-Authentifizierung (Squid)

- Optionen bei `configure`

```
--enable-auth="basic"
--enable-basic-auth-helpers="LDAP PAM"
--enable-external-acl-helpers="ldap_group unix_group"
```

- Zugriffsregel in `squid.conf`

```
auth_param basic program /usr/squid/bin/squid_ldap_auth -v 3
-b "o=SCHULE,dc=my-domain,c=de" -f "(uid=%s)" 192.168.0.1

auth_param basic children 20
auth_param basic realm Internet-Proxy-Server
acl all2 proxy_auth REQUIRED src 0.0.0.0/0.0.0.0
http_access allow all2
```

LDAP-Befehle (Überblick)

- ldapadd — fügt neue Einträge hinzu
- ldapmodify — ändert bestehende Einträge
- ldapsearch — sucht im LDAP-Baum
- ldapcompare — nur vergleichen
- ldapmodrdn — Eintrag umbenennen (RDN)
- ldapdelete — Eintrag löschen
- ldappasswd — Passwort ändern

Rechte im LDAP

- Allgemein: in `/etc/openldap/slapd.conf` wird festgelegt

```
access to <what>
    by <who> <access> <control>
```

```
<what> : dn[.regex|exact]=Objekt
         filter=<filter>
         attr(s)=<Attributliste>

<who> : anonymous | users | self
        dn[.regex]=Objekt
        dnattr=Attribut
        group[.regex]=LDAP-Gruppe
        peername[.regex]=IP/Name
```

Rechte im LDAP

```
access to <what>
    by <who> <access> <control>
```

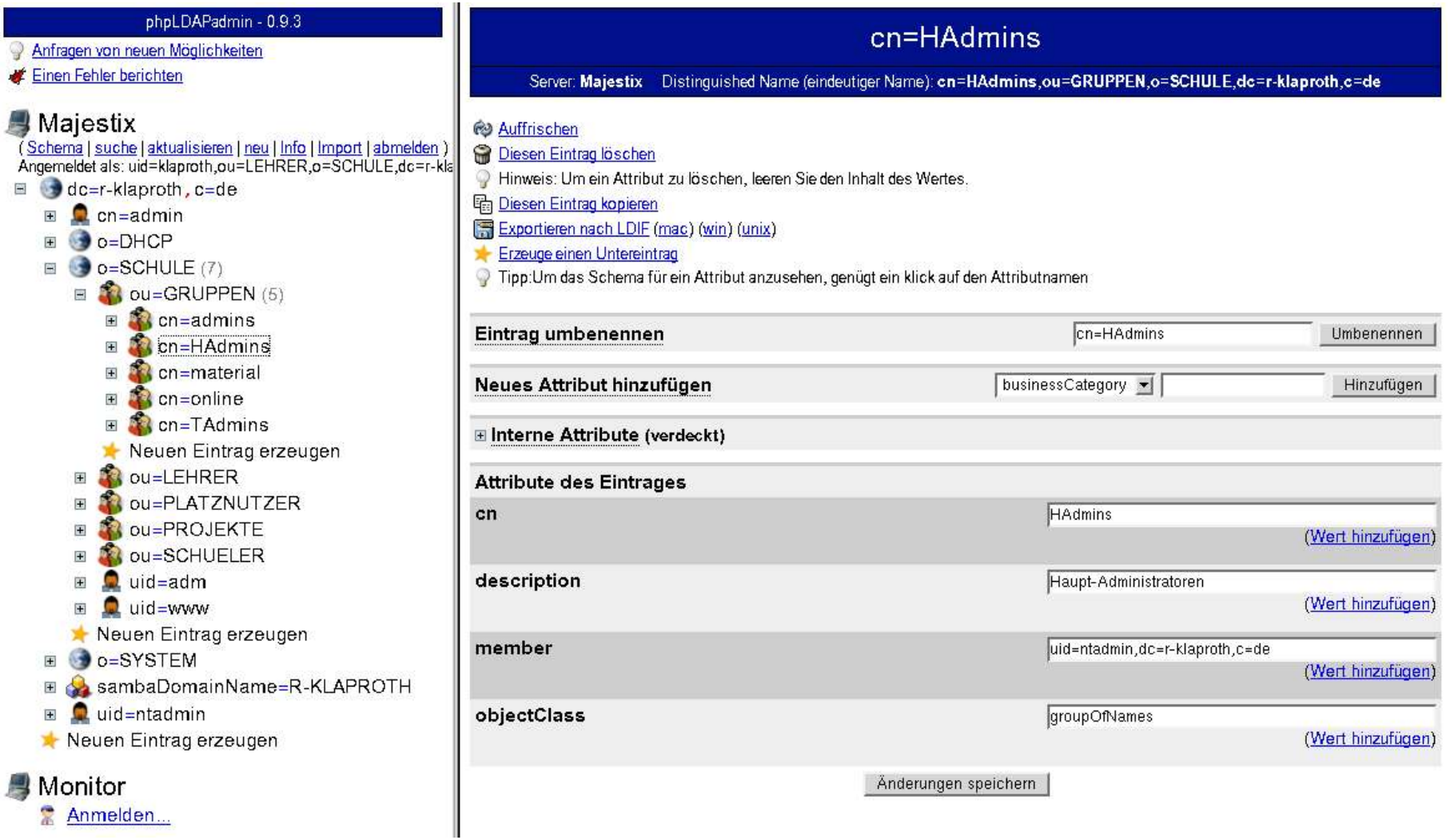
`<access>` :	`none`	Zugriff verweigern
	`auth`	Passwort prüfen
	`compare`	Vergleichen (Ja/Nein-Ergebnis)
	`search`	ob Objekt (Attribut) existiert
	`read`	Lesezugriff
	`write`	Schreibzugriff
`[+|-][0|x|c|s|r|w]`		

`<control>` :	`stop`	(Normalfall) sofort beenden
	`break`	mit nächstem Regelsatz fortsetzen
	`continue`	sofort weitersuchen

Beispiel: Arktur-Schulserver

```
# To let PAM authenticate
access to attr=userPassword
        by self write
        by anonymous auth
        by * none

access to attr=shadowLastChange
        by self write
        by * read

# For Samba Access
access to attrs=sambaLMPassword,sambaNTPassword
        by dn="cn=ntadmin,dc=r-klaproth,c=de" write
        by self write
        by * none

access to attrs=sambaPwdLastSet,sambaPwdMustChange,sambaLogonTime,sambaLogoffTime,sambaKickoffTime
        by dn="cn=ntadmin,dc=r-klaproth,c=de" write
        by self write
        by * read

access to attrs=sambaPwdCanChange,sambaUserWorkstations,sambaDomainName
        by dn="cn=ntadmin,dc=r-klaproth,c=de" write
        by self read
        by * read

# Projektverwaltung:
# Die "Member" im Admin-Eintrag dürfen memberUID anpassen (Mitglieder ein- und ausladen).
# Der "projektadm" darf im gesamten Subtree schreiben.

access to dn.regex="cn=(.*),ou=PGruppen,ou=PROJEKTE,o=SCHULE,dc=r-klaproth,c=de" attr=memberUid
        by group.regex="cn=$1,ou=PAdmins,ou=PROJEKTE,o=SCHULE,dc=r-klaproth,c=de" write
        by * read

access to dn.subtree="ou=PROJEKTE,o=SCHULE,dc=r-klaproth,c=de"
        by dn.exact="uid=projektadm,ou=PROJEKTE,o=SCHULE,dc=r-klaproth,c=de" write break
        by * read

access to dn.subtree="ou=PAdmins,ou=PROJEKTE,o=SCHULE,dc=r-klaproth,c=de"
        by dn.exact="uid=projektadm,ou=PROJEKTE,o=SCHULE,dc=r-klaproth,c=de" write break
        by * none
```

Beispiel: LDAP-Gruppe



Ausblick: ACI

- Nachteil bislang: Rechte sind in der slapd.conf festgelegt
- ACI: Access Control Instruction – Rechte im LDAP-Baum selbst verwaltet
- in der `slapd.conf` wird festgelegt
  `access to * by aci write break`
- Recht im Objekt selbst eintragen:

```
objectClass: openLDAPacl
OpenLDAPaci: 1#entry#grant;r,w,s,c;[all]#group#cn=admins,ou=groups,o=acme
OpenLDAPaci: 2#entry#grant;r,w,s,c;userPassword,mail;r,s,c;[all]#access-
id#uid=user1,ou=people,l=dallas,o=acme>
```

Systemverwaltung mit LDAP

weitere Möglichkeiten von LDAP

- Bind (Nameserver):
  Domain-Daten im LDAP
- sendmail:
  Aliases, Mappings und Classes im LDAP
- Adressbücher mit LDAP
- DHCP mit LDAP
- ...

Beispiel DHCP-Server

- Patch für ISC-dhcpd unter:
  http://www.venaas.no/ldap/bind-sdb/
- Konfigurationsdatei `/etc/dhcpd.conf`

```
ldap-server     "localhost";
ldap-port       389;
ldap-username   "cn=dhcpd,o=DHCP,dc=r-klaproth,c=de";
ldap-password   "ne2tidoou";
ldap-base-dn    "o=DHCP,dc=r-klaproth,c=de";
ldap-method     dynamic;

# In dieser Datei landet die zusammengebaute Konfiguration
ldap-debug-file "/var/log/dhcpd-ldap-startup.log";
```

- Demonstration

Sicherheit: SSL und TLS

- Verschlüsselung der gesamten Verbindung
- Sicherheit in zwei Stufen:
  a) nur Server arbeitet mit Zertifikat
  b) auch der Client muss ein Zertifikat haben
- LDAP: Protokoll mit und ohne SSL
  a) LDAP v2: Port 636 für ldaps://
  b) LDAP v3: Standard-Port 389 kann beide Protokolle

SSL: Zertifikate erstellen

- CA erstellen: (in `/etc/ssl`)
  `CA.pl -newca` `cacert.pem`
- Server-Zertifikat erzeugen:
  `CA.pl -newcert` `newreq.pem`
- Server-Zertifikat signieren:
  `CA.pl -signcert` `newcert.pem`
- Passwort entfernen:
  `openssl rsa -in newreq.pem -out ldapkey.pem`
- umbenennen:
  `newcert.pem` in `ldapcert.pem`

SSL einbinden

- In `/etc/openldap/slapd.conf` einbinden:

```
TLSCertifikateFile    /etc/ssl/ldapcert.pem
TLSCertifikateKeyFile /etc/ssl/ldapkey.pem
TLSCACertifikateFile  /etc/ssl/CA/cacert.pem
```

- In der Startdatei einbinden:
  (falls LDAP v2 genutzt wird)
  `slapd -h "ldap:// ldaps://"`

SASL und Kerberos

- Für höhere Ansprüche lassen sich SASL (Simple Authentifikation and Secure Layer) und GSSAPI (Kerberos)-Authentifizierung einbinden
- Dadurch entfallen häufige Passworteingaben
- Einrichtung relativ aufwändig
- mind. ab Windows 2000 sicher unterstützt

```
sasl-regexp  <search pattern> <replacement pattern>

sasl-regexp uid=(.*),ou=*,o=SCHULE,dc=r-klaproth,c=de
            uid=$1,ou=Person,dc=r-klaproth,c=de
```

Backup der LDAP-Daten

- Abbild erstellen:
  `slapcat -f gesamt.ldif`
  Erzeugt eine ldif-Datei mit allen, auch sonst unsichtbaren Attributen (Ersteller, Datum,...)

- Einspielen (Achtung! LDAP nicht gestartet!)
  `slapadd -l gesamt.ldif`
  Erzeugt die LDAP-Datenbank neu.

Replikation zweier LDAP (slurpd)

- Datenbank des ersten LDAP-Servers kopieren
- In der `slapd.conf` des Master-LDAP einbinden:

```
replogfile /var/log/LDAP/slave.example.replog

replica uri=ldaps://slave.example.com:636
        binddn="cn=Replicator,dc=r-klaproth,c=de"
        bindmethod=simple credentials=secret
```

- In der `slapd.conf` des Slave-LDAP einbinden:

```
updatedn   "cn=Replicator,dc=r-klaproth,c=de"
updateref  "ldaps://master.example.com:636"
```

- slapd auf Master und Slave starten
- auf dem Master den slurpd starten

Ende

Vielen Dank für Ihre Aufmerksamkeit

Links

- OpenLDAP:
  http://www.openldap.org/
- phpLdapAdmin:
  http://phpldadadmin.sourceforge.net
- LDAP verstehen:
  http://www.mitlinx.de/ldap/index.html
- Arktur-Schulserver:
  http://arktur.schul-netz.de/