

# 17. LDAP

logue

# Contents

- LDAP?
- Installation
- Configuration
- Managing
- Practice

# LDAP?

- **Lightweight Directory Access Protocol**
- TCP/IP 위에서 디렉터리를 조회하고 수정하는 응용 프로토콜
- Directory??

```
drwx----- 4 root root 4096 Jul  4 06:01 .  
drwxr-xr-x 21 root root 4096 Feb  3 2010 ..  
-rw----- 1 root root  203 Jan 26 11:54 .bash_history
```

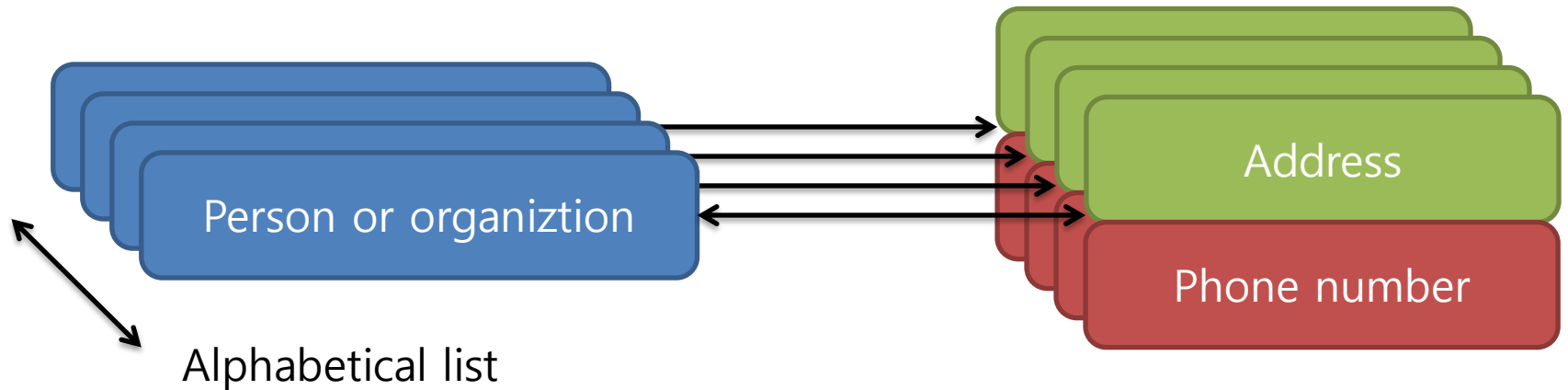


## 인터넷 프로토콜 스위트

응용 계층	BGP, DHCP, DNS, FTP, HTTP, IMAP, IRC, <b>LDAP</b> , MGCP, NNTP, NTP, POP3, RIP, RTP, RTSP, SDP, SIP, SMTP, SNMP, SOAP, SSH, TELNET, XMPP, ...
전송 계층	TCP, UDP, DCCP, SCTP, RSVP, ...
네트워크 계층	IP(v4/v6), ICMP, IGMP, ARP/RARP, ...
데이터링크 계층	MAC(이더넷, 토큰링, FDDI), PPP, ...
물리적 계층	EIA RS-232, EIA RS-422, EIA RS-449, EIA RS-485, ...

# LDAP?

- Directory
  - An organized set of records
  - Ex. Telephone directory



# LDAP?

- Directory service
  - 컴퓨터 네트워크의 사용자와  
네트워크 자원에 대한 정보를 저장하고  
조직하는 응용 소프트웨어

# LDAP?

- X.500

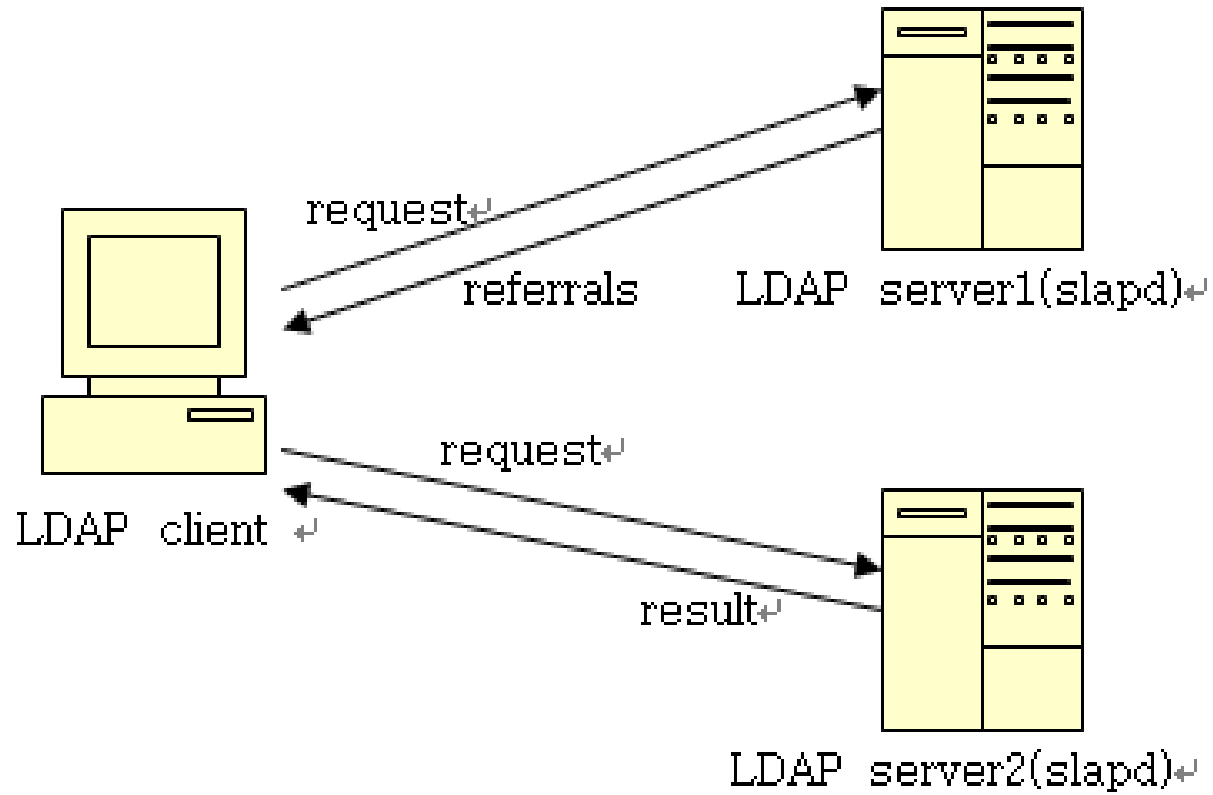
- 전자 디렉토리 서비스를 위한 네트워크 표준
- 1988년에 처음 등장



# LDAP?

- X.500
  - DAP(Directory Access Protocol)
  - DSP(Directory System Protocol)
  - DISP(Directory Information Shadowing Protocol)
  - DOP(Directory Operational Bindings Management Protocol)

# LDAP?

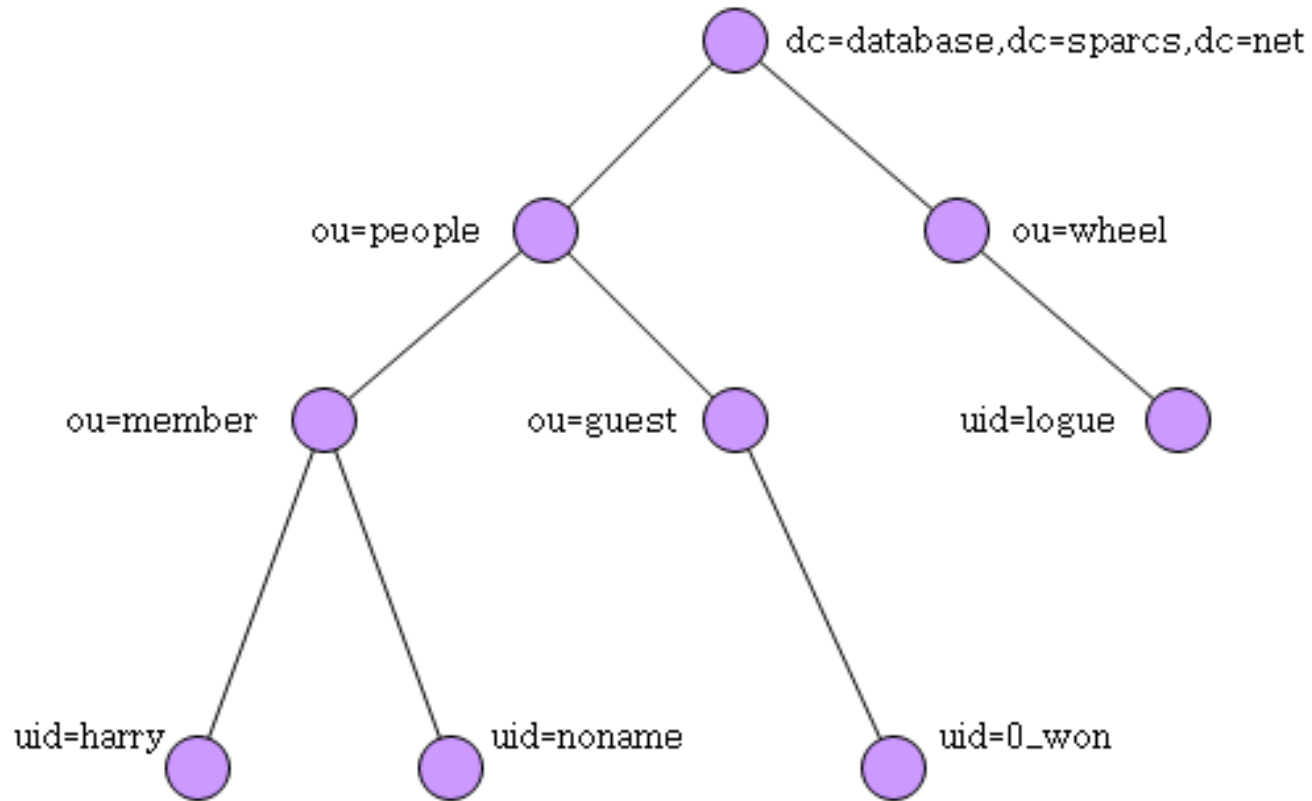


# LDAP?

- Directory Information Tree
  - A directory is a **tree** of directory entries.
  - An entry consists of a **set of attributes**.
  - An attribute has **a name and one or more values**. The attributes are defined in a schema.
  - Each entry has a unique identifier: its Distinguished Name(**DN**).
    - **RDN**

# LDAP?

- Directory Information Tree



# LDAP?

- Entry

```
dn: cn=Do-Guk Kim, ou=People, dc=sparcs, dc=org
cn: Do-Guk Kim
gidNumber: 200
homeDirectory: /home/logue
loginShell: /bin/bash
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
uidNumber: 4002
User Name: logue
```

# LDAP?

- Available backend types
  - Data Storage backends
    - bdb : BerkeleyDB
    - ldif : built on plain text LDIF files
  - Proxy backends
    - ldap : simple proxy to other LDAP servers
    - passwd : uses a passwd and group data
  - Dynamic backends
    - shell : invokes shell scripts for LDAP requests

# LDAP?

- Common usage of LDAP
  - Centralization of user and group information
  - Authenticate users locally
  - Authenticate users in a web application
  - Create a shared address directory for mail agents

# Installation

```
$ sudo apt-get install slapd ldap-utils
```

By default slapd is configured with minimal options needed to run the slapd daemon.

# Configuration

The **cn=config DIT** is used to dynamically configure the slapd daemon.

```
$ sudo ldapadd -Y EXTERNAL -H ldapi:///
-f /etc/ldap/schema/cosine.ldif
```

```
$ sudo ldapadd -Y EXTERNAL -H ldapi:///
-f /etc/ldap/schema/nis.ldif
```

```
$ sudo ldapadd -Y EXTERNAL -H ldapi:///
-f /etc/ldap/schema/inetorgperson.ldif
```

# Configuration

- backend.sparcs.org.ldif

```
# Load dynamic backend modules
dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: back_hdb

# Database settings
dn: olcDatabase=hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {1}hdb
olcSuffix: dc=sparcs,dc=org
olcDbDirectory: /var/lib/ldap
olcRootDN: cn=admin,dc=example,dc=com
```

# Configuration

- backend.[sparcs.org](https://sparcs.org).ldif

```
root@i-10-1-1-14:~# slappasswd
```

```
New password:
```

```
Re-enter new password:
```

```
{SSHA}otkHcuPvZDGTKFt0EVZV4gNgzSboNY+S
```

# Configuration

- backend.sparcs.org.ldif

```
olcRootPW: {SSHA}otkHcuPvZDGTKFt0EVZV4gNgzSboNY+S
olcDbConfig: set_cachesize 0 2097152 0
olcDbConfig: set_lk_max_objects 1500
olcDbConfig: set_lk_max_locks 1500
olcDbConfig: set_lk_max_lockers 1500
olcDbIndex: objectClass eq
olcLastMod: TRUE
olcDbCheckpoint: 512 30
olcAccess: to attrs=userPassword by dn="cn=admin,dc=sparcs,dc=org"
write by anonymous auth by self write by * none
olcAccess: to attrs=shadowLastChange by self write by * read
olcAccess: to dn.base="" by * read
olcAccess: to * by dn="cn=admin,dc=sparcs,dc=org" write by * read
```

# Configuration

- ACL(Access Control List)

```
olcAccess: <access directive>
<access directive> ::= to <what>
    [by <who> [<access>] [<control>] ]+
<what> ::= * |
    [dn[.<basic-style>]=<regex> | dn.<scope-style>=<DN>]
    [filter=<ldapfilter>] [attrs=<attrlist>]
<basic-style> ::= regex | exact
<scope-style> ::= base | one | subtree | children
<attrlist> ::= <attr> [val[.<basic-style>]=<regex>] | <attr> , <attrlist>
<attr> ::= <attrname> | entry | children
<who> ::= * | [anonymous | users | self
    | dn[.<basic-style>]=<regex> | dn.<scope-style>=<DN>]
    [dnattr=<attrname>]
    [group[/<objectclass>[/<attrname>][.<basic-style>]]=<regex>]
    [peername[.<basic-style>]=<regex>]
    [sockname[.<basic-style>]=<regex>]
    [domain[.<basic-style>]=<regex>]
    [sockurl[.<basic-style>]=<regex>]
    [set=<setspec>]
    [aci=<attrname>]
<access> ::= [self]{<level>|<priv>}
<level> ::= none | disclose | auth | compare | search | read | write | manage
<priv> ::= {=|+|-}{m|w|r|s|c|x|d|O}+
<control> ::= [stop | continue | break]
```

# Configuration

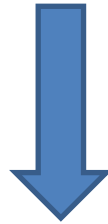
- ACL(Access Control List)
  - olcAccess: to <Entry> | <Attribute>  
by <DN><PERM>  
[by <DN><PERM> ...]
  - olcAccess: to \*  
by self write  
by anonymous auth  
by \* read

# Configuration

- ACL(Access Control List)
  - 동일한 액세스 항목을 중복 설정할 수 없다.
  - 넓은 범위에 관한 ACL을 뒤에 놓아야 한다.
  - Comma(,) 앞뒤로 공백이 없어야 한다.
  - ACL이 복잡할수록 검색 속도가 느려진다.

# Configuration

```
$ sudo ldapadd -Y EXTERNAL -H ldapi:///
-f backend.sparcs.org.ldif
```



```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=module,cn=config"

adding new entry "olcDatabase=hdb,cn=config"
```

# Configuration

- frontend.sparcs.org.ldif

```
# Create top-level object in domain
dn: dc=sparcs,dc=org
objectClass: top
objectClass: dcObject
objectclass: organization
o: SPARCS
dc: sparcs
description: Wheel Seminar LDAP Example

# Admin user.
dn: cn=admin,dc=sparcs,dc=org
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword: {SSHA}otkHcuPvZDGTKFt0EVZV4gNgzSboNY+S
```

# Configuration

```
$ sudo ldapadd -x -D  
cn=admin,dc=sparcs,dc=org -W -f  
frontend.sparcs.org.ldif
```



```
Enter LDAP Password:  
adding new entry "dc=sparcs,dc=org"  
adding new entry "cn=admin,dc=sparcs,dc=org"
```

# Managing

- Idap-utils

```
$ <command> -D <DN of the entry>  
-W -f <ldif file path>
```

# Managing

- Idapadd

```
$ Idapadd -D "cn=admin,dc=sparcs,dc=org"  
-W -f test.ldif
```

```
<test.ldif>
```

```
dn: cn=test,dc=sparcs,dc=org  
objectClass: inetOrgPerson  
cn: test  
sn: Kim
```

```
root@i-10-1-1-14:~# ldapadd -D "cn=admin,dc=sparcs,dc=org" -W -f test.ldif  
Enter LDAP Password:  
adding new entry "cn=test,dc=sparcs,dc=org"
```

# Managing

- Idapsearch

```
$ Idapsearch -b base [options] filter [attributes]
```

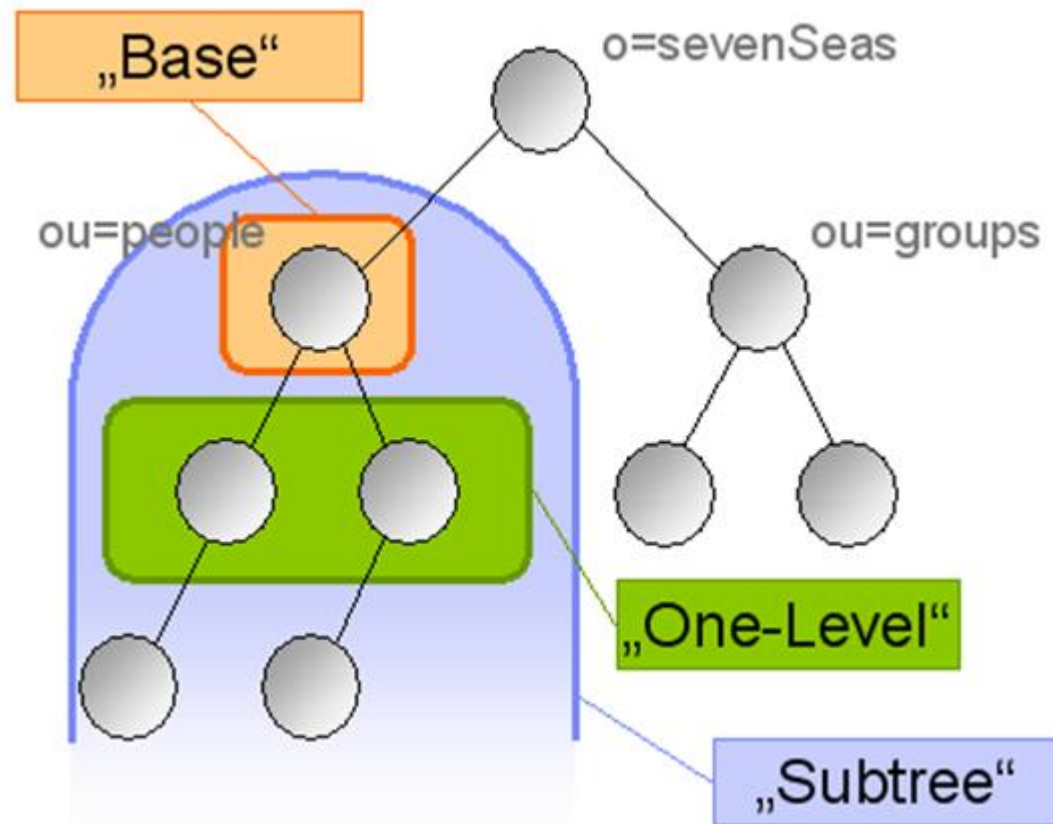
Some Useful Options!

-s: scope of the search. [base | one | sub]

-x: use simple authentication instead of SASL

# Managing

- Idapsearch



# Managing

- Idapsearch

- Filter

- Equality : "uid=logue"
    - Substring: "uid=\*gue"
    - Approximate: "uid~ =log"
    - Less than, greater then: "uid > =noname"
    - And: "&(uid=logue)(gidNumber=200)"
    - Or: |, Not: !, ...

# Managing

- ldapsearch
  - Ex) Return all entries

```
root@i-10-1-1-14:~# ldapsearch -x -b "dc=sparcs,dc=org" -s sub "objectclass=*"
# extended LDIF
#
# LDAPv3
# base <dc=sparcs,dc=org> with scope subtree
# filter: objectclass=*
# requesting: ALL
#
# sparcs.org
dn: dc=sparcs,dc=org
objectClass: top
objectClass: dcObject
objectClass: organization
o: SPARCS
dc: sparcs
description: V2h1ZWwgU2VtaW5hciBMREFOIEV4YW1wbGUg
```

# Managing

- ldapsearch
  - Ex) Find specific entry

```
root@i-10-1-1-14:~# ldapsearch -x -b "dc=sparcs,dc=org" -s sub "(&(cn=test)(sn=Kim))"
# extended LDIF
#
# LDAPv3
# base <dc=sparcs,dc=org> with scope subtree
# filter: (&(cn=test)(sn=Kim))
# requesting: ALL
#
# test, sparcs.org
dn: cn=test,dc=sparcs,dc=org
objectClass: inetOrgPerson
cn: test
sn: Kim

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

# Managing

- ldapmodify

```
$ ldapmodify -D "cn=admin,dc=sparcs,dc=org"  
-W -f testmod.ldif
```

```
<testmod.ldif>
```

```
dn: cn=test,dc=sparcs,dc=org  
Changetype: modify  
Replace: sn  
sn: Lee
```

```
root@i-10-1-1-14:~# ldapmodify -D "cn=admin,dc=sparcs,dc=org" -W -f testmod.ldif  
Enter LDAP Password:  
modifying entry "cn=test,dc=sparcs,dc=org"
```

# Managing

- Idapdelete

```
$ Idapdelete -D "cn=admin,dc=sparcs,dc=org"  
-W "cn=test,dc=sparcs,dc=org"
```

# Managing

- Idapmodrdn

```
$ Idapmodrdn -D
```

```
"cn=admin,dc=sparcs,dc=org" -W (-r)
```

```
"cn=test,dc=sparcs,dc=org" "cn=temp"
```

```
dn: cn=temp,dc=sparcs,dc=org  
objectClass: inetOrgPerson  
cn: test  
cn: temp  
sn: Lee
```

# Managing

- LDIF(LDAP Data Interchange Format)
  - 디렉토리 엔트리 표현 형식

```
dn: cn=Do-Guk Kim, ou=People, dc=sparcs, dc=org
cn: Do-Guk Kim
gidNumber: 200
homeDirectory: /home/logue
loginShell: /bin/bash
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
uidNumber: 4002
User Name: logue
```

# Managing

- LDIF(LDAP Data Interchange Format)
  - 데이터 변경 형식

```
dn: cn=Do-Guk Kim, ou=People, dc=sparcs, dc=org
changetype: modify
replace: cn
cn: Bakwi Jang
```

```
<DN of the entry>
changetype: [modify | add | delete]
(if changetype is modify)[replace | add | delete]: <attribute>
Then enter the value of the attribute if necessary.
```

# Managing

- Schema

- LDAP에서의 schema 파일은 objectClass를 정의한다. 용도에 따라 최적화한 LDAP 서비스를 위해서는 schema 작성에 대해서도 알아야 하지만 흔히 쓰이는 Linux에서의 사용자 인증 목적에서는 새로운 schema 작성이 필요 없으므로 넘어가도록 하겠다.

# Practice

- LDAP Authentication

- 지금까지의 내용들로 LDAP 서버를 구축할 수 있다. 클라이언트는 여러 가지 종류가 있다. 예를 들자면 Linux에서 인증용으로 사용할 수도 있고 trac에도 LDAP 플러그인이 있다. 여기서 LDAP 서버 구축 및 LDAP을 통한 Linux 로그인을 실습해본다.

# Practice

- LDAP Authentication
  - 먼저 Installation과 Configuration 파트에서 설명한 대로 LDAP 서버 기초 설정을 완료한다.

# Practice

- LDAP Authentication
  - \$ sudo apt-get install migrationtools
  - \$ mv /usr/share/perl5/migrate\_common.ph /usr/share/migrationtools/

# Practice

- LDAP Authentication

- <migrate\_common.ph>

- \$DEFAULT\_MAIL\_DOMAIN = "sparcs.org";

- \$DEFAULT\_BASE = "dc=sparcs,dc=org";

- # cd /usr/share/migrationtools/

- # ./migrate\_group.pl /etc/group ~/group.ldif

- # ./migrate\_passwd.pl /etc/passwd

- ~/passwd.ldif

# Practice

- LDAP Authentication
  - # vi ~/people\_group.ldif

```
dn: ou=People, dc=sparcs, dc=org
ou: People
objectclass: organizationalUnit
```

```
dn: ou=Group, dc=sparcs dc=org
ou: Group
objectclass: organizationalUnit
```

# Practice

- LDAP Authentication

- # cd

- # ldapadd -D "cn=admin,dc=sparcs,dc=org"  
-W -f ~/people\_group.ldif

- # ldapadd -D "cn=admin,dc=sparcs,dc=org"  
-W -f ~/group.ldif

- # ldapadd -D "cn=admin,dc=sparcs,dc=org"  
-W -f ~/passwd.ldif

# Practice

- LDAP Authentication
  - \$ apt-get install libnss-ldap libpam-ldap nss-updatedb nscd ldap-auth-client

```
Should debconf manage LDAP configuration? Yes
LDAP server Uniform Resource Identifier: ldapi:///127.0.0.1
Distinguished name of the search base: dc=sparcs,dc=org
LDAP Version to use: 3
Make local root Database admin: Yes
Does the LDAP database require login? No
LDAP account for root: cn=admin,dc=sparcs,dc=org
LDAP root account password: <LDAP 비밀번호>
```

# Practice

- LDAP Authentication
  - `</etc/ldap.conf>`

host 127.0.0.1

nss\_base\_passwd ou=People,dc=sparcs,dc=org

nss\_base\_shadow ou=People,dc=sparcs,dc=org

nss\_base\_group ou=Group,dc=sparcs,dc=org

# Practice

- LDAP Authentication
  - `</etc/auth-client-config/profile.d/ldap-auth-config>`

```
pam_auth=auth      sufficient  pam_ldap.so
      auth        required    pam_unix.so nullok_secure use_first_pass
pam_account=account sufficient  pam_ldap.so
      account     required    pam_unix.so
pam_password=password sufficient  pam_ldap.so
      password    required    pam_unix.so nullok obscure min=4 max=8 md5
pam_session=session required    pam_unix.so
      session     required    pam_mkhome.so skel=/etc/skel/
      session     optional   pam_ldap.so
      session     optional   pam_foreground.so
```

`# command auth-client-config -a -p lac_ldap`

# Practice

- LDAP Authentication

- 이제 설정이 끝났다. 서버와 클라이언트가 연동되었고 인증 또한 잘 될 것이다.

- 현재 서버의 passwd 정보와 LDAP의 정보가 같아 확인이 어려우니 한번 passwd로 비밀번호를 바꿔보자. 비밀번호를 바꾼 후 LDAP password information changed 구문이 뜨는 것을 보면 LDAP으로 인증을 하고 있다는 사실을 알 수 있다.

코  
쑤

우 휘 이 크...

# Practice

- LDAP Authentication
  - LDAP을 사용한 인증을 할 경우 adduser로 사용자를 추가할 경우 자동으로 추가되지 않기 때문에 추가적으로 사용자 정보를 디렉토리에 추가해줘야 제대로 작동한다.

# Practice

- LDAP Authentication
  - # adduser <아이디>
  - adduser를 한 후 로그인을 시도해보자. 분명 서버의 passwd파일에는 사용자가 추가되었겠지만 LDAP에는 추가가 되지 않았으므로 접속이 안될 것이다.
  - migrate\_passwd.pl 을 이용해 새로 추가된 사용자의 정보만을 담은 ldif 파일을 생성한 후 ldapadd를 해주면 된다.

# Practice

- LDAP Authentication

- 사용자가 비밀번호를 바꾸었을 때도 LDAP에 갱신을 해줘야 새 비밀번호로 다른 곳에서도 접속할 수가 있다.

- <test.ldif>

```
dn: uid=elaborate,ou=People,dc=sparcs,dc=org
changetype: modify
replace: userpassword
userPassword: {crypt}$1$cSAAS32$4BY2TsdFASDuvS07yNxhasyddau0
```

```
# ldapmodify -W -D
```

```
"cn=admin,dc=sparcs,dc=org" -f ~/test.ldif
```

죄 짜 꼬  
죄 짜 꼬

**수고하셨습니다**

# Reference

- SPARCS Wheel Wiki  
<https://sparcs.kaist.ac.kr/wheel/wiki/Processes/NFS%2BLDAP>
- Ubuntu Server Guide  
<https://help.ubuntu.com/10.04/serverguide/C/serverguide.pdf>

# Reference

- OpenLDAP Software 2.4 Guide  
<http://www.openldap.org/doc/admin24/>
- Reference about Idapsearch  
<http://tille.garrels.be/training/ldap/ch03.html>

# Reference

- Wikipedia OpenLDAP  
<http://en.wikipedia.org/wiki/OpenLDAP>
- LDAP에 대한 모든 것(워드 문서)  
<http://50001.com/sub/down/ldap.doc>
- 2010년 훔 세미나 LDAP by harry