Integrating HP-UX Authentication with Windows 2000 Active Directory

Doug Lamoureux
Technical Consultant
Systems Networking and Security Lab
Hewlett-Packard

Agenda

- Problem description
- Authentication and account management history
- A solution and technologies
- Making it Work
- The “big” picture
- Other technologies
- Limitations

The Problem

- Multiple accounts for the same user
- Duplicate user and group administration
- Multiple password policies
- Account synchronization

Resulting in

- Increased administration costs
- User confusion – multiple logins and passwords
- Increased security risks

Agenda

- ✓ *Problem description*
- **Authentication and account management history**
- A solution and technologies
- Making it Work
- The "big" picture
- Other technologies
- Limitations

History

- Why have we not integrated users in the 2 worlds before?
  - No need
  - Proprietary technologies used to store and authenticate users

HP-UX Authentication and Account Management History

Yesterday

Account management and authentication technology are tightly integrated

- Files
  - insecure
  - not scaleable
  - localized
- NIS
  - insecure
  - not scalable
  - distributed

HP-UX Authentication and Account Management History

Yesterday *(cont.)*

- NIS+
  - Secure
  - More scalable
  - Distributed
  - Too complex to administer
  - Not widely accepted
  - No future – being discontinued

HP-UX Authentication and Account Management History

Today

- Account management and user authentication can now be de-coupled
  - LDAP (account management & authentication)
    - scalable
    - distributed
    - standards based
    - secure and insecure
  - Kerberos (authentication)
    - secure
    - distributed
    - standards based
    - scalable

Windows Authentication and Account Management History

- **Yesterday**
  - LAN manager
  - NT Chap
  - NTLM
- **Today**
  - LDAP
  - Kerberos

Agenda

- ✓ *Problem Description*
- ✓ *Authentication and Account Management history*
- ❑ **A solution and technologies**
- ❑ Making it Work
- ❑ The "big" picture
- ❑ Other technologies
- ❑ Limitations

A Solution

- Common authentication protocol:
  Kerberos
- Common access protocol:
  LDAP
- Common data repository:
  Active Directory

Solution Technologies

HP-UX

- Plugable Authentication Module *(PAM)*
- Name Service Switch *(NSS)*
- **Kerberos** *(pam_krb5)*
- **LDAP** and LDAP-UX Integration *(NSS ldap)*

Windows 2000

- Active Directory (**LDAP/KRB**)
- Services for Unix *(SFU)*
- **Kerberos** Server *(KDC)*

Technologies: **Kerberos**

Is

- A network authentication protocol
- Uses a trusted 3rd party (KDC) to distribute "tickets"
- Can be used to encrypt data between clients and servers

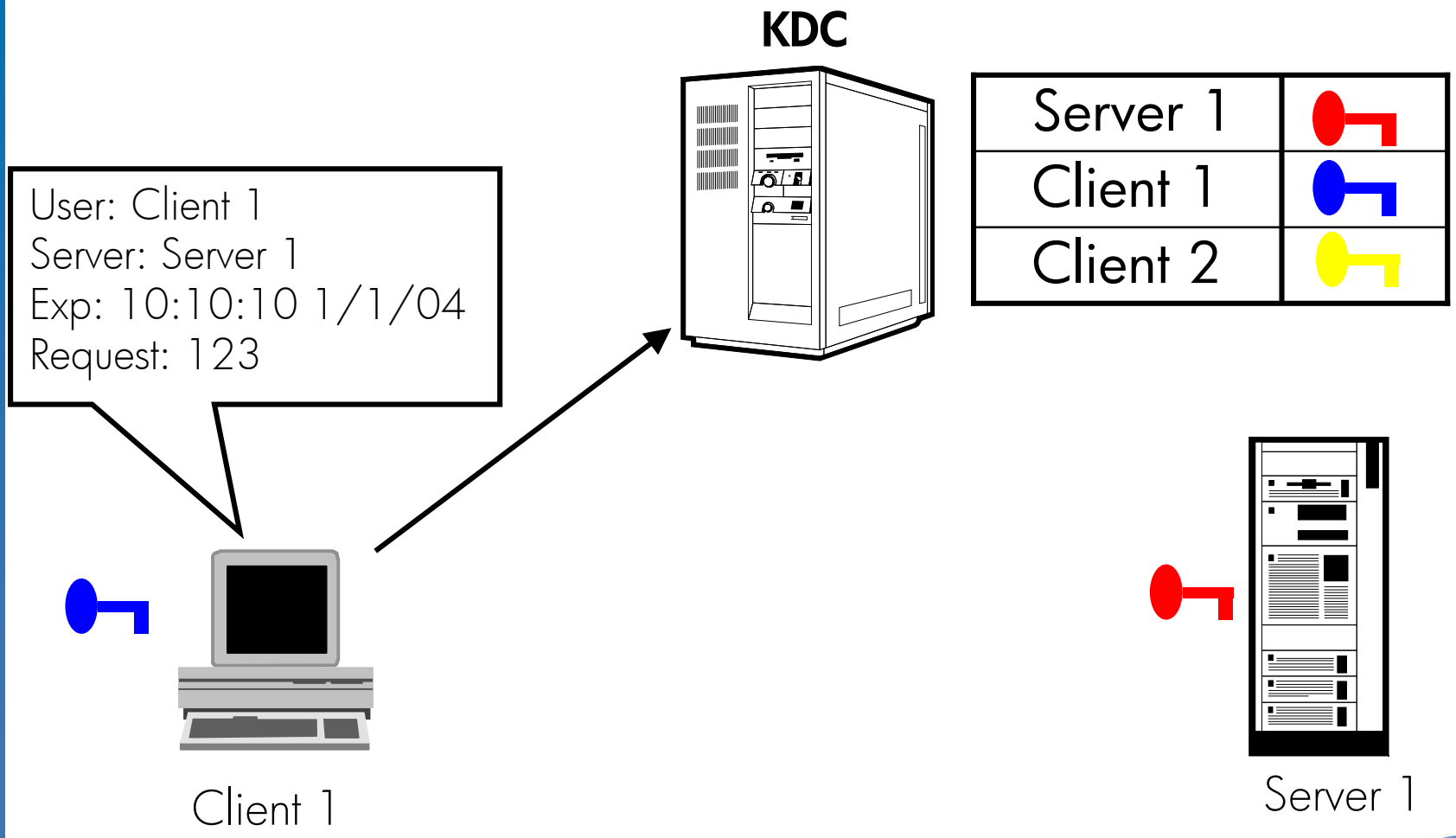
Is not

- A network authorization protocol
- A repository for user account information

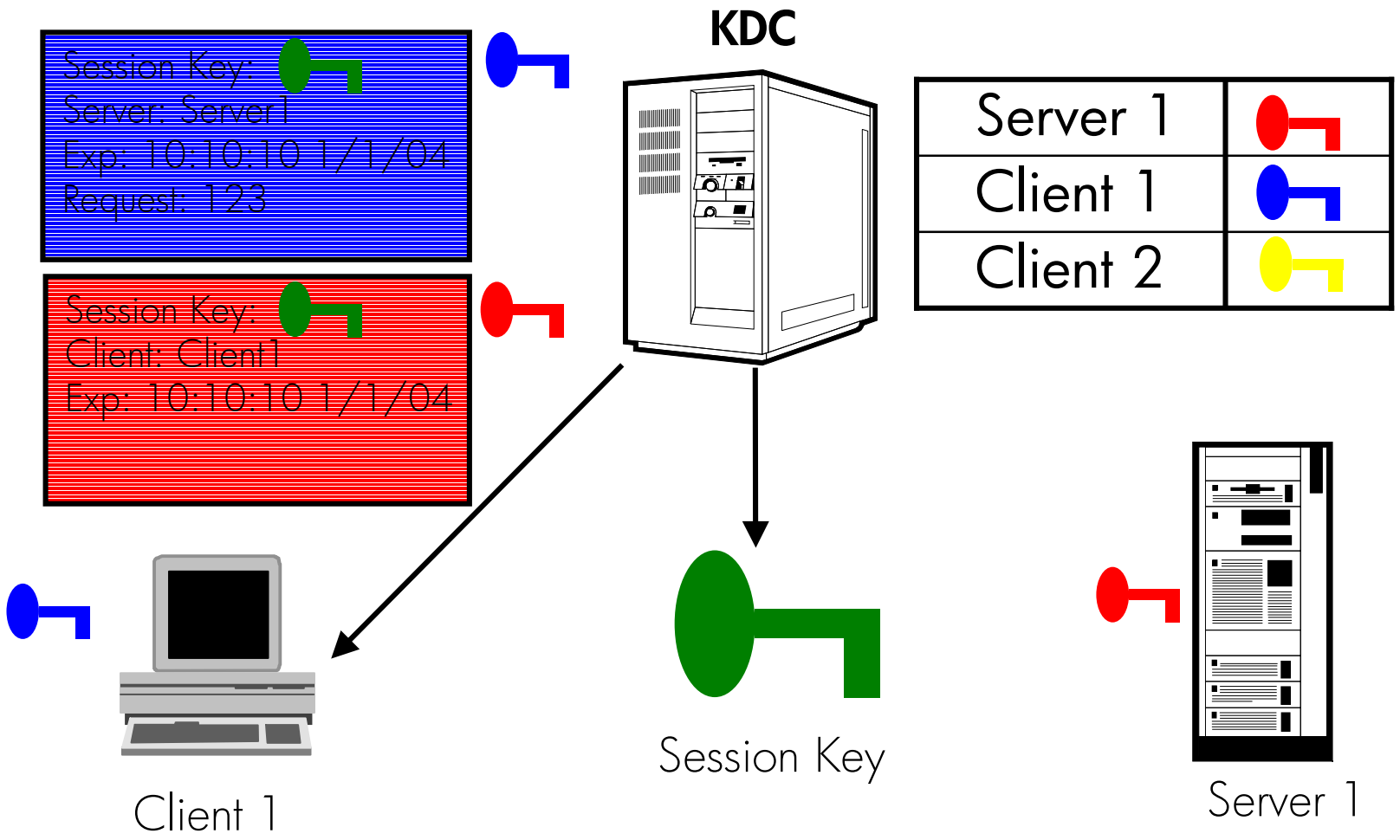
Kerberos: Session Tickets

- To access a "service" clients request a ticket for that service from the KDC
- The KDC generates session ticket for that client to present to that service to identify itself
- Session Ticket includes a key that is used as a shared secret between the client and service
- These tickets usually have a lifetime associated with them
- *Can* contain authorization data

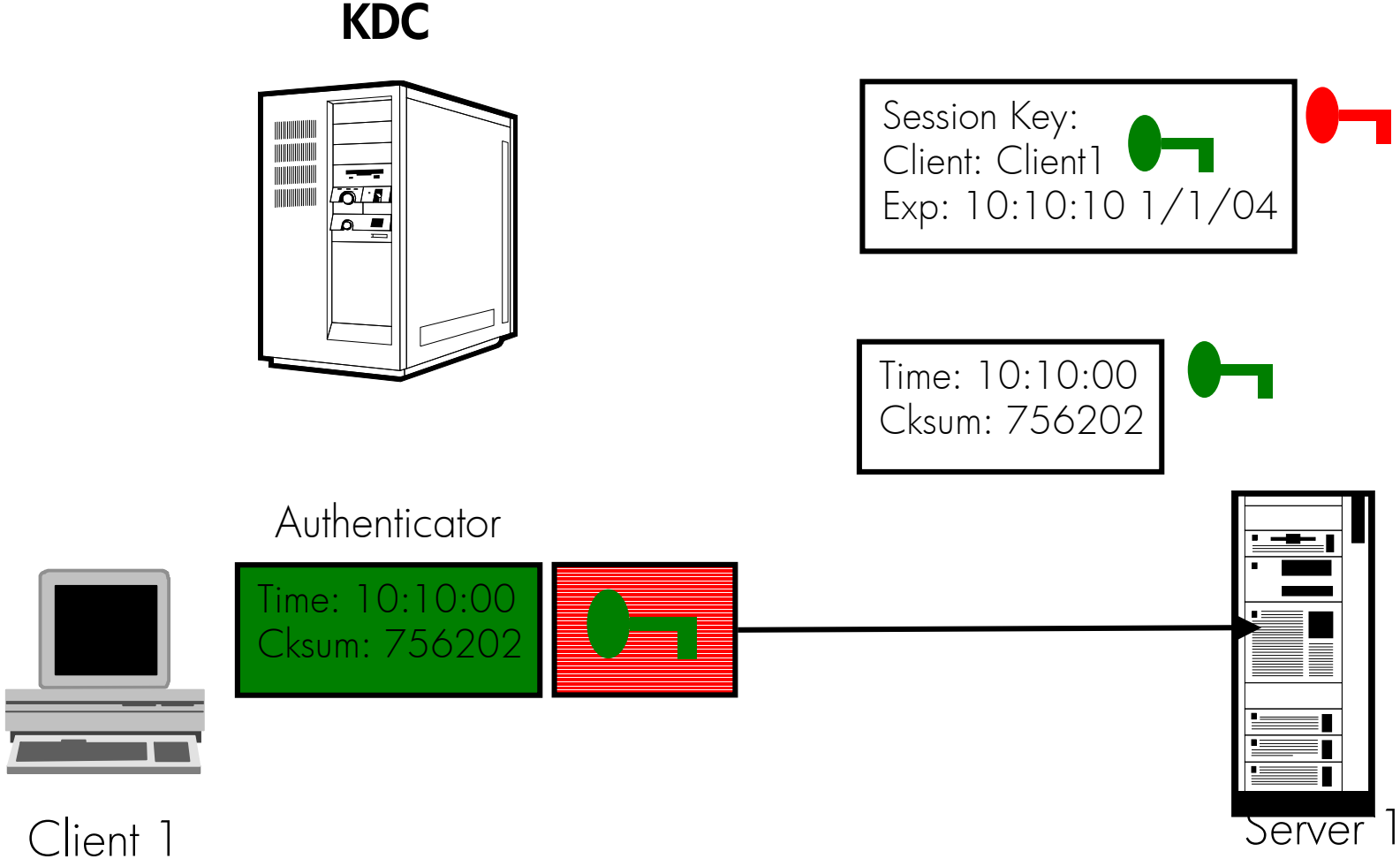
Kerberos – obtaining and using a ticket (1)



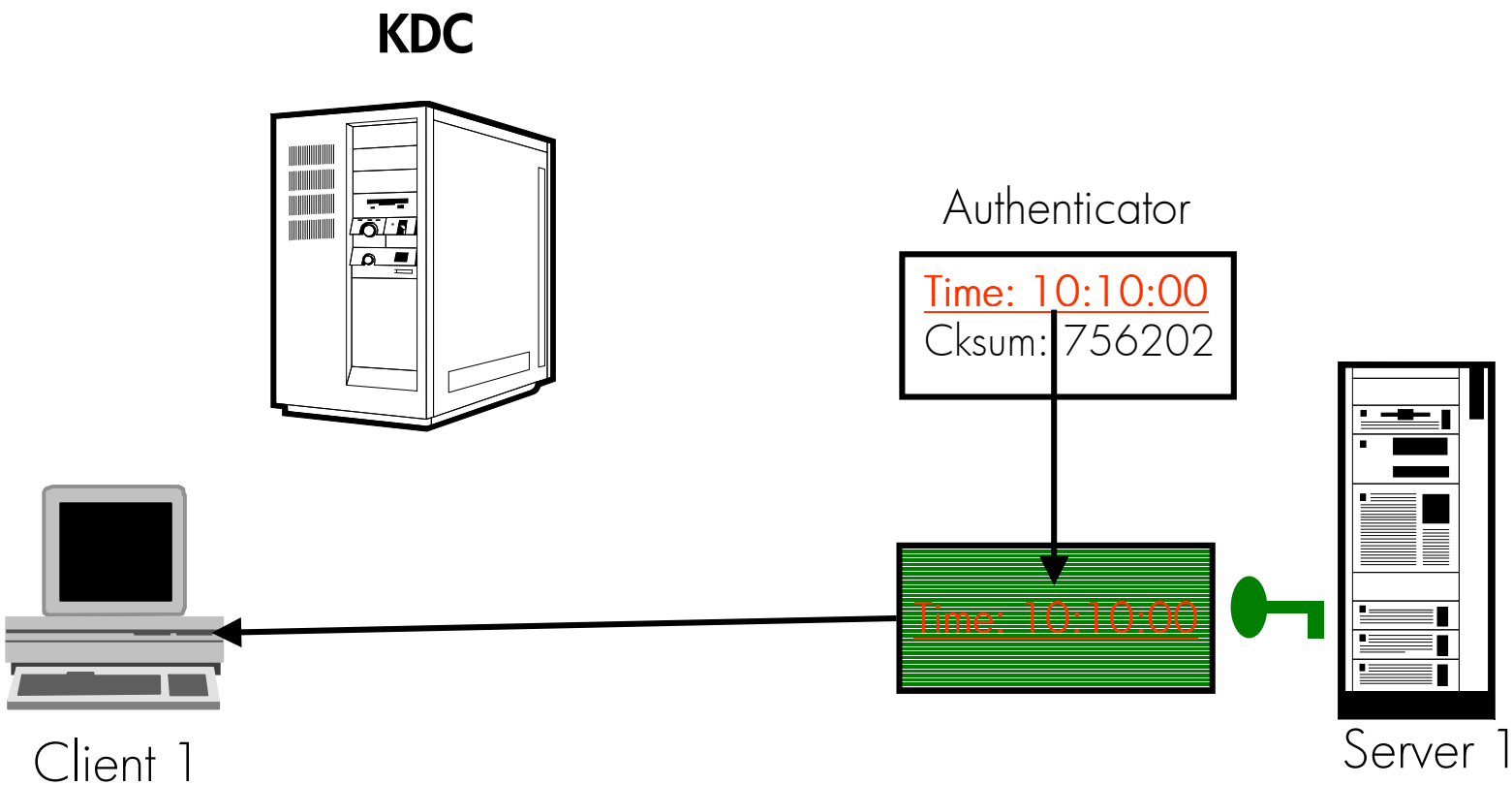
Kerberos: Obtaining and using a ticket (2)



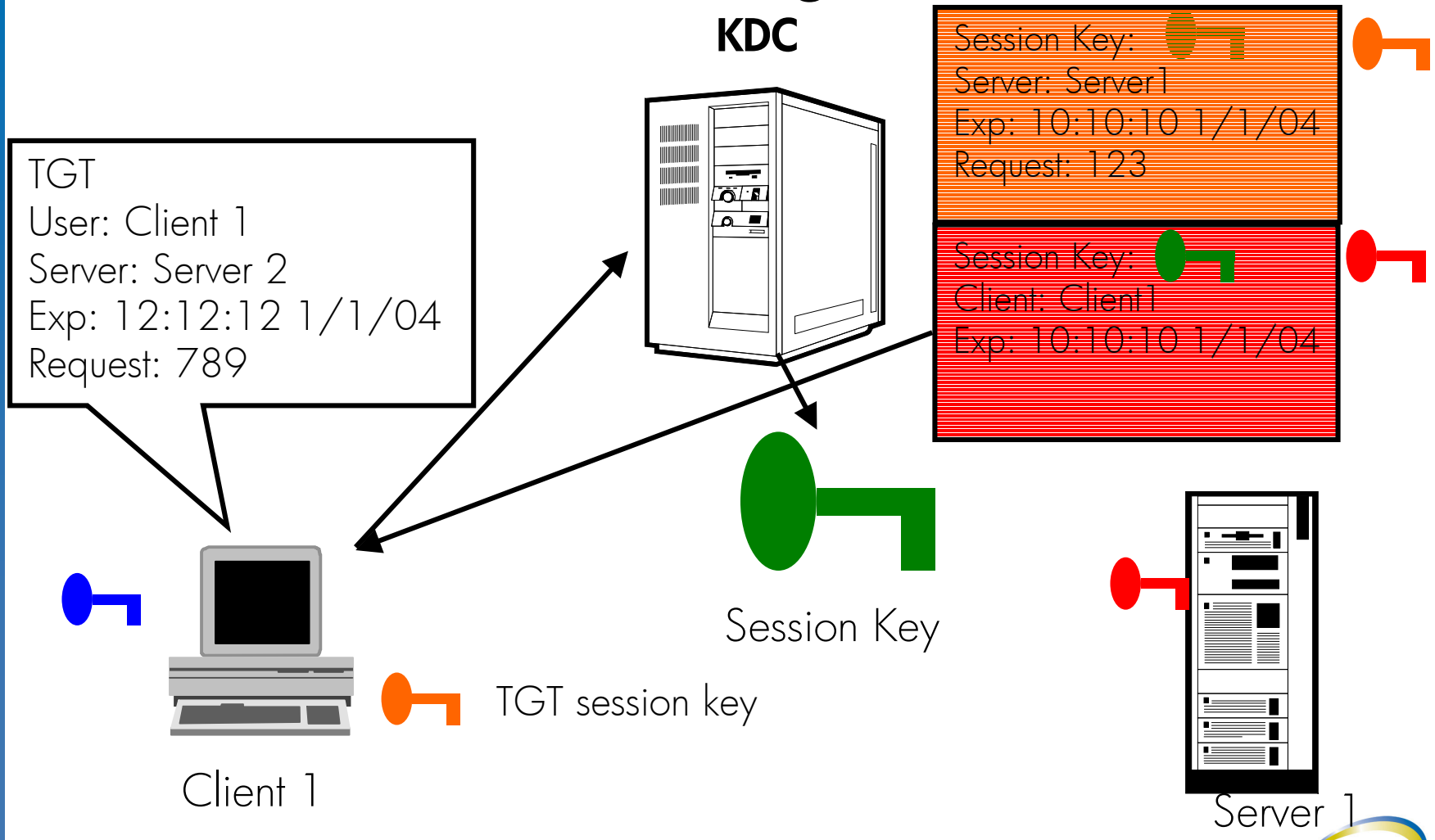
Kerberos: Obtaining and using a ticket (3)



Kerberos: Obtaining and using a ticket (4) Mutual Authentication



Kerberos: Ticket Granting Ticket (TGT) Service



Kerberos: Client Configuration

Kerberos configuration file: **/etc/krb5.conf**

```
[libdefaults]
        default_realm = ACME.COM
        default_tkt_enctypes = DES-CBC-CRC
        ccache_type = 2
        default_keytab_name = FILE:/etc/krb5.keytab

[realms]
        ACME.COM = {
                kdc = kdc1.west.acme.com:88
                kdc = kdc2.west.acme.com:88
                admin_server = kdc1.west.acme.com
        }

[domain_realm]
        .acme.com = ACME.COM
        .realm.com = ACME.COM
        .net = KRB.NET
```

Technologies: **LDAP**

- LDAP (v3) –> Lightweight Directory Access PROTOCOL defined in a set of RFC's outlined in RFC 3377
- Based on the X.500 (OSI Directory Access Protocol) standard, but much simpler (LDAP supports TCP/IP)
- Defines how a client accesses information stored in a Directory, NOT how the information is stored

What is a Directory?

- Is an LDAP Directory a Database?

  Yes it is a *type* of DB, but not a relational DB like Oracle, Sybase

LDAP Directory	**Relational Database**
Hierarchical	Rows & Columns
Variable/Multiple attributes	Structured Attributes
Tuned for read performance	Tuned for OLTP (read/write) performance

Directory Servers

- Historically used to hold contact information (phone book) and Web Portal registrations
- Now Directory Servers have become a centralized storage for Enterprise and Extranet data
  - Server configuration
  - X.509 certificates
  - Customer Data
  - DNS Data
  - More..
- MS Windows 2000, Novell eDirectory use Directories for NOS storage

What Makes Up a Directory?

- **Attribute** – Defines a piece of data within an objectclass
- **Objectclass** – The definition of an object type stored in the Directory (similar to a typedef in C)
- **Schema** –The rules that define how data is stored within the Directory (Extendable)
- **DIT (Directory Information Tree)** – All of the entries within the Directory Tree

Attributes

- Defines a single piece of data
- An attribute definition contains many fields:
  - OID (Object Identifier): Must be unique within the Directory Schema, defined in ASN.1 format
  - Name of the attribute (also unique)
  - Description of the attribute
  - Matching Rule and Qualifier *(how)*
  - Syntax
  - Single/Multi valued flag

Attribute Example

```
( 2.16.840.1.113730.3.1.39
  NAME 'preferredLanguage'
  DESC 'preferred written or spoken language for a
                person'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  SINGLE-VALUE )
```

Common Attributes

- cn – Common Name (John Smith)
- sn – Family/Sur Name (Smith)
- uid – User ID (jsmith)
- c – Country Name (CA)
- o – Organization (Hockey Town)
- l – Locality (Sudbury)
- member – Member of a group (DN)
  (uid=jsmith,ou=users,o=acme.com)
- givenName – Given Name (John)
- mobile – Mobile Phone (+1 555 473 1234)

Objectclass

- Defines the structure of an object
  - Name
  - Valid attributes (required or optional)
  - Type
  - Superior objectclasses (attributes are inherited from superior objectclasses)
- Multiple Types of objectclasses
  - Structural: Basic building block of an object
  - Auxiliary: Used to add additional data to the object
  - Abstract: Superclass (top, alias) of objects which define the basic blocks of the Data Model (objects are not created with this objectclass)

Objectclass Example

(2.5.6.2
  NAME 'country'
  DESC 'Standard LDAP objectclass'
  SUP top
  MUST (c)
  MAY (searchGuide $ description)
  X-ORIGIN 'RFC 2256')

Common Objectclasses

- OrganizationalUnit - OU
- Organization - O
- Domain - DC
- Locality - L
- Country - C

Object Example

dn: cn=Joe Admin, cn=Users, dc=hp, dc=com

objectclass: top

objectclass: person

objectclass: organizationalPerson (SUP Person)

objectclass: inetOrgPerson (SUP organizationalPerson)

userpassword: {MD4}1Vfgr/kBfbGn1nbnaqUZdg==

description: Joe Admin User

sn: Admin

cn: Joe Admin

title: Administrator

employeenumber: 123456

givenname: Joe

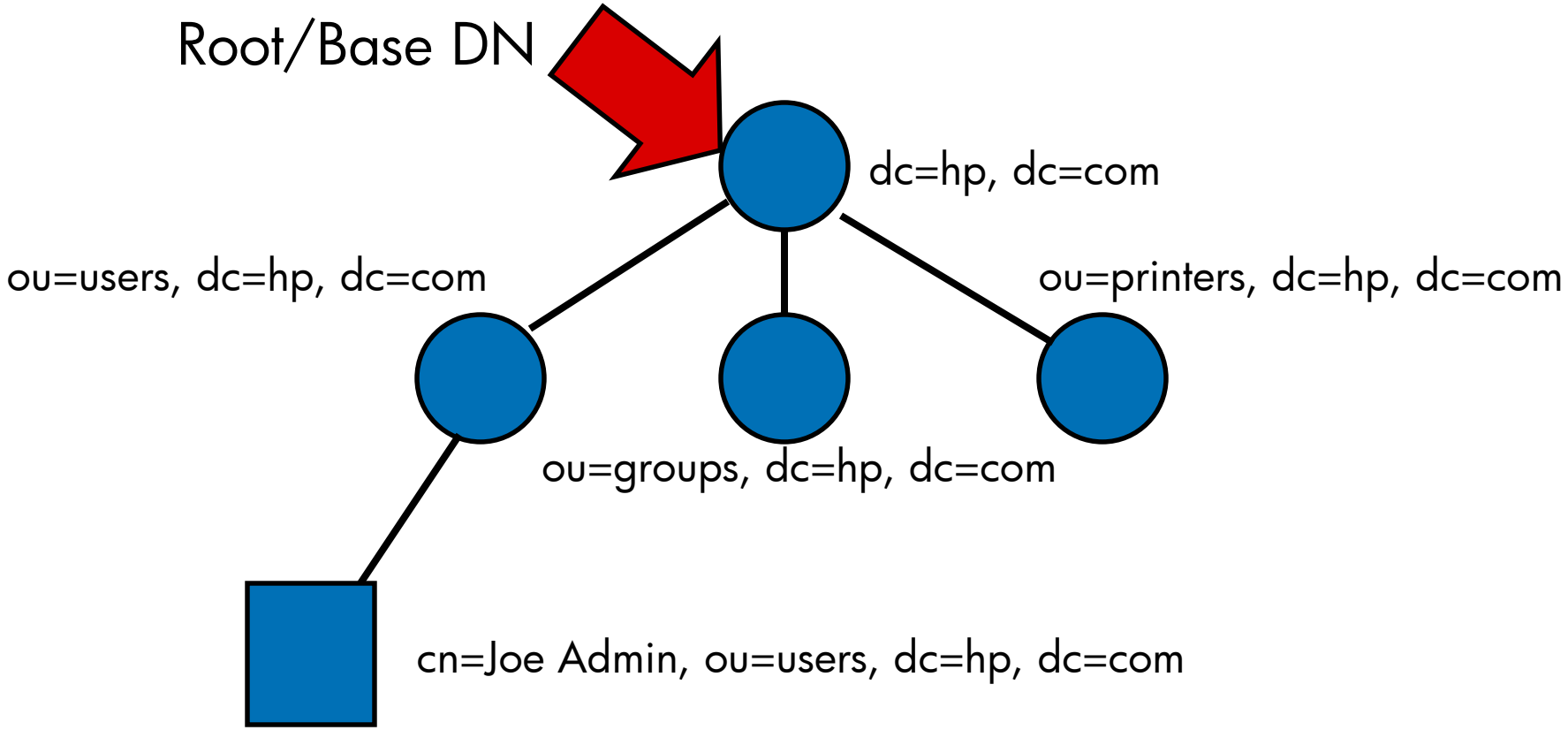
mail: joe@hp.com

uid: joe

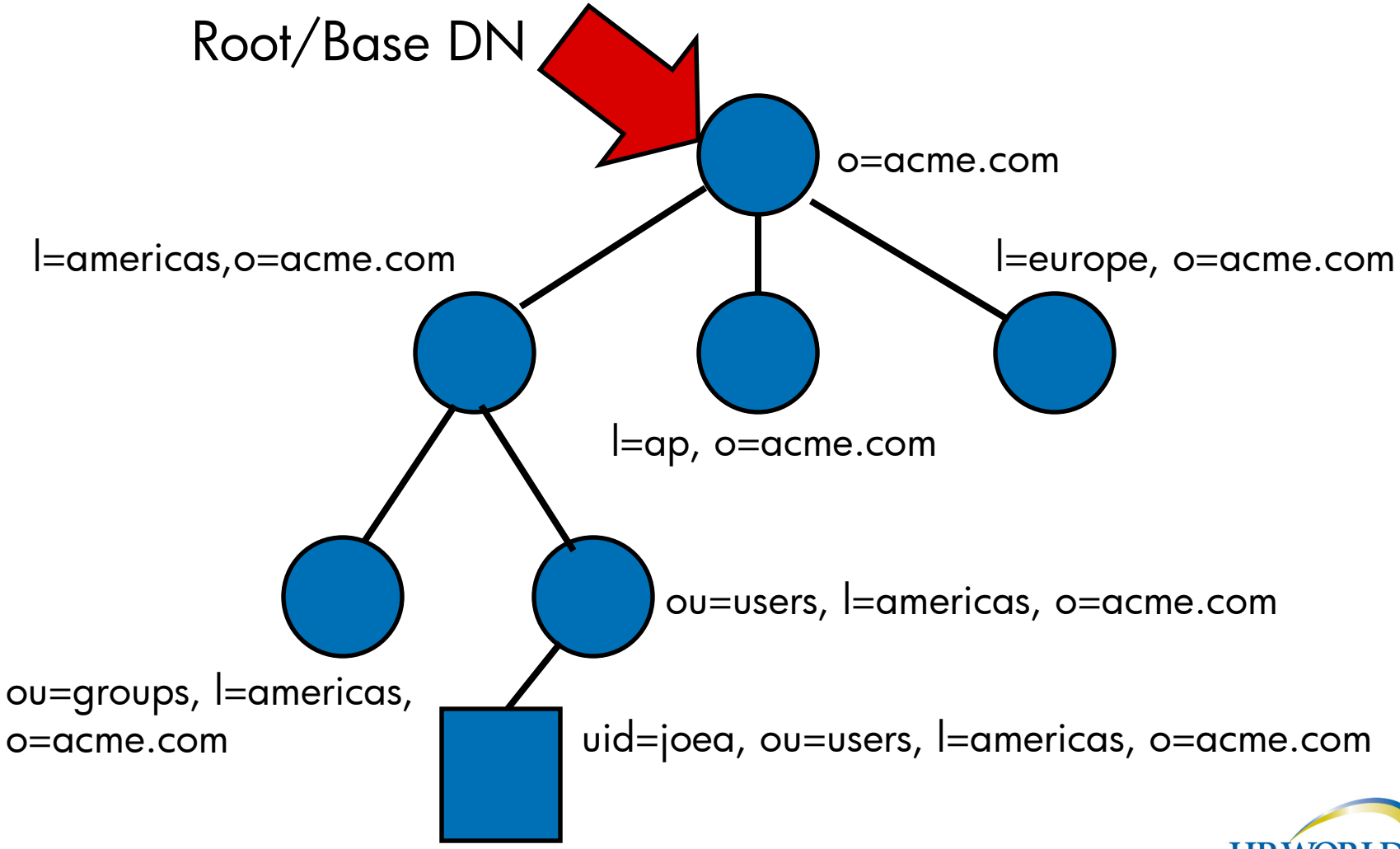
What does a Directory Look like?

- An LDAP Directory is Hierarchical
- Single "Root" (BASE)
  - Can be shallow and wide, narrow and deep or somewhere in between
  - Similar to a DNS Tree or Posix/Unix filesystem
- A collection of leaf and container objects
- Provides Access Control to the data stored within the Directory

Directory Layout



Directory Layout (2)



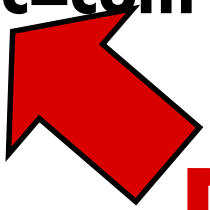
Naming Objects in the Directory

- **Distinguished Name (DN):**
  The full name of the object that uniquely identifies it within the scope of the DIT
- **Relative Distinguished Name (RDN):**
  The name that uniquely identifies it within the scope of the superior object *(**NOT** superior objectclass)*

dn: cn=Joe Admin, cn=Users, dc=hp, dc=com

objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
userpassword: {SSHA}1Vfgr/kBfbGn1nbnaqUZdg==
description: Joe Admin User
sn: Admin

cn: Joe Admin



DN

RDN

Access to the Directory

- Before accessing data within the Directory the user/application must “Bind” to the Directory
- Binding to the Directory is either authenticated or anonymous
- Applications can use SSL/TLS to “secure” communication between the LDAP client and Directory Server
- Additional authentications can be used depending on support by the Directory Server and LDAP client (usually defined by Extended Operations)

Access Control

- Most, if not all, Directory Servers provide a method in which access to data can be secured
- Actions that can be typically controlled by ACL's:
  - read
  - write
  - search
  - add
  - delete
  - selfwrite
  - compare
- Unfortunately each Directory Server implements them differently

Searching the Directory

- Using LDAP Search filters client applications ask the Directory Server to search for matches to objects that meet the clients requirements
- LDAP search filters are made up of one or more Boolean expressions
- Wildcards are valid for substring searches
- Searches can be limited in scope
- Example:

  (|(&(cn=j*)(sn=d*))(cn=p*))

  find all entries that

  - have a cn that starts with j **and** an sn that starts with d
  - have a cn that starts with p

Replication

- For High Availability and Load Balancing multiple Directories can be deployed
- Multiple Directory Servers are deployed as:
  - Master/Slave – (Single read/write, Multiple read)
  - Multi-Master – (Multiple read/write)
  - Combination of both
  - Referrals
- Some Directory Servers do not support Multi-Master
- Like ACL's each Directory Servers has implemented replication differently
- IETF working on LDUP (LDAP Duplication/Update Protocol)

RFC 2307

- The Network Information Service (NIS) schema defined in rfc 2307
  - posixAccount
  - posixGroup
  - other nis “maps” such as services (ipservices), protocols (ipprotocols), etc…

RFC 2307 posixAccount

Required attributes:

- cn *(common name)*
- uid *(user id, name)*
- uidnumber *(user id number)*
- gidnumber *(group id number)*
- homedirectory *(home directory)*

Additional attributes:

- userpassword *(password)*
- loginshell *(login shell)*
- gecos *(name, phone number, location)*
- description

RFC 2307 posixAccount

Example in ldif *(ldap data interchange format)* format:

```
dn: uid=jdoe, ou=People, ou=ldap-ux, dc=acme, dc=com
uid: jdoe
cn: John Doe
objectclass: top
objectclass: account
objectclass: posixAccount
loginshell: /usr/bin/ksh
uidnumber: 223
gidnumber: 20
homedirectory: /home/jdoe
gecos: John Doe,,,
userpassword: {crypt}ask4kskHhFl=
```

RFC 2307 posixGroup

Required attributes:

- cn *(common name, group name)*
- gidnumber *(group id number)*

Additional attributes:

- userpassword *(group password)*
- memberuid *(uid, name, of group members)*
- description

RFC 2307 posixGroup

Example in ldif *(ldap data interchange format)* format:

dn: cn=users, ou=Group, ou=ldap-ux, dc=acme,dc=com
objectclass: posixGroup
objectclass: top
cn: users
gidnumber: 20
memberuid: root
memberuid: jdoe
memberuid: jcool

HP-UX Technologies: **PAM**

Plugable Authentication Module

- Allows server/service developers to write their authentication code to a standard api
- Allows system administrators to choose which authentication module(s) to authenticate users with
- By "stacking" modules a user can be authenticated by multiple modules
- Allows new authentication technology to be added without modifying existing applications

PAM Configuration

- PAM configuration file:
  /etc/pam.conf
- Each entry is a single line containing:

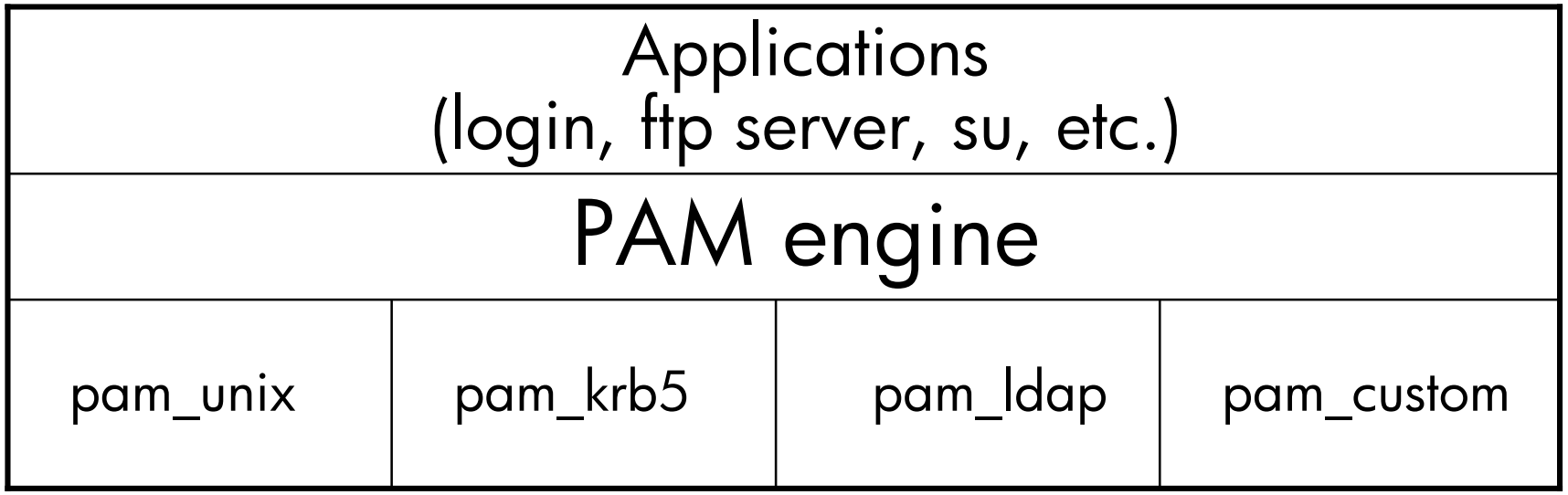
  service_name – a service such as login, dtlogin, ftp
  module_type – the type of module (auth, account, session & password)
  control_flag – controls how stacking is interpreted
  module_path – path to the library of this module
  options – a list of options to passed to the module

PAM configuration (cont.)

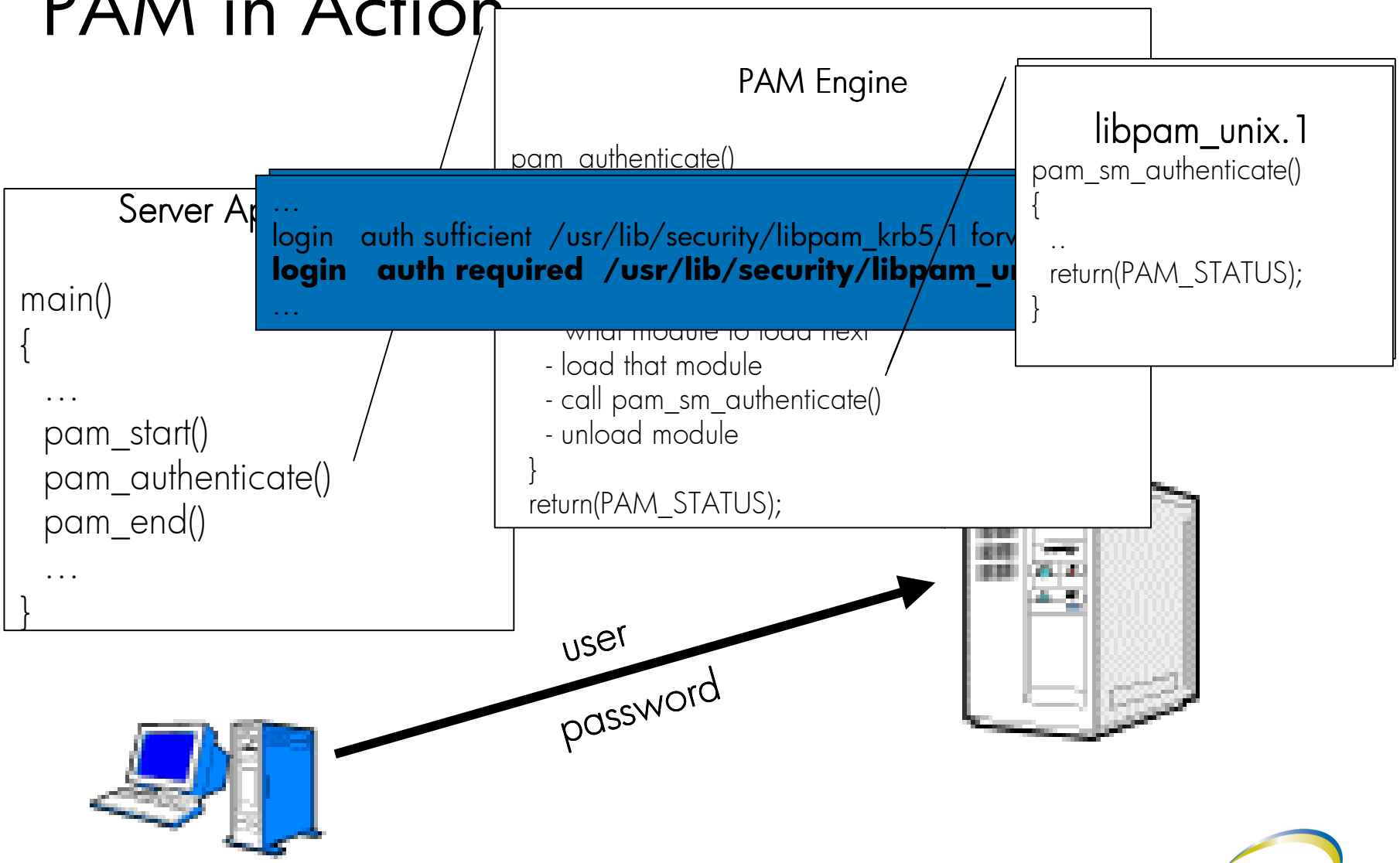
Example:

```
login auth  sufficient   /usr/lib/security/libpam_krb5.1 forwardable
login auth  required   /usr/lib/security/libpam_unix.1 try_first_pass
ftp   auth  sufficient   /usr/lib/security/libpam_krb5.1
ftp   auth  required   /usr/lib/security/libpam_unix.1 try_first_pass

login       password required /usr/lib/security/libpam_krb5.1
login       password required /usr/lib/security/libpam_unix.1
passwd      password required /usr/lib/security/libpam_krb5.1
passwd      password required /usr/lib/security/libpam_unix.1
```

PAM Architecture



PAM in Action



HP-UX Technology: **NSS**

Name Service Switch

- Used to determine the source from which various system calls (i.e. getpwent) will attempt to retrieve data from
- Multiple sources can be configured
- Possible to configure different actions for each status returned from each source

NSS

2 different types of NSS API's

- Searches
  - look for a specific entry
  - quick response
- Enumeration
  - look at all entries in a "database"
  - increase network load
  - increase "server" load

NSS Configuration

- NSS configuration file:
  /etc/nsswitch.conf
- Each entry is a single line containing:

  database – the name associated with the type of information to be looked up (i.e. hosts, passwd)
  source – the module name used to query (multi-valued)
  criteria – controls what should be done for each possible return status (optional)

NSS Configuration (example)

hosts: dns [NOTFOUND=continue TRYAGAIN=return] files nis

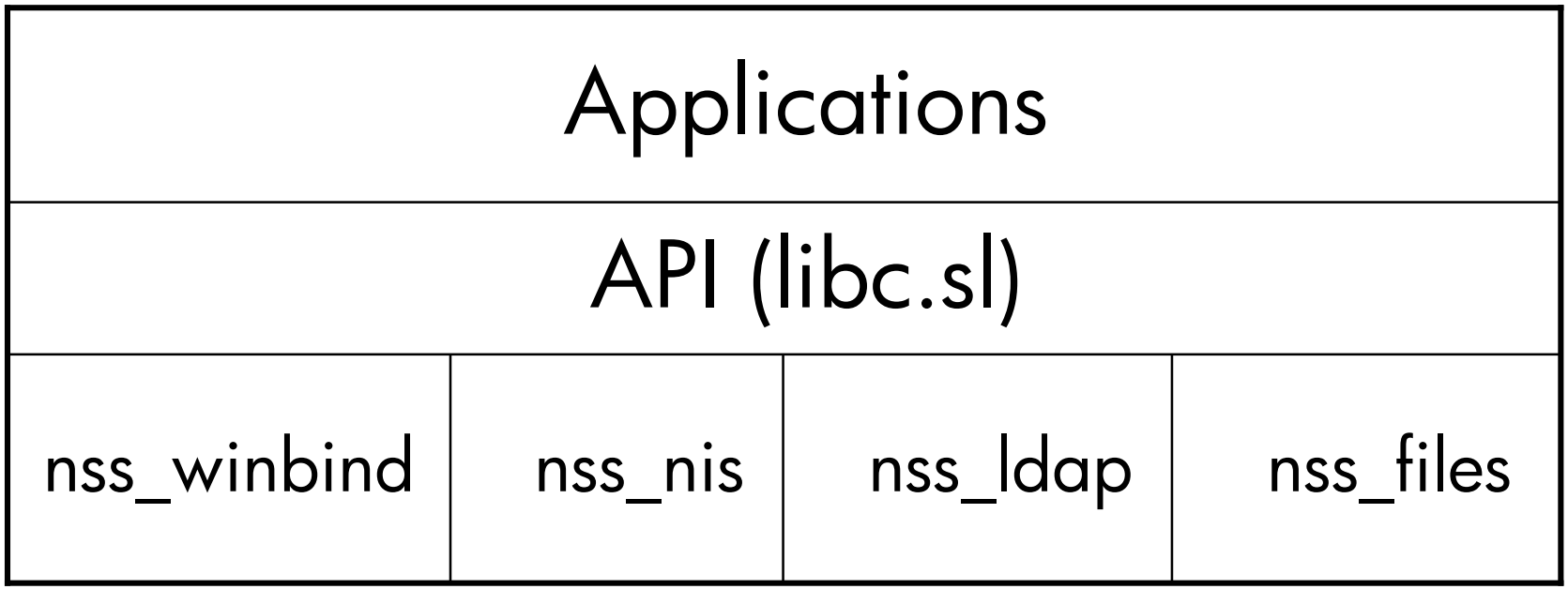
passwd: ldap [NOTFOUND=continue TRYAGAIN=continue] files

group: ldap [NOTFOUND=continue TRYAGAIN=continue] files

networks: files nis

protocols: nis [NOTFOUND=continue TRYAGAIN=return] files

NSS Architecture



HP-UX Technology: **LDAP-UX Integration**

LDAP-UX Integration consists of 2 products

- NIS/LDAP Gateway
  - Allows clients that don't support ldap natively to use an LDAP directory to store/retrieve users and groups
  - The gateway replaces existing NIS slaves
- LDAP-UX Client
  - Allows HP-UX systems direct access to Posix information in LDAP v3 compliant directories
  - Allows users to be authenticated against an LDAP v3 directory
  - LDAP Printer configuration

LDAP-UX Client

- **nss_ldap** module used to access OS information, typically nis "maps" like passwd, group, hosts, etc.) in an LDAP directory
- **pam_ldap** module provides LDAP based authentication
- LDAP Client Daemon (ldapclientd)
  - Data caching
  - Persistent connections
  - Updates local configuration profile

LDAP-UX Client

- Flexible profile configuration
  - attribute mapping
  - search filters
  - configurable access
  - multiple profiles
- Command line LDAP tools
  - ldapentry
  - ldapmodify
  - ldapsearch
  - etc.
- LDAP C-SDK
- Migration Tools

LDAP Attribute Mapping

Allows one attribute to be mapped to another:

```
attributemap: passwd:userpassword=*NULL*
attributemap: passwd:gecos=cn building telephonenumber
attributemap: passwd:uidnumber=employeeId
attributemap: passwd:homedirectory=msSFUHomeDirectory
```

LDAP Search Filters

- Allows the default LDAP search parameters to be refined when looking up entries in the directory
- Used to restrict access to systems

```
servicesearchdescriptor:passwd:ou=ATC,dc=hp,dc=com?sub? \
        (&(objectclass=posixAccount)(employeetype=engineer))
servicesearchdescriptor:pam:ou=ATC,dc=hp,dc=com?sub? \
        (&(objectclass=posixAccount)(employeetype=engineer))
# ldapsearch -b "ou=ATC,dc=hp,dc=com" uid=bsmith employeetype
  employeetype: operator
# ldapsearch -b "ou=ATC,dc=hp,dc=com" uid=dougl employeetype
  employeetype: engineer
```

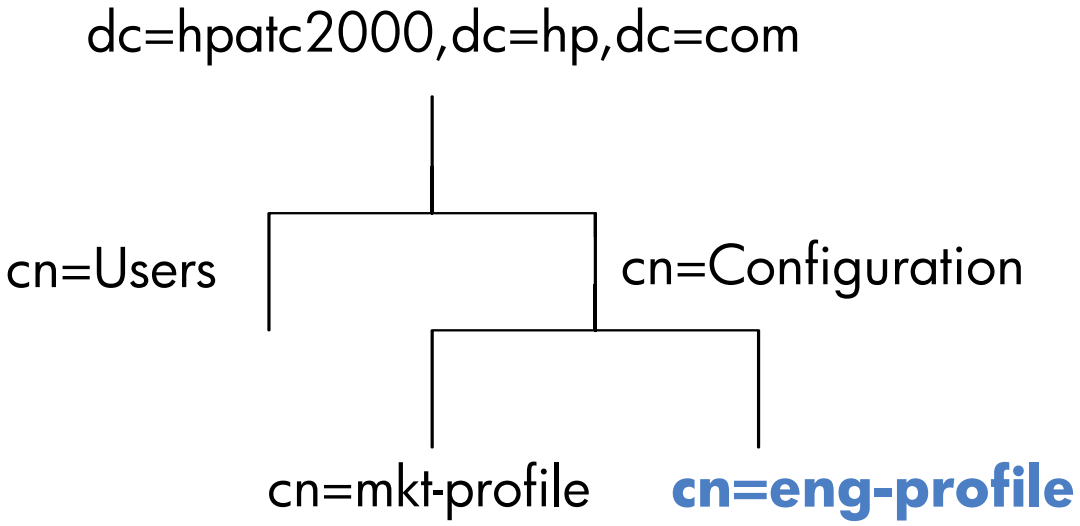
Configurable Directory Access

- Determines how the LDAP-UX Client Daemon (ldapclientd) bind to the directory during searches
  - anonymous
  - proxy user

Multiple LDAP-UX Profiles

- Multiple profiles can be created based on enterprise needs
- Using multiple profiles allows all users to be stored in a central directory but restrict access to a system or group of systems

Configuration Profile



Configuration Profile

dn: **cn=eng-profile, CN=Configuration, DC=hpatc2000, DC=hp, DC=com**
cn: eng-profile
distinguishedName:
CN=eng-profile, CN=Configuration,DC=hpatc2000,DC=hp,DC=com
objectClass: top
objectClass: DUAConfigProfile
objectGUID:: hbt8w23nXES2B6DVx9fkvA==
whenChanged: 20040708233940.0Z
whenCreated: 20040708232445.0Z
attributeMap: services:ipserviceprotocol=msSFU30IpServiceProtocol
attributeMap: services:ipserviceport=msSFU30IpServicePort
attributeMap: services:cn=msSFU30Name msSFU30Aliases
attributeMap: hosts:iphostnumber=msSFU30IpHostNumber
attributeMap: hosts:cn=cn msSFU30Aliases
attributeMap: networks:ipnetworknumber=msSFU30IpNetworkNumber
attributeMap: networks:cn=cn msSFU30Aliases

Configuration Profile

On the LDAP-UX Client

LDAP-UX Bootstrap file: **/etc/opt/ldapux/ldapux_client.conf**

LDAP_HOSTPORT="192.1.1.1:389"
PROFILE_ENTRY_DN=
"**cn=eng-profile, CN=Configuration, DC=hpatc2000, DC=hp, DC=com**"

Configuration Profile

On the LDAP-UX Client:

- Download a local copy of the profile in LDIF format: **/etc/opt/ldapux/dapux_profile.ldif**
- Run create_profile_cache to create binary profile cache file: **/etc/opt/ldapux/ldapux_profile.bin**
- LDAP-UX client daemon (ldapclientd) reads configuration into memory

Windows Technology: **Active Directory**

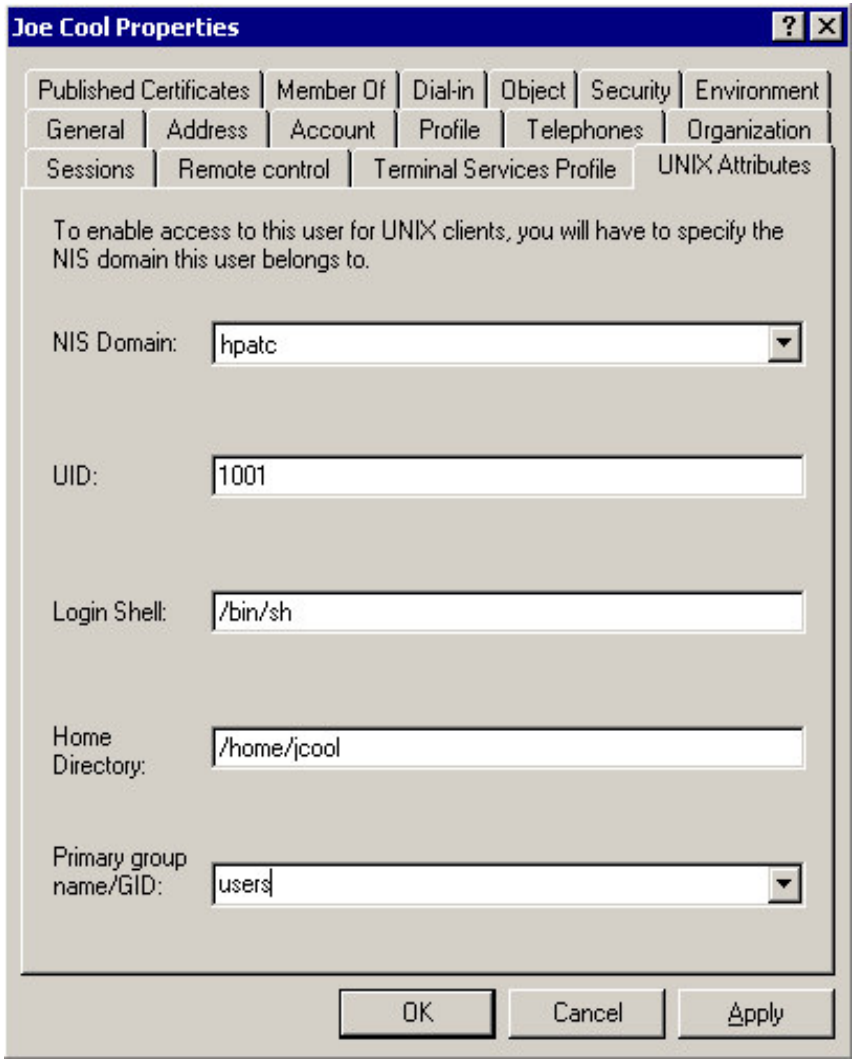
- Central component of the Windows 2000/3 networking architecture
- Allows enterprise resource to be stored to and retrieved from a distributed, replicated location
- Centralized point of user and group management
- Supports the LDAP protocol for remote access to directory objects
- Supports SASL(GSSAPI/Kerberos) for authenticated directory access
- Multi-master

Windows Technology: **Services For Unix (SFU)**

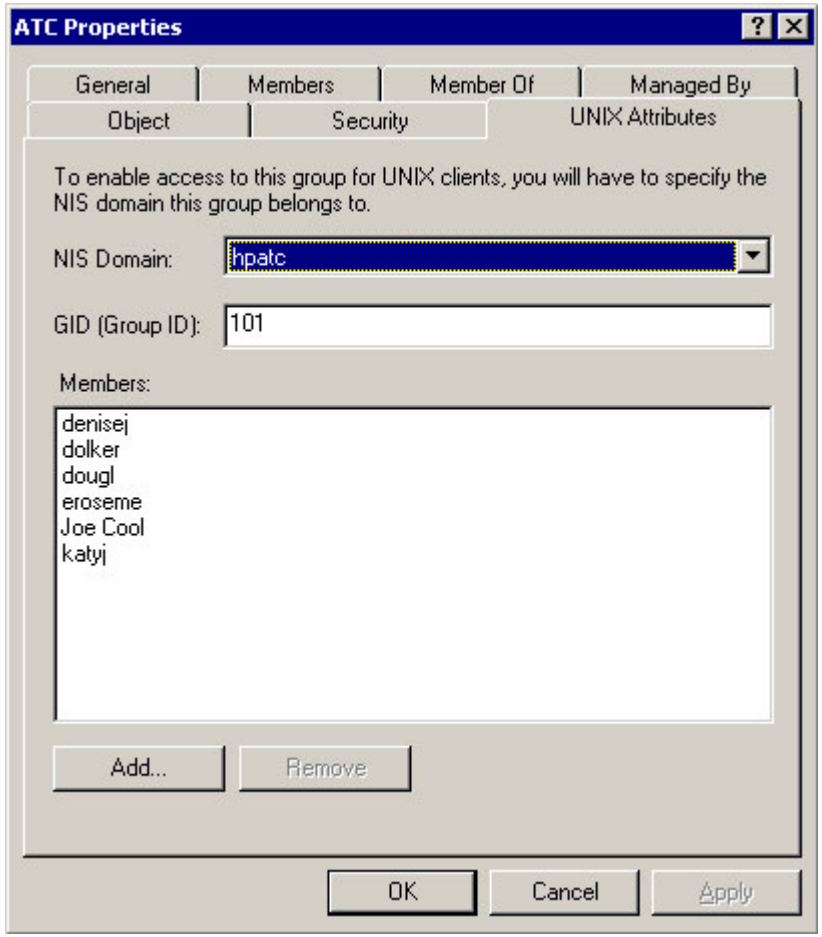
- Provides several "Unix" centric services for the Windows 2000 platform
- Extends the Active Directory's schema to include Posix attributes (RFC 2307 "***like***")
- Adds support for Unix attribute management in the "active directory users and computers" tool

NOTE: Only the schema update and configuration tool enhancements are used. All other SFU Services can be disabled

Services For Unix: User Management



Services For Unix: Group Management



RFC 2307 vs. SFU 2.0/3.x

posixAccount		
Attribute	**RFC 2307**	**SFU 2.0/3.x**
user name	uid	msSFUname/ msSFU30Name
user id number	uidnumber	uidnumber/ msSFU30UidNumber
primary group id number	gidnumber	gidnumber/ msSFU30GidNumber
users login directory	homedirectory	msSFUHomeDirectory/ msSFU30HomeDirectory
users password	userpassword	msSFUPassword/ msSFU30Password
users login shell	loginshell	loginShell/ msSFU30LoginShell
users gecos information	gecos	gecos/ msSFU30Gecos

SFU2307 vs. SFU 2.0/3.x

posixGroup

Attribute	RFC 2307	SFU 2.0/3.x
group name	cn	cn
group id number	gidnumber	gidNumber/ msSFU30GidNumber
group password	userpassword	-none-
group members	memberuid	posixmember, memberuid/ posixMember, sSFU30MemberUid

Agenda

- ✓ *Problem Description*
- ✓ *Authentication and Account Management history*
- ✓ *A solution and technologies*
- **Making it Work**
- The "big" picture
- Other technologies
- Limitations

Making it Work: Planning

Configuration Worksheet	
Kerberos REALM	HPATC2000.HP.COM
Directory administrator DN	cn=administrator, cn=users, dc=hpatc2000, dc=hp, dc=com
Directory/KDC server host	hpatcwin2k3.rose.hp.com
Directory server port	389
Configuration profile DN	cn=ldapux-profile, cn=configuration, dc=hpatc2000, dc=hp, dc=com
Base DN of user data	ou=unix, dc=hpatc2000, dc=hp, dc=com
Proxy user DN	cn=ldapux-proxy, cn=users, dc=hpatc2000, dc=hp, dc=com

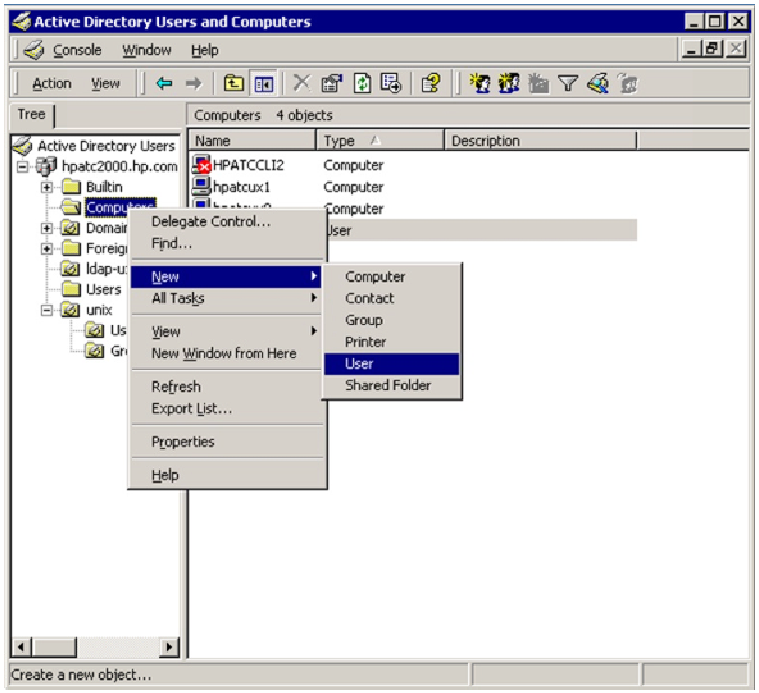
Making it Work: Active Directory

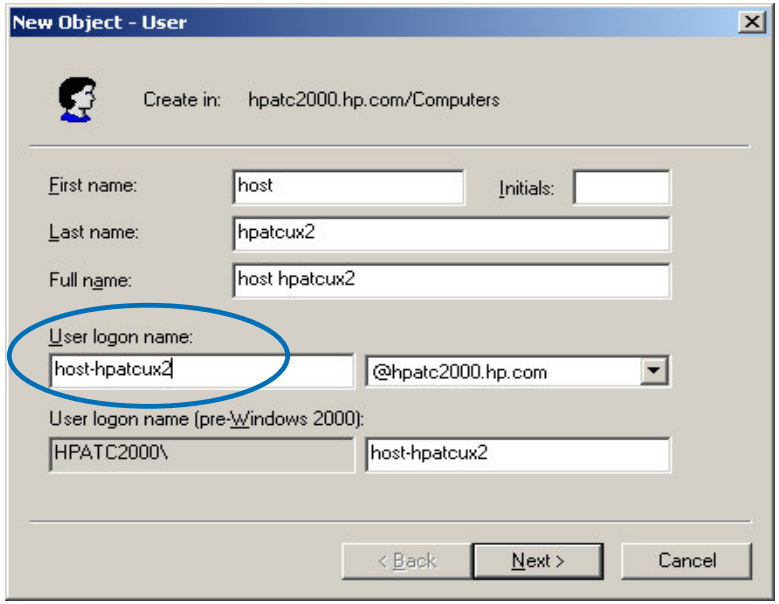
1. Install SFU
2. Create a "host" account for your HP-UX system *(Kerberos)*
3. Create an LDAP-UX proxy user *(LDAP-UX)*
4. Create a "Test User" *(Unix)*
5. Create keytab file for your HP-UX "host" account
6. Securely transfer the keytab file to your HP-UX host

Making it Work: Active Directory (2)

Create a HP-UX Host Account:

- Run Active Directory Users and Computers configuration tool
- Select the container to hold the user – Right Click – New – User

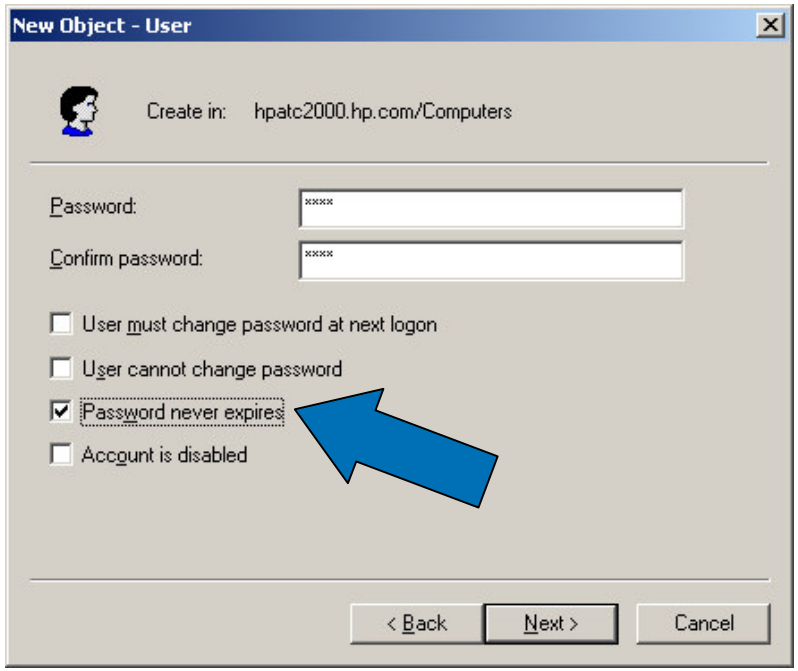


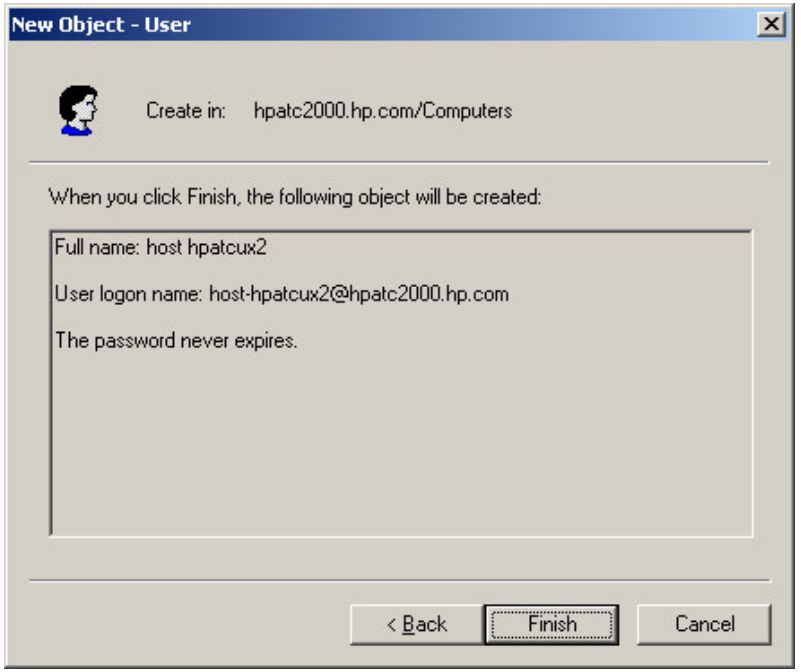


Making it Work: Active Directory (2)

Create a HP-UX Host Account:

- Run Active Directory Users and Computers configuration tool
- Select the container to hold the user – Right Click – Add – User

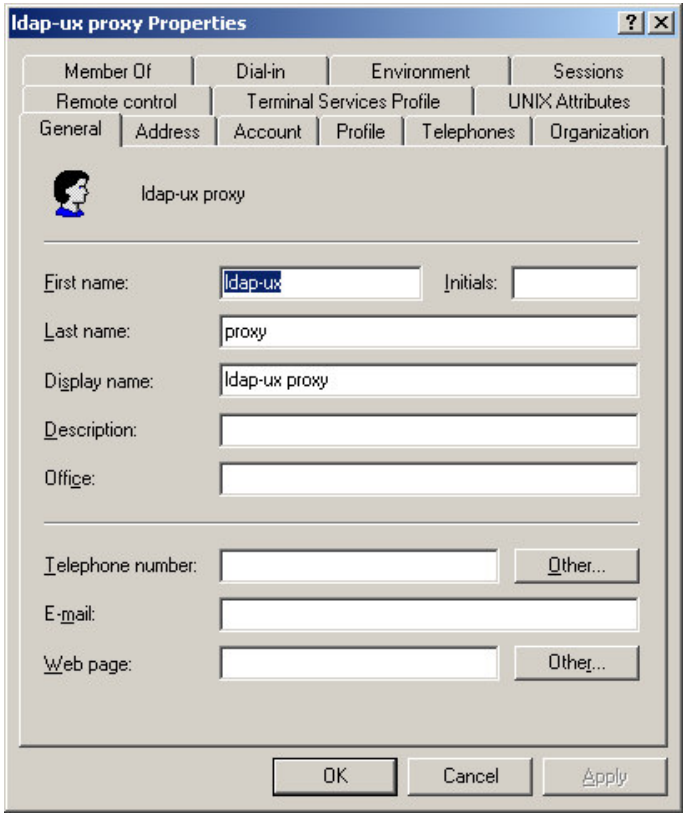




Making it Work: Active Directory (3)

Create an LDAP-UX Proxy User:

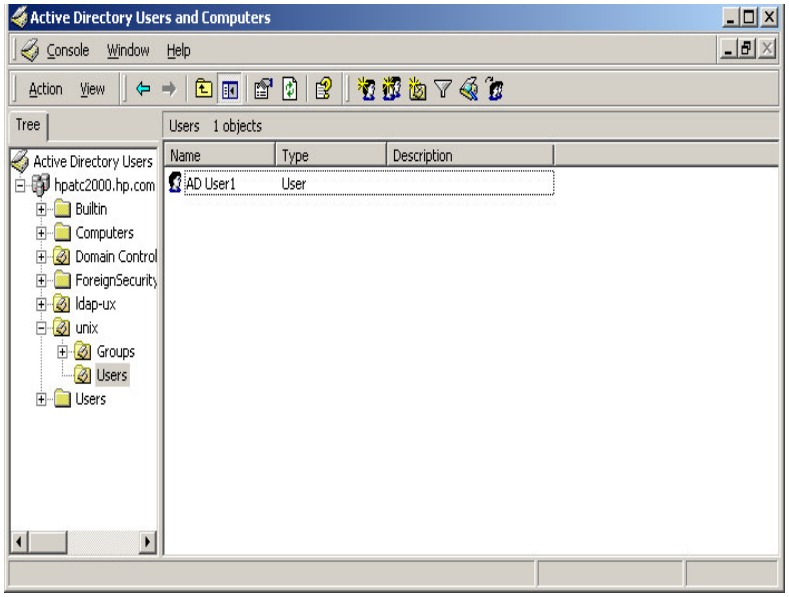
- Run Active Directory Users and Computers configuration tool
- Select the container to hold the user – Right Click – New – User

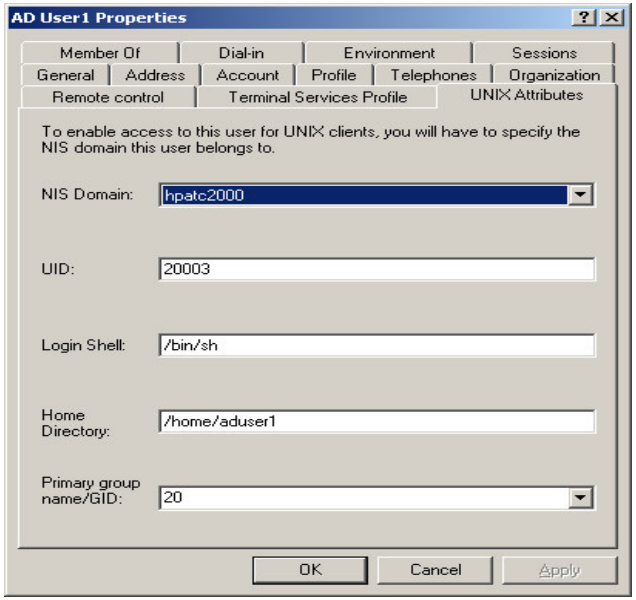


Making it Work: Active Directory (4)

Create a "Test" User:

- Run Active Directory Users and Computers configuration tool
- Select the container to hold the user – Right Click – New – User
- Add Posix/Unix Attributes after the user is created





Making it Work: Active Directory (5)

Create a Keytab File for the HP-UX Host Account:

- Use the Windows program ktpass (Installed with Support Tools)

C:\tmp\keytabs> ktpass
-princ **host/hpatcux2.rose.hp.com@HPATC2000.HP.COM**
-mapuser **host-hpatcux2** -pass ldap -crypto **DES-CBC-CRC**
-out hpatcux2.keytab

Successfully **mapped host/hpatcux2.rose.hp.com to host-hpatcux2**.
Key created.
Output keytab to hpatcux2.keytab:
Keytab version: 0x502
keysize 69 host/hpatcux2.rose.hp.com@HPATC2000.HP.COM ptype 1 (KRB5_NT_PRINCIPAL) vno 1 etype 0x1 (DES-CBC-CRC) keylength 8 (0x9e735e238f254931)

Account has been set for DES-only encryption.

Making it Work: HP-UX

1. Install LDAP-UX & PAM Kerberos
2. Configure pam_kerberos
   1. modify /etc/pam.conf
   2. create /etc/krb5.conf
   3. install keytab file (from Active Directory)
3. Configure LDAP-UX Client Services
   1. run setup program
   2. modify /etc/nsswitch.conf
4. Migrate users and groups into Active Directory

Making it Work: HP-UX (1)

Install LDAP-UX and Pam Kerberos:

```
# swlist -l product LdapUxClient PAM-Kerberos
# Initializing...
# Contacting target "hpatcux2"...
#
# Target:  hpatcux2:/
#

  LdapUxClient          B.03.30        LDAP-UX Client Services
  PAM-Kerberos          B.11.11.13     PAM-Kerberos Version 1.11
```

Making it Work: HP-UX (2)

Configure Kerberos:

```
# /opt/krbsup/bin/krb_config HPATC2000.HP.COM hp.com
   hpatcwin2k3.rose.hp.com

/etc/krb5.conf will have the following entries:
default realm = HPATC2000.HP.COM
default domain = hp.com
 kdc host = hpatcwin2k3.rose.hp.com

/etc/krb5.conf already exists.
Proceed (move to krb5.conf.bak)? [y/n] y
...
Run krbval -v to find out which entry your programs will use.
#
```

Making it Work: HP-UX (2)

```
# cat /etc/krb5.conf
[libdefaults]
        default_realm = HPATC2000.HP.COM
        kdc_req_checksum_type = 2
        default_keytab_name = /etc/krb5.keytab
        default_etypes = des-cbc-crc
        default_etypes_des = des-cbc-crc
        default_tkt_enctypes = des-cbc-crc
        default_tgs_enctypes = des-cbc-crc

[realms]
        HPATC2000.HP.COM = {
                kdc = hpatcwin2k3.rose.hp.com
                kpasswd_server = hpatcwin2k3.rose.hp.com:464
                default_domain = hp.com
        }

[domain_realm]
        .hp.com = HPATC2000.HP.COM
```

Making it Work: HP-UX (2)

Kerberos Client Validation

```
# cp /etc/pam.krb5 /etc/pam.conf
# pamkrbval
Validating the pam configuration files
--------- --- --- ----------- -----
Validating the /etc/pam.conf file
[PASS] : The validation of config file: /etc/pam.conf passed
[NOTICE] : The validation of config file: /etc/pam_user.conf is not done
          as libpam_updbe library is not configured

Validating the kerberos config file
--------- --- ------- ----- -----
[PASS] : Initialization of kerberos passed

Connecting to default Realm
 --------- - ------ ----
[PASS] : Default Realm is issuing tickets

 Validating the keytab entry for the host service principal
 --------- --- ----- ----- --- -- ---- ------ --------
[PASS] : The keytab validation is successful
```

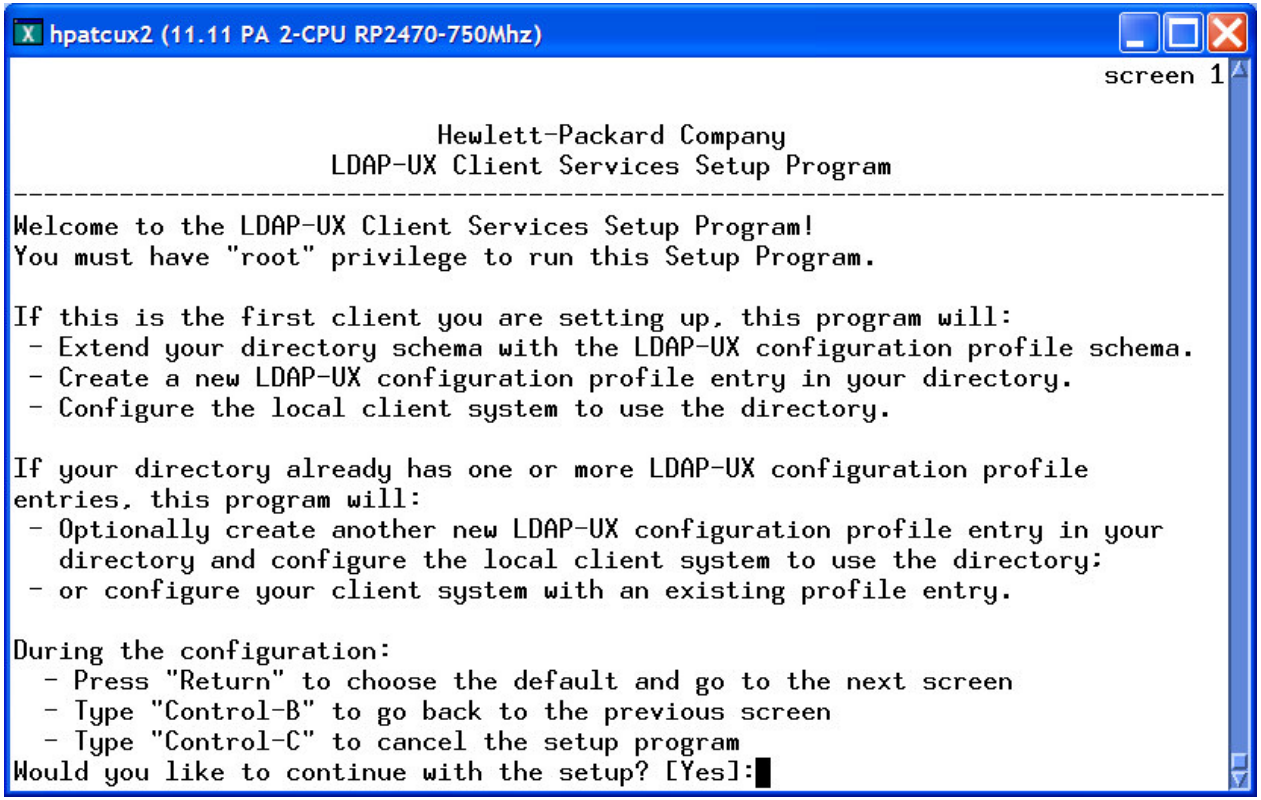
Making it Work: HP-UX (2)

Kerberos Authentication Validation

```
# kinit aduser1
Password for aduser1@HPATC2000.HP.COM:

# klist
Ticket cache: /tmp/krb5cc_0
Default principal: aduser1@HPATC2000.HP.COM

Valid starting       Expires              Service principal
07/09/04 13:19:18  07/09/04 23:18:21
                                  krbtgt/HPATC2000.HP.COM@HPATC2000.HP.COM
```

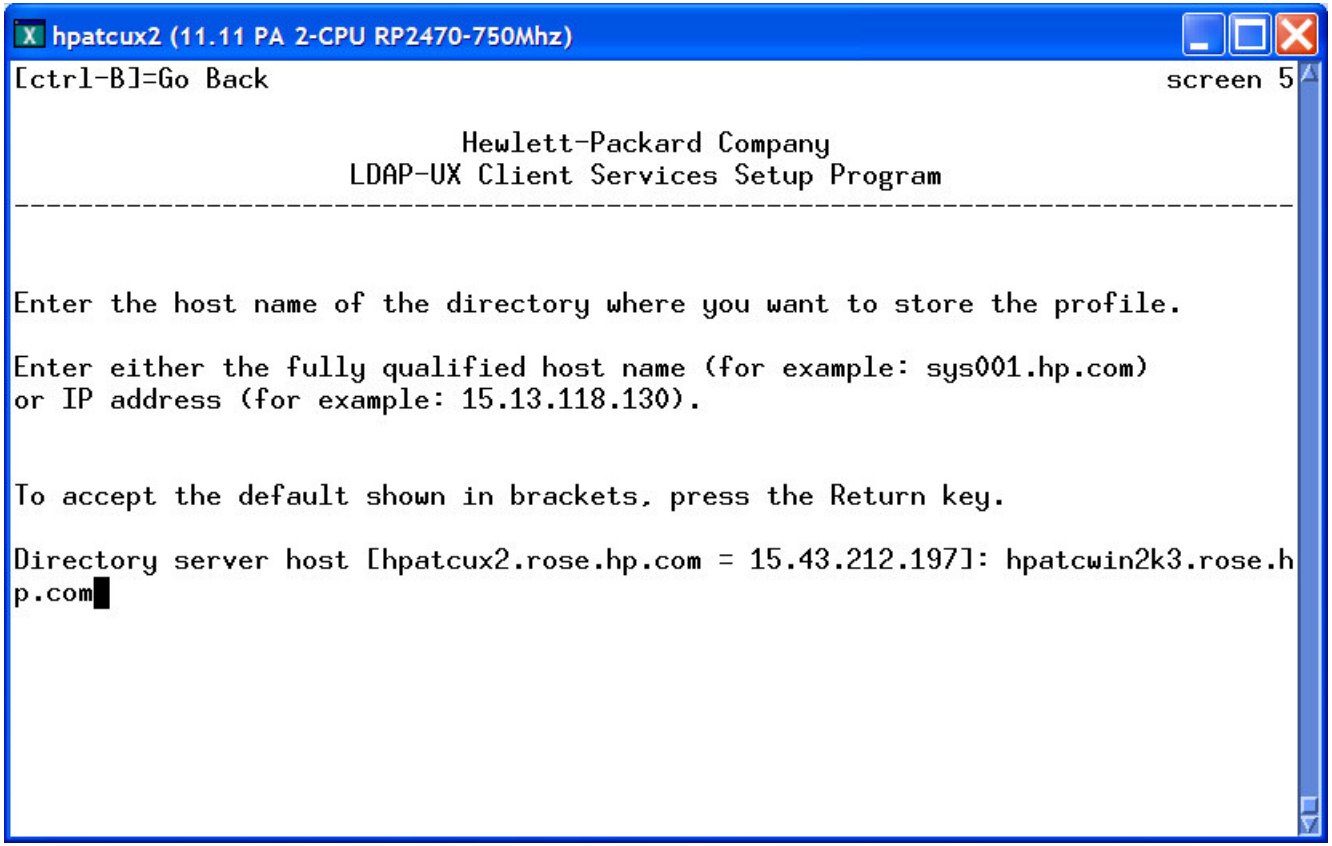
Making it Work: HP-UX (3)

LDAP-UX Configuration: /opt/ldapux/config/setup



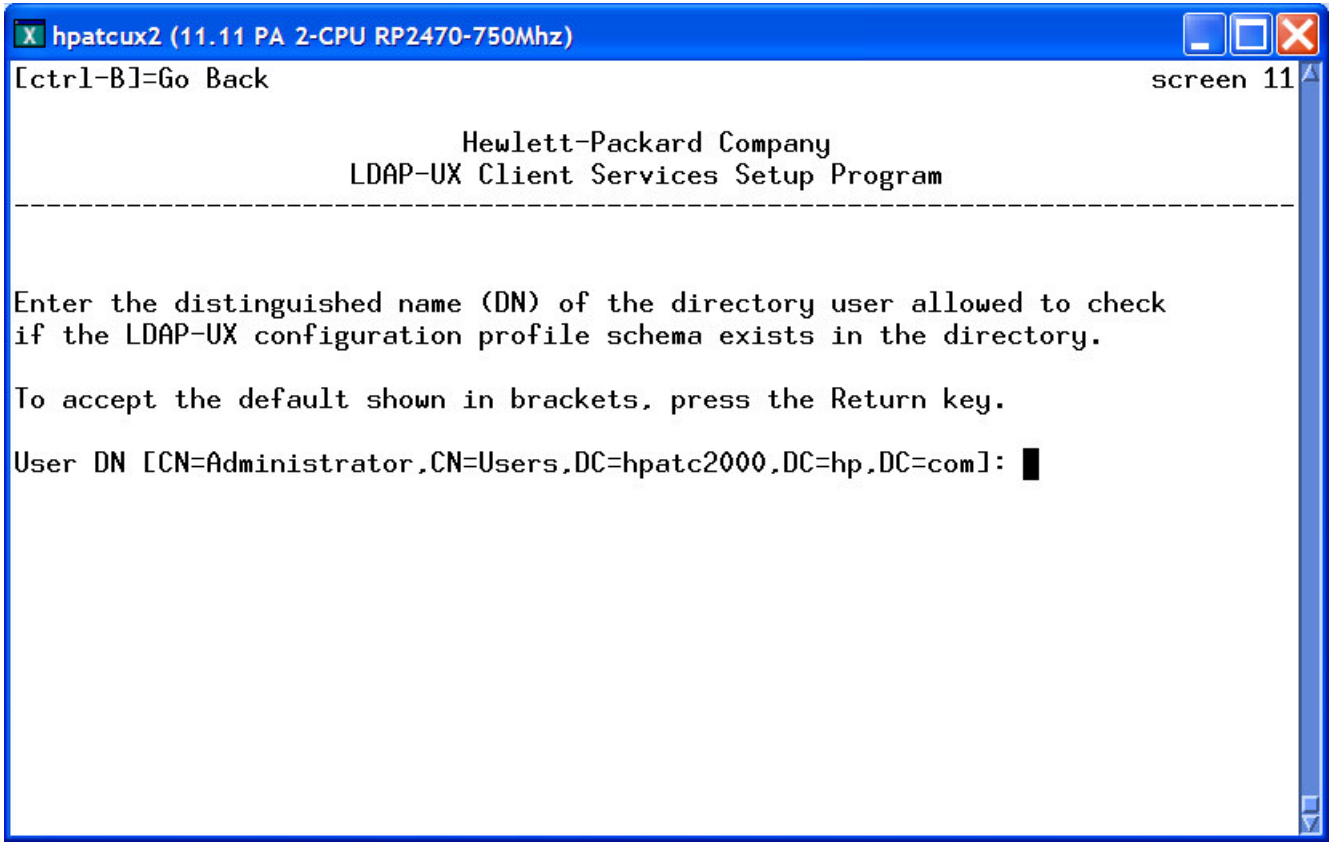
Making it Work: HP-UX (3)

LDAP-UX Configuration: Directory Server



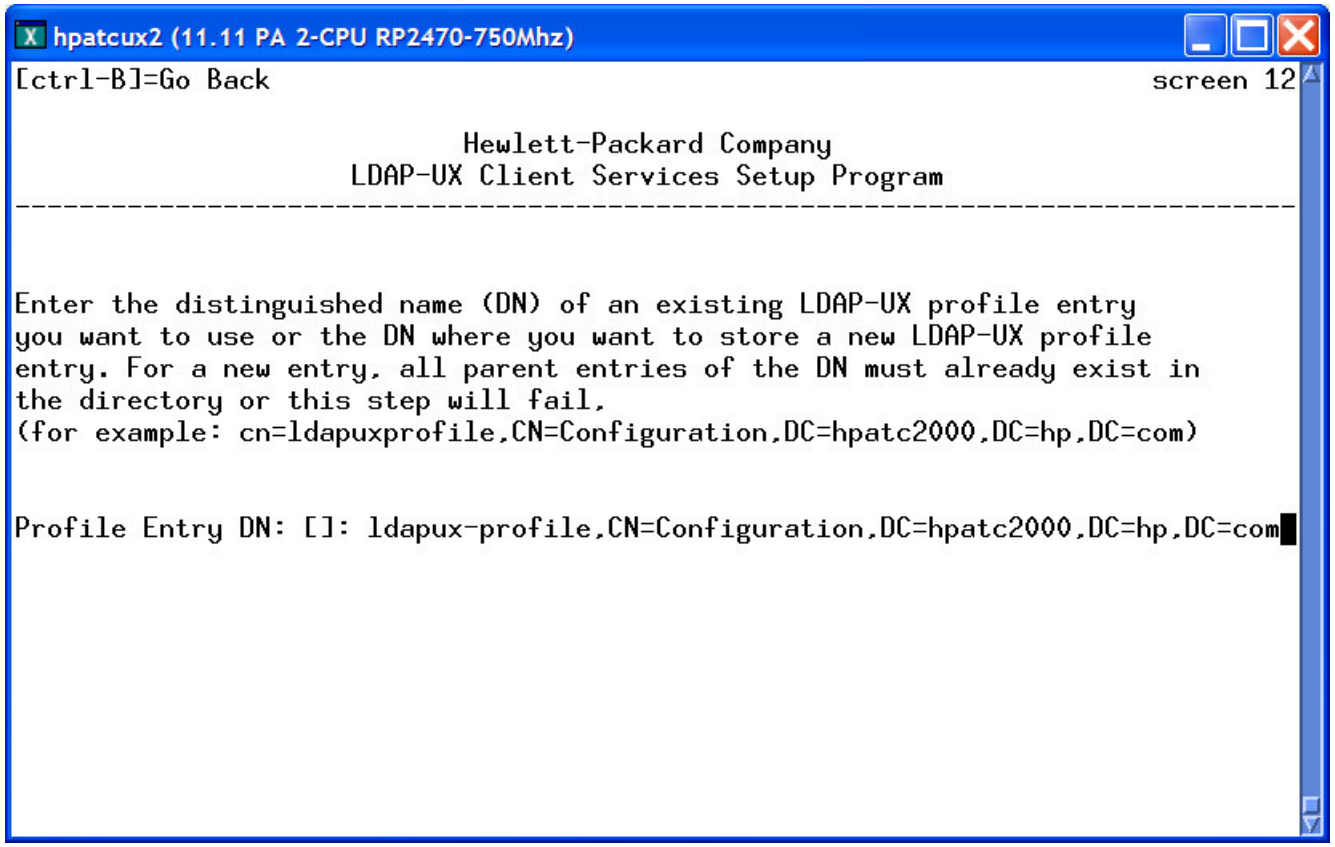
Making it Work: HP-UX (3)

LDAP-UX Configuration: Schema Update



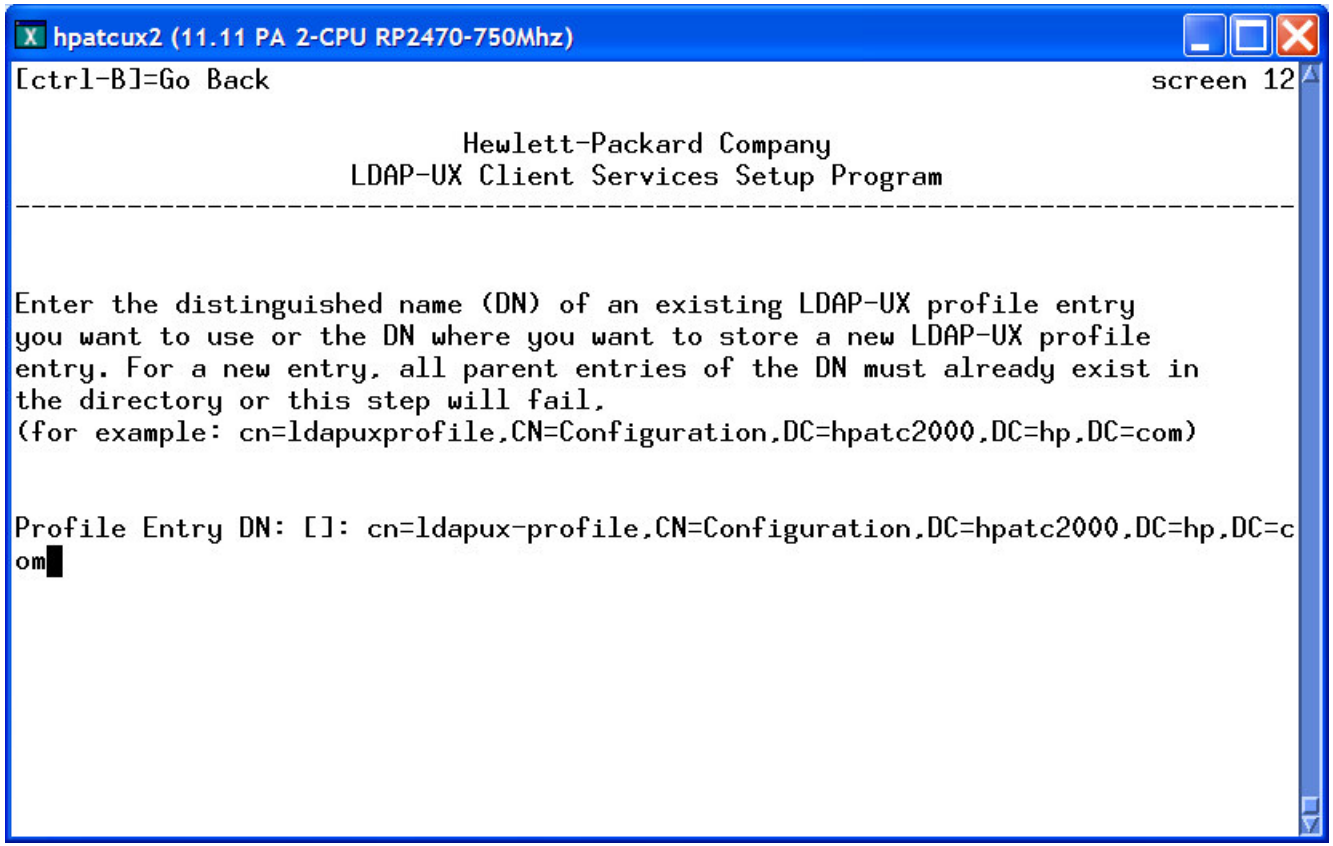
Making it Work: HP-UX (3)

LDAP-UX Configuration: Profile Configuration



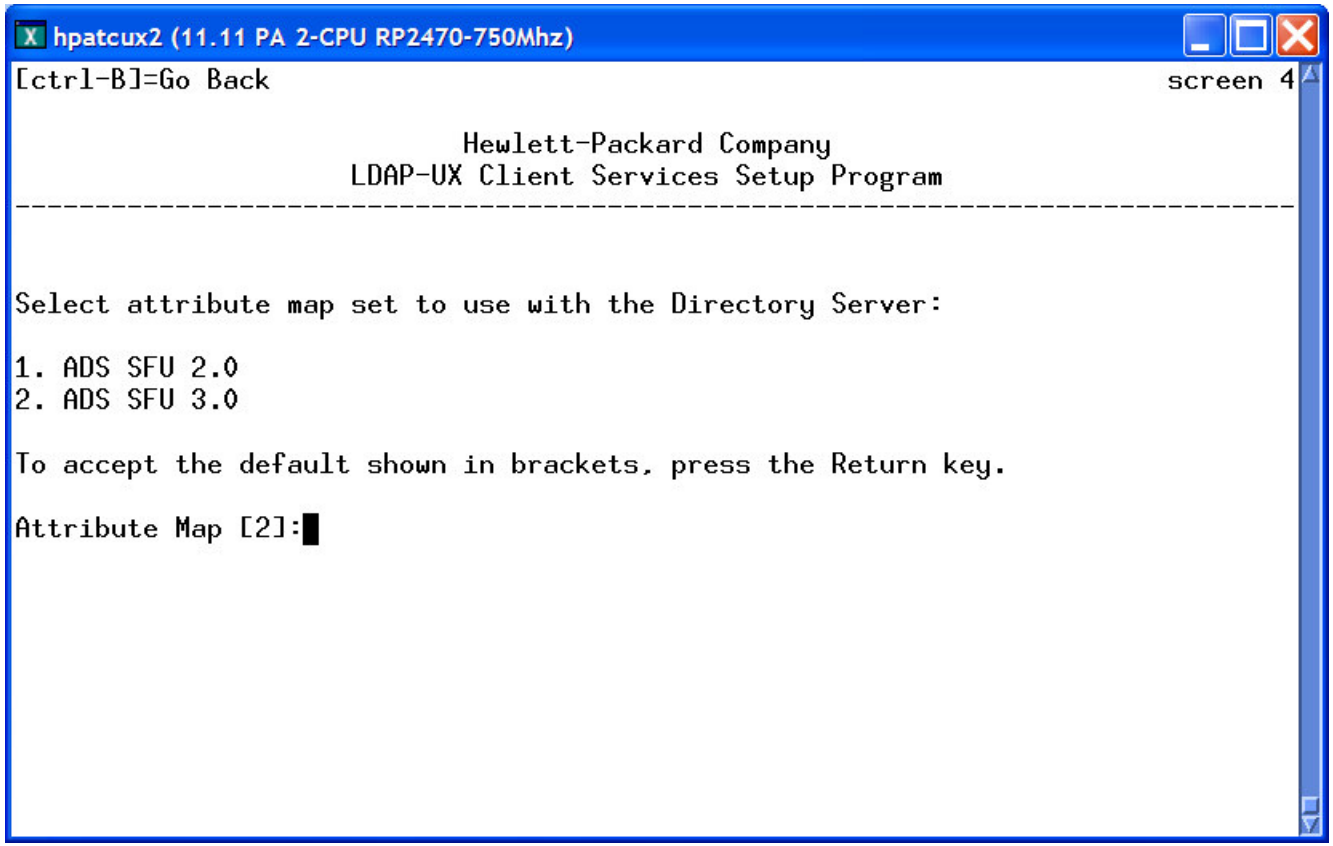
Making it Work: HP-UX (3)

LDAP-UX Configuration: Profile Configuration



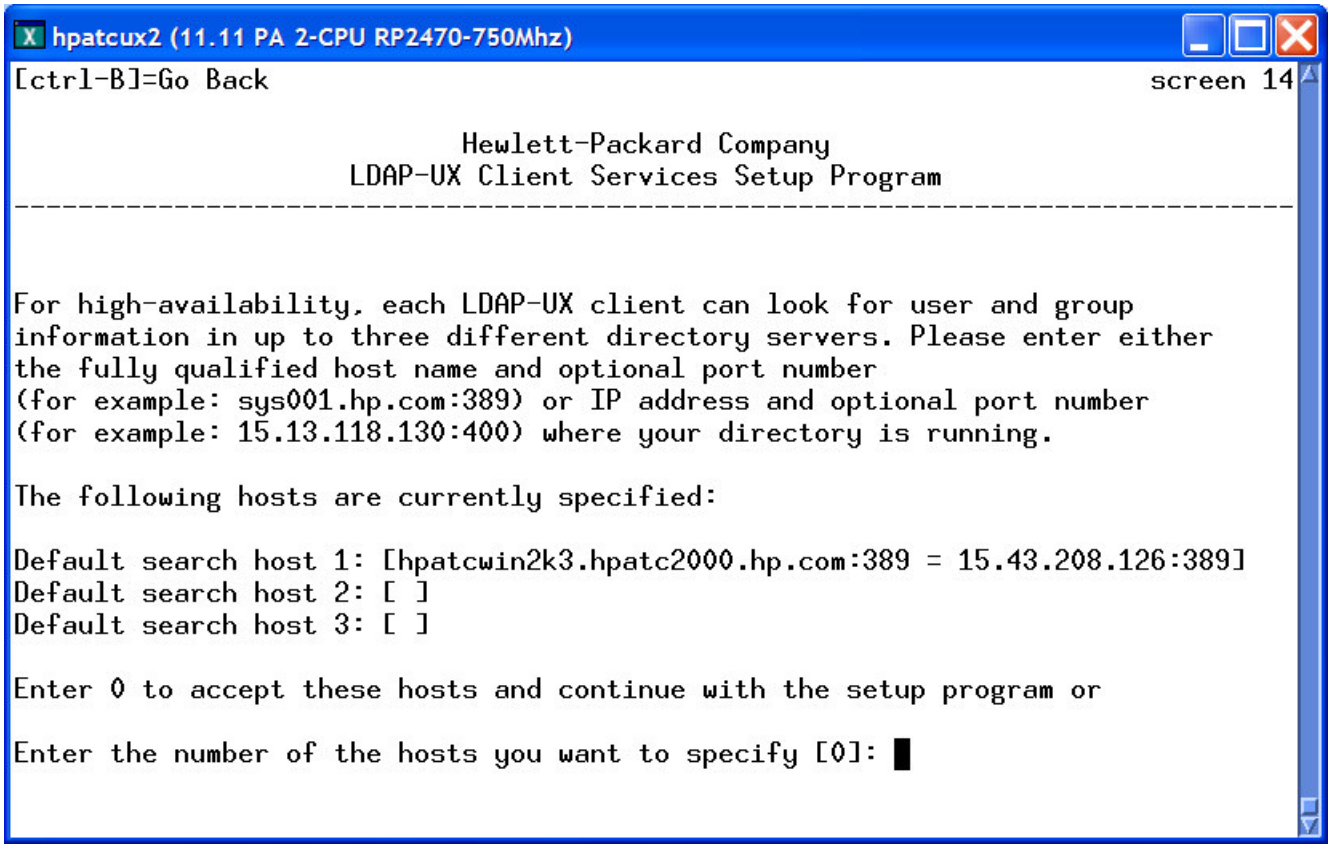
Making it Work: HP-UX (3)

LDAP-UX Configuration: SFU Mapping



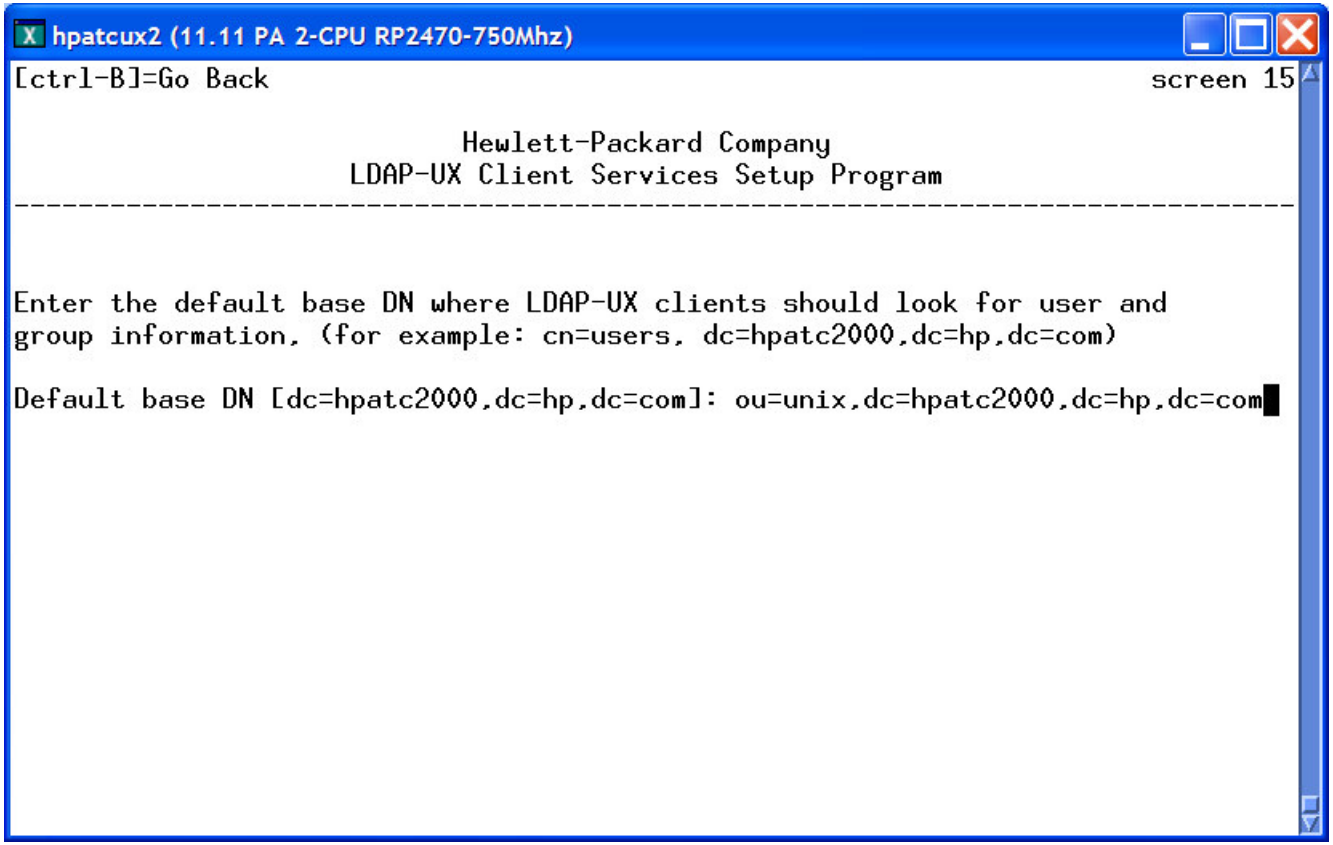
Making it Work: HP-UX (3)

LDAP-UX Configuration: Additional Directory Servers



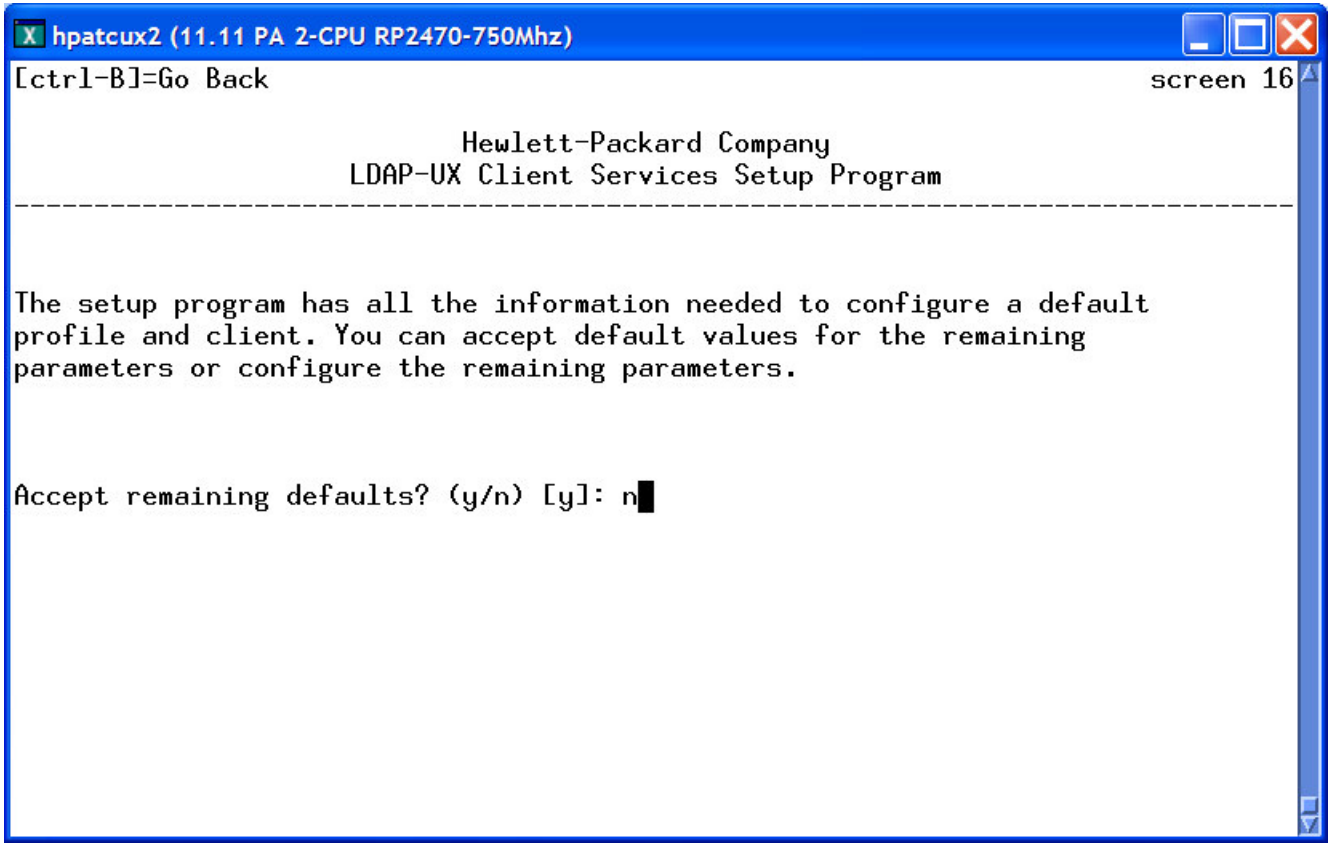
Making it Work: HP-UX (3)

LDAP-UX Configuration: Default Search Base



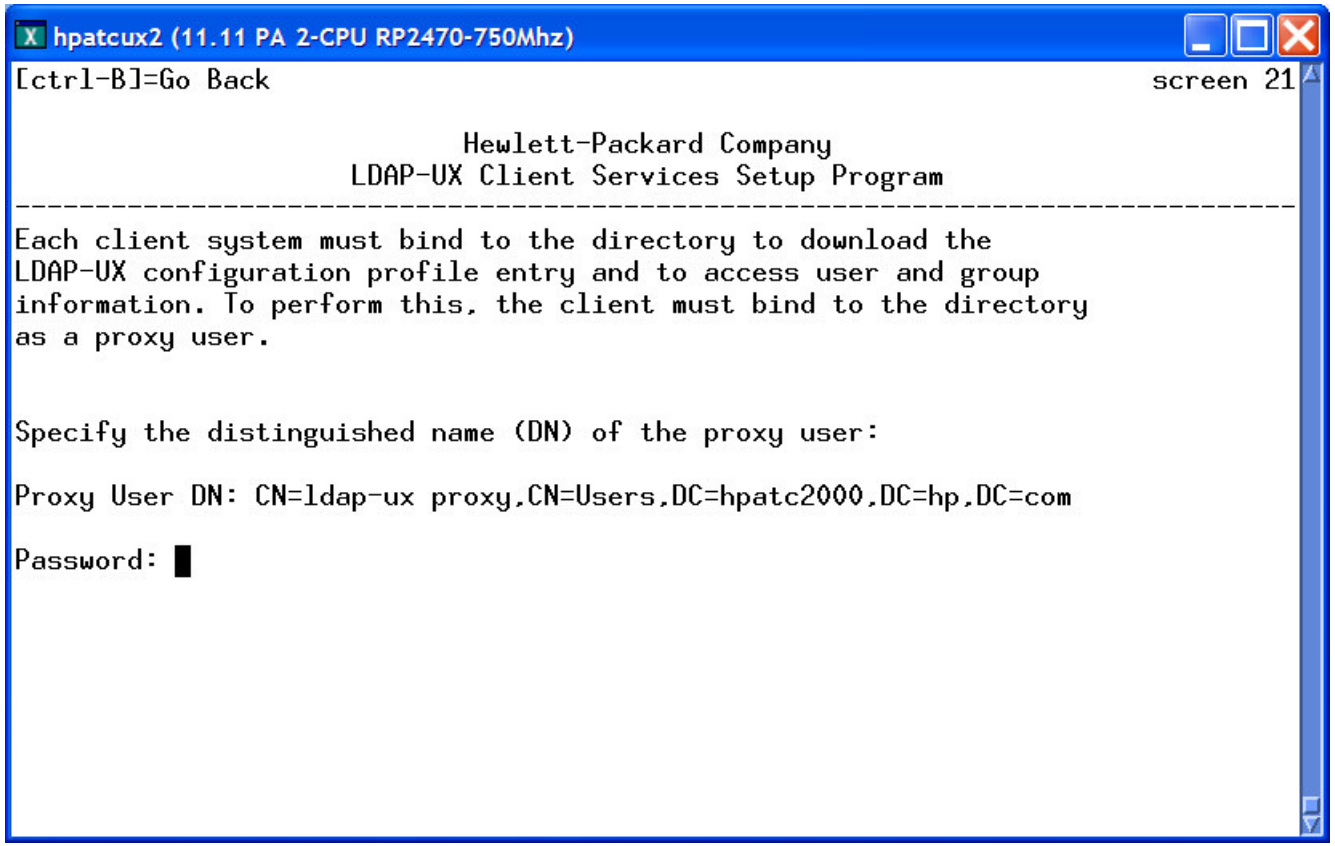
Making it Work: HP-UX (3)

LDAP-UX Configuration: More configuration options



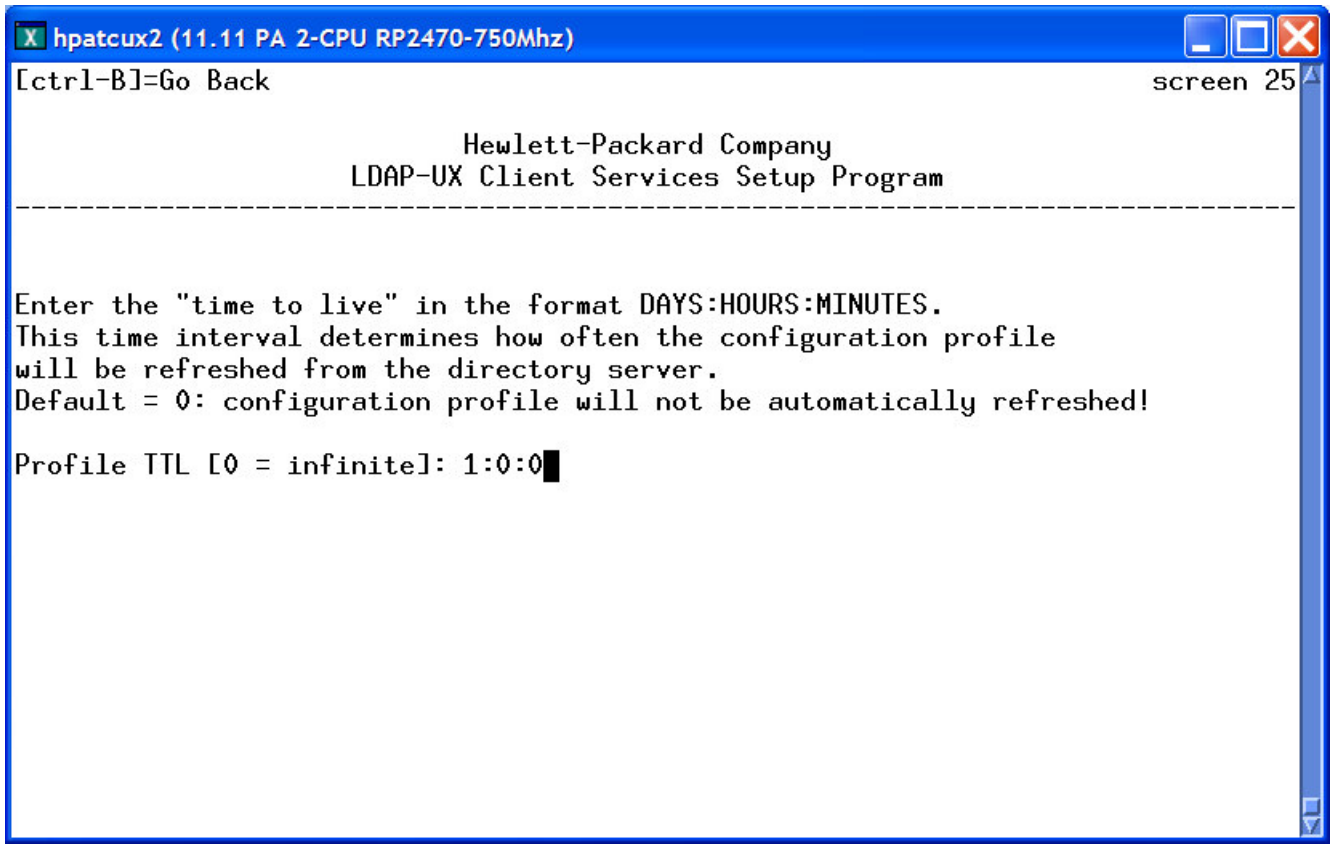
Making it Work: HP-UX (3)

LDAP-UX Configuration: Proxy User



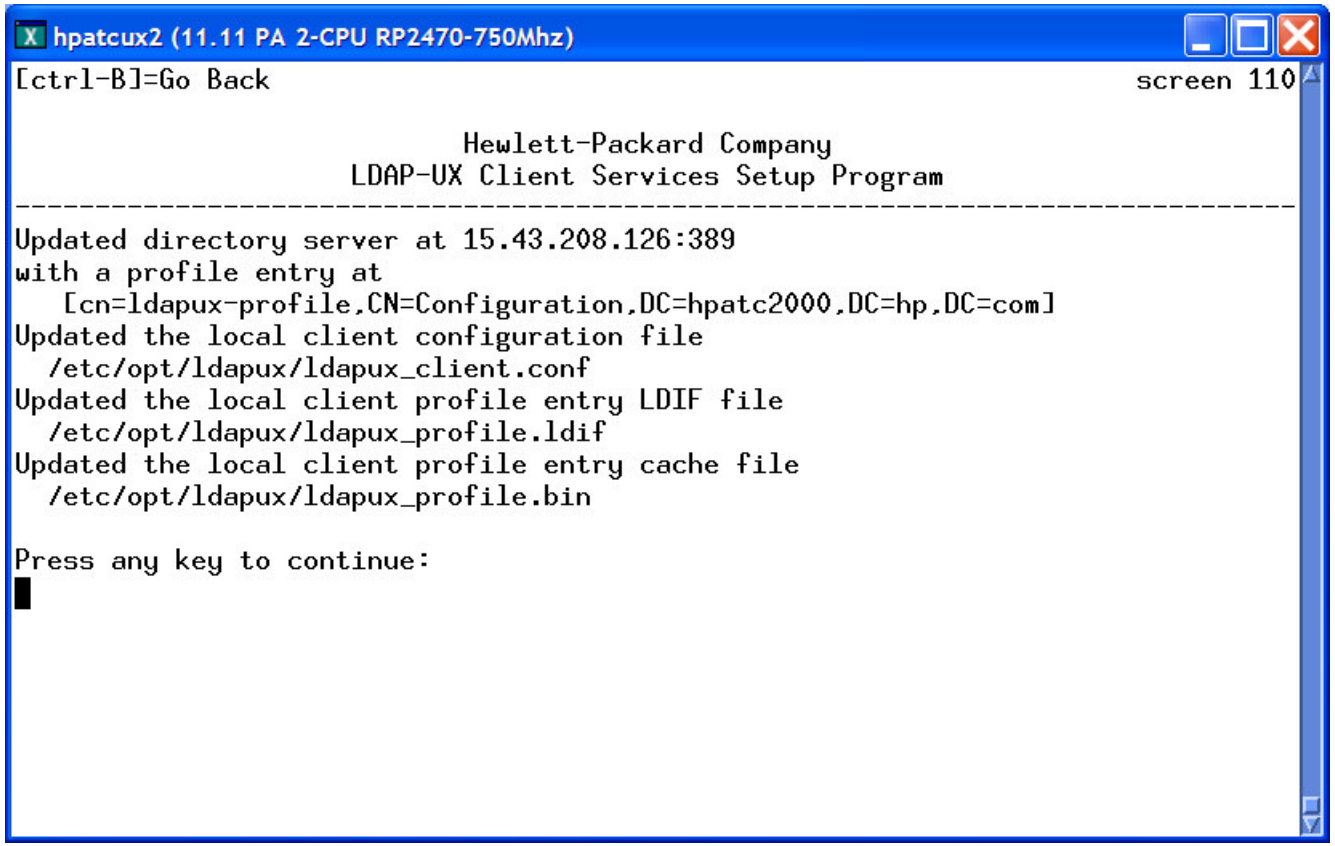
Making it Work: HP-UX (3)

LDAP-UX Configuration: Profile Refresh (TTL)



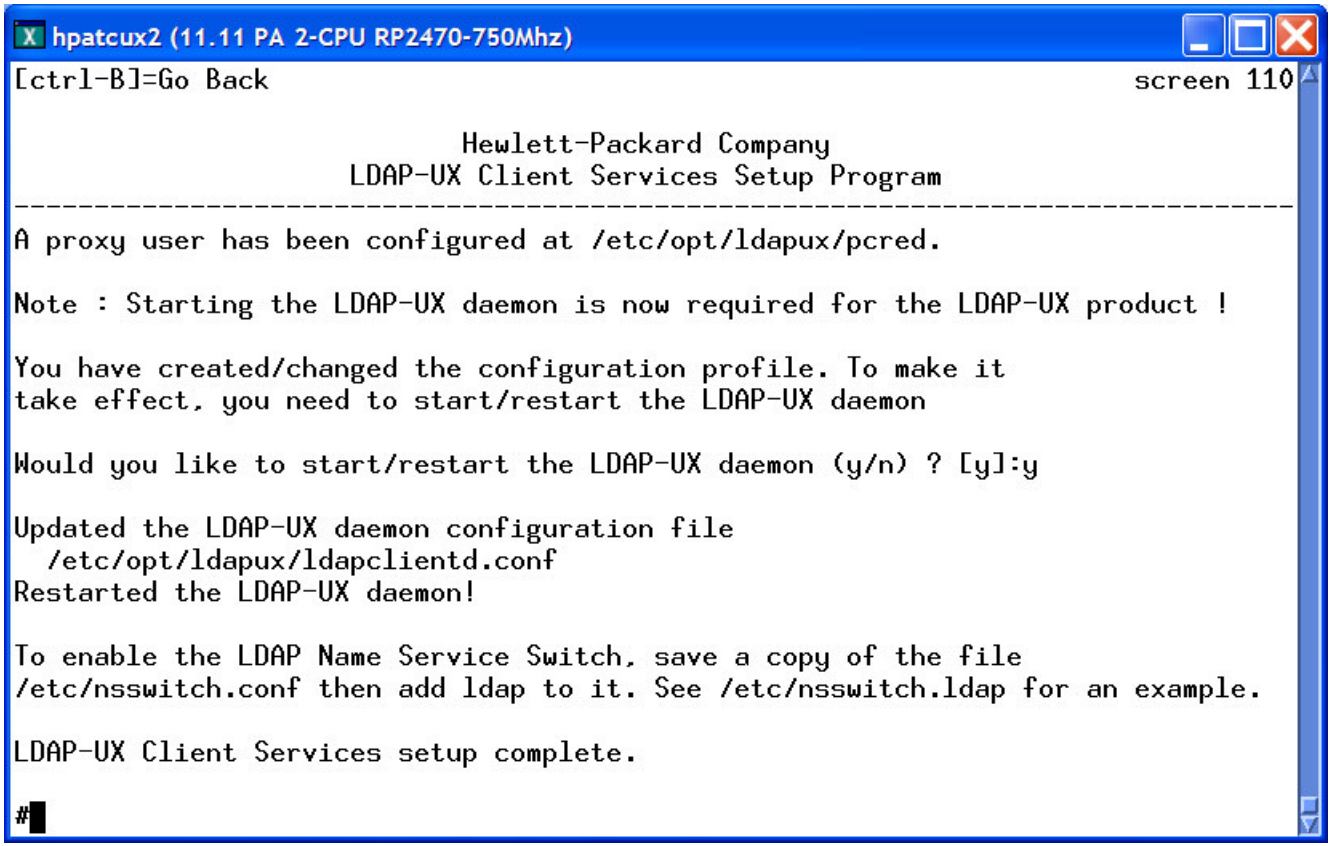
Making it Work: HP-UX (3)

LDAP-UX Configuration: Create the Profile



Making it Work: HP-UX (3)

LDAP-UX Configuration: Restart the LDAP-UX client



Making it Work: HP-UX (3)

LDAP-UX: Configuration Verification

LDAP Client (ldapclientd) Credentials:

/opt/ldapux/config/ldap_proxy_config -v
File Credentials verified – valid

LDAP-UX Configuration Profile:

/opt/ldapux/config/display_profile_cache

LDAP-UX Client Services

Global Information from the Configuration Profile
==========
host[:port]: 15.43.208.126:389
default search base: ou=unix,dc=hpatc2000,dc=hp,dc=com
…

Making it Work: HP-UX (3)

LDAP-UX: Proxy User Permission Verification *(Assuming SFU 3.0)*

```
#  ldapsearch -h hpatcwin2k3.rose.hp.com  -b ou=unix,dc=hpatc2000,dc=hp,dc=com \
                -w PwD -D "cn=ldap-ux proxy, cn=users, dc=hpatc2000, dc=hp, dc=com" \
                msSFU30Name=aduser1 \
                msSFU30GidNumber msSFU30HomeDirectory msSFU30LoginShell
                msSFU30Name msSFU30UidNumber

version: 1

dn: CN=AD User1,OU=Users,OU=unix,DC=hpatc2000,DC=hp,DC=com

msSFU30GidNumber: 20

msSFU30HomeDirectory: /home/aduser1

msSFU30LoginShell: /bin/sh

msSFU30Name: aduser1

msSFU30UidNumber: 20003
```

Making it Work: HP-UX (3)

LDAP-UX: Lookup Verification

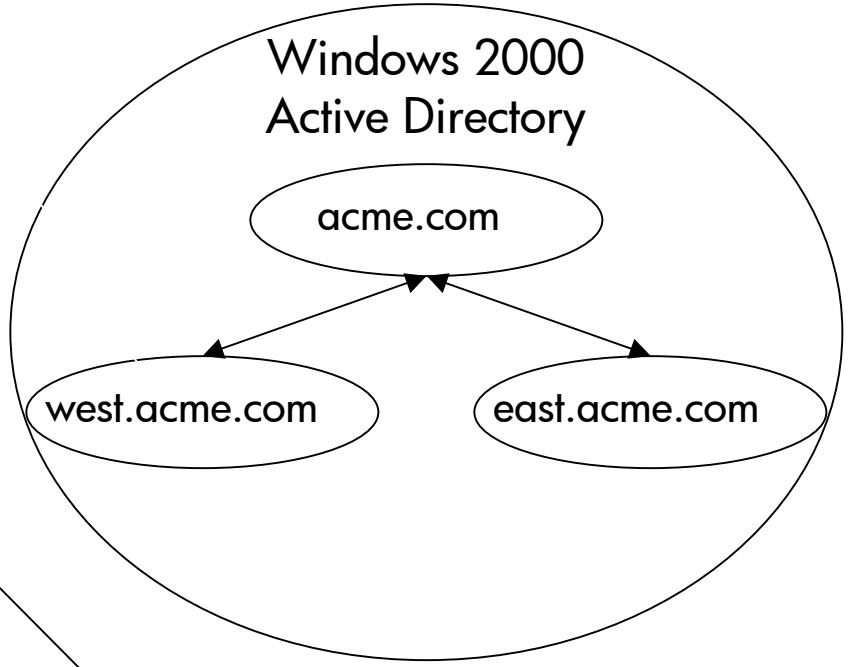
```
# /usr/contrib/bin/nsquery passwd aduser1 ldap

Using "ldap" for the passwd policy.

Searching ldap for aduser1
User name: aduser1
User Id: 20003
Group Id: 20
Gecos:
Home Directory: /home/aduser1
Shell: /bin/sh

Switch configuration: Terminates Search
#
```

Agenda

- ✓ *Problem description*
- ✓ *Authentication and account management history*
- ✓ *A solution and technologies*
- ✓ *Making it Work*
- ❑ **The "big" picture**
- ❑ Other technologies
- ❑ Limitations

The Big (simple) Picture

user: Joe Cool
name: jcool
uid: 201
gid: 20
home: /home/jcool
sid: 123404039393
...



HP-UX workstation/server

Windows Client

Agenda

- ✓ *Problem description*
- ✓ *Authentication and account management history*
- ✓ *A solution and technologies*
- ✓ *Making it Work*
- ✓ The "big" picture
- ❑ **Other technologies**
- ❑ Limitations

Other Technologies

- Secure Internet Services
- HP CIFS Server
- SAP
- Apache Webserver
- Any "pamized" application

Secure Internet Services (SIS)

- A set of traditional *"kerberized"* internet services (both clients and servers) that include ftp, telnet & rlogin
- Allows secure authentication to remote services without requiring passwords be sent across the network
- Comes disabled on the system by default
- Both clients and servers can use non-Kerberized protocol
- Best used in conjunction with pam_kerberos **not** instead of
- Shares most configuration with pam_kerberos/Kerberos client

HP CIFS Server (Samba)

- Allows windows clients to “mount” network drives from an HP-UX server
- Because Unix accounts are stored in the active directory there is no need to map a Unix account to a windows account *(user map file)*
- Authentication is done using Kerberos
- More information Session #3112 (yesterday): HP CIFS Server with Samba 3.0 and Windows Server 2003

SAP

- Allows automatic/transparent authentication of the SAP GUI (Windows) to the SAP Server (HP-UX)
- Kerberos (GSSAPI) is used by the SAP Application Server to authenticate the client connecting
- No passwords sent across the network
- http://www.sap-ag.de/partners/icc/scenarios/technology/bc-snc.asp

Apache Webserver

- LDAP
  - **auth_ldap** module allows Apache to authenticate users against the Active Directory over LDAP (user/password supplied by client)
  - Configuration is flexible enough to map user name attributes
- GSSAPI/SPNEGO
  - Browser (Mozilla, IE) obtains a Kerberos Ticket for the webserver to authenticate the client
  - Provides Single Sign On to Web based applications

PAM'ized Applications

- Any application that writes to the PAM API can authenticate users against active directory

Agenda

- ✓ *Problem description*
- ✓ *Authentication and account management history*
- ✓ *A solution and technologies*
- ✓ *Making it Work*
- ✓ The "big" picture
- ✓ Other technologies
- ❑ **Limitations**

Limitations

- No support for nested groups
- Windows 2003 not currently supported
- Netgroup maps are not supported with SFU

HP Documentation

- Add links to software.hp.com

LDAP RFC's

- 2307 An Approach for Using LDAP as a Network Information Service
- 2251 Lightweight Directory Access Protocol (v3)
- 2252 LDAP(v3) Attribute Syntax Definitions
- 2253 LDAP(v3) UTF-8 Rep of Distinguished Names
- 2254 String Representation of LDAP Search Filters
- 2255 The LDAP URL Format
- 2256 Sum. of X.500(96) User Schema for LDAPv3
- 2829 Authentication Methods for LDAP
- 2830 Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security
- INTERNET-DRAFT LDAP-BIS

Kerberos RFC's

- 1510 The Kerberos Network Authentication Service (V5)
- 1964 The Kerberos Version 5 GSS-API Mechanism
- 2623 NFS Version 2 and Version 3 Security Issues and the NFS Protocol's Use of RPCSEC_GSS and Kerberos V5
- 2942 Telnet Authentication: Kerberos Version 5

PAM RFC

- OSF-RFC 86.0

HP WORLD 2004

Solutions and Technology Conference & Expo

Co-produced by:

interex

shared knowledge • shared power

encompass

AN HP USER GROUP

RECOMMENDED TRAINING VENUE FOR THE

HP Certified Professional