Directory Services

A thorough analysis

Overview

- *Welcome*
- *Basic Concepts*
- *What is a directory service?*
- *Open Directory and Mac OS X*
- *Open Directory in Mac OS X Server*

Overview (cont'd)

- *Identification and Authorization in Mac OS X Server*
- *Authentication in Mac OS X Server*
- *Replication in Mac OS X Server*
- *Mac OS X and Active Directory*
- *Providing directory services to windows and unix clients*

Concepts

- *Basic knowledge that will help us later*
  - *Encryption: symmetric vs. asymmetric*
  - *Unix Architecture*

Directory Services Basics

or, what on earth are we doing here?

Directory Services Basics

- *What is a directory?*
- *What is a directory service?*

Big Words, simple concepts

- *Access to resources in a multi-user OS depends on 3 distinct but independent concepts*
  - *identification*
  - *authorization*
  - *authentication*

logging in: one analysis

- *Credentials are presented*
- *?? Magic*
- *User is logged in*

logging in: a better analysis

- *Credentials are presented*
- *User account is located (user is identified)*
  - *This could include determining how the user should be authenticated*
- *User is authenticated*
- *Authorization is determined*

The Air Port Example

- *The concepts of identification, authorization and identification don’t just apply to information technology.*
- *The Air Port is a good example*
  - *What is being protected*
  - *Where do identification, authorization and authentication take place?*
  - *Where does the analogy fall down?*

A little history (/etc files, identification, authorization and authentication)

- *Account and password hash in /etc/passwd*
- *Groups in /etc/group*
- *Password Shadowing*
- *limitations of /etc approach*

Where do directories fit?

- *Directories are generally useful data stores; we're looking at them in a fairly specific role*
- *Basically a replacement for /etc/passwd*
    - *albeit with added functionality*

Open Directory and Mac OS X

Mac OS X’s client-side Directory Services architecure

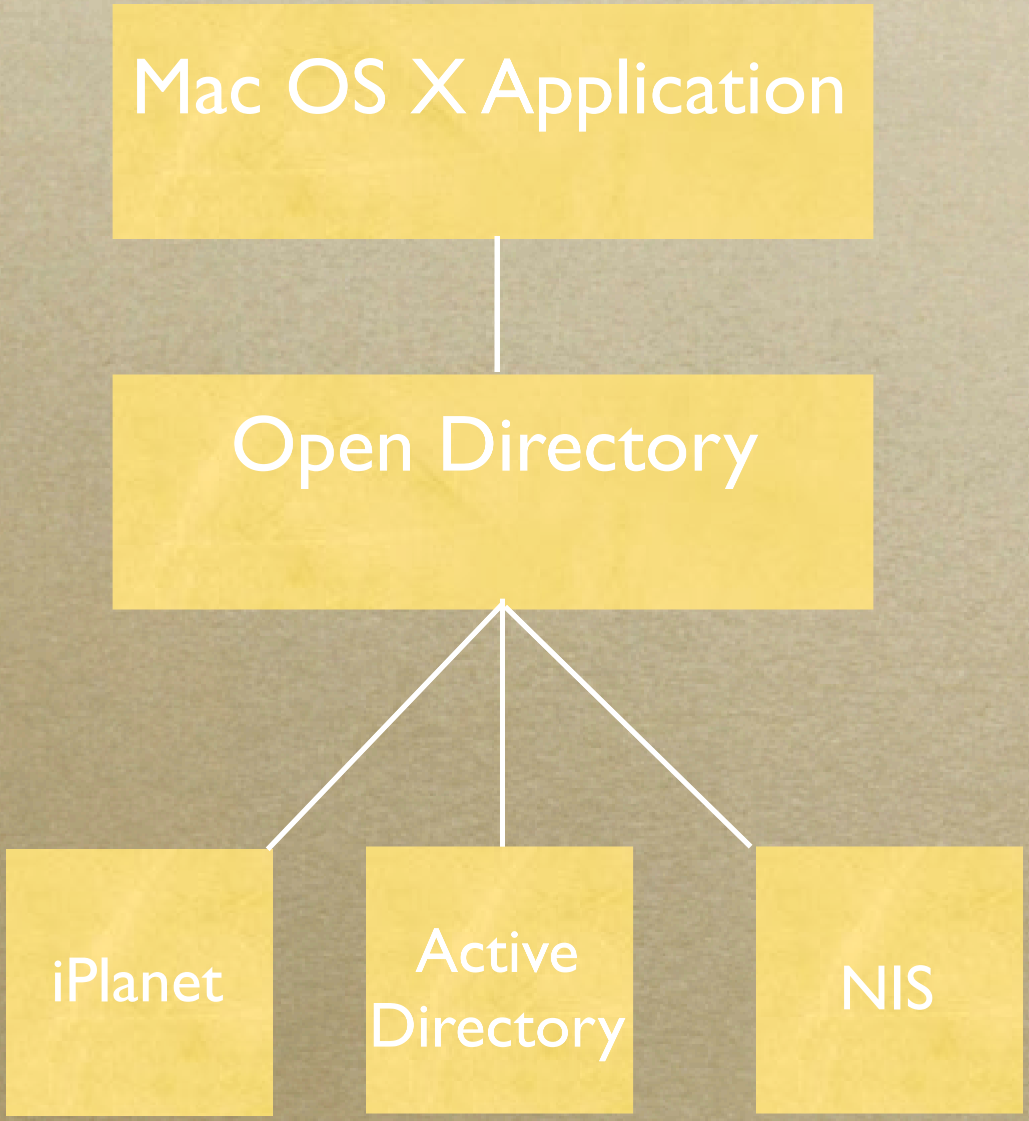
Overview

- *A note on naming*
- *Open Directory architecture*
- *component Open Directory processes*

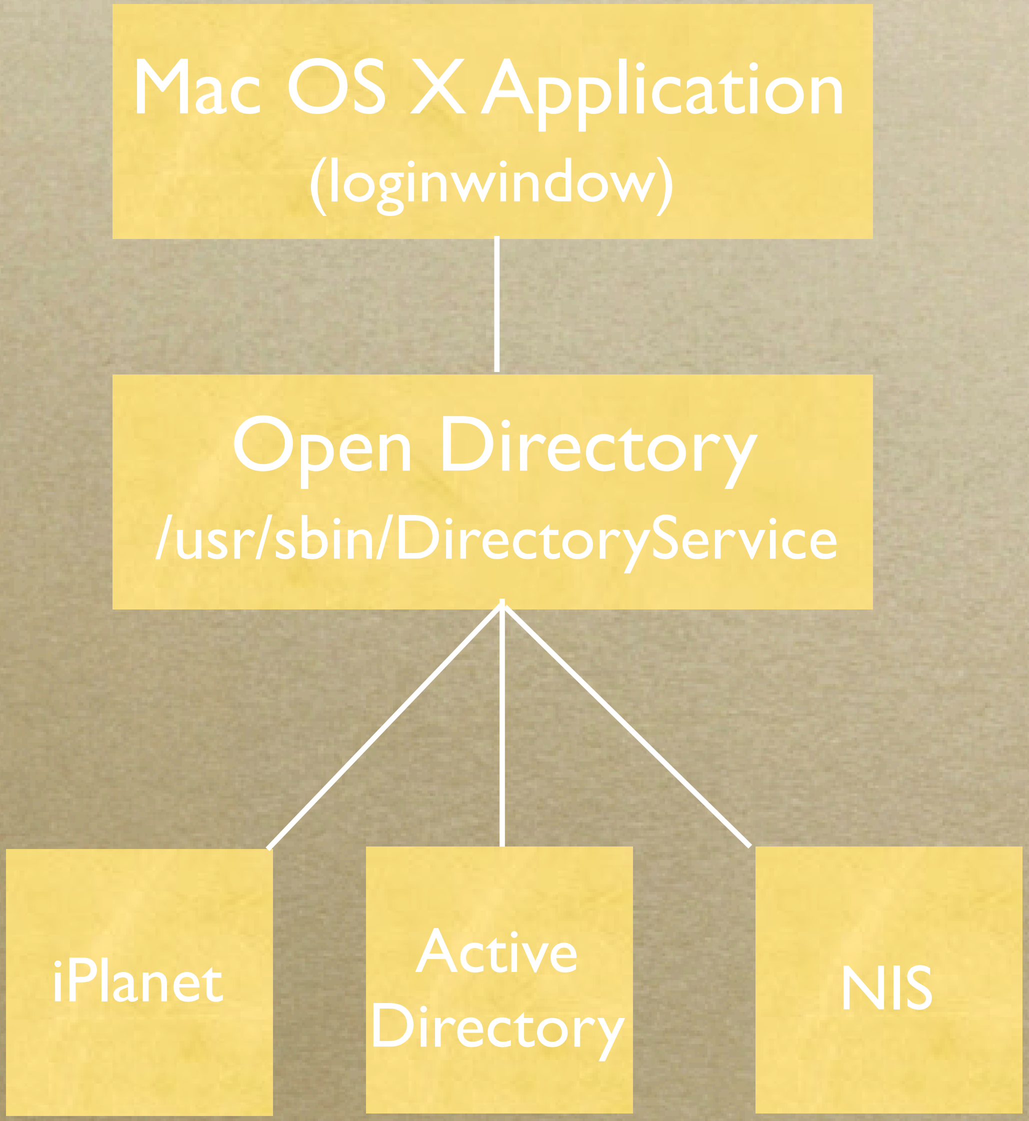
What is Open Directory?

- *OS-wide component providing identification and (in some cases) authorization and authentication*
- *Mac OS X’s rosetta stone*

Architecture: the general view



Architecture: more in-depth



Mac OS X Application

(loginwindow)

Open Directory

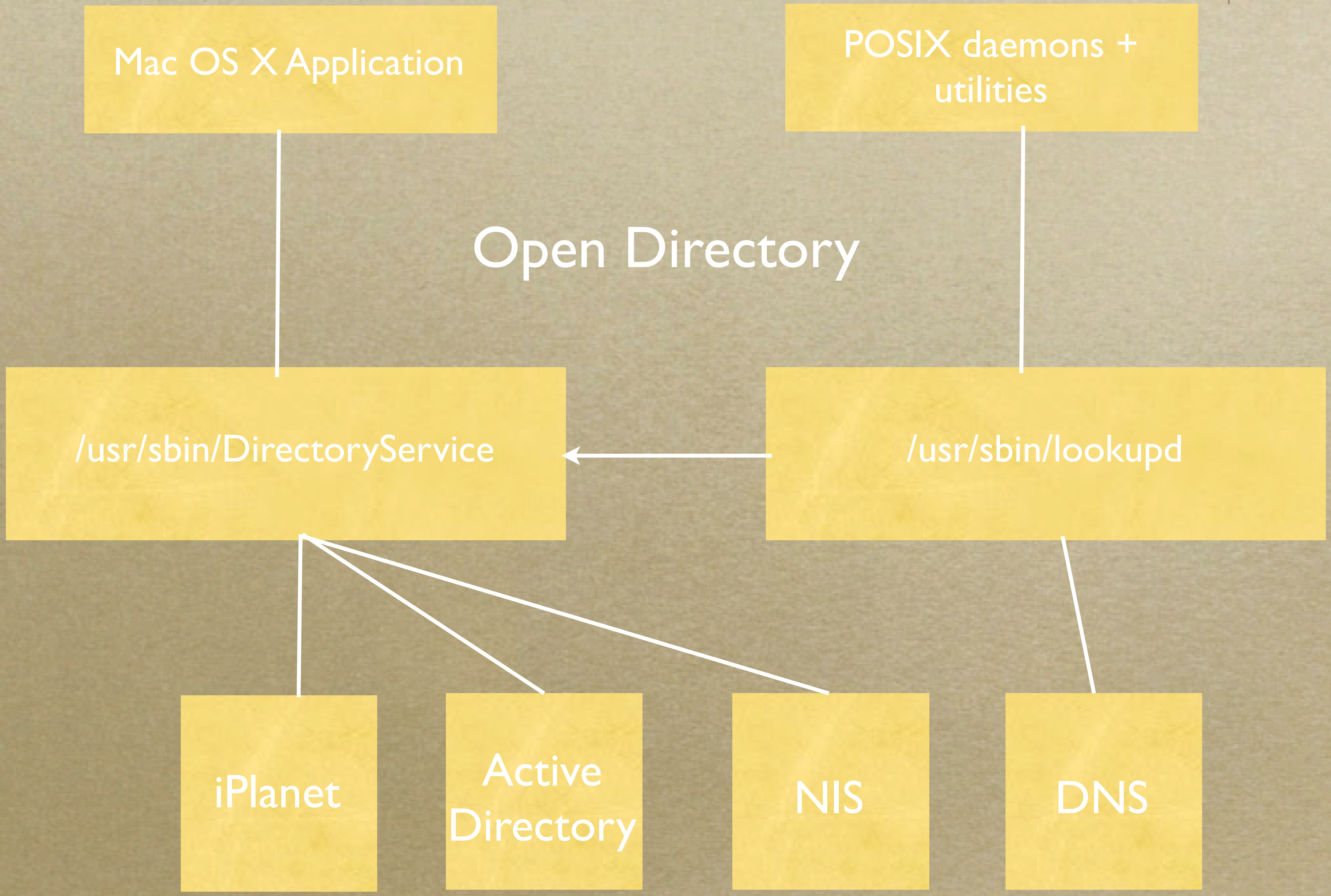
/usr/sbin/DirectoryService

iPlanet

Active Directory

NIS

Architecture: an honest schematic



Why 2 services: a little history

- *libc and source-level compatibility*
- *depends on standard system calls*

Why 2 services?

- *DirectoryService sounds an awful lot like lookupd*
- *DirectoryService is built on the limitations of lookupd*
  - *difficult to extend*
  - *no authentication support*
  - *read-only*

Why 2 services?

- *if lookupd has so many problems, why use it at all?*
  - *DirectoryService has no DNS plug-in*
  - *making changes to libc can be problematic*

DirecoryService daemon in depth

- */usr/sbin/DirecoryService*
  - *startup*
  - *Configuration Files*
  - *plug-in's*

Search Plug-in

- *In many ways, the heart of DirectoryService*
- *Determines which nodes should be searched in which order*

NetInfo Plug-in

- *Always searched first*
- *capable of searching local and remote NetInfo domains*
- *Not searched unless DirectoryService has a non-local node in its search path*

LDAPv3 Plug-in

- *Generalized method for accessing LDAP directories*
- *Covered in more depth in Identification and Mac OS X Server*

Active Directory Plug-in

- *New in Panther: Accesses AD*
- *Covered in more depth in Active Directory Integration*

BSD / NIS Plug-in

- *Why would anyone use this?*

Other Plug-in

- *Service Discovery Plug-in’s*
- *Contacts*

Monitoring DirectoryService

- *dscl*
- *Logging*
  - *Error log*
  - *Server log: /L/P/DirectoryService/.DSTCPListening*
  - *Debug log: sudo killall -USR1 DirectoryService*
  - *API logging: sudo killall -USR2 DirectoryService*

lookupd in (a little less) depth

- */usr/sbin/lookupd*
  - *startup*
  - *Configuration Files*
  - *categories and agents*

lookupd

- *services libc calls*
- *agents: libc calls revolve around various classes of objects*
  - *users*
  - *groups*
  - *hosts*

lookupd agents

- *agents query specific data sources*
  - *NIAgent: NetInfo*
  - *DSAgent: DirectoryService*
  - *FFAgent: specific /etc files*
  - *DNSAgent: dns lookups*

categories and agents: putting it together

- *different agents can be applied to different categories in specific orders*
- *lookupd -configuration*

Authentication + lookupd

- *lookupd has no explcite authentication support*
- *history: how authentication used to work*
- *this is a highly solved problem*
- *apple leverages PAM*

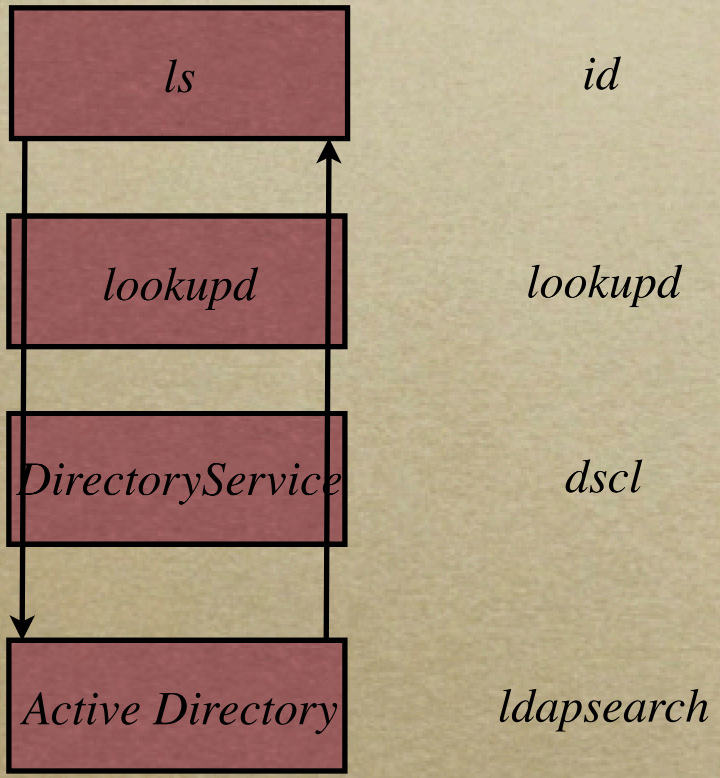
lookupd configuration

- *3 options*
  - */etc/lookupd*
  - *netinfo://config/lookupd*
  - *netinfo://locations/lookupd*
- *Common configuration changes*

lookupd monitoring

- *lookupd: debug mode (-d)*
- *lookupd: query mode (-q)*
- *lookupd logging*

The Open Directory responder chain



Open Directory and Mac OS X Server

Providing Identification, Authorization and Authentication services.

Overview

- *This is an Intro Chapter*
- *Open Directory Server architecture and management*
- *Creating an Open Directory Master*
- *Mac OS X as a member server*

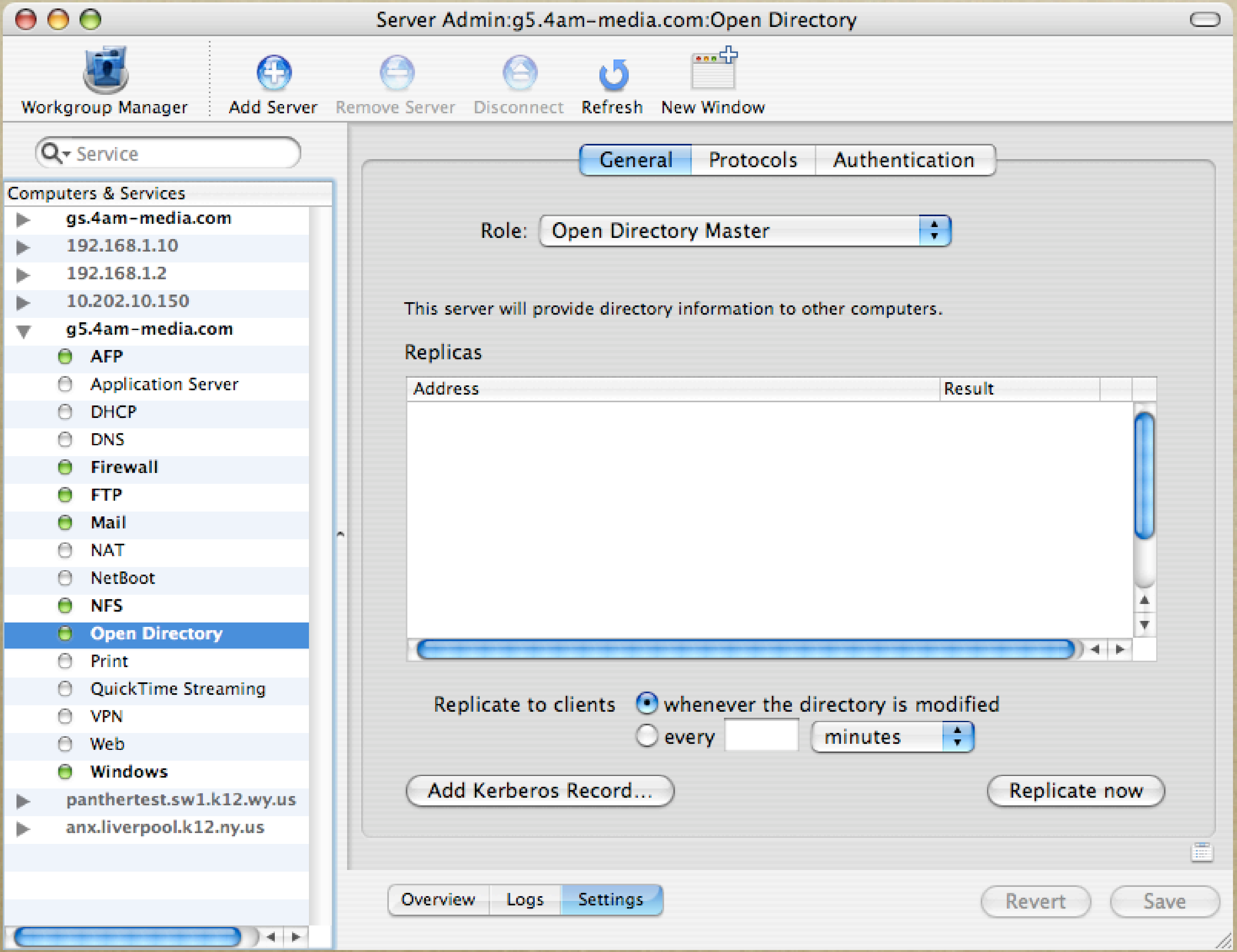
What is Open Directory Server?

- *Provides Identification, Authorization and Authentication services*
- *LDAPv3 (OpenLDAP), Password Server and Kerberos (MIT)*
- *robust configuration architecture*

Open Directory Roles

- *Roles describe specific, well-known configuration states*

Open Directory Management: Server Admin



Server Admin:g5.4am-media.com:Open Directory

Role: Open Directory Master

This server will provide directory information to other computers.

Replicas

Address	Result

Replicate to clients ● whenever the directory is modified
○ every minutes

Creating an Open Directory Master

- *Creates LDAP Directory and KDC*
- *Kerberizes Services on Master*
- *Copies Admin user to new domain*
  - *creates root in shared domain*

Mac OS X Server as an Open Directory Client

- *multiple servers, one domain*
- *Directory Access*

Identification in Open Directory Server

leveraging LDAP

Overview

- *LDAP as a Protocol*
- *OpenLDAP on Mac OS X Server*
- *Exploring directory data in Mac OS X*

LDAP: What is it?

- *Light Weight Directory Access Protocol*
- *Standardized way to access data*
- *Does not imply a particular data storage method*

LDAP: A Protocol Analysis

LDAP: Terms

- *Schema*
- *ObjectClass*
- *attribute*
- *distinguished name*
- *relative distinguished name*

LDAP: Utilities

- *ldapsearch*
- *ldapadd*
- *slapcat and slapadd*
- *ldapper and java ldap browser*

OpenLDAP in Mac OS X

- *OpenLDAP 2.1.22*
- *Startup:*
  - */etc/hostconfig*
  - */System/Library/StartupItems/LDAP*
- */usr/libexec/slapd*

OpenLDAP Configuration

- */etc/openldap*
  - *slapd.conf*
  - *slapd_macosx.conf*
  - *schema/*
  - *ldap.conf*

OpenLDAP Performance: Caching

- *2 Kinds of caching*
  - *BerkelyDB (DB_CONFIG)*
  - *cachesize (slapd.conf)*
- *Databases aren’t that big /var/db/openldap/openldap-data*
- *Just Cache the Whole thing*

OpenLDAP Performance: Indexing

- *Indices support specific kinds of searches*
  - *eq, pres, approx, sub*
- *Several common searches aren't indexed*
- *Modify config file, stop server, run slapindex, re-start server*

OpenLDAP Security

- *SSL*
- *SASL Binds*
- *Access Controls*

Mac OS X LDAP Data

dc=4am-media,dc=com

cn=users *cn=config* *cn=groups* *cn=people* *cn=autoserversetup* *cn=locations*

cn=preset_computer_lists *cn=mounts* *cn=aliases* *cn=machines* *cn=computers*

cn=preset_users *cn=printers* *cn=computer_lists* *cn=preset_groups*

Authentication in Open Directory Server

Kerberos and Password Server

Overview

- *Kerberos: Single Sign On*
- *Password Server: Challenge Response Authentication*

Kerberos: What is it?

- *Network authentication mechanism*
  - *Shared Secret*
  - *Trusted 3rd Party*
  - *Single Sign On*
- *Kerberos assumes that every packet will be captured and attacked*

LDAP: A Protocol Analysis

Kerberos: Terms

- *KDC*
- *realm*
- *encryption type*

Kerberos in Mac OS X

- *MIT Kerberos 1.3.1*
- */usr/sbin/krb5kdc*
- */usr/sbin/kadmind*
- *Startup:*
  - */etc/watchdog.conf*

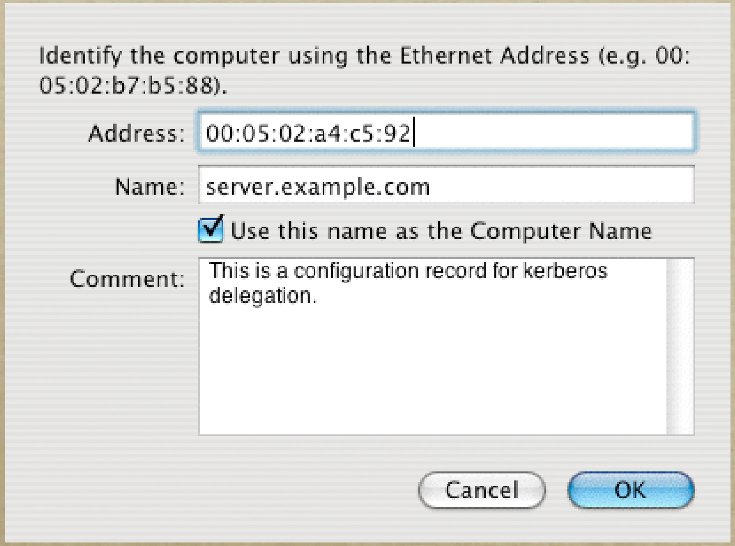
Kerberos Config Files and Databases

- */var/db/krb5kdc*
  - *.k5.REALM*
  - *principal*
  - *kdc.conf*
  - *kadm5.acl*
  - *kadm5.keytab*

Single Sign-On

- *Set up automatically on Master and Replicas*
- *Must be manually set up on other servers*
- *For practical purposes, distributing the secret between KDC and Service*
- *Let’s review Kerberos Authentication*

Single Sign-On: Graphical Set Up



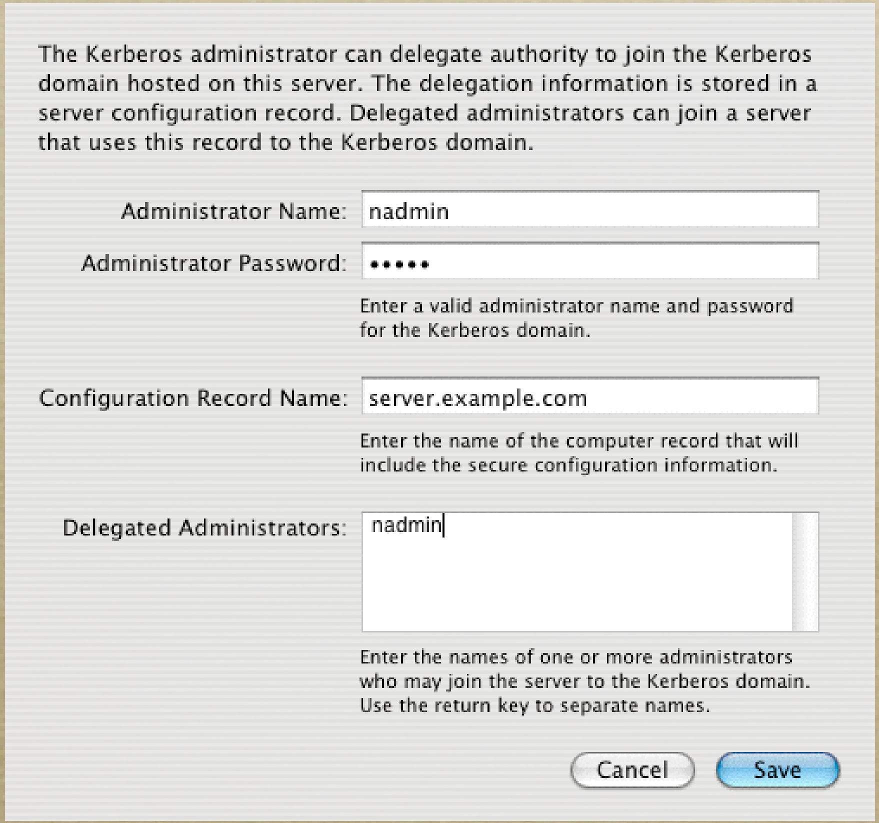
Identify the computer using the Ethernet Address (e.g. 00:05:02:b7:b5:88).

Address: 00:05:02:a4:c5:92

Name: server.example.com

☑ Use this name as the Computer Name

Comment: This is a configuration record for kerberos delegation.



The Kerberos administrator can delegate authority to join the Kerberos domain hosted on this server. The delegation information is stored in a server configuration record. Delegated administrators can join a server that uses this record to the Kerberos domain.

Administrator Name: nadmin

Administrator Password: •••••

Enter a valid administrator name and password for the Kerberos domain.

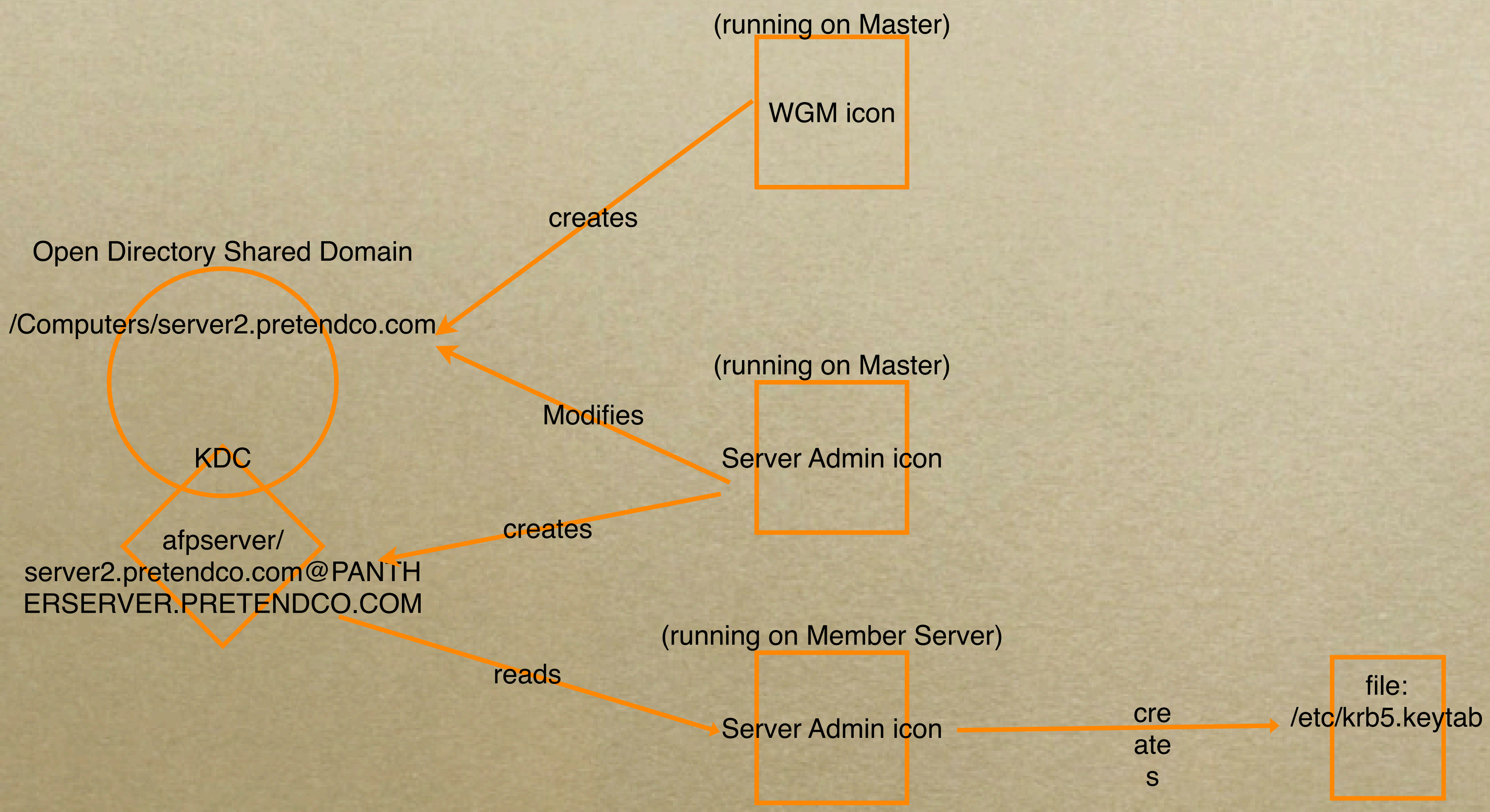
Configuration Record Name: server.example.com

Enter the name of the computer record that will include the secure configuration information.

Delegated Administrators: nadmin

Enter the names of one or more administrators who may join the server to the Kerberos domain. Use the return key to separate names.

Single Sign-On: Behind the Scenes ...



(running on Master)

WGM icon

creates

Open Directory Shared Domain

/Computers/server2.pretendco.com

(running on Master)

Modifies

Server Admin icon

KDC

creates

afpserver/
server2.pretendco.com@PANTH
ERSERVER.PRETENDCO.COM

(running on Member Server)

reads

Server Admin icon

cre
ate
s

file:
/etc/krb5.keytab

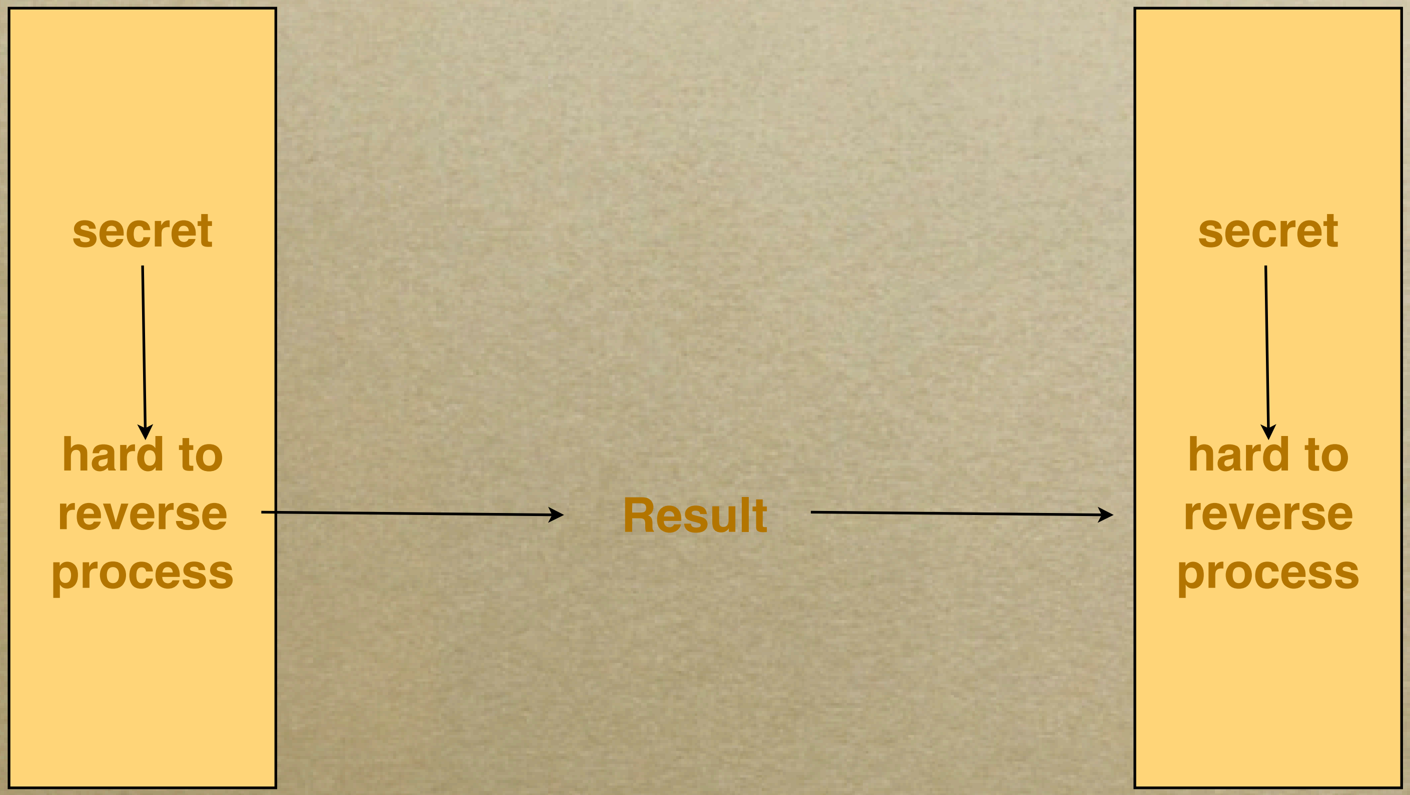
Password Server

- *Supports multiple network authentication mechanisms*
- *Based on SASL: Simple Authentication and Security Layer*

Password Server: architecture

- */usr/sbin/PasswordService*
- *Startup: /etc/watchdog.conf*
- */Library/Logs/PasswordService*
- */Library/Preferences/com.apple.passwordserver*

Challenge-Response Authentication



secret

hard to reverse process

Result

secret

hard to reverse process

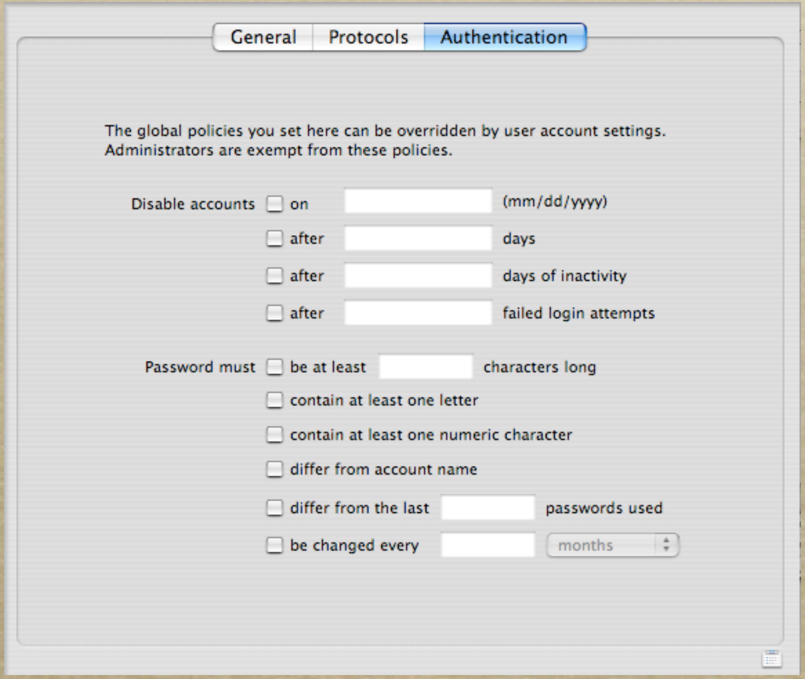
correct / incorrect

Password Server: Plug-in's

- *DHX*
- *CRAM-MD5*
- *MS Chap v2*
- *NTLMv1 and LANMANAGER*
- *apop*
- *WebDAV Digest*

Password Server: Policies

- *Global*
- *Per-User*



The global policies you set here can be overridden by user account settings.
Administrators are exempt from these policies.

Disable accounts □ on ______ (mm/dd/yyyy)
□ after ______ days
□ after ______ days of inactivity
□ after ______ failed login attempts

Password must □ be at least ______ characters long
□ contain at least one letter
□ contain at least one numeric character
□ differ from account name
□ differ from the last ______ passwords used
□ be changed every ______ months

Password Server: Utilities

- *mkpassdb*
- *pwpolicy*
- *NeST*

Password Server-KDC Synchronization

- *Changing Passwords in Kerberos*
  - *kadmind calls mkpassdb*
  - *policies are not synchronized*
- *Changing Password in Open Directory*
  - *PasswordService calls kadmin.local*
  - *some policies are synchronized*

Talking to Password Server

- *Password Server listens on port 106 and 3659*
- *It will accept and reply to text commands (telnet)*

Replication in Open Directory Server

higher availability

Overview

- *replica creation*
- *ldap replication*
- *Password Server replication*
- *Replica Discovery*

Creating a Replica

- *Server Admin*
- *slapconfig*

Creating a Replica: Behind the Scenes

- *Authentication is checked*
- *local ldap server and KDC (if it exists) is deleted*
- *master ldap server is stopped, slapcat’d, and started again.*
- *ldap and kdc dump are scp’d to replica*

Creating a Replica: Behind the Scenes

- *slave ldap server is started*
- *slave kdc is started, passwordserver replication begins.*

LDAP Replication

- *Replication is not part of the LDAP standard*
- *Apple leverages OpenLDAP's replication*
- *Changes on the master are written to a log, which the slurpd daemon then pushes out to replicas (using credentials in slapd's config file)*

Password Server Replication

- *PWS replication is multi-master*
- *Replication occurs on change or interval*
- *no replication partners: everyone talks to everyone*
- *communication is encrypted with a shared keypair*

Password Server Replication

- *timestamp issues*

Replica Discovery

- *List of replicas in DSLDAPv3PlugInConfig.plist*
- *What do you know? How was your weekend?*
- *replicas contacted in parallel*

Replica: Caveats

- *This is not a load balancing technology*
- *Interval applies to Password Server only*
- *Client can talk to different server for LDAP, PWS and Kerb*
- *slapd configurations must be manually updated*

Mac OS X, Mac OS X Server and Active Directory

Fitting in, not standing out... we mean it this time

Overview

- *Active Directory Plug-in: Features*
- *Active Directory Plug-in: Architecture*
- *Active Directory Plug-in: Mac OS X Server*
- *Single Sign-On*
- *MCX Strategies*

Active Directory Plug-in: Features

- *Accesses AD Much like a PC would*
- *Password Policy Enforcement*
- *Flexible Home Directory options*
- *UniqueID options*
- *User Caching*
- *AD Group Administration*

Active Directory Plug-in: Architecture

- */Library/Preferences/DirectoryService*
  - *Active Directory.plist*
  - *ADGroupCache.plist*
  - *winbindd.conf*
- */usr/sbin/dsconfigad*

Active Directory Plug-in: Mac OS X Server

- *winbindd: proxy authentication*
- *single sign-on*

Active Directory Plug-in: the binding process

- *dns lookup (_ldap._tcp.domain.com)*
- *temporary edu.mit.kerberos*
- *kerberized connection using credentials*
- *site policy determines closest DC*
- *second edu.mit.kerberos is built*

Active Directory Plug-in: the binding process

- *new connection is used to search for computer account*
- *computer account is joined or created*
- *if an existing account is joined, the path you specified might not be honored.*

Single Sign On and Active Directory: AFP, FTP, SSH, Mail

- *Method 1 (architecturally cleanest)*
- *Uses Machine account*
  - *obtain password using tdbtool*
  - *use with ktpass to create service principals*
  - *Fix Machine's userPrincipalName*
  - *combine keytabs*
  - *modify service-specific config files*

Single Sign On and Active Directory: AFP, FTP, SSH, Mail

- *Method 2 (most consistent)*
    - *uses specific account for service*
    - *each account is used with ktpass to create service principals*
    - *keytabs are combined*
    - *modify service-specific config files*
        - *survives computer account re-creation*

Single Sign On and Active Directory

- *SMB: Leveraging Samba*
- *(usually) Just join account and edit /etc/smb.conf*
  - *spnego = yes*
  - *security = ads*
  - *workgroup = ADD*
  - *realm = ADS.4AM-MEDIA.COM*

AD and Managed Client Data (MCX)

- *Schema Modification (lets get this over with)*
- *Golden Triangle*

MCX: Using Computer Lists

- *Simplest method for providing MCX data*
- *Computer Lists do not require server-side AD integration*
- *Preserves KDC functionality in Open Directory*

MCX: Using OD Groups

- *Adding AD users to Open Directory groups*
- *Requires Server-Side integration*
- *Open Directory KDC should be disabled*

Open Directory Server: Windows and Unix clients

Leveraging Mac OS X Directory Services

Overview

- *Mac OS X as a PDC*
- *PGina*
- *Unix clients: nss_ldap and pam_krb5*

Mac OS X PDC

- *Integrated with PasswordServer*
- *leverages Open Directory user accounts*
- *Promotion:*
  - *turns on virtual homes*

profiles, scripts and policies

- *Profiles: /Users/Profiles*
- *login scripts:*
  - */etc/logon*
  - *kixtart*
- *Policies (group policy, etc)*
  - *NT Policy Mgr for Domains*

Mac OS X PDC: drawbacks

- *8 yr old technology*
- *Only supports insecure authentication*
- *AD is more functional and widely deployed*
- *Not replicated*
- *Only OD Master may be a PDC*

PDC: Joining

- *On Join, Windows hosts are added to the ‘Windows Computers” list*
- *account names end in $*

pGina

- *Windows software that allows for various authentication methods (including LDAP)*
- *Caches user account locally*
- *Can mount Mac OS X Home Dir over SMB*

Unix clients: identification and authentication

- *Identification: Name Service Switch*
  - *nss_ldap*
- *Authentication: PAM*
  - *pam_krb5 (preferred)*
  - *pam_ldap*