NetBoot, LDAP and HomeLinks in Math MacOSX Labs

Jeff Kopmanis
Manager, MathIT
UM UNIX Admins, June 16, 2005
Previously, all Solaris 8 Labs (103 machines)

LSA AFS Home Directories

Fully managed

Access only for Math Course Students

Maple, MATLAB, Netscape 4.78, etc...
GUI (Sun CDE) was unfamiliar and hard for students to use
Solaris 8 Problems

- GUI (Sun CDE) was unfamiliar and hard for students to use
- Printing through lp/lpr was difficult
GUI (Sun CDE) was unfamiliar and hard for students to use

Printing through lp/lpr was difficult

StarOffice 6 was not 100% compatible with Microsoft Office
GUI (Sun CDE) was unfamiliar and hard for students to use

Printing through lp/lpr was difficult

StarOffice 6 was not 100% compatible with Microsoft Office

Sparse numbers of applications and application availability for 64-bit Solaris 8
Spring 2003

LSA Instructional Technology Grant (Mar/03)
LSA Instructional Technology Grant (Mar/03)

UNIX-centric: synctree? RsyncX? Radmind?
LSA Instructional Technology Grant (Mar/03)

LSA Instructional Technology Grant (Mar/03)

UNIX-centric: synctree? RsyncX? Radmind?

Standalone, CD Load
Spring 2003

LSA Instructional Technology Grant (Mar/03)

UNIX-centric: syntree? RsyncX? Radmind?

Standalone, CD Load
LSA Instructional Technology Grant (Mar/03)

UNIX-centric: synctree? RsyncX? Radmind?

Standalone, CD Load

Apple Computer: John Hickey & Interns

Netboot + Workgroup Manager

Radmind as fallback option
LSA Instructional Technology Grant (Mar/03)

UNIX-centric: synctree? RsyncX? Radmind?

Standalone, CD Load

Apple Computer: John Hickey & Co.

Netboot + Workgroup Manager

Radmind as fallback option
24 iMac G4 800MHz 512MB RAM

Xserve G4 1GHz 512MB RAM

100Mbps to each iMac, 1Gbps to Xserve

First image: MacOS 10.2.3 (Jaguar) <3GB

2 minute or less boot time!!

Online by mid-June!

F03, W04, F04: Math 216 - EXCLUSIVE, Math 215 - REQUESTED

3 LSA-IT Training Sessions

2 Apple Certification Courses

Lots of misc. Math Dept orientations and
Evaluation: 1 Year Later...

A Very Successful Project
Evaluation: 1 Year Later...

A Very Successful Project

Thank You, John, Sheri and all of Apple Computer!
Evaluation: 1 Year Later...

A Very Successful Project

But, we also learned some things...
Evaluation: 1 Year Later...

Saving Bookmarks: highly unusual and misunderstood by students
Evaluation: 1 Year Later...

- Saving Bookmarks: highly unusual and misunderstood by students
- A Volatile Desktop: erased on logout
Evaluation: 1 Year Later...

- Saving Bookmarks: highly unusual and misunderstood by students
- A Volatile Desktop: erased on logout
- Image bloat: 6+ GB (still 2 minute boot times!)
Evaluation: 1 Year Later...

- Saving Bookmarks: highly unusual and misunderstood by students
- A Volatile Desktop: erased on logout
- Image bloat: 6+ GB (still 2 minute boot times!)
- Cross-realm K5 trusts not supported in Jaguar
Evaluation: 1 Year Later...

- Saving Bookmarks: highly unusual and misunderstood by students
- A Volatile Desktop: erased on logout
- Image bloat: 6+ GB (still 2 minute boot times!)
- Cross-realm K5 trusts not supported in Jaguar
- Jaguar getting very long in tooth, both on server and client
Evaluation: 1 Year Later...

118 machines on a single G4 Xserve doesn’t work (Apple’s 100 machine limit is TRUE)
118 machines on a single G4 Xserve doesn’t work (Apple’s 100 machine limit is **TRUE**)

Server Settings app is hopelessly broken under Jaguar
118 machines on a single G4 Xserve *doesn’t* work (Apple’s 100 machine limit is **TRUE**)

Server Settings app is hopelessly broken under Jaguar

512MB RAM in Xserve is inadequate for anything over 50 netboot clients
Evaluation: 1 Year Later...

- 118 machines on a single G4 Xserve *doesn’t* work (Apple’s 100 machine limit is **TRUE**)
- Server Settings app is hopelessly broken under Jaguar
- 512MB RAM in Xserve is inadequate for anything over 50 netboot clients
- **UPGRADE TIME!!!**
Summer, 2004

- Panther Load (MacOSX 10.3.5)
- Easy to reproduce image: LSA SNI-based
- OpenAFS 1.2.10a
- Layered LDAP: UMOD & Math
- LSA LabHomeDirs.pkg (Sites-based)
- “Roaming Profiles” via HomeLinks
Used UMOD for the bulk of our directory information (IFS home, shell, Full Name, etc)
Used UMOD for the bulk of our directory information (IFS home, shell, Full Name, etc).

Workgroup Manager and NetBoot both require and use an LDAP database.
Used UMOD for the bulk of our directory information (IFS home, shell, Full Name, etc).

Workgroup Manager and NetBoot both require and use an LDAP database.

OpenDirectory on NetBoot server holds above data, and...
Used UMOD for the bulk of our directory information (IFS home, shell, Full Name, etc).

Workgroup Manager and NetBoot both require and use an LDAP database.

OpenDirectory on NetBoot server holds above data, and...

Lists of uniqnames of those permitted in our labs!
LDAP: OpenDirectory settings

Configure: LDAP Settings

Search base: `dc=math,dc=lsa,dc=umich,dc=edu`

Database: `/var/db/openldap/openldap-data`

Return a maximum of 11000 search results

Search times out in 1 hours

Use SSL

SSL Certificate:

SSL Key:

CA Certificate:
LDAP: Client LDAP

- Clients point to both UMOD and Math LDAP servers.

- Order of servers is IMPORTANT!

- Math LDAP maps groups and machines (MCX records) so that only those groups and machines present are *authorized* to use the labs.
LDAP: Authentication

Choose where to search for user authentication information.

Search: Custom path

Directory Node
/NetInfo/root
/LDAPv3/nl.math.lsa.umich.edu
/LDAPv3/ldap.itd.umich.edu

Drag directory nodes into your preferred order for searching.

Remove  Add...

Click the lock to prevent further changes.
LDAP: UMOD mappings

Access this LDAPv3 server using Custom

Record Types and Attributes
- Default Attribute Types
  - Users
    - RecordName
    - RealName
    - UniqueID
    - PrimaryGroupID

Map to any items in list
- displayName
- cn

Search base:

Search in:
- all subtrees
- first level only

Write to Server...
Workgroup Manager: Machine Groups

Authenticated as sysop to directory: /LDAPv3/127.0.0.1

List Name: B727-EH

<table>
<thead>
<tr>
<th>Address</th>
<th>Description</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:0a:95:78:1f:50</td>
<td>lab0083</td>
<td>Imac-800 15&quot; w/combo drive</td>
</tr>
<tr>
<td>00:0a:95:c7:f4:40</td>
<td>lab0168</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:cb:87:b4</td>
<td>lab0150</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c0:38</td>
<td>lab0147</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c3:20</td>
<td>lab0130</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c3:34</td>
<td>lab0122</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c3:46</td>
<td>lab0103</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c3:60</td>
<td>lab0171</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c3:62</td>
<td>lab0093</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:c4:0e</td>
<td>lab0161</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:ce:d8</td>
<td>lab0095</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:d0:e6</td>
<td>lab0155</td>
<td></td>
</tr>
<tr>
<td>00:0a:95:d1:d1:74</td>
<td>lab0167</td>
<td></td>
</tr>
</tbody>
</table>
**Workgroup Manager: Intersection**

![Workgroup Manager Interface](image)

- **Authenticated as sysop to directory:** /LDAPv3/127.0.0.1

### Computer List

<table>
<thead>
<tr>
<th>Name</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guest Computers</td>
<td></td>
</tr>
<tr>
<td>Windows Computers</td>
<td></td>
</tr>
<tr>
<td>1st Floor</td>
<td></td>
</tr>
<tr>
<td>2nd Floor</td>
<td></td>
</tr>
<tr>
<td>3rd Floor</td>
<td></td>
</tr>
<tr>
<td>4th Floor</td>
<td></td>
</tr>
<tr>
<td>5th Floor</td>
<td></td>
</tr>
<tr>
<td>B069-EH</td>
<td>1026</td>
</tr>
<tr>
<td>B723-EH</td>
<td>1025</td>
</tr>
<tr>
<td>B727-EH</td>
<td>1027</td>
</tr>
<tr>
<td>B728-EH</td>
<td>1029</td>
</tr>
<tr>
<td>B735-EH</td>
<td></td>
</tr>
<tr>
<td>B737-EH</td>
<td></td>
</tr>
<tr>
<td>B743-EH</td>
<td></td>
</tr>
<tr>
<td>B745-EH</td>
<td></td>
</tr>
<tr>
<td>Math Faculty</td>
<td></td>
</tr>
<tr>
<td>Math Staff</td>
<td></td>
</tr>
<tr>
<td>Math Students</td>
<td></td>
</tr>
<tr>
<td>Gateway User</td>
<td></td>
</tr>
<tr>
<td>Kiosk User</td>
<td></td>
</tr>
</tbody>
</table>

- **Access:**
  - Allow users with local-only accounts

**Presets:** None
Lab Home Directories

- LSA started with Sites’ scripts (thanks!)
LSA started with Sites’ scripts (thanks!)
During Login, they copy default user directory template to actual user directory
Lab Home Directories

- LSA started with Sites’ scripts (thanks!)
- During Login, they copy default user directory template to actual user directory
- During Logout, they perform quota checking and cleanup.
Lab Home Directories

- LSA started with Sites’ scripts (thanks!)
- During Login, they copy default user directory template to actual user directory
- During Logout, they perform quota checking and cleanup.
- Configurable aging and quotas
Lab Home Directories

- LSA started with Sites’ scripts (thanks!)
- During Login, they copy default user directory template to actual user directory
- During Logout, they perform quota checking and cleanup.
- Configurable aging and quotas
- LabHomeDirs.pkg
Deficiencies...

Everything was local
Deficiencies...

- Everything was local
- No persistence between sessions
Deficiencies...

- Everything was local
- No persistence between sessions
- Won’t work well with a NetBooted system, since there is no persistent disk storage
Deficiencies...

- Everything was local
- No persistence between sessions
- Won’t work well with a NetBooted system, since there is no persistent disk storage

Enter HomeLinks!
HomeLinks

Dynamically maps local structure onto network storage
HomeLinks

- Dynamically maps local structure onto network storage
- Can create network locations, if not present
HomeLinks

- Dynamically maps local structure onto network storage
- Can create network locations, if not present
- XML (plist) files used for configuration
HomeLinks

- Dynamically maps local structure onto network storage
- Can create network locations, if not present
- XML (plist) files used for configuration
- PropertyList Editor becomes an easy-to-use configuration tool
HomeLinks

- Dynamically maps local structure onto network storage
- Can create network locations, if not present
- XML (plist) files used for configuration
- PropertyList Editor becomes an easy-to-use configuration tool
- Needs **plistbuddy**, a utility that Apple
Build a network path prefix ($PREFIX) for the user logging in

For each of the Directories in the Dictionary...

use **ditto** to copy from /Users/username/key to $PREFIX/value
Sample HomeLinks.plist file

- **Root**
  - Comments
  - CreationHint
  - Directories
  - HomeFS
  - HomePrefix
  - CreateIfNotPresent
- **Directories**
  - Documents
  - Library/Preferences
  - Library/Safari
  - Movies
  - Music
  - Pictures
  - Public
  - HomeAFScell
  - HomeFS
  - HomePrefix

- **Class**
  - Dictionary
  - String
  - Boolean

- **Value**
  - 6 key/value pairs
  - 4 key/value pairs
  - if Directory entry begins with / its absolute!
  - Directories are symlinks into: $HomePrefix/
  - AFS, SMB
  - =AFSASH, =SMBHASH, (pathname to home)
  - Yes
  - 7 key/value pairs
  - Private/MATHLAB/Documents
  - Private/MATHLAB/Library/Preferences
  - Private/MATHLAB/Library/Safari
  - Private/MATHLAB/Movies
  - Private/MATHLAB/Music
  - Private/MATHLAB/Pictures
  - Public
  - umich.edu
  - AFS
  - =AFSASHASH
Constructing a Prefix

- **HomeFS** is a flag for how to interpret things. Currently only AFS is implemented.
- **HomePrefix** determines what comes before the user’s uniqname in the path.
- =AFSHASH denotes the UM double-hash
- HomeAFScell is the cell name
Prefix examples:

<table>
<thead>
<tr>
<th>Constructs</th>
<th>/afs/umich.edu/user/k/o/kopmanis</th>
</tr>
</thead>
<tbody>
<tr>
<td>HomeAFScell</td>
<td>String ▼ umich.edu</td>
</tr>
<tr>
<td>HomeFS</td>
<td>String ▼ AFS</td>
</tr>
<tr>
<td>HomePrefix</td>
<td>String ▼ =AFSHASH</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constructs</th>
<th>/afs/lsa.umich.edu/user/k/o/kopmanis</th>
</tr>
</thead>
<tbody>
<tr>
<td>HomeAFScell</td>
<td>String ▼ lsa.umich.edu</td>
</tr>
<tr>
<td>HomeFS</td>
<td>String ▼ AFS</td>
</tr>
<tr>
<td>HomePrefix</td>
<td>String ▼ =AFSHASH</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constructs</th>
<th>/home/exports/kopmanis</th>
</tr>
</thead>
<tbody>
<tr>
<td>HomeFS</td>
<td>String ▼ NFS</td>
</tr>
<tr>
<td>HomePrefix</td>
<td>String ▼ /home/exports</td>
</tr>
</tbody>
</table>
Directory Dictionary

- Works with key-value pairs
- Keys are the “from” or “source” location
- Key maps to /Users/username/key
- Values are the “to” or “destination” location
- Value maps to $PREFIX/value
key-value examples:

<table>
<thead>
<tr>
<th>Directory</th>
<th>Dictionary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documents</td>
<td>Dictionary</td>
</tr>
<tr>
<td>Library/Preferences</td>
<td>Private/MATHLAB/Library/Preferences</td>
</tr>
<tr>
<td>Library/Safari</td>
<td>Private/MATHLAB/Library/Safari</td>
</tr>
<tr>
<td>Movies</td>
<td>Private/MATHLAB/Movies</td>
</tr>
<tr>
<td>Music</td>
<td>Private/MATHLAB/Music</td>
</tr>
<tr>
<td>Pictures</td>
<td>Private/MATHLAB/Pictures</td>
</tr>
<tr>
<td>Public</td>
<td>Public</td>
</tr>
</tbody>
</table>

With our previous umich.edu AFS prefix:

/Users/kopmanis/Documents

/afs/umich.edu/user/k/o/kopmanis/Private/MATHLAB/Documents

/Users/kopmanis/Public

/afs/umich.edu/user/k/o/kopmanis/Public

/Users/kopmanis/Library/Preferences

/afs/umich.edu/user/k/o/kopmanis/Private/MATHLAB/Library/Preferences
HomeLinks will build a structure according to the plist configuration file

If “Yes”, this flag signals that directories should be created in the target location if not present

Each directory is tested along the way

If $PREFIX is not present, no creation will be possible, so no mapping is performed (Dec04)
Build a network path prefix ($PREFIX) for the user logging in

For each of the Directories in the Dictionary...

use **ditto** to copy from /Users/username/key to $PREFIX/value
Wait a Minute!!!

What about Kerberos tickets and AFS tokens and file ACLs and permissions ?!
Panther Security Sessions

OSX processes run as root or other admin users

MacOS 10.3.x
Panther Security Sessions

- OSX processes run as root or other admin users
- User authenticates at login to create a Security Session, which includes AFS tokens.

MacOS 10.3.x

User

AFS tokens
Panther Security Sessions

- OSX processes run as root or other admin users
- User authenticates at login to create a Security Session, which includes AFS tokens.
- HomeLinks, run by MacOSX, uses `sudo` to reach into user’s Security Session to use AFS tokens of that user
Panther Security Sessions

- HomeLinks uses ITCS-Sites/LSA LoginHooks package to provide /etc/hooks structures.
- LI81.HomeLinks is run at login time.
- The magic sudo call is made in LI81.HomeLinks.
- LI81.HomeLinks is a wrapper for the real script in /etc/HomeLinks/HomeLinks.
#!/bin/bash
sudo -u $1 /etc/hooks/HomeLinks/Homelinks $1
#!/bin/bash

# First arg should be user shortname
export UNIQNAME=$1
export UNIQUID=$(id -u $UNIQNAME)
export DEBUG=0
export CONFIGFILE="/etc/hooks/HomeLinks/HomeLinks.plist"
export PLB="/usr/bin/PlistBuddy"

. /etc/radmind.defaults

function PLB_get () {
  $PLB -c "print $1" $CONFIGFILE | grep -v "Not Exist"
}

function PLB_get_Prefix () {
  user=$1
  HomePrefix=""
  HomePrefix="":HomePrefix""
  then
    local HomePrefix="":AFS"
    then
      echo /afs/$AFScell/user/${user:0:1}/${user:1:1}/$user/
    else
      HomePrefix="":SMB"
    fi
  fi
  if [ "$HomePrefix" = ":AFSHASH" ]
    then
      echo /afs/$AFScell/user/${user:0:1}/${user:1:1}/$user/
    else
      echo $HomePrefix
    fi
  fi
  echo $HomePrefix
}

function link_directory () {
  # $APPLE is the "Apple-standard" directory location we're going to map
  APPLE=$1
  # $REAL is the "real" location we'll be mapping to
  REAL=$2
  APPLEPATH=/Users/$UNIQNAME/$APPLE
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "APPLEPATH=$APPLEPATH"; fi
  if [ ${REAL:0:1} = '/' ]
    then
      PREFIX=""
    else
      PREFIX=$HomePrefix
    fi
  fi
  BUILDPATH=$PREFIX$REAL
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "BUILDPATH=$BUILDPATH"; fi
  # # and, if it doesn't (and we have $CreateIfNotPresent is set), create the target directory and permit it to the user
  if [ "$CreateIfNotPresent" = "true" ]
    then
      if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Create: $BUILDPATH"; fi
      mkdir -p $BUILDPATH
    fi
  fi
  # finally, make the link
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Link: ln -s $BUILDPATH $APPLEPATH"; fi
  ln -sf $BUILDPATH $APPLEPATH
}

#######################################################
# MAIN #
#######################################################

if [ ${UNIQUID} -lt 1000 ]
then
  /usr/bin/logger -is -t HomeLinks -p user.info "Nothing to be done for $UNIQNAME (id=$UNIQUID)"
  exit
fi

export CreateIfNotPresent=""$PLB_get "":CreateIfNotPresent"
if [ "$CreateIfNotPresent" = "true" ]
then
  # finally, make the link
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Link: ln -s $BUILDPATH $APPLEPATH"; fi
  ln -sf $BUILDPATH $APPLEPATH
else
  /usr/bin/logger -is -t HomeLinks -p user.info "ERROR: Unsupported Home FS!"
fi

# # and, if it doesn't (and we have $CreateIfNotPresent is set), create the target directory and permit it to the user
if [ "$CreateIfNotPresent" = "true" ]
then
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Create: $BUILDPATH"; fi
  mkdir -p $BUILDPATH
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Copy: $APPLEPATH to $BUILDPATH"; fi
  ( cd $APPLEPATH ; tar cf - . ) | ( cd $BUILDPATH ; tar xpf - )
fi

# remove any existing directory first
if [ -d $APPLEPATH ]
then
  rm -rf $APPLEPATH
  if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Remove: $APPLEPATH"; fi
fi

# FINALLY, make the link
if [ "$DEBUG" = "1" ]; then /usr/bin/logger -is "Link: ln -s $BUILDPATH $APPLEPATH"; fi
ln -sf $BUILDPATH $APPLEPATH

}
Needs to test for the existence of HomeDir:
A non-existent HomeDir results in a login that “jumps off a cliff”
Needs to test for the existence of HomeDir: A non-existent HomeDir results in a login that “jumps off a cliff”

Needs to check for the success of the ditto runs: quotas or other limitations result in an incomplete template copy, possibly fatal to the session
HomeLinks.mpkg

Complete UM-Installable Meta-Package containing:

HomeLinks v2
LoginHooks
LabHomeDirs
plistbuddy

AVAILABLE SOON

https://www.math.lsa.umich.edu/software
(requires UMICH Kerberos)
Questions?

Jeff Kopmanis, MathIT
kopmanis@umich.edu
https://www.math.lsa.umich.edu/software