Table of Contents

[1. Resources 2](#_Toc382918305)

[2. Machines and Packages 2](#_Toc382918306)

[3. Users 3](#_Toc382918307)

[4. NTP & Time Sync 3](#_Toc382918308)

[5. DNS Setup 4](#_Toc382918309)

[5.1. Need to Setup the DNS Servers for Kerberos 4](#_Toc382918310)

[5.2. Testing DNS Setup 4](#_Toc382918311)

[6. Kerberos 4](#_Toc382918312)

[6.1. Installation of MIT Kerberos KDC on Debian 4](#_Toc382918313)

[6.1.1. Testing MIT Kerb on Debian 5](#_Toc382918314)

[6.2. Installation of Heimdal Kerberos KDC on OpenBSD 5](#_Toc382918315)

[6.3. Adding Hosts on OpenBSD/Heimdal 6](#_Toc382918316)

[6.4. Adding Hosts on Linux/MIT 6](#_Toc382918317)

[6.5. Add NFS Entries 6](#_Toc382918318)

[6.6. Adding Samba entries 7](#_Toc382918319)

[6.7. Adding Users on Linux/MIT 7](#_Toc382918320)

[6.8. Slave kerberos2.setup 7](#_Toc382918321)

[6.9. Push the database from master with hprop: 8](#_Toc382918322)

[6.10. Kerberos Client Setup 9](#_Toc382918323)

[6.10.1. Kerberos Client Setup on Debian 9](#_Toc382918324)

[6.10.2. Kerberos Client Setup on Ubuntu 10](#_Toc382918325)

[6.10.3. Kerberized SSH on Linux Hosts 11](#_Toc382918326)

[6.10.4. Windows Kerberos Client Setup 11](#_Toc382918327)

[6.11. Kerberized Programs 11](#_Toc382918328)

[6.11.1. Kerberized Telnet 12](#_Toc382918329)

[7. Samba 12](#_Toc382918330)

[7.1. Installation of Samba 12](#_Toc382918331)

[7.2. Adding Samba Users 13](#_Toc382918332)

[adduser username 13](#_Toc382918333)

[8. NFSv4 Exports 13](#_Toc382918334)

[8.1. Debian NFS Client 13](#_Toc382918335)

[8.2. NFSv4 Server 14](#_Toc382918336)

[9. LDAP 15](#_Toc382918337)

[9.1. Debian LDAP Server 15](#_Toc382918338)

[9.2. (Directory service) with TLS 16](#_Toc382918339)

[ypldap -- YP map server using LDAP backend 22](#_Toc382918340)

[More Kerberos Setup 22](#_Toc382918341)

# Resources

http://www.debian-administration.org/articles/570

# Machines and Packages

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Hostname | IP Address | OS/Arch | Roles | Packages |
| Fubaria | 172.16.6.187 | Sparc x64, OpenBSD | slave DNS, backup KDC | named, kdc |
| darwin | 172.16.6.189 | Sparc x64, FreeBSD | MIT Kerberos KDC | kdc, kadmind, kpasswdd |
| gigantor | 172.16.6.2 | X86/win2k3 |  |  |
| Netlab1 | 172.16.6.3 | Sparc x64, Debian Wheezy | LDAP Server, Samba PDC |  |
| Netlab2 | 172.16.6.31 | Sparc x64, Debian |  |  |
| Nightshade | 172.16.6.185 | X86\_64, debian wheezy | client |  |
| lanparty | 172.16.6.6 | X86\_64, Ubuntu 13.10 | Minecraft server, NFS exports, file server |  |

OpenAFS

OpenLDAP has to have support for SASL and GSSAPI. SASL can use GSSAPI to extend it's auth mechanisms. SASL and GSSAPI are frameworks that various authentication providers can be plugged into, for example Kerberos or NTLM.

Here's an example to help make this a little clearer (brutally simplified for clarity's sake):

1. Client connects to server and says, "I support SASL! How should I authenticate myself?"
2. Server receives the connection and responds, "I also support SASL, and can use these mechanisms, in descending order of preference: GSSAPI, CRAM-MD5, PLAIN."
3. Client responds, "Of the choices, I'd like to use GSSAPI."
4. Server responds "GSSAPI? Capital. I support Kerberos and NTLM."
5. Client responds "Let's use Kerberos. Here's my encrypted ticket etc. etc."

# Users

|  |  |  |  |
| --- | --- | --- | --- |
| Name | UID | Groups(GId) |  |
| Franklin | 1000 | 1000(users), |  |
| alex | 1010 |  |  |
| ethan | 1009 |  |  |
| Fubaria | 1012 |  |  |

franklin:x:1000:1000:franklin,,,:/home/franklin:/bin/bash

userftp:x:1001:1001::/home/FTP-shared:/bin/false

minecraft:x:1002:60::/home/minecraft:/bin/bash

armagetronad:x:1003:1003::/home/armagetronad:/bin/sh

l4d:x:1004:60::/home/l4d:/bin/bash

kenf:x:1005:1002:Ken Felix,,,:/home/kenf:/bin/bash

ossec:x:1006:1004::/usr/local/var/ossec:/bin/false

ossecm:x:1007:1004::/usr/local/var/ossec:/bin/false

ossecr:x:1008:1004::/usr/local/var/ossec:/bin/false

fubaria:x:1012:1007:fubaria,,,:/home/fubaria:/bin/bash

phreakinggeek:x:1013:1007:a man,,,:/home/phreakinggeek:/bin/bash

tf2server:x:1014:1009:Team Fortress 2,,,:/home/tf2server:/bin/bash

useradd -d /home/ethan -g fubaria -M -p ed1tty -s /bin/bash –u 1009 ethan

useradd -d /home/alex -g fubaria -M –p alexdiaz02 -s /bin/bash –u 1010 alex

useradd -d /home/fubaria -g fubaria -M -p fub2002 -s /bin/bash -u 1012 fubaria

useradd -d /home/phreakinggeek -g fubaria -M -p phr3ak -s /bin/bash -u 1013 phreakinggeek

useradd -d /home/laura -g fubaria -M -p Macur1z -s /bin/bash -u 1015 laura

# NTP & Time Sync

apt-get install ntp

ntpq -p

# DNS Setup

NOTE: The command “**hostname –f” should return FQDN on all machines.** A common reason is that the /etc/hosts file contains a line with "127.0.0.1 nfsclient1 localhost.localdomain localhost" or similar.

# Need to Setup the DNS Servers for Kerberos

The DNS servers need to have special entries for Kerberos. Add this to the forward lookup file.

kerberos IN CNAME darwin.bitsmasher.net.

kerberos1 IN CNAME darwin.bitsmasher.net.

kerberos2 IN CNAME fubaria.bitsmasher.net.

\_kerberos IN TXT BITSMASHER.NET

\_kerberos.\_udp IN SRV 00 00 08 darwin.bitsmasher.net.

; \_kerberos.\_udp IN SRV 10 00 88 fubaria.bitsmasher.net.

\_kerberos.\_tcp IN SRV 00 00 88 darwin.bitsmasher.net.

;\_kerberos.\_tcp IN SRV 10 00 88 fubaria.bitsmasher.net.

; designate the master KDC:

\_kerberos-master.\_udp IN SRV 00 00 88 darwin.bitsmasher.net.

; designate the kadmind server

\_kerberos-adm.\_tcp IN SRV 00 00 749 darwin.bitsmasher.net.

; designate the kpasswdd server

\_kpasswd.\_udp IN SRV 00 00 464 darwin.bitsmasher.net.

\_ldap.\_tcp.dc.\_msdcs SRV 0 0 636 gigantor.bitsmasher.net.

Make sure you can resolve both keyserver aliases from a third machine before you proceed.

# Testing DNS Setup

root@darwin:/root> getent hosts badreligion

172.16.6.50 badreligion.bitsmasher.net

root@darwin:/root> getent hosts 172.16.6.50

172.16.6.50 badreligion.bitsmasher.net

# Kerberos

## Installation of MIT Kerberos KDC on Debian

sudo apt-get install krb5-{admin-server,kdc}

**sudo mkdir /var/log/kerberos**

To the end of /etc/krb5.conf:

[logging]  
 kdc = [FILE:/var/log/kerberos/krb5kdc.log](file:///\\var\log\kerberos\krb5kdc.log)  
 admin\_server = [FILE:/var/log/kerberos/kadmin.log](file:///\\var\log\kerberos\kadmin.log)  
 default = [FILE:/var/log/kerberos/krb5lib.log](file:///\\var\log\kerberos\krb5lib.log)

Start the kadmin daemon:

root@nightshade:/var/log# ps aux | grep kadmind

root 2900 0.0 0.0 41704 0 ? Ss 2013 0:00 /usr/sbin/kadmind -P /var/run/kadmind.pid

Or restart:

**invoke-rc.d krb5-admin-server restart**

**invoke-rc.d krb5-kdc restart**

# Testing MIT Kerb on Debian

root@nightshade:/etc# kadmin.local

Authenticating as principal root/admin@BITSMASHER.NET with password.

kadmin.local: listprincs

kadmin.local: **addprinc root/admin**

# Installation of Heimdal Kerberos KDC on OpenBSD

(Secure storage of user and machine passwords)

/etc/krb5.conf (FreeBSD)

/usr/libexec/kdc --detach

<http://jurjenbokma.com/ApprenticesNotes/kerberos_on_OpenBSD.html>

#/var/heimdal (on openbsd, FreeBSD)

#Do “tail –f /var/heimdal/kdc.log” on Darwin to debug KDC issues.

mkdir /var/heimdal; kstash –random-key; kadmin -l init BITSMASHER.NET;

# verify the key

ktutil --keytab=/var/heimdal/m-key list

# copy the key to the slave

root@darwin:/var/heimdal> kadmin -l

kadmin> list \*

default

kadmin/admin

kadmin/hprop

kadmin/changepw

changepw/kerberos

krbtgt/BITSMASHER.NET

Logs:

/var/heimdal/kdc.log

/var/heimdal/kadmind.log

# Adding Hosts on OpenBSD/Heimdal

You have to create an entry for each host, then on the admin server, you export the keytab over to that box. This will allow for things like passwd-less SSH if they have done kinit.

root@darwin:/etc> kadmin -l

kadmin> add --random-key host/fubaria.bitsmasher.net

kadmin> getprincs \*

host/fubaria.bitsmasher.net

# extract a copy of the key to be placed on the principal

kadmin> ext --keytab=/etc/kerberosV/krb5.keytab host/fubaria.bitsmasher.net

kadmin> add --random-key host/gigantor.bitsmasher.net

kadmin> ext --keytab=/etc/krb5.keytab host/gigantor.bitsmasher.net

kadmin> add --random-key host/darwin.bitsmasher.net

kadmin> ext --keytab=/etc/krb5.keytab host/darwin.bitsmasher.net

root@darwin:/etc> ktutil list

FILE:/etc/krb5.keytab:

# Adding Hosts on Linux/MIT

kadmin: ank -kvno kvno -randkey host/badreligion.local.bitsmasher.net

kadmin: ank -kvno kvno -randkey nfs/mail.local.bitsmasher.net

addprinc -kvno kvno -randkey cifs/lanparty.local.bitsmasher.net

ktadd -e rc4-hmac:normal cifs/lanparty.local.bitsmasher.net

# Add NFS Entries

root@darwin:/root> kadmin

kadmin> add --random-key nfs/nightshade.bitsmasher.net

root/admin@BITSMASHER.NET's Password:

Max ticket life [1 day]:

Max renewable life [1 week]:

Principal expiration time [never]:

Password expiration time [never]:

Attributes []:

# extract a copy of the key to be placed on the principal

kadmin> ext --keytab=/etc/krb5.keytab [nfs/nightshade@BITSMASHER.NET](mailto:nfs/nightshade@BITSMASHER.NET)

root@darwin:/etc/kerberosV> scp /etc/krb5.keytab 172.16.6.6:/etc

root@172.16.6.6's password:

krb5.keytab 100% 6044 5.9KB/s 00:00

Make sure only the root user can access the keytab:

$ chown root:root /etc/krb5.keytab

$ chmod 0600 /etc/krb5.keytab

# Adding Samba entries

kadmin> add -r [cifs/netlab1.bitsmasher.net@BITSMASHER.NET](mailto:cifs/netlab1.bitsmasher.net@BITSMASHER.NET)

# extract a copy of the key to be placed on the principal

kadmin> ext --keytab=/etc/krb5.keytab \*/[netlab1.bitsmasher.net@BITSMASHER.NET](mailto:netlab1.bitsmasher.net@BITSMASHER.NET)

root@darwin:/root> ktutil --keytab=/etc/krb5.keytab list

Make sure only the root user can access the keytab:

$ chown root:root /etc/krb5.keytab

$ chmod 0600 /etc/krb5.keytab

# Adding Users on Linux/MIT

From nightshade:

kadmin: addprinc -kvno kvno -pw changeme adh4wk

kadmin: ktadd –k /etc/krb5.keytab adh4wk

(now copy /etc/krb5.keytab to all the hosts)

## Slave kerberos2.setup

/etc/kerberosV/krb5.conf (OpenBSD)

mkdir /var/heimdal

/etc/rc.d/kdc -f start

Add line to /etc/inetd.conf:

krb\_prop stream tcp nowait root /usr/libexec/hpropd hpropd

Then restart it:

/etc/rc.d/inetd reload

## Push the database from master with hprop:

Make use you are running hpropd on the slave Kerberos server now. From the master, you run this to push out the database:

# create the hprop key on the master:

kadmin> add --random-key hprop/gigantor.bitsmasher.net

# now export it to the keytab file

kadmin> ext --keytab=/etc/krb5.keytab hprop/gigantor.bitsmasher.net

# transfer the keytab file to the slave:

scp Darwin:/etc/krb5.keytab gigantor:/etc/kerberosV

# now try to send it over to the slave

/usr/libexec/hprop -E kerberos2

Mar 21 22:52:25 gigantor hpropd[26491]: Connection from 172.16.6.189

Mar 21 22:52:26 gigantor hpropd[26491]: Received 19 principals

# add line to the cron on master

\*/15 \* \* \* \* /usr/libexec/hprop -E kerberos2

Needs to add the LDAP principals to the Kerberos database.

To use the GSSAPI mechanism with *slapd*(8) one must create a service key with a principal for *ldap* service within the realm for the host on which the service runs. For example:

kadmin> add –r ldap/gigantor.bitsmasher.net

When *slapd*(8) runs, it must have access to this key. This is generally done by placing the key into a keytab, such as /etc/krb5.keytab.

kadmin> ext --keytab=/etc/openldap/ldap.keytab ldap/gigantor.bitsmasher.net

To use the GSSAPI mechanism to authenticate to the directory, the user obtains a Ticket Granting Ticket (TGT) prior to running the LDAP client. When using OpenLDAP client tools, the user may mandate use of the GSSAPI mechanism by specifying -Y GSSAPI as a command option.

For the purposes of authentication and authorization, *slapd*(8) associates a non-mapped authentication request DN of the form:

uid=<principal>,cn=<realm>,cn=gssapi,cn=auth

Continuing our example, a user with the Kerberos principal kurt@EXAMPLE.COM would have the associated DN:

uid=kurt,cn=example.com,cn=gssapi,cn=auth

and the principal ursula@FOREIGN.REALM would have the associated DN:

uid=ursula,cn=foreign.realm,cn=gssapi,cn=auth

## Kerberos Client Setup

# Kerberos Client Setup on Debian

Install the Kerberized telnet, rsh, rlogin, and ftp like so:

apt-get install krb5-clients krb5-user

Set up an RSH server

sudo apt-get install krb5-rsh-server

Apt-get install libpam-krb5 libpam-smbpass libnss-ldap

Instead of using LDAP PAM as described in [LDAP/PAM](https://wiki.debian.org/LDAP/PAM), set up PAM to authenticate using Kerberos. Install libpam-krb5 and then proceed to /usr/share/doc/libpam-krb5/README.Debian.gz which has great directions to get going. If your Kerberos environment was properly set up above, then you should have logins working nicely.

/etc/ldap/ldap.conf

/etc/libnss-ldap.conf

http://www.debian-administration.org/articles/284

cacert.pem is the file you want to distribute to your clients.

root@netlab1:/etc/ldap# scp /etc/ssl/certs/ca-certificates.crt 172.16.6.185:/etc/ssl/certs/

Test keys:

openssl x509 -in cacert.pem -noout -text

openssl x509 -in cacert.pem -noout -dates

openssl x509 -in cacert.pem -noout -purpose

TLSCACertificateFile /etc/ssl/certs/cacert.pem

TLSCertificateKeyFile /etc/ssl/private/server-key.pem (cat netlab1.privkey.pem netlab1.cert.pem > netlab1-key.pem; mv netlab1-key.pem /etc/ssl/private/server-key.pem)

TLSCertificateFile /etc/ssl/certs/server-cert.pem (netlab1.cert.pem)

 A private key in key.pem.

 A certificate in cert.pem.

 A combined private key and certificate in key-cert.pem .

# Kerberos Client Setup on Ubuntu

<https://help.ubuntu.com/10.04/serverguide/kerberos.html>

sudo apt-get install heimdal-clients

**dpkg-reconfigure krb5-config**

Suggested packages:

heimdal-docs heimdal-kcm

The following packages will be REMOVED:

krb5-admin-server krb5-kdc krb5-user

The following NEW packages will be installed:

heimdal-clients libhdb9-heimdal libkadm5clnt7-heimdal libkadm5srv8-heimdal

libkafs0-heimdal libotp0-heimdal libsl0-heimdal

0 upgraded, 7 newly installed, 3 to remove and 15 not upgraded.

Need to get 543 kB of archives.

After this operation, 602 kB of additional disk space will be used.

Do you want to continue [Y/n]? y

# Kerberized SSH on Linux Hosts

To enable native Kerberos authentication in OpenSSH, make sure the following commands are present in the sshd\_config file (typically found at /etc/ssh):

KerberosAuthentication yes

GSSAPIAuthentication yes

GSSAPICleanupCredentials yes

Be sure to restart the SSH daemon after making these changes.

In the file /etc/ssh/ssh\_config, make sure you have these set like so:

GSSAPIAuthentication yes

GSSAPIDelegateCredentials yes

For passwordless scp:

scp -o GSSAPIAuthentication=yes /etc/krb5.keytab netlab1:/etc

# Windows Kerberos Client Setup

C:\Users\franklin>ksetup /dumpstate

Machine is not configured to log on to an external KDC. Probably a workgroup member

Failed to create Kerberos key: 5 (0x5)

C:\Users\franklin>ksetup /addkdc BITSMASHER.NET kerberos1.bitsmasher.net

Failed to create Kerberos key: 5 (0x5)

Failed to open Kerberos Key: 0x5

Failed /AddKdc : 0xc0000001

C:\Users\franklin>kinit franklin@BITSMASHER.NET

Password for franklin@BITSMASHER.NET:

C:\Users\franklin>klist

Current LogonId is 0:0x86acd

Cached Tickets: (0)

# Kerberized Programs

Using “ksu: instead of “su”

#### Change password on OpenBSD

root@gigantor:/var/run/saslauthd> passwd -K root

root@BITSMASHER.NET's Password:

New password:

Verifying - New password:

Reply from server: Password changed

root@gigantor:/var/run/saslauthd>

#### Change Password on FreeBSD

root@darwin:/etc/ssh> kpasswd

root@BITSMASHER.NET's Password:

New password:

Verifying - New password:

Verify failure

root@darwin:/etc/ssh>

# Kerberized Telnet

http://www.stacken.kth.se/~thn/ktelnet/

# Samba

## Installation of Samba

addprinc -kvno kvno -randkey cifs/lanparty.local.bitsmasher.net

ktadd -e rc4-hmac:normal cifs/lanparty.local.bitsmasher.net

Need to install samba-ldap tools.

FreeBSD: [net/smbldap-tools](http://www.Freshports.org/net/smbldap-tools)

C:\Users\franklin>nltest /dsgetdc:bitsmasher

DC: \\GIGANTOR

Address: \\GIGANTOR

Dom Name: BITSMASHER

The command completed successfully

root@gigantor:/var/log> net rpc join -S gigantor -Uroot

Enter root's password:

Creation of workstation account failed

Unable to join domain BITSMASHER.NET.

smbldap-useradd -t 0 -w gigantor$

root@gigantor:/etc> net rpc testjoin

get\_schannel\_session\_key: could not fetch trust account password for domain 'BITSMASHER.NET'

net\_rpc\_join\_ok: failed to get schannel session key from server GIGANTOR for domain BITSMASHER.NET. Error was NT\_STATUS\_CANT\_ACCESS\_DOMAIN\_INFO

Join to domain 'BITSMASHER.NET' is not valid: NT\_STATUS\_CANT\_ACCESS\_DOMAIN\_INFO

## Adding Samba Users

# adduser username

smbpasswd -a username

mkdir /home/samba/profiles/username

chown username:users /home/samba/profiles/username

# NFSv4 Exports

NFSv4 needs machine credentials for the server and every client, which wants to use the NFSv4 security features. Create the credentials for the nfs-server and all nfs-clients on the Kerberos KDC and distribute the extraced keys with scp to the destination

To verify that you have the /nfs/host key, use the command “klist -e -k /etc/krb5.keytab”:

klist -e -k -t /etc/krb5.keytab

(Shows the “kvno” version number on all the keys. If you don’t have the latest, you will need to ~~run ktadd on the lagging host~~copy the latest /etc/krb5.keytab from nightshade.

Quick Check:

klist -e -k -t /etc/krb5.keytab | grep BIT | cut -f4 -d' '| sort |uniq -c

## Debian NFS Client

If you make changes, be sure to:

sudo service nfs-kernel-server restart

First update the file “/etc/idmapd.conf “:

# line 6: uncomment and change to your domain name  
Domain = local.bitsmasher.net

Now restart “nfs-common”

root@client:~# /etc/init.d/nfs-common restart

Try mounting the filesystem:

mount -t nfs4 -o sec=krb5 lanparty.local.bitsmasher.net:/export/home /home

Now fix /etc/fstab:

lanparty:/export/home /home nfs4 \_netdev,auto 0 0

## NFSv4 Server

You have to specify the server in the keytab and copy it to all machines first:

ank -kvno kvno -randkey nfs/lanparty.local.bitsmasher.net

ktadd -e rc4-hmac:normal nfs/lanparty.local.bitsmasher.net

<https://help.ubuntu.com/community/NFSv4Howto>

Ubuntu: sudo apt-get install nfs-kernel-server nfs-common

root@lanparty:/var/log# mkdir /export

root@lanparty:/var/log# mkdir /export/home

root@lanparty:/var/log# mount --bind /home /export/home

You can configure the directories to be exported by adding them to the /etc/exports file. Two lines for each export:

/export/home 172.16.6.0/24(rw,nohide,insecure,no\_subtree\_check,async,anonuid=65534,anongid=65534)

/export/home gss/krb5(rw,nohide,insecure,no\_subtree\_check,async,anonuid=65534,anongid=65534)

service nfs-kernel-server restart; service idmapd restart

ufw allow from 172.16.6.0/24 to any port 2049

Edit /etc/default/nfs-common:

# Do you want to start the gssd daemon? It is required for Kerberos mounts.

NEED\_GSSD=YES

# Start the idmapd daemon as it needed for NFSv4

NEED\_IDMAPD=YES

Edit /etc/idmapd.conf and set Domain = bitsmasher.net

You need the gss kernel modules on nfs-servers and nfs-clients.

# modprobe rpcsec\_gss\_krb5

Enable SVCGSSD in nfs-kernel-server configuration:

/etc/default/nfs-kernel-server

Now restart:

service /etc/init.d/nfs-kernel-server restart

# LDAP

# Debian LDAP Server

apt-get install slapd samba-doc ldap-utils libnss-ldap phpldapadmin nslcd gnutls-bin nscd

<https://wiki.debian.org/LDAP/OpenLDAPSetup>

Verify the samba schema:

ldapsearch -LLLQY EXTERNAL -H ldapi:/// -b cn=schema,cn=config "(objectClass=olcSchemaConfig)" dn

<https://wiki.debian.org/LDAP/NSS>

apt-get install slapd samba-doc ldap-utils libnss-ldap phpldapadmin nslcd gnutls-bin nscd

Configure LDAP in /etc/nsswitch.conf

view openldap config:

ldapsearch -Y EXTERNAL -H ldapi:/// -b "cn=config"

To modify configuration use the command:

* ldapmodify -Y EXTERNAL -H ldapi:/// -f <file.ldif>

<http://172.16.6.3/phpldapadmin>

nslcd should be restarted if any changes are made to /etc/nss-ldapd.conf.

ldapsearch -x -h localhost -b "dc=bitsmasher,dc=net" "(objectClass=\*)"

netstat -tunlp | grep slapd

# (Directory service) with TLS

ldapsearch -x -d 1

root@netlab1:/var/log# gnutls-cli-debug -p 636 localhost

Resolving 'localhost'...

Connecting to '::1:636'...

Checking for SSL 3.0 support... no

Checking whether %COMPAT is required... yes

Checking for TLS 1.0 support... no

Checking for TLS 1.1 support... no

Checking fallback from TLS 1.1 to... failed

Checking for TLS 1.2 support... no

Checking whether we need to disable TLS 1.0... yes

Server does not support any of SSL 3.0, TLS 1.0 and TLS 1.1

Ldapsearch with –x with use plaintext. This is on port 389.

To use TLS, you omit the –x flag and pass the ldapsearch the –Z or –ZZ command. For this to work you have to have the certificates in place, use ldaps://, and port 636.

For SSL, You omit the –x flag. Without -x, you get SASL binds, but that means you need to have CyrusSASL or something else installed. So far I have not been using CyrusSASL.

For openBSD, install the package:

/usr/ports/sysutils/login\_ldap

Next, edit: /etc/login.conf (check the man page too: man login\_ldap) (this is for freebsd, openbsd)

ldap:\

:auth=-ldap:\

:x-ldap-server=ldap.bitsmasher.net,,ssl:\

:x-ldap-basedn=ou=Users,dc=bitsmasher,dc=net:\

:x-ldap-filter=(&(objectclass=posixAccount)(uid=%u)):\

:tc=default:

#### For FreeBSD LDAP client setup:

<http://www.freebsd.org/doc/en/articles/ldap-auth/client.html>

cd /usr/ports/net/openldap24-client; make install clean

The configuration file for the **OpenLDAP** libraries is /usr/local/etc/openldap/ldap.conf.

BASE dc=bitsmasher,dc=net

URI ldaps://gigantor.bitsmasher.net/

SIZELIMIT 12

TIMELIMIT 15

DEREF never

# TLS configuration

TLS\_CACERT /usr/local/etc/openldap/ssl/ca.crt

TLS\_CIPHER\_SUITE HIGH:MEDIUM:+SSLv3

mkdir /usr/local/etc/openldap/ssl

copy ca.crt from gigantor:/etc/openldap/ssl into the new directory.

++++++++++++++++++++++++++++++++++++++++++++++++++

/etc/openldap/slapd.conf

ldapsearch -x -b "" -s base -LLL supportedSASLMechanisms

http://www.bayour.com/LDAPv3-HOWTO.html#4.5.3.6.Modify%20the%20LDAP%20database%20to%20allow%20simple%20bind%20as%20user.|outline

# Testing OpenLDAP, simple/anonymous bind, with SSL/TLS

# NOTE: ***requires*** the FQDN one connects to matches the one in the certificate.

ldapsearch -H ldaps://localhost -x -b "" -s base -LLL supportedSASLMechanisms

# Testing OpenLDAP, using your Kerberos ticket

ldapsearch -H ldaps://gigantor.bitsmasher.net/ -I -b "" -s base -LLL supportedSASLMechanisms

# Testing OpenLDAP, using your Kerberos ticket, with SSL/TLS

verify that a SSL and TLS works with SASL to by using -ZZ and -H parameters to the above **ldapsearch** command line. The difference between -Z and -ZZ is that the later requires the operation to be successful.

# Testing OpenLDAP, simple user bind, with SSL/TLS

root@gigantor:/var/log> ldapsearch -x -D 'uid=root,ou=Users,dc=bitsmasher,dc=net' -W -b "" -s base -LLL -H ldaps://gigantor.bitsmasher.net/ supportedSASLMechanisms

Enter LDAP Password: (instinct)

++++++++++++++++++++++++++++++++++++++++++++++++

ldapsearch -x -LL -b 'ou=Groups,dc=bitsmasher,dc=net' -s sub

ldapsearch -x -b '' -s base '(objectclass=\*)' namingContexts

ldapsearch -x -b"dc=bitsmasher,dc=net" uid=franklin userPassword

ldapsearch -b 'dc=bitsmasher,dc=net' -x

ldapwhoami -x -D 'uid=franklin,ou=Users,dc=bitsmasher,dc=net' –W

root@gigantor:/var/run/saslauthd> ldapwhoami -x -D 'uid=root,ou=Users,dc=bitsmasher,dc=net' -W

Enter LDAP Password: (instinct)

dn:uid=root,ou=Users,dc=bitsmasher,dc=net

# Do a “kinit” and hit enter at the prompt:

# SASL/GSSAPI authentication started

# SASL Interaction

# Please enter your authorization name:

ldapsearch -I -b"dc=bitsmasher,dc=net" uid=franklin userPassword

[root@gigantor ~]# /usr/local/sbin/smbldap-populate

Populating LDAP directory for domain BITSMASHER (S-1-5-21-1668329231-3356681009-513988110)

(using builtin directory structure)

adding new entry: dc=bitsmasher,dc=net

adding new entry: ou=Users,dc=bitsmasher,dc=net

adding new entry: ou=Groups,dc=bitsmasher,dc=net

adding new entry: ou=Computers,dc=bitsmasher,dc=net

adding new entry: ou=Idmap,dc=bitsmasher,dc=net

adding new entry: uid=root,ou=Users,dc=bitsmasher,dc=net

adding new entry: uid=nobody,ou=Users,dc=bitsmasher,dc=net

adding new entry: cn=Domain Admins,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Domain Users,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Domain Guests,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Domain Computers,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Administrators,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Account Operators,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Print Operators,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Backup Operators,ou=Groups,dc=bitsmasher,dc=net

adding new entry: cn=Replicators,ou=Groups,dc=bitsmasher,dc=net

adding new entry: sambaDomainName=BITSMASHER,dc=bitsmasher,dc=net

Please provide a password for the domain root:

Changing UNIX and samba passwords for root

New password:

Retype new password:

[root@gigantor ~]#

ldapsearch -x -LL -b 'ou=Groups,dc=bitsmasher,dc=net' -s sub

[root@gigantor sbin]# smbldap-groupadd -a -g 1500 fubaria

[root@gigantor sbin]# groupadd -g 1500 fubaria

[root@gigantor samba]# net groupmap add rid=1500 unixgroup="fubaria" ntgroup="fubaria"

adding entry for group fubaria failed!

[root@gigantor share]# smbldap-groupadd -a -g 666 fubaria

dn: cn=fubaria,ou=Groups,dc=bitsmasher,dc=net

objectClass: top

objectClass: posixGroup

objectClass: sambaGroupMapping

cn: fubaria

gidNumber: 666

sambaSID: S-1-5-21-1668329231-3356681009-513988110-2333

sambaGroupType: 2

displayName: fubaria

[root@gigantor smbldap-tools]# smbldap-groupadd -g 1500 fubaria

[root@gigantor smbldap-tools]# smbldap-useradd -w -u 3000 badreligion$

[root@gigantor smbldap-tools]# smbldap-useradd -w -u 3001 alexpc$

[root@gigantor smbldap-tools]# smbldap-useradd -w -u 3002 conky$

ldapsearch -x -LL -b 'ou=Users,dc=bitsmasher,dc=net' -s sub

#smbldap-useradd -a -u 2000 -G 513 -N Franklin -S Diaz -c "Franklin Diaz" franklin

#this one works a little better

smbldap-useradd -a -m -c "franklin" franklin

[root@gigantor smbldap-tools]# smbldap-usershow franklin

dn: uid=franklin,ou=Users,dc=bitsmasher,dc=net

objectClass: top,person,organizationalPerson,inetOrgPerson,posixAccount,shadowAccount,sambaSamAccount

cn: franklin

sn: franklin

givenName: franklin

uid: franklin

uidNumber: 1001

gidNumber: 513

homeDirectory: /home/franklin

loginShell: /bin/ksh

gecos: franklin

sambaLogonTime: 0

sambaLogoffTime: 2147483647

sambaKickoffTime: 2147483647

sambaPwdCanChange: 0

displayName: franklin

sambaSID: S-1-5-21-1668329231-3356681009-513988110-3002

sambaPrimaryGroupSID: S-1-5-21-1668329231-3356681009-513988110-513

sambaLogonScript: logon.bat

sambaHomeDrive: H:

sambaLMPassword: 4A0E0BE6DCE7A2294C995AFC52163EA2

sambaAcctFlags: [U]

sambaNTPassword: AB788F89785B6387B900F5F106D895B9

sambaPwdLastSet: 1363643438

sambaPwdMustChange: 1367531438

userPassword: {SSHA}ncVoQBoBSmLXJikgIQcDAaI0d+hLeVF5

[root@gigantor smbldap-tools]#

[root@gigantor kerberosV]# smbldap-useradd -t 0 -w darwin$

[root@gigantor kerberosV]# net groupmap list

Domain Admins (S-1-5-21-1668329231-3356681009-513988110-512) -> 512

Domain Users (S-1-5-21-1668329231-3356681009-513988110-513) -> 513

Domain Guests (S-1-5-21-1668329231-3356681009-513988110-514) -> 514

Domain Computers (S-1-5-21-1668329231-3356681009-513988110-515) -> 515

Administrators (S-1-5-32-544) -> \_openldap

Account Operators (S-1-5-32-548) -> 548

Print Operators (S-1-5-32-550) -> 550

Backup Operators (S-1-5-32-551) -> 551

Replicators (S-1-5-32-552) -> 552

Users (S-1-5-32-545) -> 2000

fubaria (S-1-5-21-1668329231-3356681009-513988110-2333) -> 666

root@gigantor:/root> ldapsearch -I -Y GSSAPI -H ldaps:/// -b ""

SASL/GSSAPI authentication started

SASL Interaction

Please enter your authorization name: root

ldap\_sasl\_interactive\_bind\_s: Local error (-2)

additional info: SASL(-1): generic failure: GSSAPI Error: Miscellaneous failure (see text) (Server (ldap/gigantor.bitsmasher.net@BITSMASHER.NET) unknown)

root@gigantor:/root>

/var/www/phpldapadmin

## ypldap -- YP map server using LDAP backend

Needs portmap, ypldap, and ypbind daemons. The ypserv is not needed. The config file is /etc/ypldap.conf

<http://www.kernel-panic.it/openbsd/pdc/pdc4.html#pdc-4.3>

root@darwin:/root> domainname bitsmasher.net

root@darwin:/root> cat /etc/defaultdomain

cat: /etc/defaultdomain: No such file or directory

root@darwin:/root> echo "bitsmasher.net" > /etc/defaultdomain

test it with:

# **ypldap –dv**

It should display the ldap users and information and if it don't shows or you get any error don't panic just continue the further steps.

openBSD

cd /usr/ports/sysutils/login\_ldap; make install

update /etc/login.conf

echo ‘bitsmasher.net’ > /etc/defaultdomain

echo ‘bitsmasher.net’ > /etc/yp/ldap.bitsmasher.net

$ sudo vipw

$ sudo tail -1 /etc/master.passwd

+:::::::::/bin/ksh

$ sudo vi /etc/group

$ tail -1 /etc/group

+:::

# More Kerberos Setup

root@gigantor:/root> kadmin -l

kadmin> init BITSMASHER.NET

kadmin> list \*

To enable Kerberos binding to the ~~OpenLDAP~~ Debian server, create a principal ldap/earth.sample.com and add that to the keytab:

kadmin add -r ldap/earth.sample.com

Using ktutil on Debian:

~~ktutil get ldap/earth.sample.com~~

# ktutil

ktutil: rkt /etc/krb5.keytab

ktutil: list

add -policy service -randkey ldap/ldap.bitsmasher.net

root@darwin:/root> kadmin -l

kadmin> add -r ldap/gigantor.bitsmasher.net

kadmin> ext ldap/gigantor.bitsmasher.net

kadmin> add -r ldap/darwin.bitsmasher.net

kadmin> ext ldap/darwin.bitsmasher.net

kadmin> add -r ldap/fubaria.bitsmasher.net

kadmin> ext ldap/fubaria.bitsmasher.net

kadmin> ext ldap/mail.bitsmasher.net

kadmin> ext ldap/www.bitsmasher.net

kadmin> add -r ldap/badreligion.bitsmasher.net

kadmin> ext ldap/badreligion.bitsmasher.net

#### Using Kerberos Authentication with LDAP

You should now be able to use tools, such as ldapsearch, with Kerberos authentication automatically.

root@gigantor:/etc/kerberosV> ldapsearch -b ou=Users,dc=bitsmasher,dc=net '(uid=franklin)'

SASL/GSSAPI authentication started

ldap\_sasl\_interactive\_bind\_s: Other (e.g., implementation specific) error (80)

additional info: SASL(-1): generic failure: Could not open db