FreeBSD Administration - Support #727

Install OpenLDAP Server on FreeBSD

01/18/2016 06:20 PM - Daniel Curtis

Status:	Closed	**Start date:**	01/18/2016
Priority:	Normal	**Due date:**	
Assignee:	Daniel Curtis	**% Done:**	100%
Category:	Directory Server	**Estimated time:**	2.00 hours
Target version:	FreeBSD 9	**Spent time:**	4.00 hours

Description

This is a guide on installing an OpenLDAP server on FreeBSD 9.

Prepare the Environment

- Make sure the system is up to date:

```
pkg update && pkg upgrade
portsnap fetch extract
```

- Install portmaster:

```
pkg install portmaster
pkg2ng
```

Install OpenLDAP Server

- Install the openldap24-server package from the ports tree:

```
portmaster net/openldap24-server
```

  - **NOTE**: Make sure to enable **[X] GSSAPI**, **[X] PPOLICY**, **[X] MEMBEROF**, **[X] DYNLIST**, **[X] DYNGROUP**, **[X] REFINT**, **[X] SHA2**, **[X] SASL**, and **[X] UNIQUE** during the openldap24-server port configuration.

- Edit the OpenLDAP Client config file:

```
vi /usr/local/etc/openldap/ldap.conf
```

  - Change the BASE to your own environment:

```
BASE dc=example,dc=com
URI ldap:// ldaps://

# SIZELIMIT 0 indicates unlimited search size
SIZELIMIT 0
TIMELIMIT 15
DEREF never
```

- Change the default password:

```
slappasswd -h "{SSHA}" >> /usr/local/etc/openldap/slapd.conf
```

- Edit the OpenLDAP Server config file:

```
vi /usr/local/etc/openldap/slapd.conf
```

  - And change as necessary on each server:

```
include /usr/local/etc/openldap/schema/core.schema
include /usr/local/etc/openldap/schema/cosine.schema
include /usr/local/etc/openldap/schema/corba.schema
include /usr/local/etc/openldap/schema/inetorgperson.schema
include /usr/local/etc/openldap/schema/nis.schema
include /usr/local/etc/openldap/schema/collective.schema
include /usr/local/etc/openldap/schema/openldap.schema
include /usr/local/etc/openldap/schema/duaconf.schema
include /usr/local/etc/openldap/schema/dyngroup.schema
include /usr/local/etc/openldap/schema/misc.schema
include /usr/local/etc/openldap/schema/pmi.schema
include /usr/local/etc/openldap/schema/ppolicy.schema

pidfile /var/run/openldap/slapd.pid
argsfile /var/run/openldap/slapd.args

logfile /var/log/slapd.log
loglevel 256

modulepath /usr/local/libexec/openldap
moduleload back_mdb

disallow bind_anon
require authc

database mdb

suffix "dc=example,dc=com"
rootdn "cn=Manager,dc=example,dc=com"

directory /var/db/openldap-data
maxsize 1073741824

access to attrs=userPassword
        by self write
        by anonymous auth
        by dn.base="cn=Manager,dc=example,dc=com" write
        by * none

access to *
        by self write
        by dn.base="cn=Manager,dc=example,dc=com" write
        by * read

# Indices to maintain
index objectClass  eq
index uid          eq
index uidNumber    eq
index uniqueMember eq
index gidNumber    eq
index cn           eq
index memberUid    eq

rootpw {SSHA}A6ia1SPQlY4J5qWBUkPg1qqiwZHrL0mb

overlay         memberof
memberof-dangling    drop
memberof-refint      TRUE
```

- Edit the rc.conf file:

```
vi /etc/rc.conf
```

  - And add the follow to the end of the file:

```
slapd_enable="YES"
slapd_flags='-h "ldapi://%2fvar%2frun%2fopenldap%2fldapi/ ldap://0.0.0.0/"'
slapd_sockets="/var/run/openldap/ldapi"
```

- Start slapd:

```
service slapd start
```

- Test the slapd configuration using an anonymous connection:

```
ldapsearch
```

  Example output, this is expected to cause an error:

```
ldap_bind: Inappropriate authentication (48)
    additional info: anonymous bind disallowed
```

- Test the slapd configuration to demonstrate a successful connection using an authorized user:

```
ldapsearch -D "cn=Manager,dc=example,dc=com"
```

  - *Example output*:

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# search result
search: 3
result: 32 No such object

# numResponses: 1
```

Populate the LDAP Server

- Create the domain template file:

```
vi ~/example.com.ldif
```

  - And add the following:

```
dn: dc=example,dc=com
```

```
objectclass: dcObject
objectclass: organization
o: example
dc: example

dn: cn=Manager,dc=example,dc=com
objectclass: organizationalRole
cn: Manager
```

- To import this file into the server:

```
ldapadd -D "cn=Manager,dc=example,dc=com" -W -f ~/example.com.ldif
```

- To verify the data was imported correctly using the ldapsearch command:

```
ldapsearch
```

  - *Example output*:

```
# extended LDIF
#
# LDAPv3
# base <dc=loga,dc=us> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# example.com
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
o:: bG9nYSA=
dc:: bG9nYSA=

# Manager, example.com
dn: cn=Manager,dc=example,dc=com
objectClass: organizationalRole
cn: Manager

# search result
search: 2
result: 0 Success

# numResponses: 3
# numEntries: 2
```

Add SSL to OpenLDAP

- Install openssl:

```
pkg install openssl
```

- Generate a strong SSL key and a CSR to send for signing by a CA:

```
cd /usr/local/etc
openssl req -sha512 -out ldap.example.com.csr -new -newkey rsa:4096 -nodes -keyout ldap.exampl
e.com.key
```

- Generate the DH parameters:

```
openssl dhparam -out /usr/local/etc/dhparam.pem 4096
```

- Edit the OpenLDAP Server config file:

```
vi /usr/local/etc/openldap/slapd.conf
```

  - And change as necessary on each server:

```
include /usr/local/etc/openldap/schema/core.schema
include /usr/local/etc/openldap/schema/cosine.schema
include /usr/local/etc/openldap/schema/corba.schema
include /usr/local/etc/openldap/schema/inetorgperson.schema
include /usr/local/etc/openldap/schema/nis.schema
include /usr/local/etc/openldap/schema/collective.schema
include /usr/local/etc/openldap/schema/openldap.schema
include /usr/local/etc/openldap/schema/duaconf.schema
include /usr/local/etc/openldap/schema/dyngroup.schema
include /usr/local/etc/openldap/schema/misc.schema
include /usr/local/etc/openldap/schema/pmi.schema
include /usr/local/etc/openldap/schema/ppolicy.schema

TLSCACertificateFile /usr/local/etc/ca-cert.bundle
TLSCertificateFile /usr/local/etc/ldap.example.com.crt
TLSCertificateKeyFile /usr/local/etc/ldap.example.com.key
TLSDHParamFile /usr/local/etc/dhparam.pem

pidfile /var/run/openldap/slapd.pid
argsfile /var/run/openldap/slapd.args

logfile /var/log/slapd.log
loglevel 256

modulepath /usr/local/libexec/openldap
moduleload back_mdb

disallow bind_anon
require authc

database mdb

suffix "dc=example,dc=com"
rootdn "cn=Manager,dc=example,dc=com"

directory /var/db/openldap-data
maxsize 1073741824

access to attrs=userPassword
        by self write
        by anonymous auth
        by dn.base="cn=Manager,dc=example,dc=com" write
        by * none

access to *
        by self write
        by dn.base="cn=Manager,dc=example,dc=com" write
        by * read

# Indices to maintain
index objectClass  eq
index uid          eq
```

```
index uidNumber    eq
index uniqueMember eq
index gidNumber    eq
index cn           eq
index memberUid    eq

rootpw {SSHA}A6ia1SPQlY4J5qWBUkPg1qqiwZHrL0mb

overlay         memberof
memberof-dangling    drop
memberof-refint      TRUE
```

- Set the ownership of the SSL certificate and key to the LDAP user:

```
chown ldap:ldap /usr/local/etc/ldap.example.com.{crt,key}
```

- Edit the rc.conf file:

```
vi /etc/rc.conf
```

  - And add ldaps:/// to the slapd_flags:

```
slapd_flags='-h "ldapi://%2fvar%2frun%2fopenldap%2fldapi/ ldap:/// ldaps:///"'
```

- Restart openldap:

```
service slapd restart
```

Populate the LDAP Server

- Create the People Organizational Unit ldif file:

```
vi ~/people-ou.ldif
```

  - And add the following:

```
dn: ou=People,dc=example,dc=com
objectclass: organizationalUnit
ou: People
```

- Import the People OU file into the server:

```
ldapadd -D "cn=Manager,dc=example,dc=com" -W -f ~/people-ou.ldif
```

- Create the bob user ldif file:

```
vi ~/bob.ldif
```

  - And add the following:

```
dn: cn=Bob Guy,ou=People,dc=example,dc=com
```

```
cn: Bob Guy
givenname: Bob
initials: BG
mail: bob@example.com
objectclass: inetOrgPerson
objectclass: organizationalPerson
objectclass: person
sn: Guy
uid: bob
userpassword: {MD5}X03MO1qnZdYdgyfeuILPmQ==
```

- **NOTE**: The password for bob is **password**.

Install LDAP Web Frontend

Install Nginx

- Install nginx and php56:

```
pkg install nginx php56
```

- Configure the default PHP settings

```
cp /usr/local/etc/php.ini-production /usr/local/etc/php.ini
```

- Create a configuration directory to make managing individual server blocks easier

```
mkdir /usr/local/etc/nginx/conf.d
```

- Edit the main nginx config file:

```
vi /usr/local/etc/nginx/nginx.conf
```

  - And strip down the config file and add the include statement at the end to make it easier to handle various server blocks:

```
#user  nobody;
worker_processes  1;
error_log  /var/log/nginx-error.log;

events {
    worker_connections  1024;
}

http {
    include       mime.types;
    default_type  application/octet-stream;
    sendfile        on;
    keepalive_timeout  65;

    ssl_dhparam /usr/local/etc/dhparam.pem;

    # Load config files from the /etc/nginx/conf.d directory
    include /usr/local/etc/nginx/conf.d/*.conf;
}
```

- Edit /usr/local/etc/php-fpm.conf:

```
vi /usr/local/etc/php-fpm.conf
```

- Make the following changes:

```
listen = /var/run/php-fpm.sock
listen.owner = www
listen.group = www
listen.mode = 0660
```

- Create the nginx SSL certificate bundle:

```
cat /usr/local/etc/ldap.example.com.crt /usr/local/etc/dhparam.pem > /usr/local/etc/ldap.examp
le.com.bundle.crt
```

- Harden the SSL certificate bundle permissions:

```
chown www:www /usr/local/etc/ldap.example.com.bundle.crt
```

- Add the www user to the ldap group:

```
pw user mod www -G ldap
```

- Start and enable nginx and php-fpm at boot:

```
echo 'nginx_enable="YES"' >> /etc/rc.conf
echo 'php_fpm_enable="YES"' >> /etc/rc.conf
service php-fpm start
service nginx start
```

Install LDAP Account Manager

- Install LDAP Acccount Manager:

```
pkg install ldap-account-manager
```

- Add a lam.example.com server block:

```
vi /usr/local/etc/nginx/conf.d/lam.example.com.conf
```

Add the following:

```
server {
    listen       80;
    listen       443 ssl;
    server_name  ldap.example.com;
    root         /usr/local/www/lam;
    access_log   /var/log/ldap.example.com-access.log;
    error_log    /var/log/ldap.example.com-error.log;

    ssl on;
    ssl_certificate /usr/local/etc/ldap.example.com.crt;
    ssl_certificate_key /usr/local/etc/ldap.example.com.key;
```

```
    ssl_ciphers 'AES128+EECDH:AES128+EDH:!aNULL';
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_session_cache  builtin:1000  shared:SSL:10m;
    ssl_stapling on;
    ssl_stapling_verify on;
    ssl_prefer_server_ciphers on;
    ssl_dhparam /usr/local/etc/dhparam.pem;
    add_header Strict-Transport-Security max-age=63072000;
    add_header X-Frame-Options SAMEORIGIN;
    add_header X-Content-Type-Options nosniff;

    allow 192.168.1.0/24;
    deny all;

    location ~ \.php$ {
      fastcgi_split_path_info ^(.+\.php)(/.+)$;
      fastcgi_pass unix:/var/run/php-fpm.sock;
      fastcgi_index index.php;
      fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
      include fastcgi_params;
    }

    location ~ (tmp/internal|sess|config|locale) {
      deny all;
      return 403;
    }
}
```

- Restart nginx:

```
service nginx restart
service php-fpm restart
```

- Open a web browser and go to http://lam.example.com
  1. Click on LAM Configuration -> General Settings, the default master password is **lam**; make sure to change it before going into production.
  2. Go to LAM Configuration -> Edit Server profiles, select any of the profiles and enter the password **lam**. Change the domains from the default to **dc=example,dc=com**. Click **Save** when finished.
  3. Go back to login page and log in as the Manager user

LDAP with SASL

- Install cyrus-sasl and the cyrus-sasl-ldapdb packages:

```
pkg install cyrus-sasl cyrus-sasl-ldapdb
```

- Install cyrus-sasl2-saslauthd from ports:

```
portmaster security/cyrus-sasl2-saslauthd
```

  - **NOTE**: Make sure to enable **[X] HTTPFORM** and **[X] OPENLDAP**.

- Create and edit the saslauthd config file:

```
vi /usr/local/etc/saslauthd.conf
```

  - And the following:

```
ldap_servers: ldaps://ldap.example.com
ldap_search_base: dc=example,dc=com
ldap_filter: (uid=%u)
ldap_bind_dn: cn=Manager,dc=example,dc=com
ldap_pw: SuperSecretPassword
ldap_auth_method: bind
```

- Start saslauthd, set it to use ldap as the authentication mechanism, and enable it at boot:

```
echo 'saslauthd_enable="YES"' >> /etc/rc.conf
echo 'saslauthd_flags="-a ldap"' >> /etc/rc.conf
service saslauthd start
```

- Test the connection between saslauthd and the LDAP servers by running:

```
testsaslauthd -u bob -p password
```

  - *Example output*:

```
0: OK "Success."
```

Kerberos

- Edit the kerberos config file:

```
vi /etc/krb5.conf
```

  - And adjust the parameters as needed:

```
[libdefaults]
    default_realm = EXAMPLE.COM
[realms]
    EXAMPLE.COM = {
        kdc = 192.168.1.10
        kdc = 192.168.1.10
        admin_server = 192.168.1.10
    }
[domain_realm]
    .example.com = EXAMPLE.COM
```

- Create the Kerberos database using the kstash command and enter a Master Key for security:

```
kstash
```

- Initialize the Kerberos Database with the kadmin utility using the -l option.

```
kadmin -l
init EXAMPLE.COM
```

- While still in kadmin, create a principal ‘bob’ using the add command:

```
add bob
```

- Next create an ‘admin’ principal

```
add larry/admin
```

- Access to the administration server is controlled by an ACL file, create this file in the appropriate directory with the following contents:

```
echo 'larry/admin@EXAMPLE.COM all' >> /var/heimdal/kadmind.acl
```

- Then start and enable kerberos at boot:

```
echo 'kdc_enable="YES"' >> /etc/rc.conf
echo 'kadmind_enable="YES"' >> /etc/rc.conf
service kdc start
service kadmind start
```

Resources

- http://loga.us/2014/08/16/openldap-and-multi-master-replication-in-freebsd-part-i-openldap/
- https://www.ldap-account-manager.org/static/doc/manual-onePage/index.html
- http://blog.adimian.com/2014/10/how-to-enable-memberof-using-openldap/
- https://technicalnotes.wordpress.com/2014/04/19/openldap-setup-with-memberof-overlay/
- http://ximalas.info/2014/01/10/ldap-authentication-for-subversions-svnserve-on-freebsd-using-sasl-saslauthd-and-novell-edirectory/
- http://acidx.net/wordpress/2014/06/installing-a-mailserver-with-postfix-dovecot-sasl-ldap-roundcube/

History

#1 - 01/18/2016 07:06 PM - Daniel Curtis

- Description updated

#2 - 01/18/2016 08:13 PM - Daniel Curtis

- Description updated

- Status changed from New to In Progress

- % Done changed from 0 to 30

#3 - 02/21/2016 08:54 PM - Daniel Curtis

- Description updated

#4 - 02/22/2016 05:20 PM - Daniel Curtis

- Description updated

- % Done changed from 30 to 50

#5 - 02/22/2016 07:48 PM - Daniel Curtis

- Description updated

#6 - 02/25/2016 10:25 PM - Daniel Curtis

- Description updated

#7 - 02/26/2016 11:00 PM - Daniel Curtis

- Description updated

#8 - 02/26/2016 11:07 PM - Daniel Curtis

- Description updated

#9 - 02/27/2016 12:08 PM - Daniel Curtis

- Status changed from In Progress to Resolved

- % Done changed from 50 to 100

#10 - 02/27/2016 07:23 PM - Daniel Curtis

- Description updated

#11 - 02/27/2016 07:50 PM - Daniel Curtis

- Description updated

#12 - 02/27/2016 08:48 PM - Daniel Curtis

- Description updated

#13 - 02/29/2016 10:29 PM - Daniel Curtis

- Description updated

#14 - 03/01/2016 07:58 PM - Daniel Curtis

- Description updated

#15 - 03/01/2016 09:18 PM - Daniel Curtis

- Description updated

#16 - 03/12/2016 02:10 PM - Daniel Curtis

- Status changed from Resolved to Closed

#17 - 05/27/2016 12:11 PM - Daniel Curtis

- Description updated

#18 - 05/27/2016 12:21 PM - Daniel Curtis

- Description updated

#19 - 05/27/2016 02:15 PM - Daniel Curtis

- Description updated

#20 - 05/27/2016 02:28 PM - Daniel Curtis

- Description updated