Deploy Dual Stack OpenLDAP Server 2.4 with SASL and TLS on FreeBSD 10.3

Lawrence E. Hughes
2 July 2017

This document is based heavily on a HowTo at https://project.altservice.com/issues/727.

This write-up assumes you have already deployed FreeBSD 10.3 with dual stack networking. You do not need any GUI (e.g. XWindow / Gnome), but it won't hurt if such is installed. This can work in a normal install (e.g. on hardware or VirtualBox) or in an AWS FreeBSD 10.3 instance.

If you have not already installed the FreeBSD ports files (at /usr/ports), do the following two commands now:

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

Now install *portmaster*:

```
# pkg install portmaster
```

Now use *portmaster* to download and install openldap24-server from the ports tree:

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

You will get a chance to select various options. Be sure the following are selected (x = selected by default, A = add by selecting with up/down arrow keys and pressing SpaceBar)

```
[ ] ACCESSLOG
[ ] ACI
[ ] AUDITLOG
[ ] BDB
[ ] COLLECT
[ ] CONSTRAINT
[ ] DDS
[ ] DEBUG
[ ] DEREF
[ ] DNSSRV
[ ] DYNACL
[x] DYNAMIC_BACKENDS
[A] DYNGROUP
[A] DYNLIST
[ ] FETCH
[A] GSSAPI
[ ] LMPASSWD
[x] MDB
[A] MEMBEROF
[ ] ODBC
[ ] OUTLOOK
[ ] PASSWD
[ ] PERL
[A] PPOLICY
[ ] PROXYCACHE
[A] REFINT
[ ] RELAY
[ ] RETCODE
[ ] RLOOKUPS
[ ] RWM
```

```
[A] SASL
[ ] SEQMOD
[A] SHA2
[ ] SHELL
[ ] SLAPI
[ ] SLP
[ ] SMBPWD
[ ] SOCK
[ ] SSSXLV
[x] SYNCPROV
[ ] TCP_WRAPPERS
[ ] TRANSLUCENT
[A] UNIQUE
[ ] VALSORT
```

In order to allow the “c” (country) attribute in user entries, we need to make some minor changes to the default schemas. These are found in /usr/local/etc/openldap/schemas

In core.schema, replace

```
objectclass ( 2.5.6.2 NAME 'country'
        DESC 'RFC2256: a country'
        SUP top STRUCTURAL
        MUST c
        MAY ( searchGuide $ description ) )
```

with

```
objectclass ( 2.5.6.2 NAME 'country'
        DESC 'RFC2256: a country'
        SUP top AUXILIARY
        MAY c  )
```

In cosine.schema , replace

```
objectclass ( 0.9.2342.19200300.100.4.18 NAME 'friendlyCountry'
        SUP country STRUCTURAL
        MUST friendlyCountryName )
```

with

```
objectclass ( 0.9.2342.19200300.100.4.18 NAME 'friendlyCountry'
        SUP country AUXILIARY
        MUST friendlyCountryName )
```

In the FreeBSD package install, the database is in directory /var/db/openldap-data.

If you need to restart at some point, do the following commands as root:

```
# rm –rf /var/db/openldap-data
# service slapd restart
```

Change Directory to /usr/local/etc/openldap:

```
# cd /usr/local/etc/openldap
```

The following steps assume your LDAP BaseDN is `dc=aws,dc=sixscape,dc=net`. Adjust accordingly for your BaseDN.

Edit the file ldap.conf to change the BASE (this affects the command line apps, like ldapadd and ldapsearch).

```
# uemacs ldap.conf
```

```
BASE dc=aws,dc=sixscape,dc=net
URI ldap:// ldaps://

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

Edit the file sldap.conf:

```
# uemacs sldap.conf
```

Replace the existing contents with the following:

```
include         /usr/local/etc/openldap/schema/core.schema
include         /usr/local/etc/openldap/schema/cosine.schema
include         /usr/local/etc/openldap/schema/inetorgperson.schema

# TLSCertificateFile /etc/ssl/bsd.demo.sixscape.net.crt
# TLSCertificateKeyFile /etc/ssl/private.bsd.demo.sixscape.net.key
# TLSCACertificateFile /etc/ssl/gd_bundle-g2-g1.crt

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

logfile /var/log/slapd.log
loglevel 256

# Load dynamic backend modules:
modulepath      /usr/local/libexec/openldap
moduleload      back_mdb
# moduleload    back_ldap

access to dn.base=""
        by * read

access to dn.base="cn=Users"
        by * read

access to dn.base="cn=Subschema"
        by * read

access to attrs=userPassword,userPKCS12
        by self write
        by anonymous auth
        by * none

access to dn.subtree="cn=Users,dc=aws,dc=sixscape,dc=net"
        by self write
```

```
        by users read
        by anonymous read

access to *
        by self write
        by users read
        by anonymous auth
        by * none

########################################################################
# MDB database definitions
########################################################################

database        mdb
maxsize         1073741824
suffix          "dc=aws,dc=sixscape,dc=net"
rootdn          "cn=root,dc=aws,dc=sixscape,dc=net"

# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.
# Mode 700 recommended.
directory       /var/db/openldap-data

# Indices to maintain
index   objectClass     eq

rootpw {SSHA}62QB6+JYZdawKLNIAyeh4H6hK3As81zu
```

To create the encoded root password, use slappasswd and append the output onto slapd.conf (then add rootpw before it)

```
# slappasswd –h "{SSHA}" >> slapd.conf

New password: (enter new password followed by Enter, no echo)
Re-enter new password: (enter new password again, followed by Enter, no echo)
```

Edit the file /etc/rc.conf and add the following new lines at the end of the file

```
# uemacs /etc/rc.conf
```

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

If configuring for dual stack (IPv4 + IPv6), after `"ldap://0.0.0.0"` insert `" ldap://[::]"`

Start slapd:

```
# service slapd start
```

If it doesn't come up, check /var/log/debug.log. If no error is reported, verify slapd is running:

```
# ps –ax | grep slapd

466  -  Is    0:00.01 /usr/local/libexec/slapd -h ldapi://%2fvar%2frun%2fopenl
```

Verify it is listening on port 389 on both IPv4 and IPv6:

```
# netstat -na | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         (state)
tcp6       0      0 2600:1f14:611:b6.389   2001:470:ed3a:10.61009 ESTABLISHED
tcp4       0    128 172.31.3.107.22        66.96.204.73.17396     ESTABLISHED
tcp4       0      0 127.0.0.1.25           *.*                    LISTEN
tcp4       0      0 *.22                   *.*                    LISTEN
tcp6       0      0 *.22                   *.*                    LISTEN
tcp6       0      0 *.389                  *.*                    LISTEN
tcp4       0      0 *.389                  *.*                    LISTEN
udp4       0      0 *.514                  *.*
udp6       0      0 *.514                  *.*
```

Test the server using ldapsearch:

```
# ldapsearch -W -D "cn=root,dc=aws,dc=sixscape,dc=net"
Enter LDAP Password: (enter LDAP password, created above)
# extended LDIF
#
# LDAPv3
# base <dc=aws,dc=sixscape,dc=net> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# search result
search: 2
result: 32 No such object

# numResponses: 1
```

Create initial LDIF file to populate server (on root.ldif):

```
dn: dc=aws,dc=sixscape,dc=net
objectclass: dcObject
objectclass: organization
objectclass: top
dc: aws
description: default
o: Sixscape Communications

dn: cn=Users,dc=aws,dc=sixscape,dc=net
objectclass: person
objectclass: country
objectclass:top
c: SG
cn: Users
sn: Users
```

Import root LDIF file:

```
# ldapadd -W -D "cn=root,dc=aws,dc=sixscape,dc=net" -f root.ldif
Enter LDAP Password:
adding new entry "dc=aws,dc=sixscape,dc=net"

adding new entry "cn=Users,dc=aws,dc=sixscape,dc=net"
```

Search again to verify new entries worked

```
# ldapsearch -W -D "cn=root,dc=aws,dc=sixscape,dc=net"
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=aws,dc=sixscape,dc=net> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# aws.sixscape.net
dn: dc=aws,dc=sixscape,dc=net
objectClass: dcObject
objectClass: organization
objectClass: top
dc: aws
description: default
o: Sixscape Communications

# Users, aws.sixscape.net
dn: cn=Users,dc=aws,dc=sixscape,dc=net
objectClass: person
objectClass: country
objectClass: top
c: SG
cn: Users
sn: Users
# search result
search: 2
result: 0 Success

# numResponses: 4
# numEntries: 3
```

NOTE: Do not try to install phpLDAPadmin via the pkg system – this conflicts with the OpenLDAP-SASL version and will uninstall both the OpenLDAP24-server and OpenLDAP24-client and replace them with non-SASL versions.

You can now add users with any LDAP compliant tool, such as LDAPSoft Admin Tool, SixWallet, etc.

Add SSL/TLS

Obtain SSL/TLS server cert (in this case for bsd.aws.sixscape.net). Private key should be unencrypted. You might use WinSCP to upload the cert and key, and the CA Certs to the ec2-user's home directory.

```
$ ls
DigiCert Global Root CA.crt             bsd.aws.sixscape.net.key
DigiCert SHA2 Secure Server CA.crt      bsd.aws.sixscape.net.pfx
bsd.aws.sixscape.net.crt
```

Concatenate the two CA certs into CA_Certs.pem:

```
$ cat "DigiCert Global Root CA.crt" "DigiCert SHA2 Secure Server CA.crt" >
CA_Certs.pem
```

Copy the CA Certs and the server cert into /etc/ssl and the key to /etc/ssl/private

```
# cp CA_Certs.pem /etc/ssl
# cp bsd.aws.sixscape.net.crt /etc/ssl
# cp bsd.aws.sixscape.net.key /etc/ssl/private
```

Add following lines into /usr/local/etc/openldap/slapd.conf (after #includes is fine):

```
TLSCertificateFile /etc/ssl/bsd.aws.sixscape.net.crt
TLSCertificateKeyFile /etc/ssl/private/bsd.aws.sixscape.net.key
TLSCACertificateFile /etc/ssl/CA_Certs.pem
```

Reboot server

```
# reboot
```

If it doesn't come up, check /var/log/debug.log - If it does, verify slapd is running:

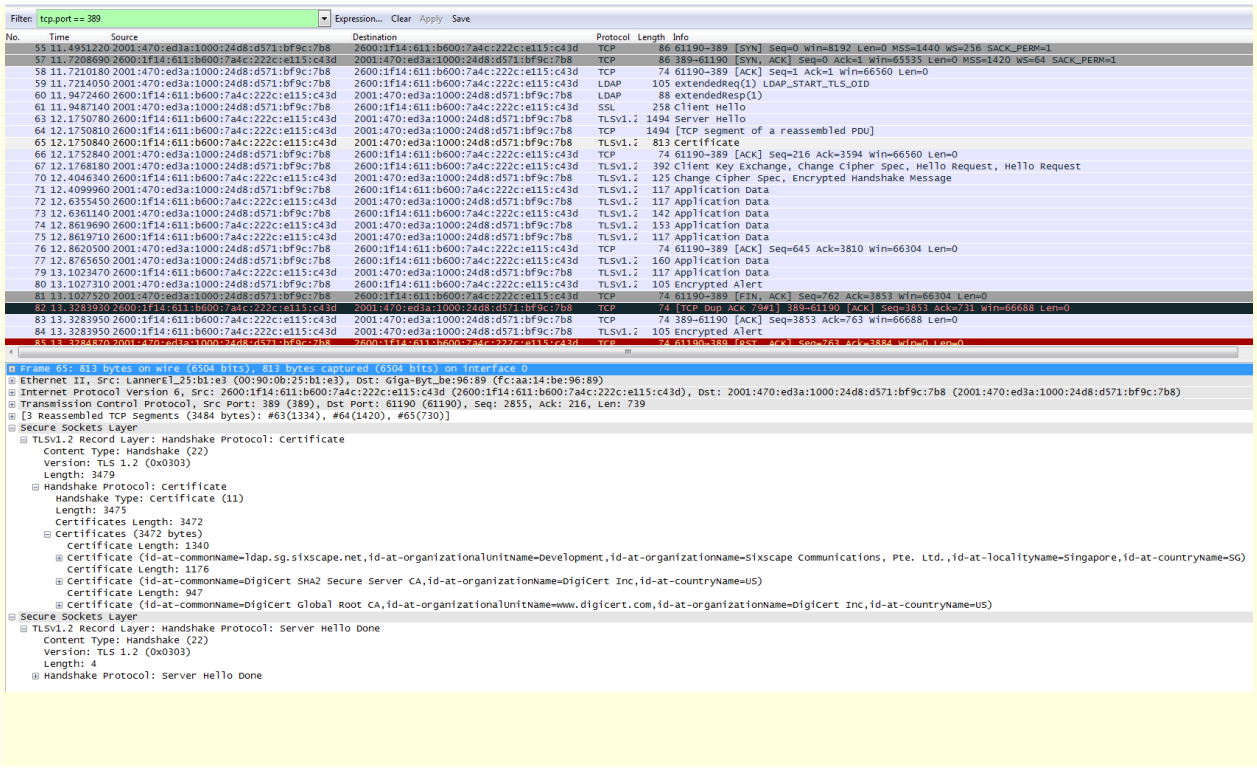
```
# ps -ax | grep slapd

466  -  Is    0:00.01 /usr/local/libexec/slapd -h ldapi://%2fvar%2frun%2fopenl
```

Verify it is listening on port 389 on both IPv4 and IPv6:

```
# netstat -na | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        (state)
tcp6       0      0 2600:1f14:611:b6.389   2001:470:ed3a:10.61009 ESTABLISHED
tcp4       0    128 172.31.3.107.22        66.96.204.73.17396     ESTABLISHED
tcp4       0      0 127.0.0.1.25           *.*                    LISTEN
tcp4       0      0 *.22                   *.*                    LISTEN
tcp6       0      0 *.22                   *.*                    LISTEN
tcp6       0      0 *.389                  *.*                    LISTEN
tcp4       0      0 *.389                  *.*                    LISTEN
udp4       0      0 *.514                  *.*
udp6       0      0 *.514                  *.*
```

Test server by connecting with LDAP over Explicit TLS on port 389. You can verify TLS is working with Wireshark:


Filter: tcp.port == 389
Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
55 11.4951220 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TCP 86 61190→389 [SYN] Seq=0 Win=8192 Len=0 MSS=1440 WS=256 SACK_PERM=1
57 11.7208690 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TCP 86 389→61190 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1420 WS=64 SACK_PERM=1
58 11.7210180 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TCP 74 61190→389 [ACK] Seq=1 Ack=1 Win=66560 Len=0
59 11.7214050 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d LDAP 105 extendedReq(1) LDAP_START_TLS_OID
60 11.9472460 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 LDAP 88 extendedResp(1)
61 11.9487140 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d SSL 258 Client Hello
63 12.1750780 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 1494 Server Hello
64 12.1750810 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TCP 1494 [TCP segment of a reassembled PDU]
65 12.1750840 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 813 Certificate
66 12.1752840 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TCP 74 61190→389 [ACK] Seq=216 Ack=3594 Win=66560 Len=0
67 12.1768180 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TLSv1.2 392 Client Key Exchange, Change Cipher Spec, Hello Request, Hello Request
70 12.4046340 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 125 Change Cipher Spec, Encrypted Handshake Message
71 12.4099960 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TLSv1.2 117 Application Data
72 12.6355450 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 117 Application Data
73 12.6361140 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TLSv1.2 142 Application Data
74 12.8619690 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 153 Application Data
75 12.8619710 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 117 Application Data
76 12.8620500 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TCP 74 61190→389 [ACK] Seq=645 Ack=3810 Win=66304 Len=0
77 12.8765650 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TLSv1.2 160 Application Data
79 13.1023470 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 117 Application Data
80 13.1027310 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TLSv1.2 105 Encrypted Alert
81 13.1027520 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 2600:1f14:611:b600:7a4c:222c:e115:c43d TCP 74 61190→389 [FIN, ACK] Seq=762 Ack=3853 Win=66304 Len=0
82 13.3283930 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TCP 74 [TCP Dup ACK 79#1] 389→61190 [ACK] Seq=3853 Ack=731 Win=66688 Len=0
83 13.3283950 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TCP 74 389→61190 [ACK] Seq=3853 Ack=763 Win=66688 Len=0
84 13.3283950 2600:1f14:611:b600:7a4c:222c:e115:c43d 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 TLSv1.2 105 Encrypted Alert
Frame 65: 813 bytes on wire (6504 bits), 813 bytes captured (6504 bits) on interface 0
Ethernet II, Src: Lannerel_25:b1:e3 (00:90:0b:25:b1:e3), Dst: Giga-Byt_be:96:89 (fc:aa:14:be:96:89)
Internet Protocol Version 6, Src: 2600:1f14:611:b600:7a4c:222c:e115:c43d (2600:1f14:611:b600:7a4c:222c:e115:c43d), Dst: 2001:470:ed3a:1000:24d8:d571:bf9c:7b8 (2001:470:ed3a:1000:24d8:d571:bf9c:7b8)
Transmission Control Protocol, Src Port: 389 (389), Dst Port: 61190 (61190), Seq: 2855, Ack: 216, Len: 739
[3 Reassembled TCP Segments (3484 bytes): #63(1334), #64(1420), #65(730)]
Secure Sockets Layer
TLSv1.2 Record Layer: Handshake Protocol: Certificate
Content Type: Handshake (22)
Version: TLS 1.2 (0x0303)
Length: 3479
Handshake Protocol: Certificate
Handshake Type: Certificate (11)
Length: 3475
Certificates Length: 3472
Certificates (3472 bytes)
Certificate Length: 1340
Certificate (id-at-commonName=ldap.sg.sixscape.net,id-at-organizationalUnitName=Development,id-at-organizationName=Sixscape Communications, Pte. Ltd.,id-at-localityName=Singapore,id-at-countryName=SG)
Certificate Length: 1176
Certificate (id-at-commonName=DigiCert SHA2 Secure Server CA,id-at-organizationName=DigiCert Inc,id-at-countryName=US)
Certificate Length: 947
Certificate (id-at-commonName=DigiCert Global Root CA,id-at-organizationalUnitName=www.digicert.com,id-at-organizationName=DigiCert Inc,id-at-countryName=US)
Secure Sockets Layer
TLSv1.2 Record Layer: Handshake Protocol: Server Hello Done
Content Type: Handshake (22)
Version: TLS 1.2 (0x0303)
Length: 4
Handshake Protocol: Server Hello Done