HOWTO: Setup FTP with TLS support

Contributed by Michael Felt

HOWTO setup AIX FTPD for TLS sessions

In this article I shall resolve a new fallacies about AIX FTP and TLS connections

Fallacy #1: Hard to do

I 'believed' this falacy because of two things: a) reading and undertanding OpenSSL documentation is like trying to pull your teeth and feel good while your are doing it; b) had not bothered to look because I could use sftp/scp instead.

Actually, setting up ftpd - to support ftps requests is quite simple: set two parameters in /etc/ftpd.cnf. The knowledge center has the whole story see ftpd.cnf.htm). Of these only two settings are vital: CERTIFICATE and CERTIFICATE_PRIVATE_KEY.

Not so hard after all...

Let's Keep it Simple!

Since I am working from a self-signed key - I am going to have both the certifiicate and the private key in one file and both paramaters are going to indicate the same file.

If you use an official (i.e., signed by a recognized CA) then you may want to keep the seperate. The process is simple (below I'll show how to see if one or both are in a file).

The key and certificate are just 'plain-text'

The certificate and the private-key need to be in the PEM format. I remember EM as standing for for "Email-Mail" (actually "Enhanced Mail") - which, to me, means something that can be mailed as text. I would have guessed the P stands for "Portable" - but actually, it stands for "Privacy". Repeating all that: for FTPD your need to set CERTIFICATE and CERTIFICATE_PRIVATE_KEY to point at PEM (Privacy Enhanced Mail) format file(s). So, the files are "just-text" - aka 7-bit ASCII codes - but encoded.

Fallacy #2: OpenSSL is hard to use

Actually, I still think this is true - at least there is a reasonable element of truth to it. That said - with adequate instructions (which I give here) - a self-signed PEM file can be made with a single command. There is a dialog - so be prepared to answer some questions!

The ONE command you need for your self-signed certificate is:

```
# openssl req -new -x509 -nodes -set_serial 2005100101 -keyout ftpd.pem -out ftpd.pem -days 365
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to 'ftpd.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:NL
State or Province Name (full name) [Some-State]:N Holland
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:rootvg
Organizational Unit Name (eg, section) []:rootvg.net
Common Name (e.g. server FQDN or YOUR name) []:www.rootvg.net
Email Address []:email@rootvg.net
```

What do we have?

To examine what was made we can look at it 'raw' and use OpenSSL. First the 'raw' PEM.

Notice the two sections clearly bordered by "-----BEGIN..." and "-----END..." lines.

```
# cat ftpd.pem
-----BEGIN PRIVATE KEY-----
MIICdgIBADANBgkqhkiG9w0BAQEFAASCAmAwggJcAgEAAoGBAL59YN+BUUbpMRrS
```

v9DBZcT5fFFbxYcwMPD//wOip7vodi4giD/e8E7VFUWOy4f/+hWD3qhPDmTL76ga

... SNIP to make invalid ...

ampTtFElAUr73DgYPYjg8koMSZBqzeHWnDzu7Cxq9r1KOA9d0/s2GwJAb+IYz5Nz

O8j+qgKe0q6LJQ5q2yqlwWHnFZTZPdqqfR2PnYwbugcht8608tjV77bLlDoKel0M

12mtIbg8fs1iSg==

-----END PRIVATE KEY-----

-----BEGIN CERTIFICATE-----

MIICzTCCAjagAwIBAgIEd4NmRTANBgkqhkiG9w0BAQsFADCBgTELMAkGA1UEBhMC

TkwxEjAQBgNVBAgMCU4gSG9sbGFuZDEPMA0GA1UECgwGcm9vdHZnMRMwEQYDVQQL

... SNIP to make invalid ...

BgkqhkiG9w0BAQsFAAOBgQBoUzEm+C0YJaK+5kz/ZLf1cJYzZ3Dijg3MT5BilpEv

NGw98l28yVweTOIFrTz9qN8kqd7/LwNV1p4cyFitB/oENfTrReUsZgca8wMpocqP

R7S8iTrWV1M09YxJzI0Dv4hia76K/dW0JeO53JOHywL0td/+SDqMMONnWElrM+8r

RQ==

-----END CERTIFICATE-----

What is really in the certificate?

(Or, in case I forgot what was 'in' my PEM file)? Use OpenSSL to read and present the data in a human readable form!

openssl x509 -in ftpd.pem -noout -text

And see:

Certificate:

Data:

Version: 3 (0x2)

Serial Number: 2005100101 (0x77836645)

Signature Algorithm: sha256WithRSAEncryption

Issuer: C=NL, ST=N Holland, O=rootvg, OU=rootvg.net, CN=www.rootvg.net/emailAddress=email@rootvg.net

```
    Validity
        Not Before: Jan 19 12:59:19 2017 GMT
        Not After : Jan 19 12:59:19 2018 GMT
    Subject: C=NL, ST=N Holland, O=rootvg, OU=rootvg.net, CN=www.rootvg.net/emailAddress=email@rootvg.net
    Subject Public Key Info:
        Public Key Algorithm: rsaEncryption
            Public-Key: (1024 bit)
            Modulus:
                00:be:7d:60:df:81:51:46:e9:31:1a:d2:bf:d0:c1:
                65:c4:f9:7c:51:5b:c5:87:30:30:f0:ff:ff:03:a2:
                a7:bb:e8:76:2e:20:88:3f:de:f0:4e:d5:15:45:8e:
                cb:87:ff:fa:15:83:de:a8:4f:0e:64:cb:ef:a8:1a:
                a3:43:4d:70:3d:24:1f:62:1d:ba:de:fc:fd:60:07:
                a6:d2:b3:65:a1:b0:c5:24:6d:3d:9c:19:cf:88:5d:
                c4:b8:4d:34:5c:ce:49:a7:94:55:57:a6:1c:13:ab:
                a1:6d:5a:1c:a5:65:88:50:12:f0:a7:22:33:fa:e8:
                e1:ec:c8:d9:46:f5:cb:66:15
            Exponent: 65537 (0x10001)
    X509v3 extensions:
        X509v3 Subject Key Identifier:
            7F:40:B1:D0:47:64:18:C5:1D:DA:8A:0A:E9:59:4C:73:0A:79:A7:FC
        X509v3 Authority Key Identifier:
            keyid:7F:40:B1:D0:47:64:18:C5:1D:DA:8A:0A:E9:59:4C:73:0A:79:A7:FC

        X509v3 Basic Constraints:
            CA:TRUE
Signature Algorithm: sha256WithRSAEncryption
    68:53:31:26:f8:2d:18:25:a2:be:e6:4c:ff:64:b7:f5:70:96:
    33:67:70:e2:8e:0d:cc:4f:90:62:96:91:2f:34:6c:3d:f2:5d:
    bc:c9:5c:1e:4c:e2:05:ad:3c:fd:a8:df:24:a9:de:ff:2f:03:
    55:d6:9e:1c:c8:58:ad:07:fa:04:35:f4:eb:45:e5:2c:66:07:
    1a:f3:03:29:a1:ca:8f:47:b4:bc:89:3a:d6:57:53:34:f5:8c:
```

49:cc:8d:03:bf:88:62:6b:be:8a:fd:d5:b4:25:e3:b9:dc:93:

87:cb:02:f4:b5:df:fe:48:3a:8c:30:e3:67:58:49:6b:33:ef:

2b:45

So, the last steps are to setup and verify ftps is working.

Fallacy #3: FTP and TLS is hard to use

I know I thought FTPS might be (after all, I already know how to use sftp/scp - why should I care or worry about ftp?) And so, the fallacy lived - at least in my mind. But, it shames me to admit - it is so extremely simple. As a client - add three characters "-s " (remember a space is also a character!). In other words, instead of "ftp localhost" use "ftp -s localhost"

Setup FTP - step 1 - Is it running?

grep ftp /etc/inetd.conf

#ftp stream tcp6 nowait root /usr/sbin/ftpd ftpd -l -d -u 027 -t 900 -T 900

#tftp dgram udp6 SRC nobody /usr/sbin/tftpd tftpd -n

If not, like here, activate it with:

"smitty otherserv"->Super Daemon (inetd)->inetd Subservers->Add an inetd Subserver and enter ftpd->Enter

or:

/usr/sbin/chsubserver
-r inetd -C /etc/inetd.conf -a -v 'ftp' -p 'tcp6' -t 'stream' -w
'nowait' -u 'root' -g '/usr/sbin/ftpd' 'ftpd -l -d -u 027 -t 900 -T
900'

Setup FTP - step 2 - verify it is active:

root@x071:[/tmp]grep ftp /etc/inetd.conf

ftp stream tcp6 nowait root /usr/sbin/ftpd ftpd -l -d -u 027 -t 900 -T 900

#tftp dgram udp6 SRC nobody /usr/sbin/tftpd tftpd -n

and/or

root@x071:[/tmp]netstat -a | grep ftp

tcp 0 0 *.ftp *.* LISTEN

Setup FTPD - step 3 - verify TLS connection

As there is no ftpd.cnf file, ftps (ftp with TLS) cannot be working:

```
root@x071:[/tmp]ls -l /etc/ftp*
-rw------- 1 root system 5 May 27  2016 /etc/ftpusers
```

Again - as a client we will start an ftps request. This is extremely simple: just add '-s'.

```
root@x071:[/tmp]ftp -s localhost
Connected to loopback.
220 x071 FTP server (Version 4.2 Fri Mar 18 06:54:37 CDT 2016) ready.
234 Using authentication type TLSv1
TLS Auth Entered.

ERROR  Error during the hand shake for the control connection
ERROR  Error setting BIO object for the control connection
FTP: Unable to authenticate to Server.
```

Setup FTPD - step 4 - create /tmp/ftpd.cnf and try again!

```
cat - >/etc/ftpd.cnf <<EOF
CERTIFICATE        /tmp/ftpd.pem
CERTIFICATE_PRIVATE_KEY    /tmp/ftpd.pem
EOF
```

And try again...

```
root@x071:[/tmp]ftp -s localhost
```

Connected to loopback.

220 x071 FTP server (Version 4.2 Fri Mar 18 06:54:37 CDT 2016) ready.

234 Using authentication type TLSv1

TLS Auth Entered.

TLS handshake succeeded, though cert had zero depth and Server signed it's own cert!

Certificate:

Data:

Version: 3 (0x2)

Serial Number: 2005100101 (0x77836645)

Issuer: C=NL, ST=N Holland, O=rootvg, OU=rootvg.net, CN=www.rootvg.net/emailAddress=email@rootvg.net

Validity

Not Before: Jan 19 12:59:19 2017 GMT

Not After : Jan 19 12:59:19 2018 GMT

Subject: C=NL, ST=N Holland, O=rootvg, OU=rootvg.net, CN=www.rootvg.net/emailAddress=email@rootvg.net

TLSv1/SSLv3 (DHE-RSA-AES256-GCM-SHA384), 256 bits

Setup FTPD - step 5 - remove SSLv3 ciphers

FTPS is working with TLS! HOWEVER, I do not want TLSv1.0/SSLv3 working - so I am going to add one more line to ftpd.cnf!

print -- "CIPHER_LIST ALL:!MD5:!SHA1" >>/etc/ftpd.cnf

And - volia!

root@x071:[/tmp]ftp -s localhost

Connected to loopback.

220 x071 FTP server (Version 4.2 Fri Mar 18 06:54:37 CDT 2016) ready.

234 Using authentication type TLSv1

TLS Auth Entered.

TLS handshake succeeded, though cert had zero depth and Server signed it's own cert!

Certificate:

Data:

Version: 3 (0x2)

Serial Number: 2005100101 (0x77836645)

Issuer: C=NL, ST=N Holland, O=rootvg, OU=rootvg.net, CN=www.rootvg.net/emailAddress=email@rootvg.net

Validity

Not Before: Jan 19 12:59:19 2017 GMT

Not After : Jan 19 12:59:19 2018 GMT

Subject: C=NL, ST=N Holland, O=rootvg, OU=rootvg.net, CN=www.rootvg.net/emailAddress=email@rootvg.net

TLSv1/SSLv3 (DHE-RSA-AES256-GCM-SHA384), 256 bits

Setup FTPD - step 6 - verify and tune SSL ciphers

Hmm, it seems falacy #2 is - 'little bit' true, because the "Subject:" lines are both the same. However, I know (because I tried) - the command "openssl ciphers ALL:!MD5:!SHA1 -v" shows only lines like:

openssl ciphers ALL:!MD5:!SHA1 -v | grep DHE-RSA-AES256-GCM-SHA384

ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=RSA Enc=AESGCM(256) Mac=AEAD

DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(256) Mac=AEAD

While both lines are TLSv1.2 (and not SSLv3/TLS1.0 as the ftp output would have me believe) I am going to accept only ECDH Key Exchanges (Kx) - and I change ftpd.cnf with:

cat - >/etc/ftpd.cnf <<EOF

CERTIFICATE /tmp/ftpd.pem

CERTIFICATE_PRIVATE_KEY /tmp/ftpd.pem

CIPHER_LIST ALL:!MD5:!SHA1!DH

EOF

Setup AIX FTP with TLS support

I hope this HOWTO helps you out. Questions or comments to Tweet to @rootvgnet