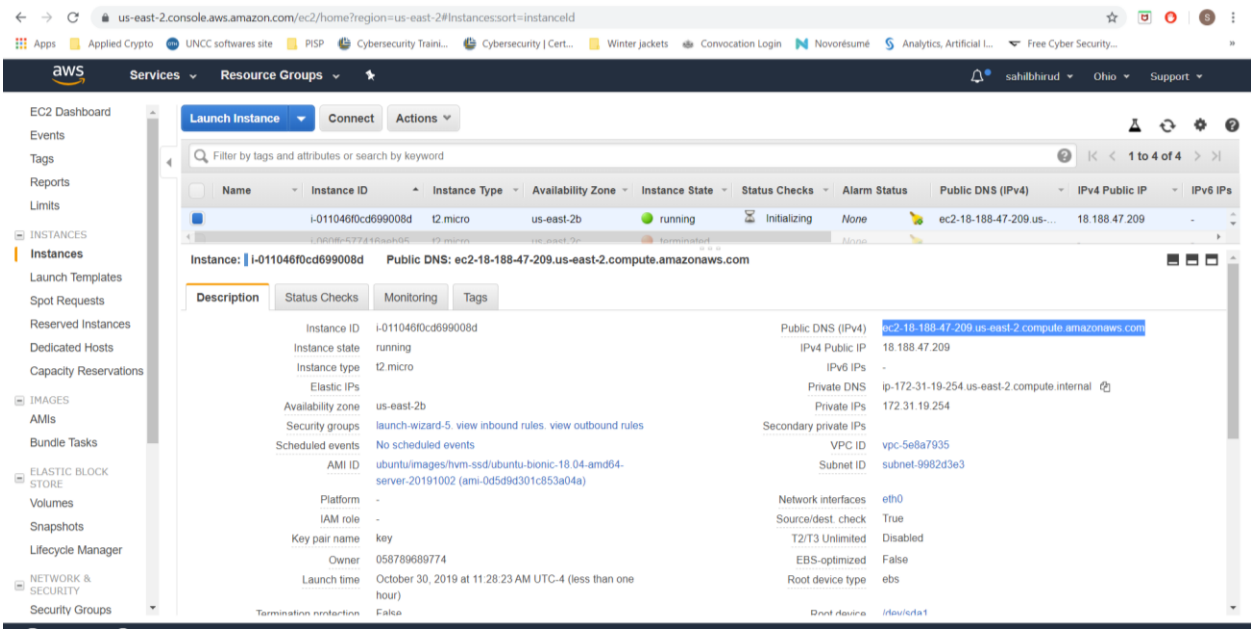
ITIS 6240 – Applied Cryptography

Sahil Bhirud

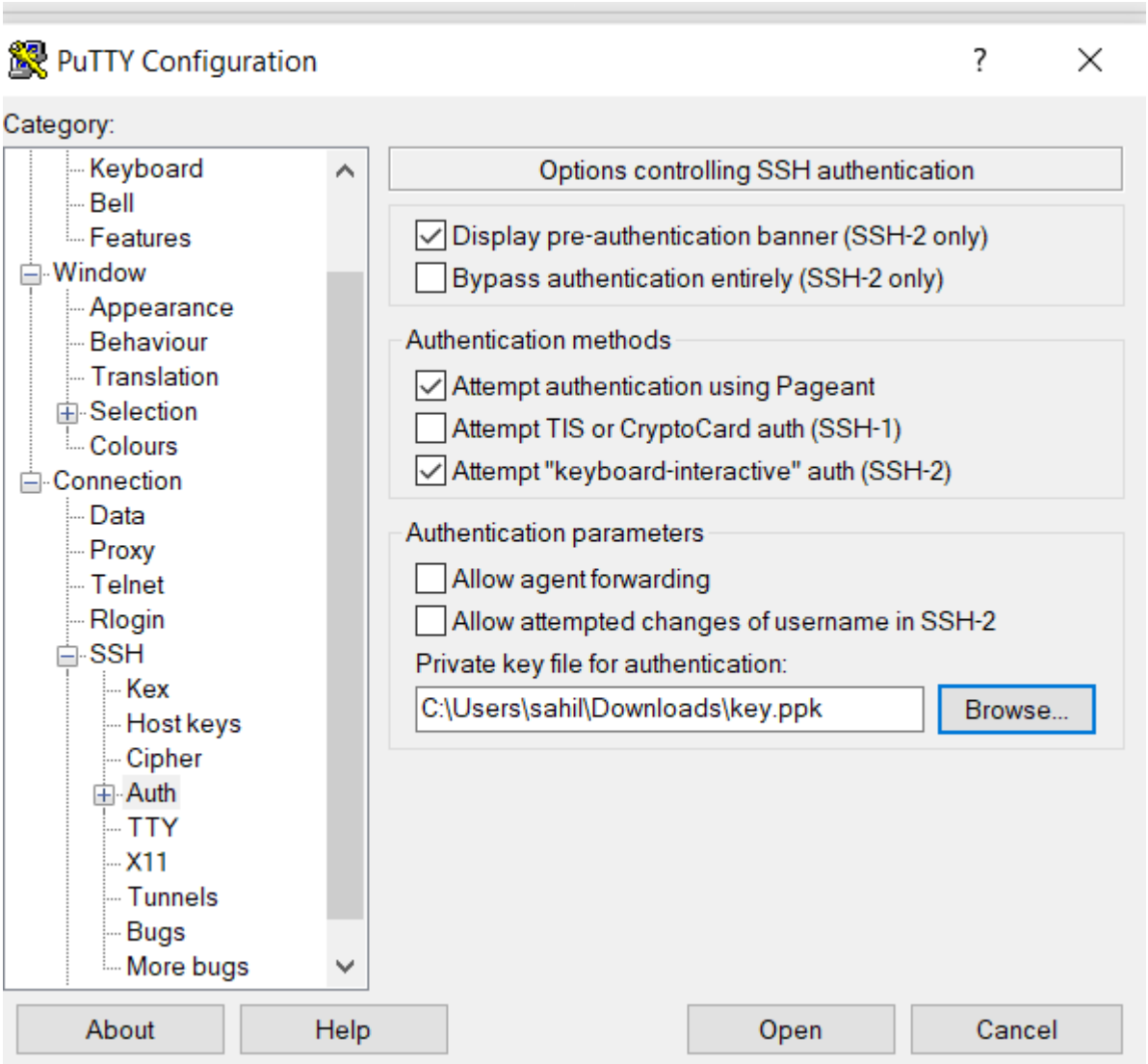
Project 1

I used an AWS instance of Ubuntu 18.04 LTS – 64-bit Linux Server. Following are the steps that I undertook to perform the project and successfully setup an Apache Server with strong cryptography.

1. I launched an AWS EC2 instance of Ubuntu 18.04 64-bit machine and opened up its ports for HTTP, HTTPS and SSH traffic.



2. I used PuTTYgen to convert the “key.pem” key that I got from AWS to Public and Private key files for the Server. Using the Public address of the AWS instance and “.ppk” file, I connected to it using PuTTY.



3. After connecting to the remote machine, I ran the commands which were mentioned in the instructions:

   sudo apt-get update

   sudo apt-get -y install make wget libssl-dev libncurses5-dev gcc

4. Then I installed Apache2 on the server. I did this by running the command:

   sudo apt-get install apache2

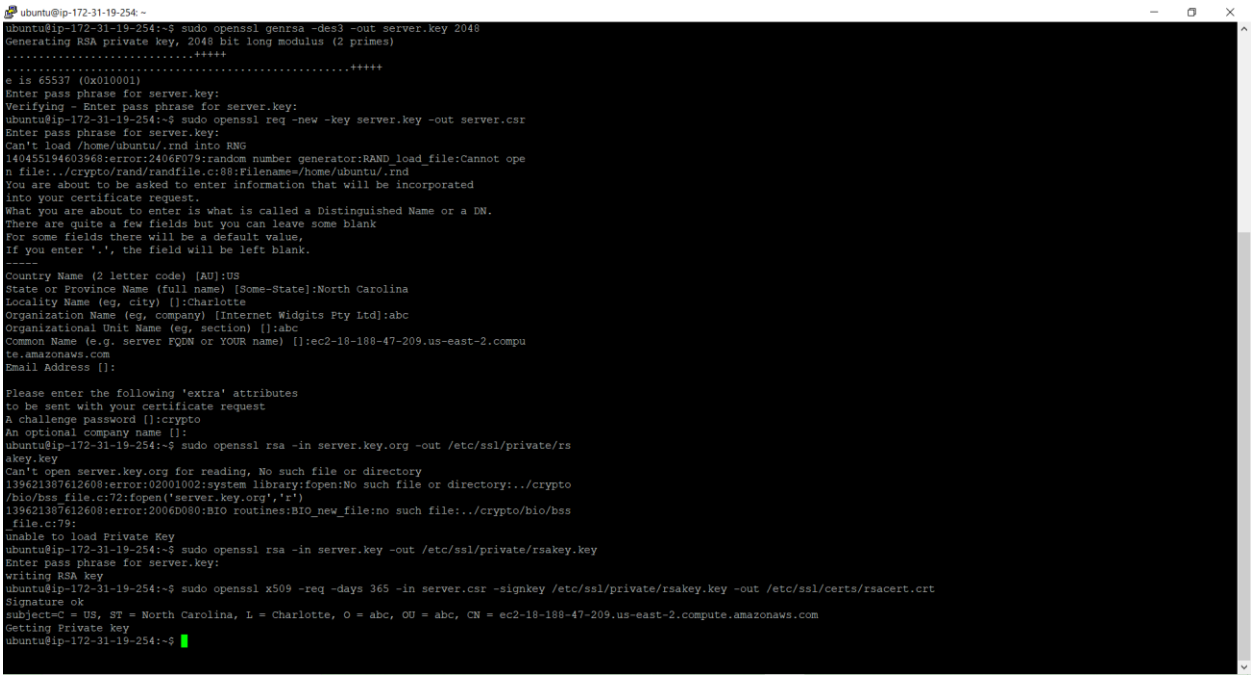
5. To create and install the SSL certificate and key, I used the following commands:

   sudo openssl genrsa -des3 -out server.key 2048

sudo openssl req -new -key server.key -out server.csr

sudo openssl -in server.key -out /etc/ssl/private/rsakey.key

sudo openssl x509 -req -days 365 -in server.csr -signkey /etc/ssl/private/rsakey.key -out /etc/ssl/certs/rsacert.crt



```
ubuntu@ip-172-31-19-254: ~
ubuntu@ip-172-31-19-254:~$ sudo openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
.............................+++++
.......................................................+++++
e is 65537 (0x010001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
ubuntu@ip-172-31-19-254:~$ sudo openssl req -new -key server.key -out server.csr
Enter pass phrase for server.key:
Can't load /home/ubuntu/.rnd into RNG
140455194603968:error:2406F079:random number generator:RAND_load_file:Cannot ope
n file:../crypto/rand/randfile.c:88:Filename=/home/ubuntu/.rnd
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]:US
State or Province Name (full name) [Some-State]:North Carolina
Locality Name (eg, city) []:Charlotte
Organization Name (eg, company) [Internet Widgits Pty Ltd]:abc
Organizational Unit Name (eg, section) []:abc
Common Name (e.g. server FQDN or YOUR name) []:ec2-18-188-47-209.us-east-2.compu
te.amazonaws.com
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:crypto
An optional company name []:
ubuntu@ip-172-31-19-254:~$ sudo openssl rsa -in server.key.org -out /etc/ssl/private/rs
akey.key
Can't open server.key.org for reading, No such file or directory
139621387612608:error:02001002:system library:fopen:No such file or directory:../crypto
/bio/bss_file.c:72:fopen('server.key.org','r')
139621387612608:error:2006D080:BIO routines:BIO_new_file:no such file:../crypto/bio/bss
_file.c:79:
unable to load Private Key
ubuntu@ip-172-31-19-254:~$ sudo openssl rsa -in server.key -out /etc/ssl/private/rsakey.key
Enter pass phrase for server.key:
writing RSA key
ubuntu@ip-172-31-19-254:~$ sudo openssl x509 -req -days 365 -in server.csr -signkey /etc/ssl/private/rsakey.key -out /etc/ssl/certs/rsacert.crt
Signature ok
subject=C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-47-209.us-east-2.compute.amazonaws.com
Getting Private key
ubuntu@ip-172-31-19-254:~$
```

6. I ran the commands mentioned below to enable the settings in the Apache service and restarted the Apache service with the modifications.

   sudo a2enmod ssl

   sudo a2enmod headers

   sudo systemctl restart apache2

```
ubuntu@ip-172-31-19-254:~$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-sig
ned certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-19-254:~$ sudo a2enmod headers
Enabling module headers.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-172-31-19-254:~$
```

7. Now, I modified the default-ssl.conf file located in /etc/apache2/sites-available/ and added the following lines to the text:

   ServerName ec2-18-188-47-209.us-east-2.compute.amazonaws.com
   SSLCertificateFile /etc/ssl/certs/rsacert.crt
   SSLCertificateKeyFile /etc/ssl/private/rsakey.key

```
ubuntu@ip-172-31-19-254: ~
  GNU nano 2.9.3                    /etc/apache2/sites-available/default-ssl.conf

<IfModule mod_ssl.c>
        <VirtualHost _default_:443>
                ServerAdmin webmaster@localhost
                ServerName ec2-18-188-47-209.us-east-2.compute.amazonaws.com

                DocumentRoot /var/www/html

                # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
                # error, crit, alert, emerg.
                # It is also possible to configure the loglevel for particular
                # modules, e.g.
                #LogLevel info ssl:warn

                ErrorLog ${APACHE_LOG_DIR}/error.log
                CustomLog ${APACHE_LOG_DIR}/access.log combined

                # For most configuration files from conf-available/, which are
                # enabled or disabled at a global level, it is possible to
                # include a line for only one particular virtual host. For example the
                # following line enables the CGI configuration for this host only
                # after it has been globally disabled with "a2disconf".
                #Include conf-available/serve-cgi-bin.conf

                #   SSL Engine Switch:
                #   Enable/Disable SSL for this virtual host.
                SSLEngine on

                #   A self-signed (snakeoil) certificate can be created by installing
                #   the ssl-cert package. See
                #   /usr/share/doc/apache2/README.Debian.gz for more info.
                #   If both key and certificate are stored in the same file, only the
                #   SSLCertificateFile directive is needed.
                SSLCertificateFile      /etc/ssl/certs/rsacert.crt
                SSLCertificateKeyFile /etc/ssl/private/rsakey.key

                #   Server Certificate Chain:
                #   Point SSLCertificateChainFile at a file containing the
                #   concatenation of PEM encoded CA certificates which form the
                #   certificate chain for the server certificate. Alternatively
                #   the referenced file can be the same as SSLCertificateFile
                #   when the CA certificates are directly appended to the server
                #   certificate for convinience.
                #SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt

                #   Certificate Authority (CA):

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-A Mark Text   M-] To Bracket   M-▲ Previous   ^B Back
^X Exit       ^R Read File   ^\ Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line M-E Redo   M-6 Copy Text   M-W WhereIs Next M-▼ Next   ^F Forward
```

I also uncommented the following lines in the same file:

***BrowserMatch "SIE [2-6]" ***

***Nokeepalive ssl-unclean-shutdown ***
Downgrade-1.0 force-response-1.0

```
ubuntu@ip-172-31-19-254: ~
  GNU nano 2.9.3                                        /etc/apache2/sites-available/default-ssl.conf

                #   approach is that mod_ssl sends the close notify alert but doesn't wait for
                #   the close notify alert from client. When you need a different shutdown
                #   approach you can use one of the following variables:
                #   o ssl-unclean-shutdown:
                #       This forces an unclean shutdown when the connection is closed, i.e. no
                #       SSL close notify alert is send or allowed to received.  This violates
                #       the SSL/TLS standard but is needed for some brain-dead browsers. Use
                #       this when you receive I/O errors because of the standard approach where
                #       mod_ssl sends the close notify alert.
                #   o ssl-accurate-shutdown:
                #       This forces an accurate shutdown when the connection is closed, i.e. a
                #       SSL close notify alert is send and mod_ssl waits for the close notify
                #       alert of the client. This is 100% SSL/TLS standard compliant, but in
                #       practice often causes hanging connections with brain-dead browsers. Use
                #       this only for browsers where you know that their SSL implementation
                #       works correctly.
                #   Notice: Most problems of broken clients are also related to the HTTP
                #   keep-alive facility, so you usually additionally want to disable
                #   keep-alive for those clients, too. Use variable "nokeepalive" for this.
                #   Similarly, one has to force some clients to use HTTP/1.0 to workaround
                #   their broken HTTP/1.1 implementation. Use variables "downgrade-1.0" and
                #   "force-response-1.0" for this.
                 BrowserMatch "MSIE [2-6]" \
                               nokeepalive ssl-unclean-shutdown \
                               downgrade-1.0 force-response-1.0

        </VirtualHost>
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

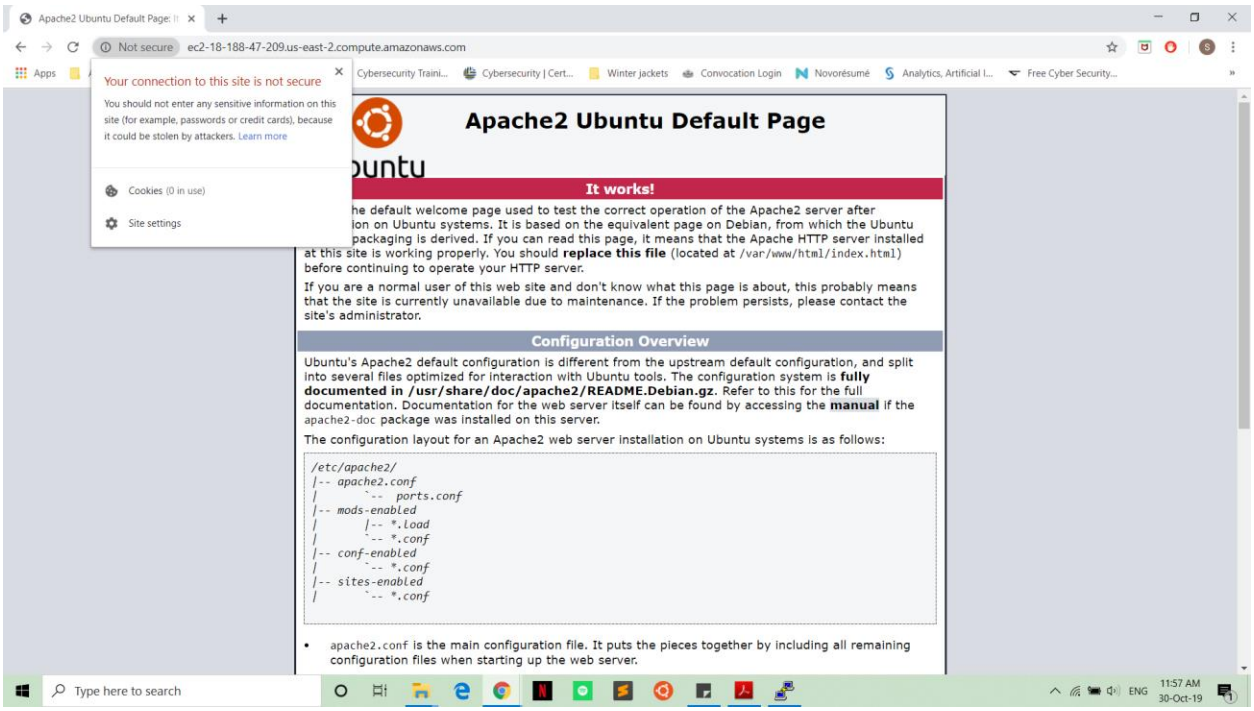
8. After modifying the “default-ssl.conf” file, I enabled it in Apache and restarted the Apache service.

   sudo a2ensite default-ssl

   sudo systemctl restart apache2

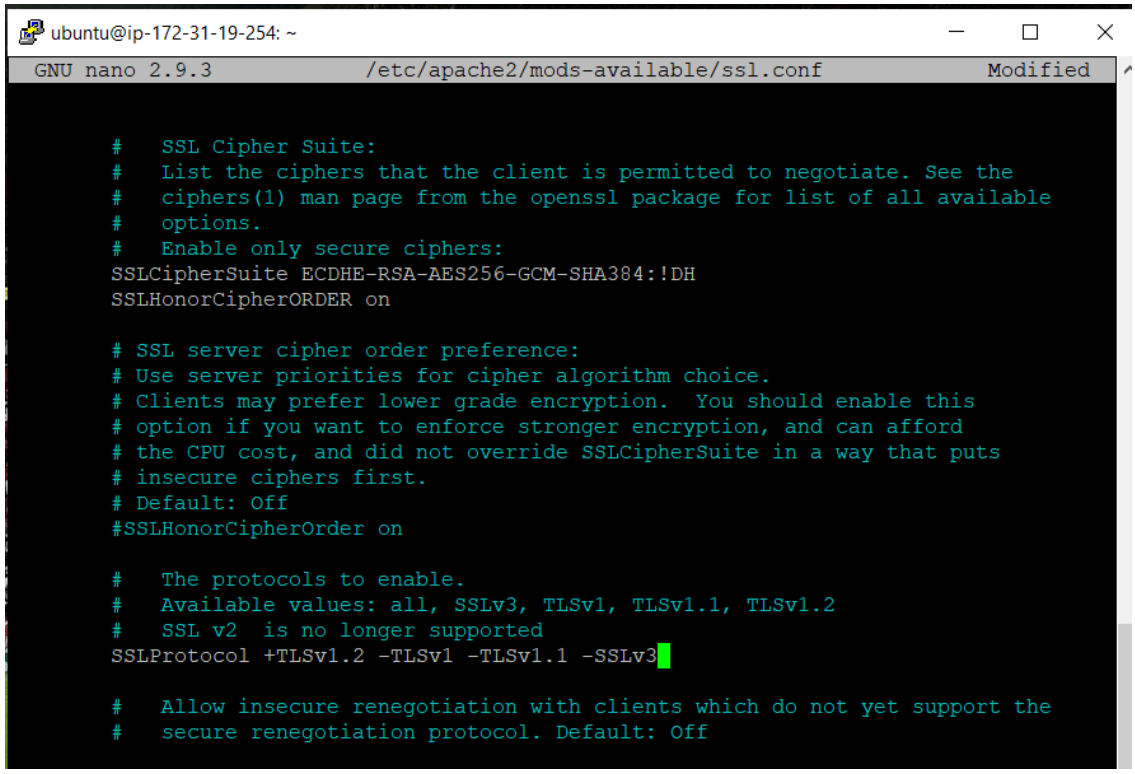
```
ubuntu@ip-172-31-19-254:~$ sudo nano /etc/apache2/sites-available/default-ssl.conf
ubuntu@ip-172-31-19-254:~$ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  systemctl reload apache2
ubuntu@ip-172-31-19-254:~$ sudo systemctl restart apache2
ubuntu@ip-172-31-19-254:~$
```

9. After doing this, I checked if the server was accessible through the browser (Google Chrome).

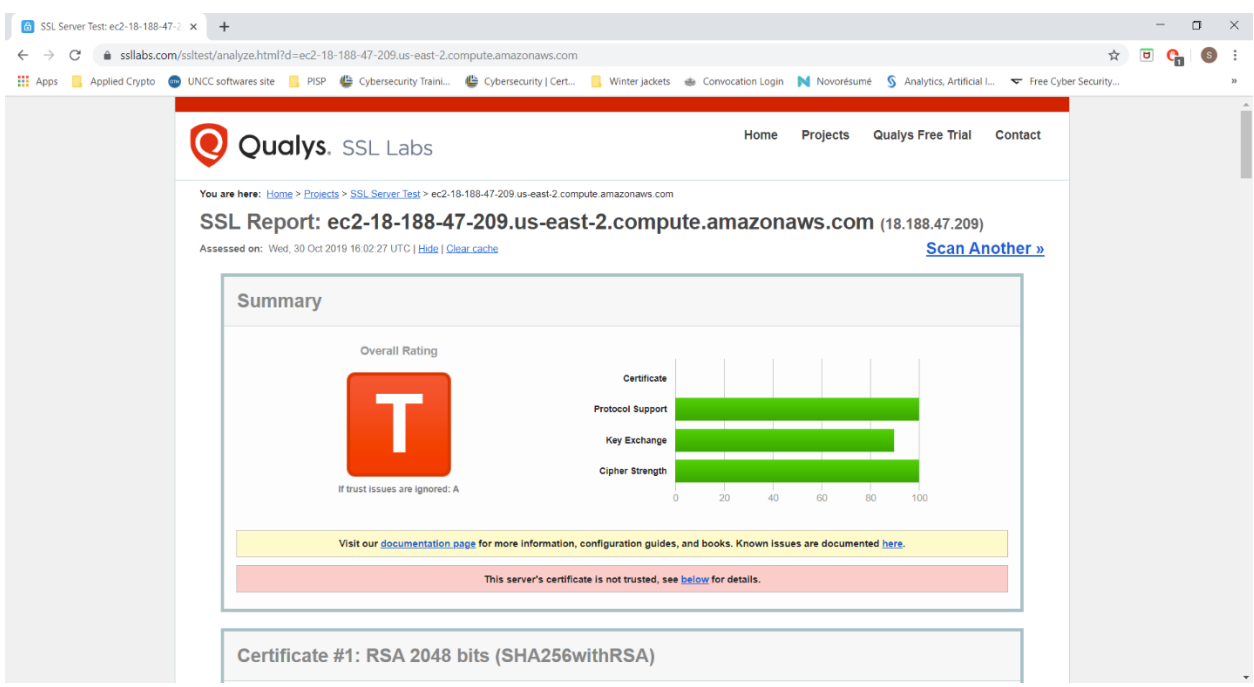


10. To change SSL configuration, I added these lines in the “ssl.conf” file located at /etc/apache2/mods-available/ssl.conf

    SSLCipherSuite ECDHE-RSA-AES256-GCM-SHA384:!DH
    SSLHonorCipherORDER on
    SSLProtocol +TLSv1.2 -TLSv1 -TLSv1.1 -SSLv3



11. I ran tests on both the sites, https://www.ssllabs.com/ssltest/ and https://www.digicert.com/help/ I have attached the screenshots of the reports of these sites respectively.



SSL Server Test: ec2-18-188-47-2

ssllabs.com/ssltest/analyze.html?d=ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Apps | Applied Crypto | UNCC softwares site | PISP | Cybersecurity Traini... | Cybersecurity | Cert... | Winter jackets | Convocation Login | Novorésumé | Analytics, Artificial I... | Free Cyber Security...

Server Key and Certificate #1

Subject	ec2-18-188-47-209.us-east-2.compute.amazonaws.com Fingerprint SHA256: b5b89ff81ef08516ba50ae9aeca126615651d8186613a87257f3f83032e27b13 Pin SHA256: 0gTL6+7Ohm7f2YWm1gGLLneiPnZ2L6LKAgM0GtXM1M=
Common names	ec2-18-188-47-209.us-east-2.compute.amazonaws.com
Alternative names	- INVALID
Serial Number	36a7c1e3d08ae7b4f65cfb565e8498075b0a5c05
Valid from	Wed, 30 Oct 2019 15:51:47 UTC
Valid until	Thu, 29 Oct 2020 15:51:47 UTC (expires in 11 months and 28 days)
Key	RSA 2048 bits (e 65537)
Weak key (Debian)	No
Issuer	ec2-18-188-47-209.us-east-2.compute.amazonaws.com Self-signed
Signature algorithm	SHA256withRSA
Extended Validation	No
Certificate Transparency	No
OCSP Must Staple	No
Revocation information	None
DNS CAA	No (more info)
Trusted	No NOT TRUSTED (Why?) Mozilla Apple Android Java Windows

Additional Certificates (if supplied)

Certificates provided	1 (945 bytes)
Chain issues	None

Certification Paths

SSL Server Test: ec2-18-188-47-2

ssllabs.com/ssltest/analyze.html?d=ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Apps | Applied Crypto | UNCC softwares site | PISP | Cybersecurity Traini... | Cybersecurity | Cert... | Winter jackets | Convocation Login | Novorésumé | Analytics, Artificial I... | Free Cyber Security...

Configuration

Protocols

TLS 1.3	No
TLS 1.2	Yes
TLS 1.1	No
TLS 1.0	No
SSL 3	No
SSL 2	No

For TLS 1.3 tests, we only support RFC 8446.

Cipher Suites

 # TLS 1.2 (we could not determine if the server has a preference)

TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030) ECDH x25519 (eq. 3072 bits RSA) FS	256

Handshake Simulation

Android 4.4.2	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Android 5.0.0	Server sent fatal alert: handshake_failure			
Android 6.0	Server sent fatal alert: handshake_failure			
Android 7.0	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Android 8.0	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Android 8.1	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Android 9.0	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
BingPreview Jan 2015	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS

Type here to search

ENG 12:13 PM 30-Oct-19

SSL Server Test: ec2-18-188-47-2

ssllabs.com/ssltest/analyze.html?d=ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Apps | Applied Crypto | UNCC softwares site | PISP | Cybersecurity Traini... | Cybersecurity | Cert... | Winter jackets | Convocation Login | Novorésumé | Analytics, Artificial I... | Free Cyber Security...

IE 11 / Win 10 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Edge 15 / Win 10 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Edge 16 / Win 10 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Edge 18 / Win 10 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Edge 13 / Win Phone 10 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Java 8u161	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Java 11.0.3	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Java 12.0.1	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
OpenSSL 1.0.1l R	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
OpenSSL 1.0.2s R	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
OpenSSL 1.1.0k R	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
OpenSSL 1.1.1c R	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Safari 6 / iOS 6.0.1	Server sent fatal alert: handshake_failure			
Safari 7 / iOS 7.1 R	Server sent fatal alert: handshake_failure			
Safari 7 / OS X 10.9 R	Server sent fatal alert: handshake_failure			
Safari 8 / iOS 8.4 R	Server sent fatal alert: handshake_failure			
Safari 8 / OS X 10.10 R	Server sent fatal alert: handshake_failure			
Safari 9 / iOS 9 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Safari 9 / OS X 10.11 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Safari 10 / iOS 10 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Safari 10 / OS X 10.12 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Safari 12.1.2 / MacOS 10.14.6 Beta R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Safari 12.1.1 / iOS 12.3.1 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH x25519 FS
Apple ATS 9 / iOS 9 R	RSA 2048 (SHA256)	TLS 1.2 > http/1.1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
Yahoo Slurp Jan 2015	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS
YandexBot Jan 2015	RSA 2048 (SHA256)	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDH secp256r1 FS

SSL Server Test: ec2-18-188-47-2

ssllabs.com/ssltest/analyze.html?d=ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Apps | Applied Crypto | UNCC softwares site | PISP | Cybersecurity Traini... | Cybersecurity | Cert... | Winter jackets | Convocation Login | Novorésumé | Analytics, Artificial I... | Free Cyber Security...

Protocol Details

DROWN	No, server keys and hostname not seen elsewhere with SSLv2 (1) For a better understanding of this test, please read this longer explanation (2) Key usage data kindly provided by the Censys network search engine; original DROWN website here (3) Censys data is only indicative of possible key and certificate reuse; possibly out-of-date and not complete
Secure Renegotiation	Supported
Secure Client-Initiated Renegotiation	No
Insecure Client-Initiated Renegotiation	No
BEAST attack	Mitigated server-side (more info)
POODLE (SSLv3)	No, SSL 3 not supported (more info)
POODLE (TLS)	No (more info)
Zombie POODLE	No (more info)
GOLDENDOODLE	No (more info)
OpenSSL 0-Length	No (more info)
Sleeping POODLE	No (more info)
Downgrade attack prevention	Unknown (requires support for at least two protocols, excl. SSL2)
SSL/TLS compression	No
RC4	No
Heartbeat (extension)	No
Heartbleed (vulnerability)	No (more info)
Ticketbleed (vulnerability)	No (more info)
OpenSSL CCS vuln. (CVE-2014-0224)	No (more info)
OpenSSL Padding Oracle vuln. (CVE-2016-2107)	No (more info)
ROBOT (vulnerability)	No (more info)
Forward Secrecy	Yes (with most browsers) ROBUST (more info)
ALPN	Yes http/1.1
NPN	No
Session resumption (caching)	Yes

SSL Server Test: ec2-18-188-47-2

ssllabs.com/ssltest/analyze.html?d=ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Apps | Applied Crypto | UNCC softwares site | PISP | Cybersecurity Traini... | Cybersecurity | Cert... | Winter jackets | Convocation Login | Novorésumé | Analytics, Artificial I... | Free Cyber Security...

Session resumption (tickets)	Yes
OCSP stapling	No
Strict Transport Security (HSTS)	No
HSTS Preloading	Not in: Chrome Edge Firefox IE
Public Key Pinning (HPKP)	No (more info)
Public Key Pinning Report-Only	No
Public Key Pinning (Static)	No (more info)
Long handshake intolerance	No
TLS extension intolerance	No
TLS version intolerance	No
Incorrect SNI alerts	No
Uses common DH primes	No, DHE suites not supported
DH public server param (Ys) reuse	No, DHE suites not supported
ECDH public server param reuse	No
Supported Named Groups	x25519, secp256r1, x448, secp521r1, secp384r1 (server preferred order)
SSL 2 handshake compatibility	No

HTTP Requests

1 https://ec2-18-188-47-209.us-east-2.compute.amazonaws.com/ (HTTP/1.1 200 OK)

Miscellaneous

Test date	Wed, 30 Oct 2019 16:01:53 UTC
Test duration	34.74 seconds
HTTP status code	200
HTTP server signature	Apache/2.4.29 (Ubuntu)
Server hostname	ec2-18-188-47-209.us-east-2.compute.amazonaws.com

SSL Server Test: ec2-18-188-47-2

ssllabs.com/ssltest/analyze.html?d=ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Apps | Applied Crypto | UNCC softwares site | PISP | Cybersecurity Traini... | Cybersecurity | Cert... | Winter jackets | Convocation Login | Novorésumé | Analytics, Artificial I... | Free Cyber Security...

HSTS Preloading	Not in: Chrome Edge Firefox IE
Public Key Pinning (HPKP)	No (more info)
Public Key Pinning Report-Only	No
Public Key Pinning (Static)	No (more info)
Long handshake intolerance	No
TLS extension intolerance	No
TLS version intolerance	No
Incorrect SNI alerts	No
Uses common DH primes	No, DHE suites not supported
DH public server param (Ys) reuse	No, DHE suites not supported
ECDH public server param reuse	No
Supported Named Groups	x25519, secp256r1, x448, secp521r1, secp384r1 (server preferred order)
SSL 2 handshake compatibility	No

HTTP Requests

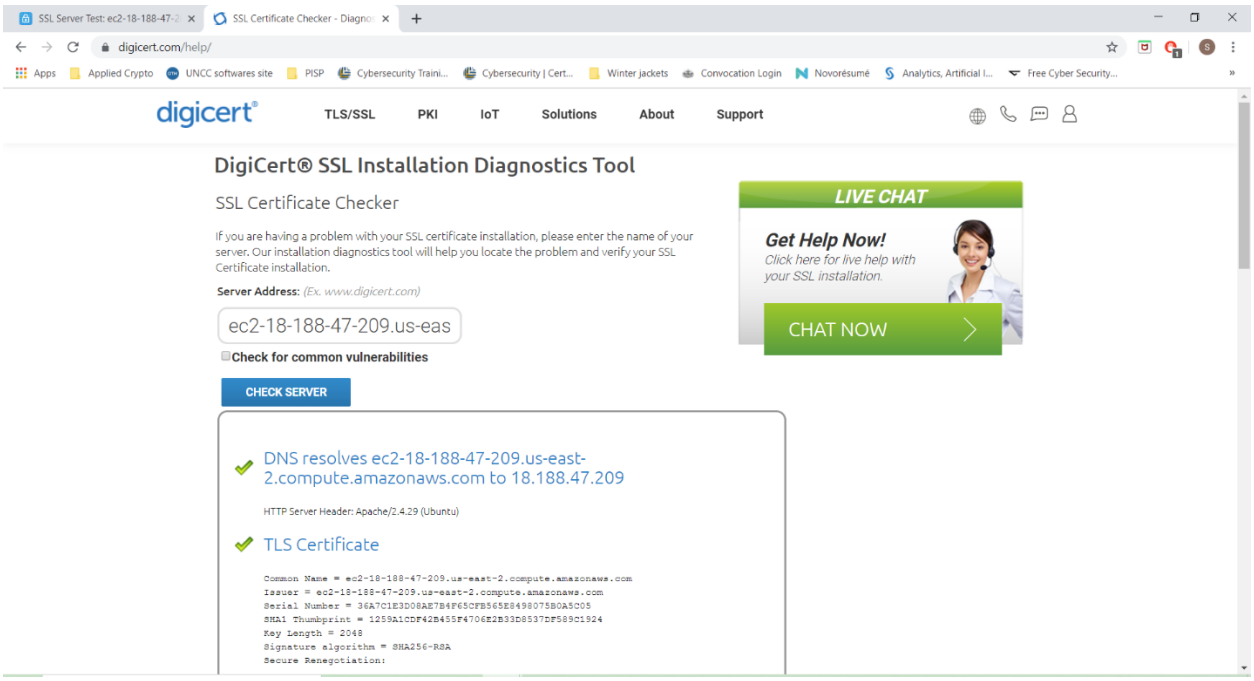
1 https://ec2-18-188-47-209.us-east-2.compute.amazonaws.com/ (HTTP/1.1 200 OK)

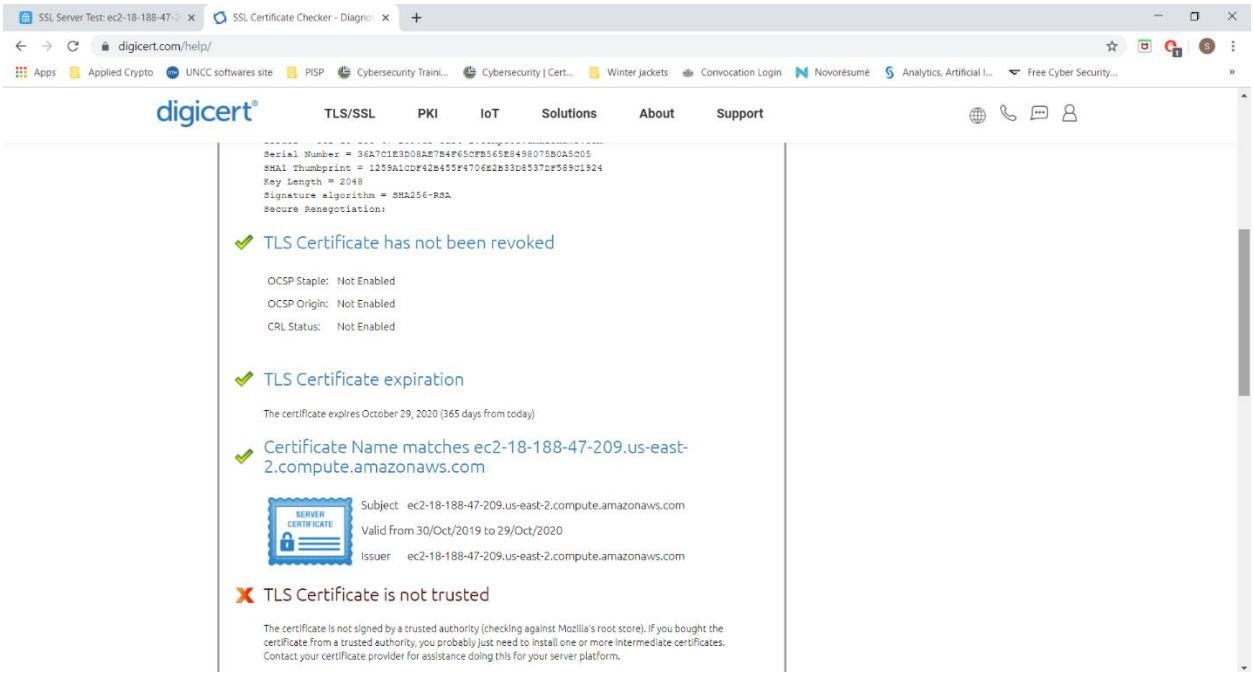
Miscellaneous

Test date	Wed, 30 Oct 2019 16:01:53 UTC
Test duration	34.74 seconds
HTTP status code	200
HTTP server signature	Apache/2.4.29 (Ubuntu)
Server hostname	ec2-18-188-47-209.us-east-2.compute.amazonaws.com

Type here to search

ENG 12:14 PM 30-Oct-19





12. I connected to my server using my SSL command line and the results of the operation is shown below:

    openssl s_client -connect ec2-18-188-47-209.us-east-2.compute.amazonaws.com:443

ubuntu@ip-172-31-19-254: ~

```
ubuntu@ip-172-31-19-254:~$ openssl s_client -connect ec2-18-188-47-209.us-east-2.comput
e.amazonaws.com:443
CONNECTED(00000005)
depth=0 C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-
47-209.us-east-2.compute.amazonaws.com
verify error:num=18:self signed certificate
verify return:1
depth=0 C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-
47-209.us-east-2.compute.amazonaws.com
verify return:1
---
Certificate chain
 0 s:C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-47-
209.us-east-2.compute.amazonaws.com
   i:C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-47-
209.us-east-2.compute.amazonaws.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIDrTCCApUCFDanwePQiue09lz7Vl6EmAdbClwFMA0GCSqGSIb3DQEBCwUAMIGS
MQswCQYDVQQGEwJVUzEXMBUGAlUECAwOTm9ydGggQ2Fyb2xpbmExEjAQBgNVBAcM
CUNoYXJsb3R0ZTEMMAoGAlUECgwDYWJjMQwwCgYDVQQLDANhYmMxOjA4BgNVBAMM
MWVjMi0xOC0xODgtNDctMjA5LnVzLWVhc3QtMi5jb2lwdXRlLmFtYXpvbmF3cy5j
b20wHhcNMTkxMDMwMTUlMTQ3WhcNMjAxMDI5MTUlMTQ3WjCBkjELMAkGAlUEBhMC
VVMxFzAVBgNVBAgMDk5vcnRoIENhcm9saW5hMRIwEAYDVQQHDAlDaGFybG90dGUx
DDAKBgNVBAoMA2FiYzEMMAoGAlUECwwDYWJjMTowOAYDVQQDDDFlYzItMTgtMTg4
LTQ3LTIwOS51cy1lYXN0LTIuY29tcHV0ZS5hbWF6b25hd3MuY29tMIIBIjANBgkq
hkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA4S9IfmpwL1u3Oqjg3m8bXTZxKs4UADCd
7CsV+q+7x24eUcD7geRQ5RsWzm0wCjoY8sjtLm5/Xn+kHEpTcP4PY3ygql2jzbSd
8qCdBcfxQYjJE1PfpEzgU4Juuf4S2rZzVobYyQqSsD42tQre2hiizeO2oNiDsNpT
vjpFywfnCx5+9THJH6DFjdtJST+NJAP/wr0W6NV4Vu3bJOXIHNL39xmjO9DhGCQz
V8iEFGnxTwtHq5udi6zCKLWIcNdDde7vCTkW6nqI12flcLQ0mBh9aJsLa7UnidWe
Quj15BpjayMbyV0H+776ikAFWLNJcmGqsz+GpGpsyrz5bhusOLsfLQIDAQABMA0G
CSqGSIb3DQEBCwUAA4IBAQCUlQzJNinjUmaLXZtod5j9ijoAIbK3KV69wqTGjXAx
T7n4SkWWZqkloKMBmJvpfKDxJskqDCZ9nUBvtRBISOTco7VyTHdk/DgzHi0cYPK
0gcJF6TxngfgUM9zrTiz0COUV3PNcAUH6KSxtNjkTSqVqCL1tHUHp3qRWtg/TgBX
82RRsn9Td0AOTzJOOApbiJNFo7L6/xln6LF0hPj/WmoPIsN8jqu9cGpkhczV7t6+
bfWL6nlxJ2A/eqBViTkk2am9eUXgHCINYdtrAQcldkm3nrkxlO4Qrq7PlI+G/0Yd
Qm3Mipc3182buGvoqMAh8Flvs78FJdZExtibg+7bqYga
-----END CERTIFICATE-----
subject=C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-
47-209.us-east-2.compute.amazonaws.com

issuer=C = US, ST = North Carolina, L = Charlotte, O = abc, OU = abc, CN = ec2-18-188-4
7-209.us-east-2.compute.amazonaws.com

---
No client certificate CA names sent
Peer signing digest: SHA256
Peer signature type: RSA-PSS
```

ubuntu@ip-172-31-19-254: ~

```
---
No client certificate CA names sent
Peer signing digest: SHA256
Peer signature type: RSA-PSS
Server Temp Key: X25519, 253 bits
---
SSL handshake has read 1654 bytes and written 444 bytes
Verification error: self signed certificate
---
New, TLSv1.2, Cipher is ECDHE-RSA-AES256-GCM-SHA384
Server public key is 2048 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
No ALPN negotiated
SSL-Session:
    Protocol  : TLSv1.2
    Cipher    : ECDHE-RSA-AES256-GCM-SHA384
    Session-ID: F4ED56375CD2473687FF8102FDDDAF0DAB7FF62B8AFB70E391E5C84B44950C76
    Session-ID-ctx:
    Master-Key: AC40FCBB56FA4845BFA48891EA4E62E2376421E178537851FA7CBD6D5B0F8C8A9BCA0AB
67DE87FB209A46575A8F0E48A
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    TLS session ticket lifetime hint: 300 (seconds)
    TLS session ticket:
    0000 - c8 ad dd 07 c9 5b e0 a6-a2 f0 9e 30 9f cd 22 62   .....[.....0.."b
    0010 - 0e 3a 98 53 29 7f 8d 07-48 32 68 3a c7 5c 32 ce   .:.S)...H2h:.\2.
    0020 - f2 61 9d cb f8 16 91 c7-b9 df 7c 77 43 e7 2a 81   .a........|wC.*.
    0030 - dc ed c0 a1 6c 93 c6 a1-2f d2 e0 b0 a1 7c 62 6a   ....l.../....|bj
    0040 - 1a f1 98 4c 11 ca 07 10-e6 11 68 80 a5 4c 49 76   ...L......h..LIv
    0050 - 8f ae 5e 17 14 8b ad 98-2b 83 2f ba ad c3 82 08   ..^.....+./.....
    0060 - 3e d3 0c 03 79 e6 9e 31-f7 cf 05 e3 7c eb f9 41   >...y..1....|..A
    0070 - 62 8d ca 6b 6f 8a ed 1b-b1 c8 5f 28 03 45 a1 4f   b..ko....._(.E.O
    0080 - 82 ff 71 93 87 f5 74 b9-11 4c aa 14 ec fc 7f 8d   ..q...t..L......
    0090 - 47 0d 9a a7 2b fb d1 b8-b8 7a 47 52 7e 23 c7 5e   G...+....zGR~#.^
    00a0 - 6e c3 7c ca 66 28 91 07-6a 35 4f f0 45 b0 de de   n.|.f(..j5O.E...
    00b0 - 56 ba 94 86 9a a5 8c 68-1e 9f eb 92 46 67 c2 c1   V......h....Fg..
    00c0 - ba ff b9 01 bf c7 20 df-e8 71 dc fc e9 15 23 35   ...... ..q....#5
    00d0 - 26 b5 95 75 86 df 78 8c-3a 8c 2d 02 11 73 a9 fd   &..u..x.:.-..s..
    00e0 - 9a 6a b7 1e bf c4 7b 74-75 07 26 68 bb 08 74 88   .j....{tu.&h..t.

    Start Time: 1572452778
    Timeout   : 7200 (sec)
    Verify return code: 18 (self signed certificate)
    Extended master secret: yes
---
closed
ubuntu@ip-172-31-19-254:~$
```