

INTRODUCTION TO SSL

- Secure socket layer- encrypts the data stream between the web server and the web client (browser)
- Uses public key cryptography (PKI)
 - Private, public key
- Certificate signed by certificate authority – prevents impersonation attacks
 - SSL X.509
 - Self-signed certificate

OBJECTIVE: creating a self-signed certificate and encrypting the data stream between a web server and a web client (website)

TOOLS

- OpenSSL- general purpose cryptography library

TASK 1 : BASIC ENCRYPTION AND DECRYPTION USING OPEN SSL

- Using various encryption algorithms and modes on the text file and image file given
- Save each one separately and observe

QUESTIONS

1. Perform 2 types of encryptions on text_file.txt using different cipher types, compare the outputs. Mention the commands used for encryption and decryption, provide screenshots of the encrypted text. (cat (filename))

Player ▾ | || ▢ | [] [] []

Terminal File Edit View Search Terminal Help ↑ 10:32 PM ⚙

sophiapineda - cybersec-server@ubuntu: ~/Desktop

```
cybersec-server@ubuntu:~$ pwd
/home/cybersec-server
cybersec-server@ubuntu:~$ cd Desktop
cybersec-server@ubuntu:~/Desktop$ ls
cybersec-server@ubuntu:~/Desktop$ ls
cybersec-server@ubuntu:~/Desktop$ ls
image.bmp  Text_file.txt
cybersec-server@ubuntu:~/Desktop$ openssl enc -aes-128-cbc -e -in Text_file.txt
-out Encrypted.enc -K 11223344556677889900 -iv 0000
cybersec-server@ubuntu:~/Desktop$ cat Encrypted.enc
BxT P</xL [가]xr [S%#D# [v [~ [R/Nu [BS* [
/'gor" [G] ≡ ^L [tL Wa [ [6 [ [M( [j86A [p [ [ # [ 'r [0 [Yp6Y
1 [B2- [^4J] [ $ &w [ [w [Z&^t [41- [ $ [q
w [UL [1 [jYpP [A^0.
h [Z [D [({ [L [u [m [ [; [y [q [ [h [r [X [
* [~ [ [n [ [M [ [e [ [r [g [ [Q [p [8 [k [ \ [z [ [55G [ [ [ > [ " [6 [g [A [ [
< [ [ [ = [r [ , [ 'r [ [ [S4 [ = [ [ Y [ . [ [E [e [ : [5 [E [K [ > [C [o [g [ [ @ [ [ < [ / [ [ $ [ [U [4 [f [x [v [
3
] [~ [ ! [ [D [o [ ' [ [ + [ - [ [C [ [Ty [ *44 [sPu [ = [ ! [ [m
[ [ @ [ [n [ [g [ [ [M [ = [ [810 [
} [ [ ? [ ( [ [z [J [ # [m [ [Y [ [N [o [ L [ [ [ ; [ etF [ \ [ [Q [ [ 歟 [ [ [ '2 [ [ ' [E [o [ + [ ^ [
b [J [ [7 [ [ // [ [ ! [ [ - [3 [ [T [ ( [i [4 [ [ ] [u [I [o [k [o [q [R [8 [
```

Videos

Downloads

Network

Browse Network

Connect to Server

CyberSec-Server-2022 (2) - VMware Workstation 16 Player (Non-commercial use only)

Player | [Icons] | 10:33 PM

Terminal File Edit View Search Terminal Help

sophiapineda - cybersec-server@ubuntu: ~/Desktop

```
cybersec-server@ubuntu:~/Desktop$ openssl enc -aes-128-cbc -e -in Encrypted.enc -out Encrypted.enc -K 11223344556677889900 -iv 0000
cybersec-server@ubuntu:~/Desktop$ cat Encrypted.enc
cybersec-server@ubuntu:~/Desktop$ openssl enc -aes-128-cbc -e -in Encrypted.enc -out Decrypted.txt -K 11223344556677889900 -iv 0000
cybersec-server@ubuntu:~/Desktop$ cat Decrypted.txt
```

- Repeating patterns
 - Scrambled letters
2. Perform 2 types of encryptions on image.bmp using different cipher types, compare the outputs. Mention the commands used for encryption and decryption, provide screenshots of the encrypted text. (sudo nano (filename))

Player ▾ | [Pause] [Full Screen] [Close]

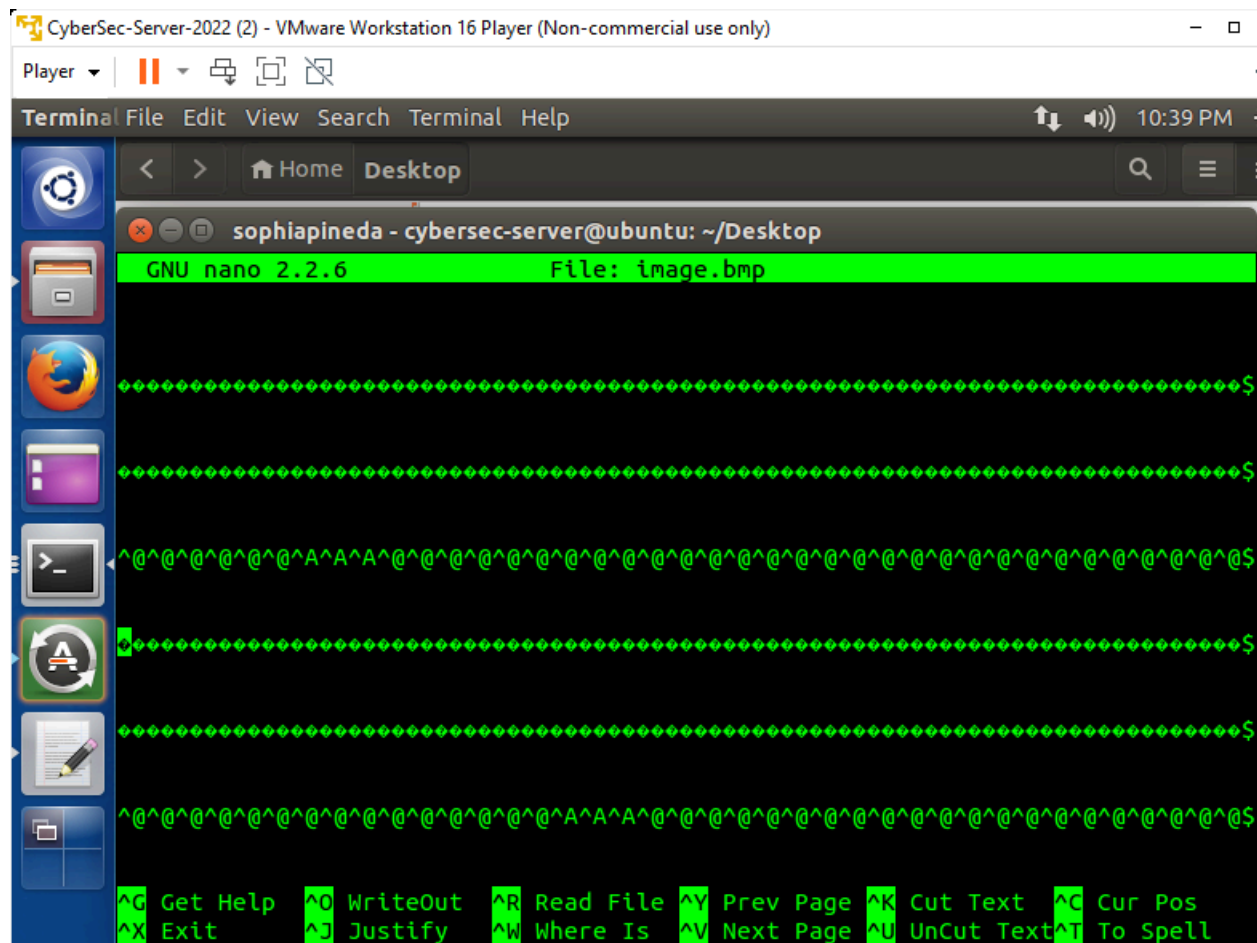
Terminal File Edit View Search Terminal Help [Volume] [Mute] 10:38 PM

sophiapineda - cybersec-server@ubuntu: ~/Desktop

```
cybersec-server@ubuntu:~$ pwd
```

sophiapineda - cybersec-server@ubuntu: ~/Desktop

cybersec-server@ubuntu:~/Desktop\$



- Unable to decrypt
- Question marks
- Hashes?

TASK 2: BECOME A CERTIFICATE AUTHORITY

- Becoming a root CA to issue others certificates (servers)- self signed

STEPS:

- Create directory
- Copy file and move it into folder
- Modify file
- Type in command to make the certificate
- Type in command to decrypt content

SCREENSHOTS

1. Command to generate self-signed certificate for the CA

CyberSec-Server-2022 (2) - VMware Workstation 16 Player (Non-commercial use only)

Player | [Icons] | 10:58 PM

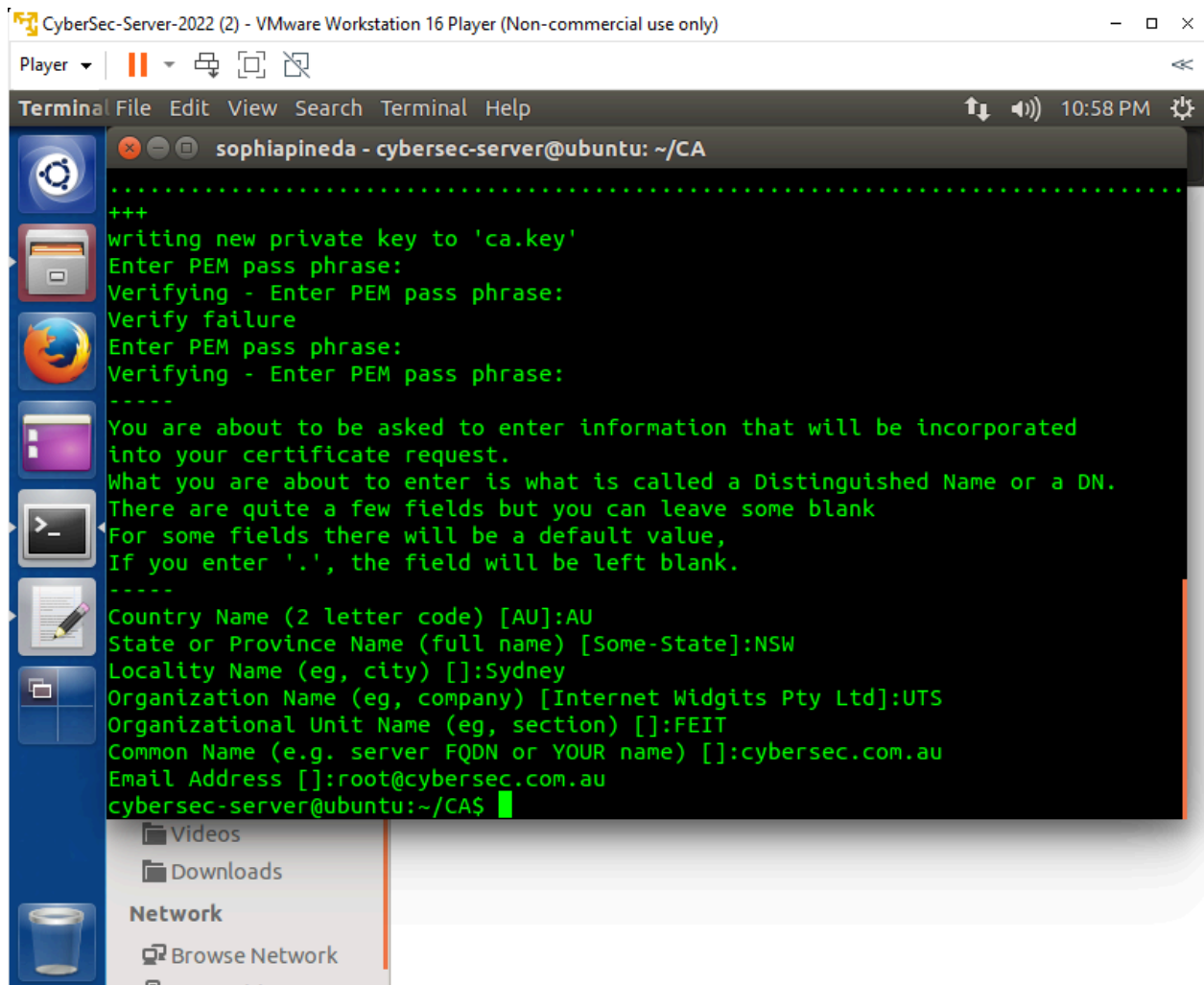
Terminal File Edit View Search Terminal Help

sophiapineda - cybersec-server@ubuntu: ~/CA

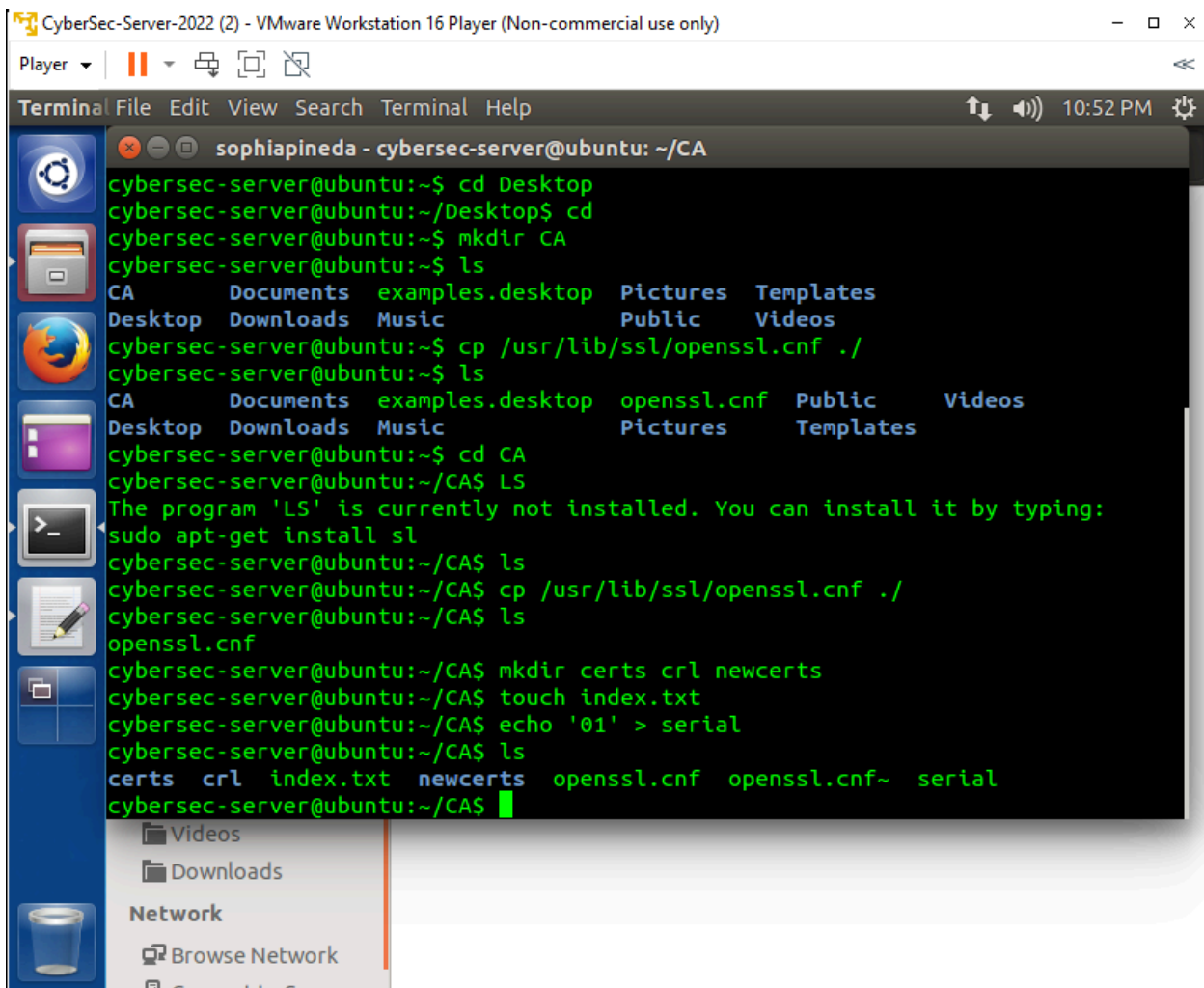
```
cybersec-server@ubuntu:~$ ls
CA      Documents  examples.desktop  Pictures  Templates
Desktop Downloads  Music             Public    Videos
cybersec-server@ubuntu:~$ cp /usr/lib/ssl/openssl.cnf ./
cybersec-server@ubuntu:~$ ls
CA      Documents  examples.desktop  openssl.cnf  Public    Videos
Desktop Downloads  Music             Pictures     Templates
cybersec-server@ubuntu:~$ cd CA
cybersec-server@ubuntu:~/CA$ ls
The program 'ls' is currently not installed. You can install it by typing:
sudo apt-get install ls
cybersec-server@ubuntu:~/CA$ ls
cybersec-server@ubuntu:~/CA$ cp /usr/lib/ssl/openssl.cnf ./
cybersec-server@ubuntu:~/CA$ ls
openssl.cnf
cybersec-server@ubuntu:~/CA$ mkdir certs crl newcerts
cybersec-server@ubuntu:~/CA$ touch index.txt
cybersec-server@ubuntu:~/CA$ echo '01' > serial
cybersec-server@ubuntu:~/CA$ ls
certs  crl  index.txt  newcerts  openssl.cnf  openssl.cnf~  serial
cybersec-server@ubuntu:~/CA$ openssl req -new -x509 -keyout ca.key -out ca.crt -
config openssl.cnf
Generating a 2048 bit RSA private key
.....+++
```

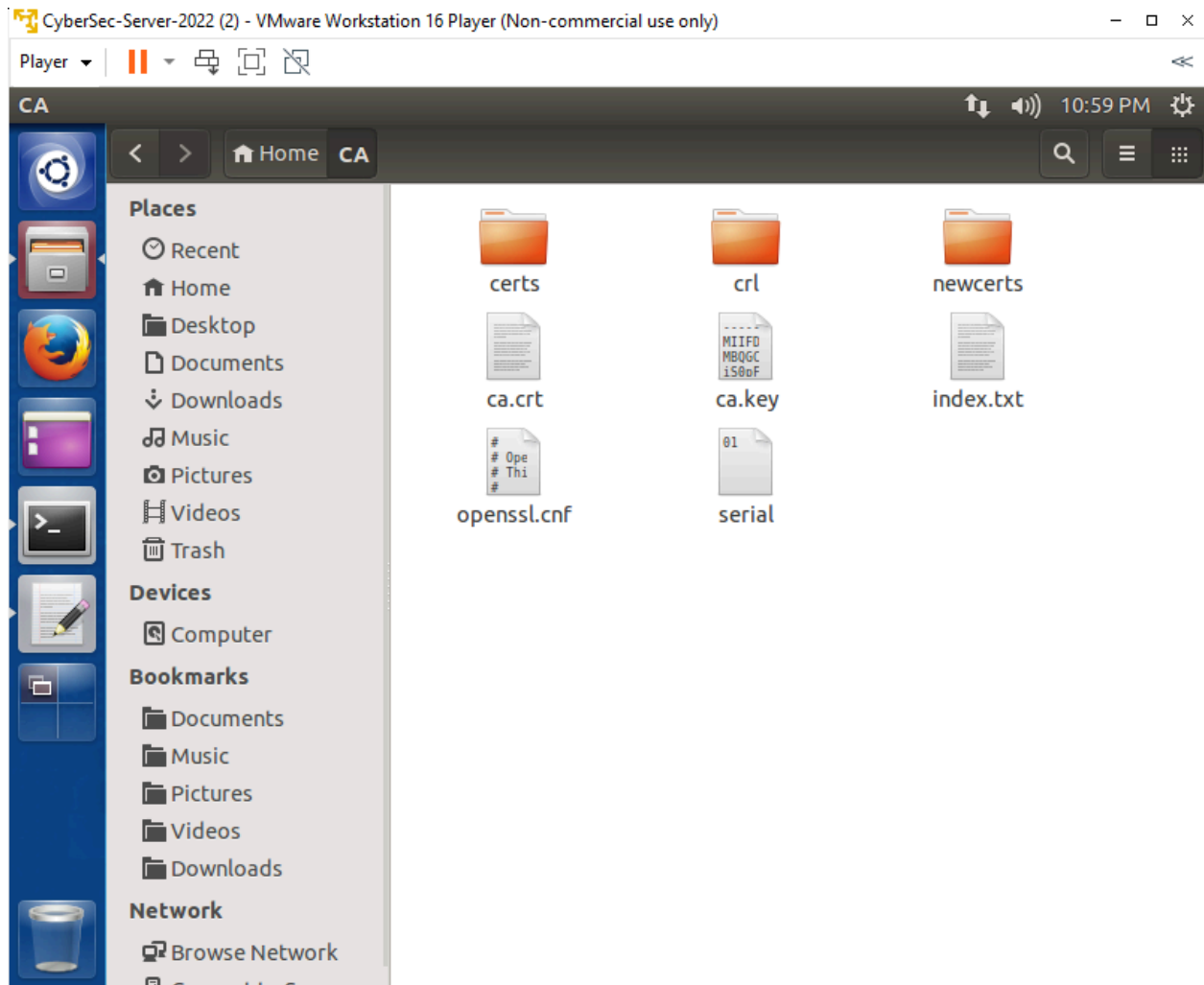
Videos
Downloads
Network
Browse Network

2. Info given for generating the self-signed certificate



3. List of files created





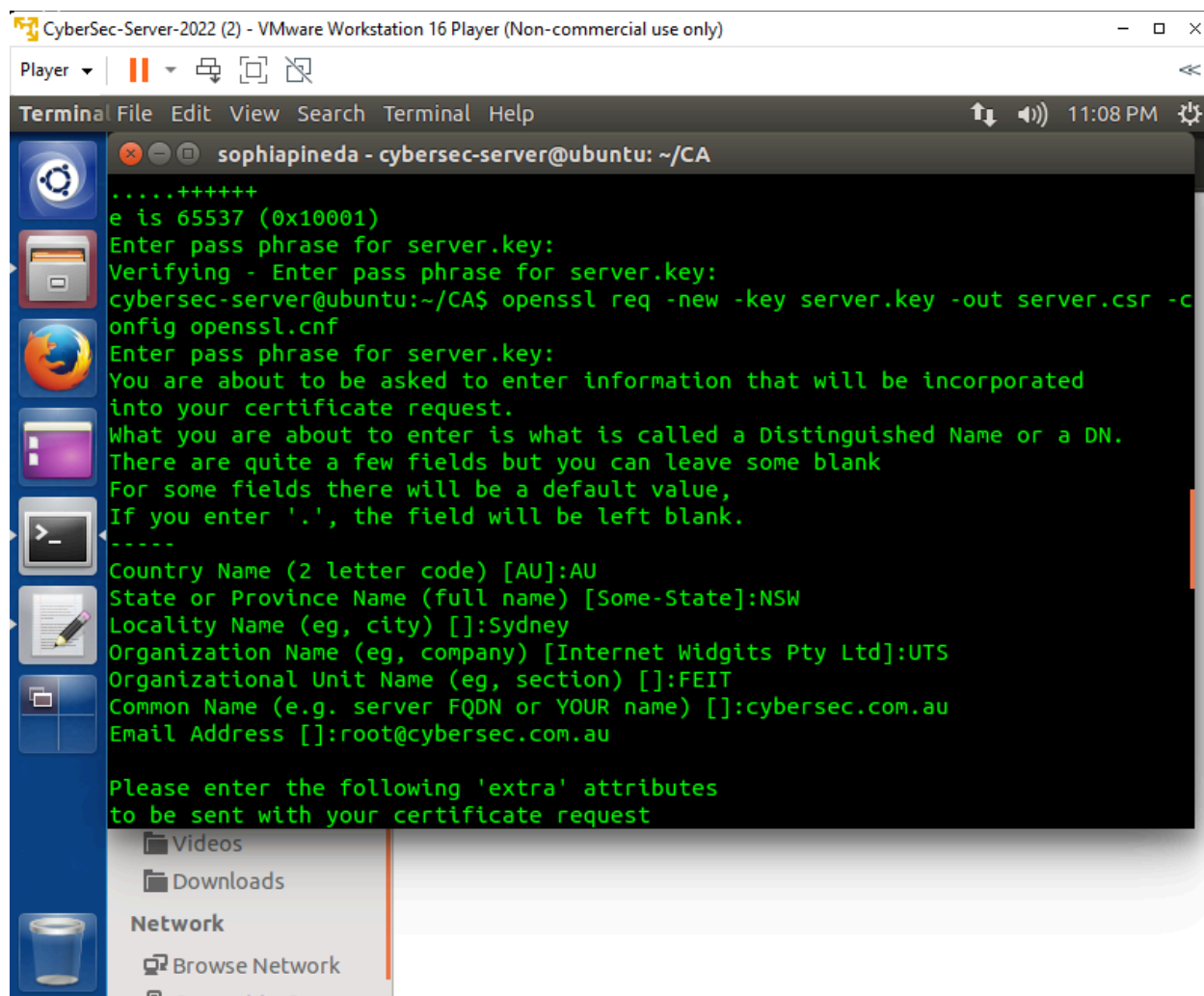
TASK 3: CREATE A CERTIFICATE FOR CYBERSEC.COM.AU

STEPS:

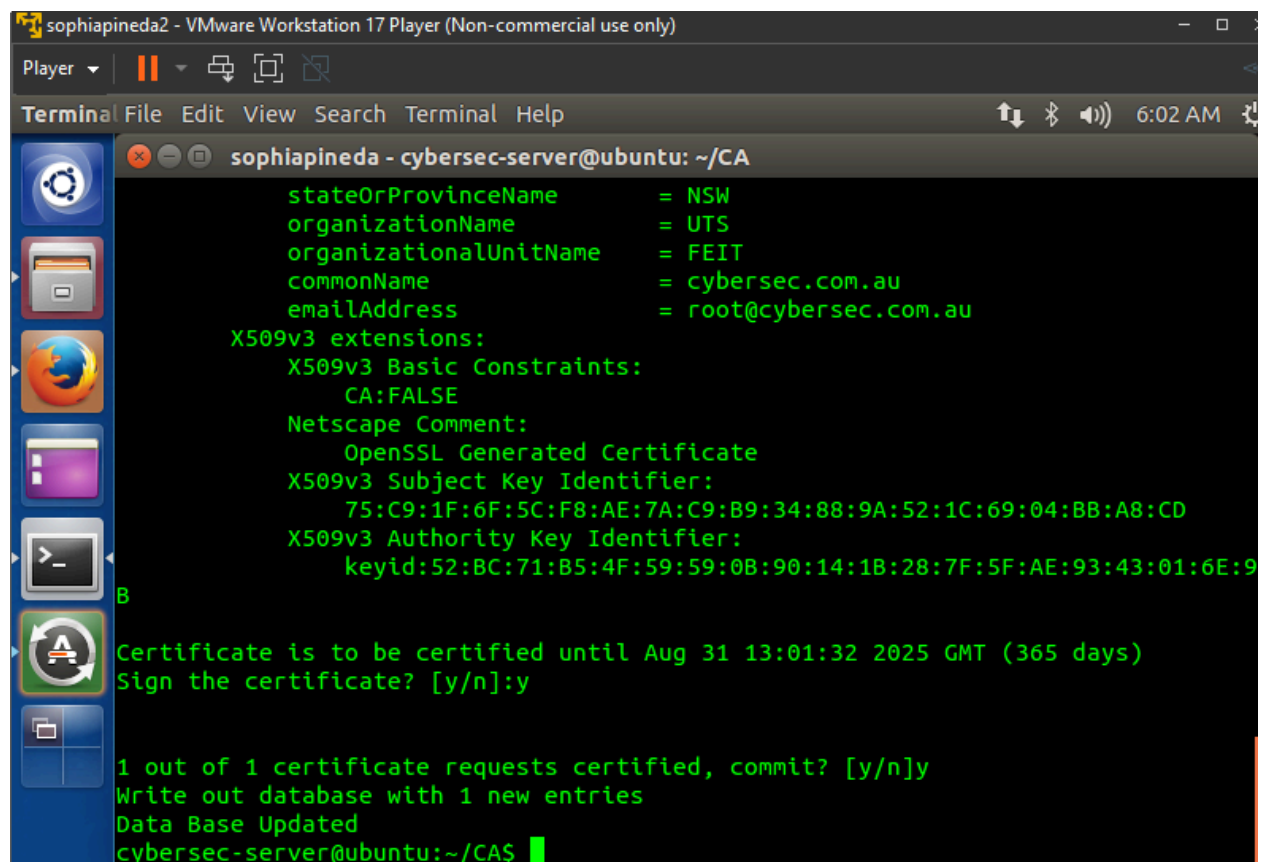
- Generate public/private key pair, use command
- Generate certificate signing request (company's public key)
- Turn certificate signing request into an X509 certificate

SCREENSHOTS

1. Command used to generate CSR



2. Info given when generating CSR

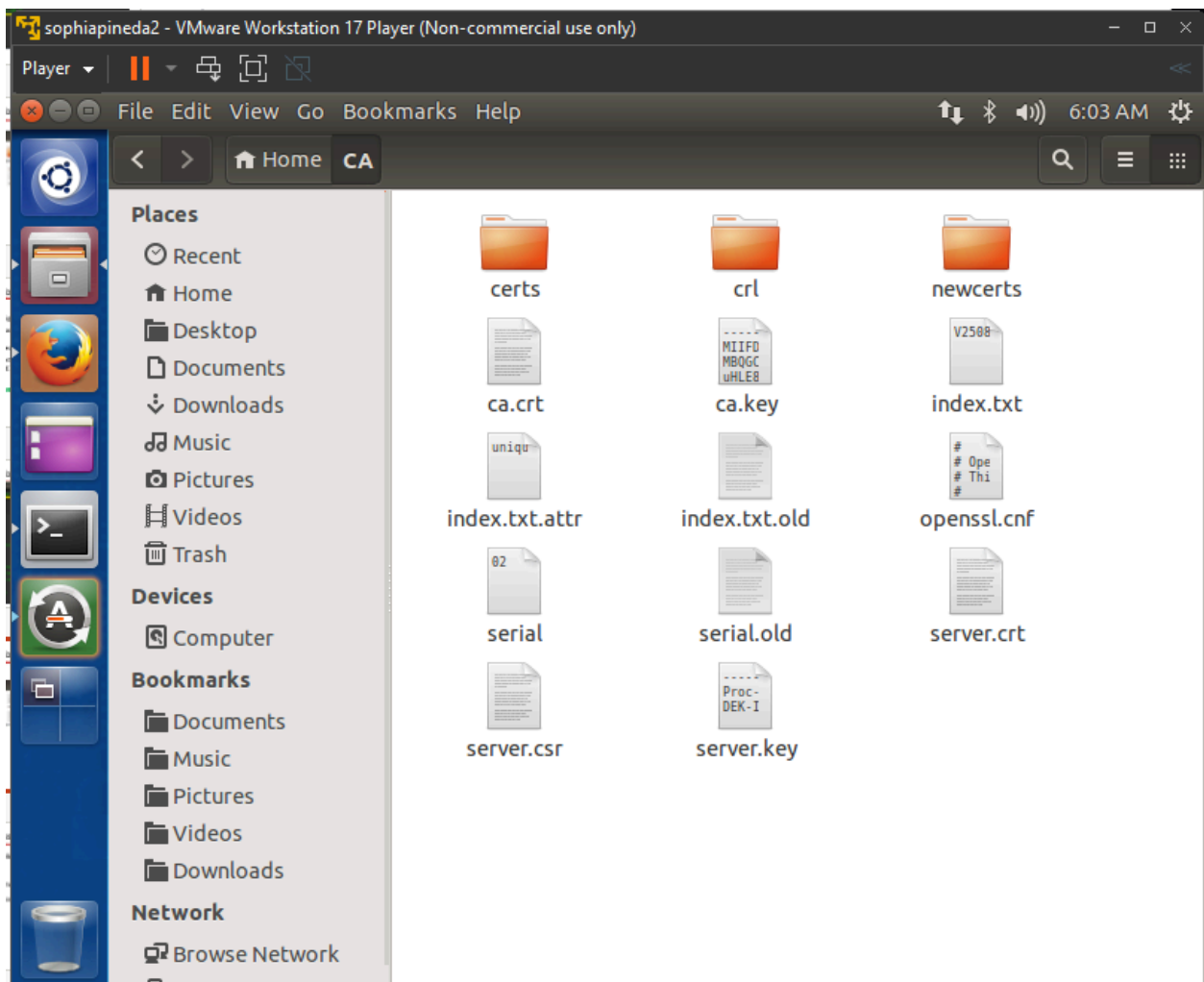


The screenshot shows a VMware Workstation 17 Player window titled 'sophiapineda2 - VMware Workstation 17 Player (Non-commercial use only)'. The interface includes a top menu bar with 'Player', 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. Below the menu is a toolbar with icons for play, pause, and other controls. The main area is a terminal window titled 'sophiapineda - cybersec-server@ubuntu: ~/CA'. The terminal output shows the configuration of a Certificate Authority (CA) using OpenSSL. The configuration includes the state or province name (NSW), organization name (UTS), organizational unit name (FEIT), common name (cybersec.com.au), and email address (root@cybersec.com.au). It also shows the X509v3 extensions, including the Basic Constraints (CA:FALSE) and the Netscape Comment (OpenSSL Generated Certificate). The terminal prompts for signing the certificate, which is confirmed with 'y'. The final output shows that 1 out of 1 certificate requests were certified, committed, and the database was updated with 1 new entry. The prompt returns to 'cybersec-server@ubuntu:~/CA\$'.

```
sophiapineda - cybersec-server@ubuntu: ~/CA
stateOrProvinceName       = NSW
organizationName          = UTS
organizationalUnitName    = FEIT
commonName                = cybersec.com.au
emailAddress              = root@cybersec.com.au
X509v3 extensions:
X509v3 Basic Constraints:
    CA:FALSE
Netscape Comment:
    OpenSSL Generated Certificate
X509v3 Subject Key Identifier:
    75:C9:1F:6F:5C:F8:AE:7A:C9:B9:34:88:9A:52:1C:69:04:BB:A8:CD
X509v3 Authority Key Identifier:
    keyid:52:BC:71:B5:4F:59:59:0B:90:14:1B:28:7F:5F:AE:93:43:01:6E:9
B
Certificate is to be certified until Aug 31 13:01:32 2025 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
cybersec-server@ubuntu:~/CA$
```

3. Files created



TASK 4: USE PKI FOR WEBSITES

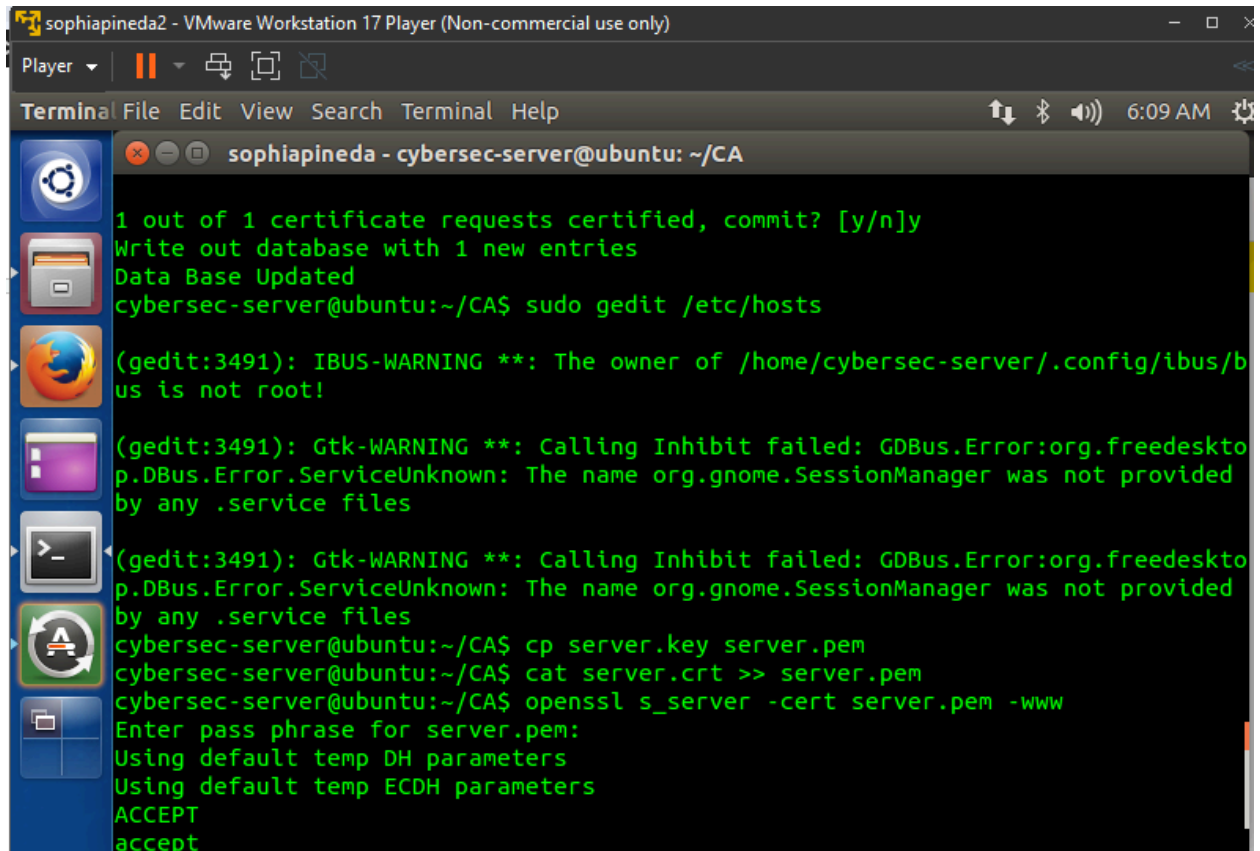
- How public-key certificates are used by websites to secure web browsing

STEPS:

- Make domain name add to local host files
- Restart system
- Launch web server with certificate generated from previous task
- Certificate + secret key
- Manually add our CA's certificate to firefox's browser

SCREENSHOTS:

1. Launching the web server using "server.pem"



```
sophiapineda2 - VMware Workstation 17 Player (Non-commercial use only)
Player
Terminal File Edit View Search Terminal Help
6:09 AM

sophiapineda - cybersec-server@ubuntu: ~/CA

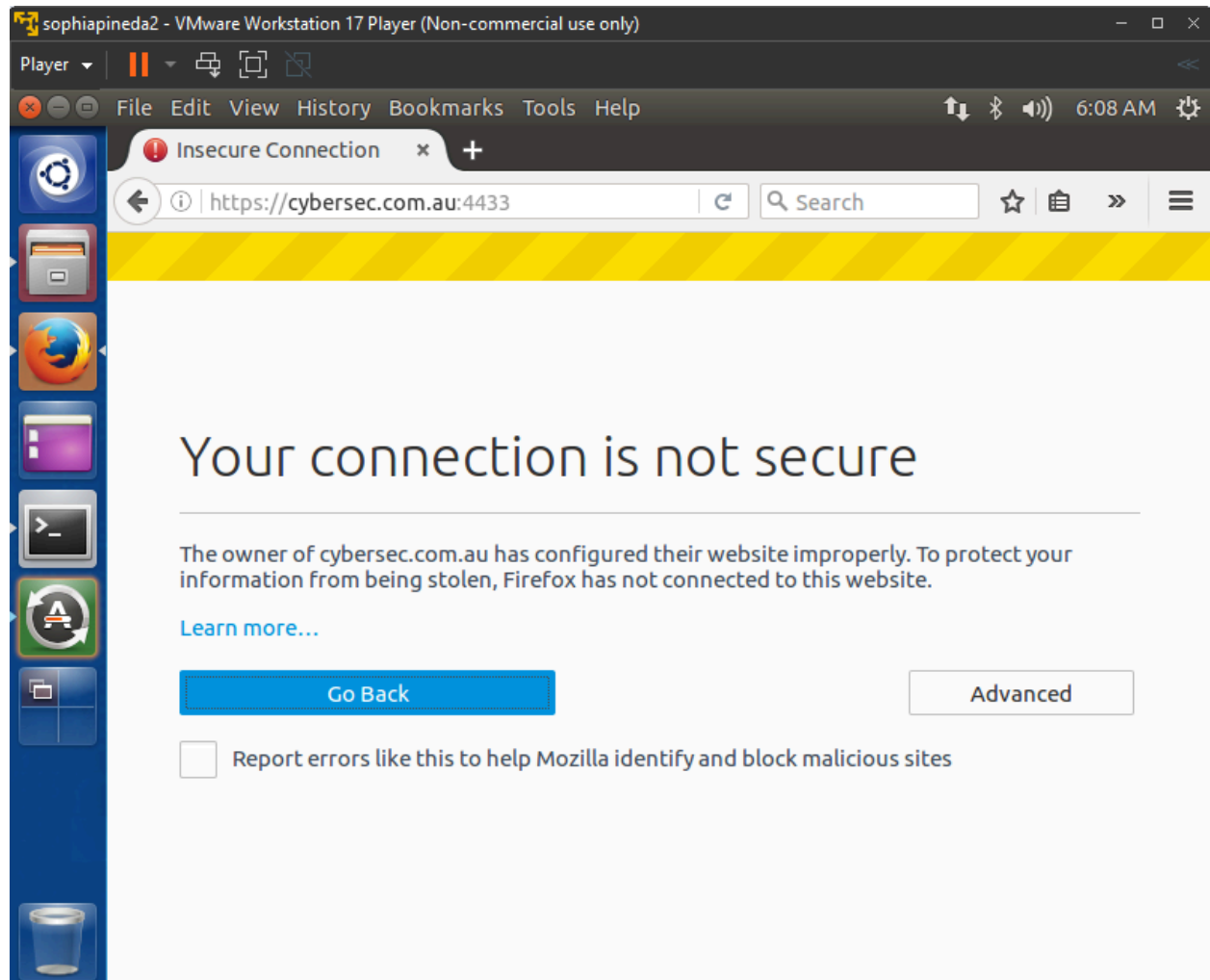
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
cybersec-server@ubuntu:~/CA$ sudo gedit /etc/hosts

(gedit:3491): IBUS-WARNING **: The owner of /home/cybersec-server/.config/ibus/b
us is not root!

(gedit:3491): Gtk-WARNING **: Calling Inhibit failed: GDBus.Error:org.freedeskt
o.p.DBus.Error.ServiceUnknown: The name org.gnome.SessionManager was not provided
by any .service files

(gedit:3491): Gtk-WARNING **: Calling Inhibit failed: GDBus.Error:org.freedeskt
o.p.DBus.Error.ServiceUnknown: The name org.gnome.SessionManager was not provided
by any .service files
cybersec-server@ubuntu:~/CA$ cp server.key server.pem
cybersec-server@ubuntu:~/CA$ cat server.crt >> server.pem
cybersec-server@ubuntu:~/CA$ openssl s_server -cert server.pem -www
Enter pass phrase for server.pem:
Using default temp DH parameters
Using default temp ECDH parameters
ACCEPT
accept
```

2. Screenshot of certificate manager showing the generated certificate for cybersec.com.au



3. Website on web browser after the addition of the certificate

sophiapineda2 - VMware Workstation 17 Player (Non-commercial use only)

Player | [Pause] [Full Screen] [Refresh] [Close]

Mozilla Firefox | [Up] [Bluetooth] [Speaker] 6:12 AM [Settings]

https://cyb...om.au:4433/ x [Preferences] x +

[Back] [Info] [Lock] | https://cybersec.com.au:4433 [Refresh] [Search] [Star] [Bookmarks] [Menu]

Ciphers common between both SSL end points:
ECDHE-ECDSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-GCM-SHA256 ECDHE-ECDSA-AES256-SHA
ECDHE-ECDSA-AES128-SHA ECDHE-RSA-AES128-SHA ECDHE-RSA-AES256-SHA
DHE-RSA-AES128-SHA DHE-RSA-AES256-SHA AES128-SHA
AES256-SHA DES-CBC3-SHA

New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES128-GCM-SHA256
SSL-Session:
Protocol : TLSv1.2
Cipher : ECDHE-RSA-AES128-GCM-SHA256
Session-ID:
Session-ID-ctx: 01000000
Master-Key: 6257C7840D2DAB5F68F3389B266044BF926BC2D59724317B53D509AC23AE900CEC0C8B483192AAC920BA9
Key-Arg : None
PSK identity: None
PSK identity hint: None
SRP username: None
Start Time: 1725109925
Timeout : 300 (sec)
Verify return code: 0 (ok)

0 items in the session cache
0 client connects (SSL_connect())
0 client renegotiates (SSL_connect())
0 client connects that finished
3 server accepts (SSL_accept())
0 server renegotiates (SSL_accept())
3 server accepts that finished
0 session cache hits
0 session cache misses
0 session cache timeouts
0 callback cache hits
0 cache full overflows (128 allowed)