

NASA HW6

NA

SSID: <https://www.zhihu.com/question/24362037>

Evil Twin: <https://ithelp.ithome.com.tw/articles/10254256>

PEAP: <https://zh.m.wikipedia.org/zh-hans/扩展认证协议>

WIFI Certificate: <https://www.quora.com/What-is-a-WIFI-certificate>

WIFI channel: <https://zh.m.wikipedia.org/zh/无线局域网信道列表>

SSID/BSSID

1. SSID是一整個無線網路服務集合的識別名稱(對同一個無線網路服務只有一個)，BSSID是對某個無線網路服務的單位的識別名稱(同一個無線網路服務中的同一個AP只有一個)，一般來說是裝置的MAC Address。
2. 是，是
3. a. 駭客利用跟使用者常用SSID相同但訊號較強的基地台，誘使使用者連接，接下來就可以透過DNS攻擊之類的方式來得到使用者的私人資訊。
b. 使用VPN讓所有流量都加密

PSK/EAP/PEAP

1. PSK是Pre-shared key的縮寫，在WIFI中就是利用預先設定的密碼作為驗證手段，而EAP是利用帳號密碼的Pair連上Radius伺服器做驗證，PEAP是加密版(TLS)的EAP，利用Server端的PKI來作加密。
三者差別就是PSK只使用密碼，EAP使用帳號密碼對來登入，PEAP就是把EAP加密。
2. Personal, 只有密碼，連線方便，但是沒有身份紀錄，因此相較於企業更適合用於個人

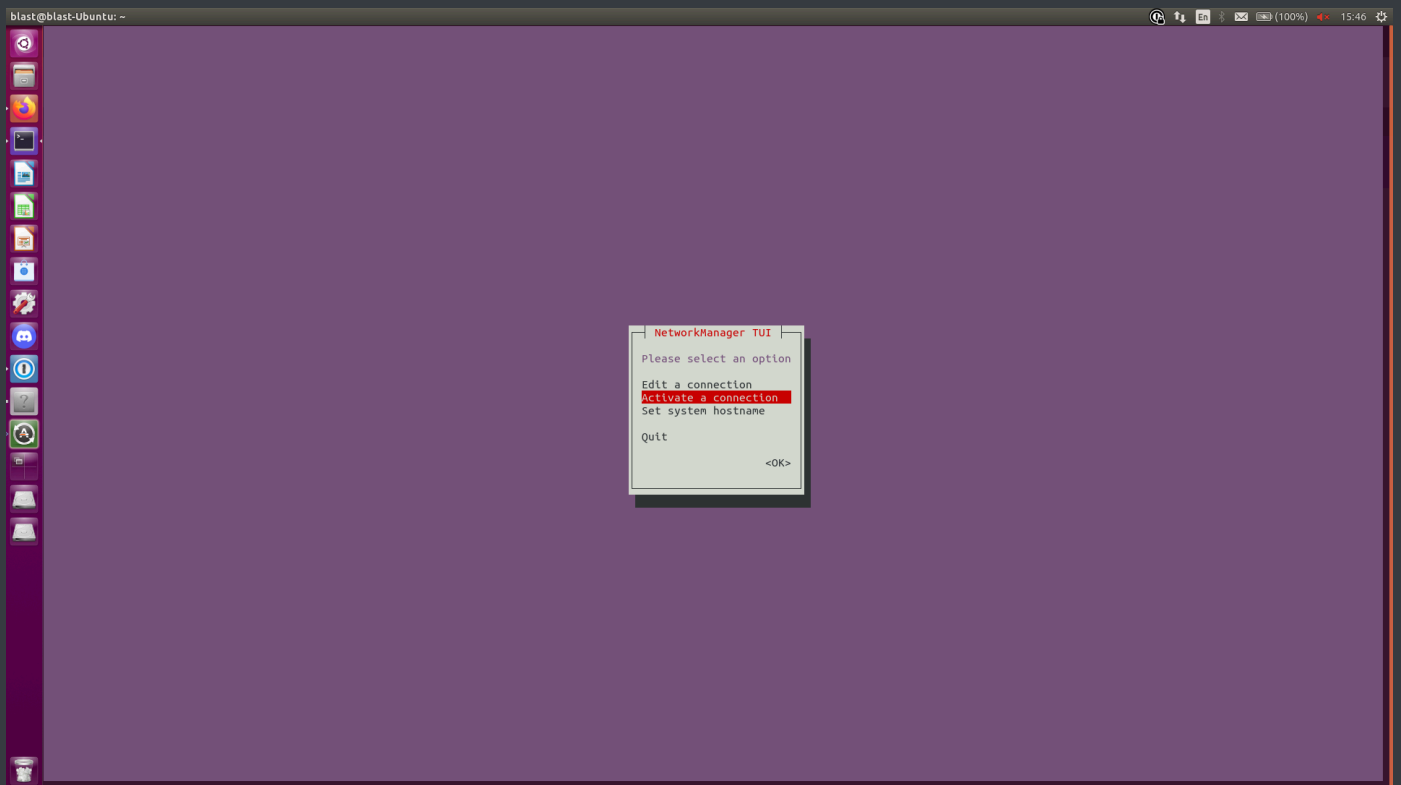
WiFi Certificate

1. WiFi-Certificate 是來驗證WiFi熱點的身份，確定該WiFi熱點就是我們要的，而不是被偽造的，並且同樣利用證書，對登入及註冊的過程作加密保護。
2. CA就是負責簽發，管理WiFi-Certificate的機構，負責驗證證書持有者的正確性並簽發證書，防止證書被修改

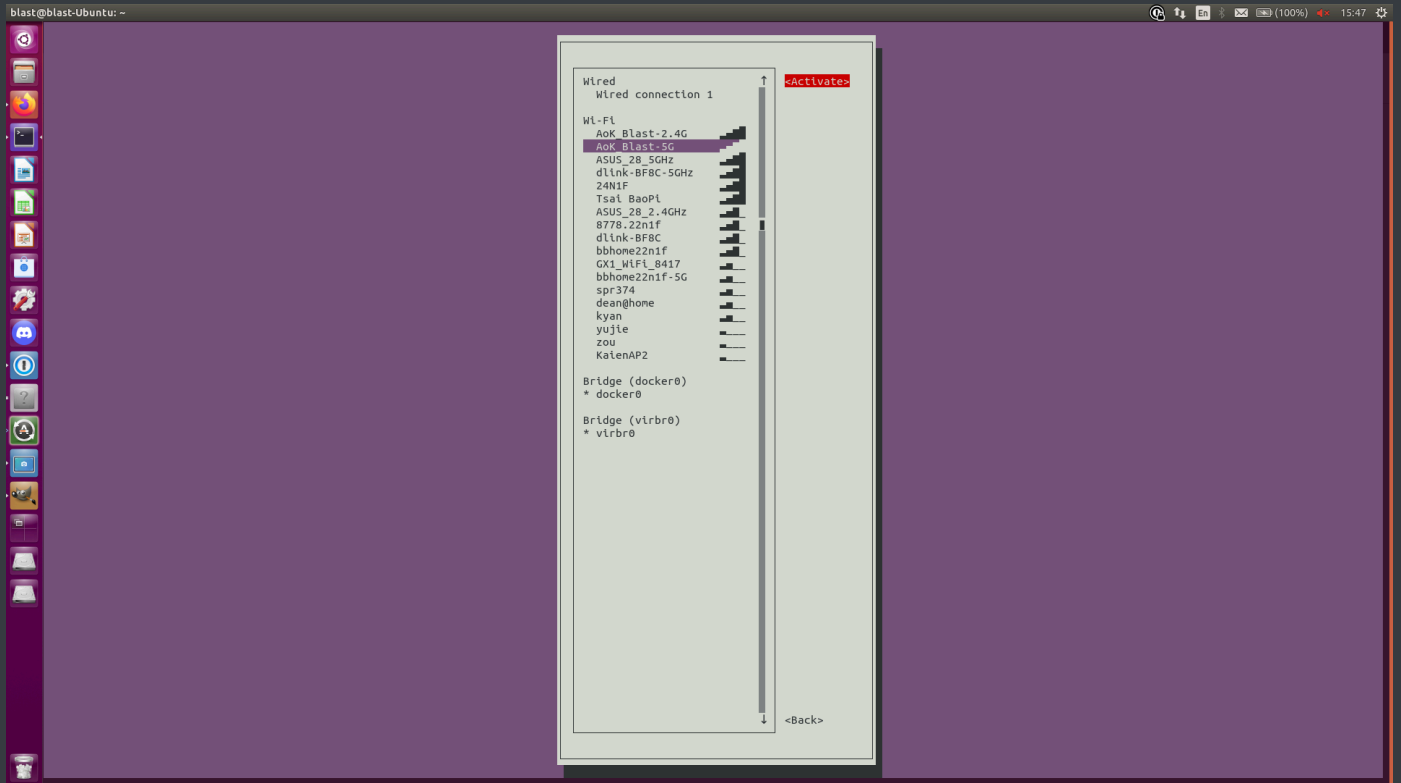
Connect to WiFi with terminal

```
nmtui
```

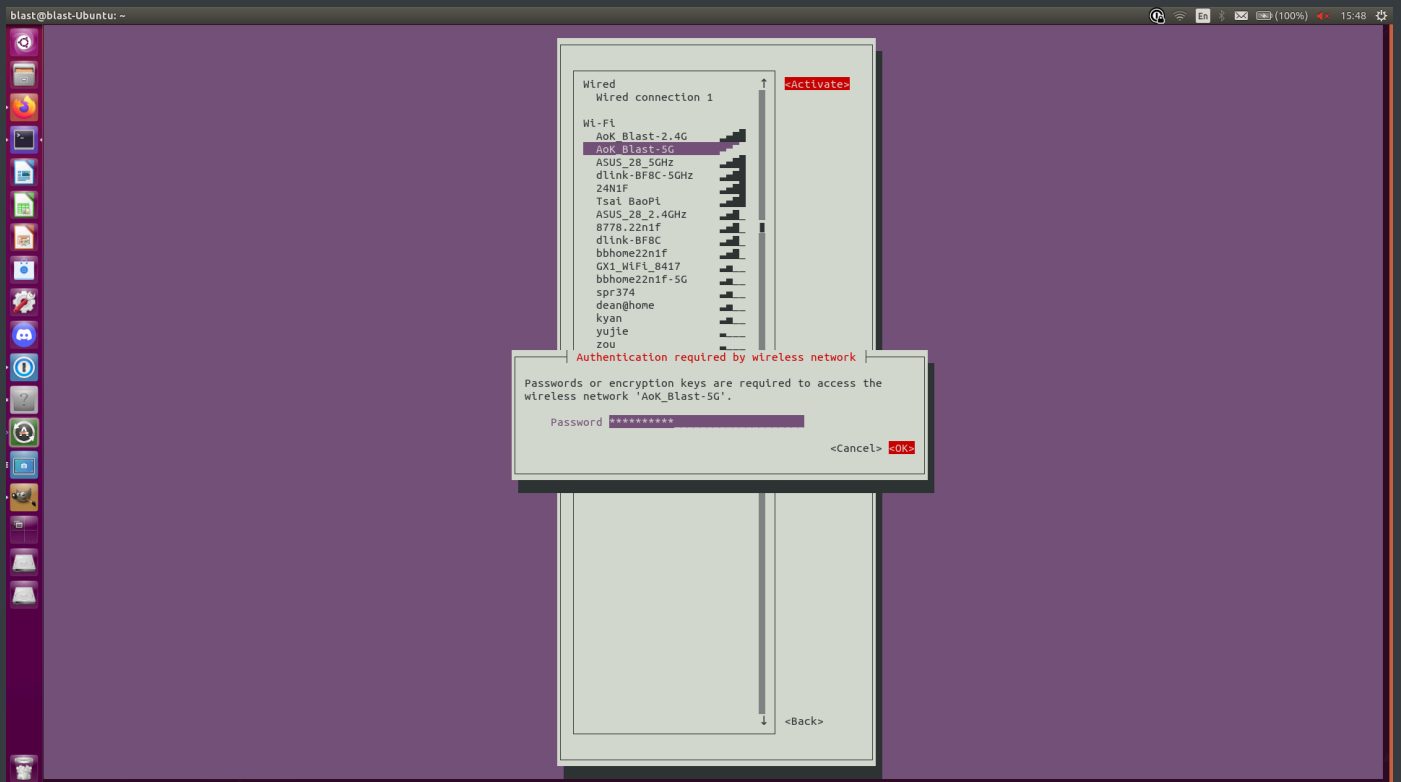
選擇啟用連線



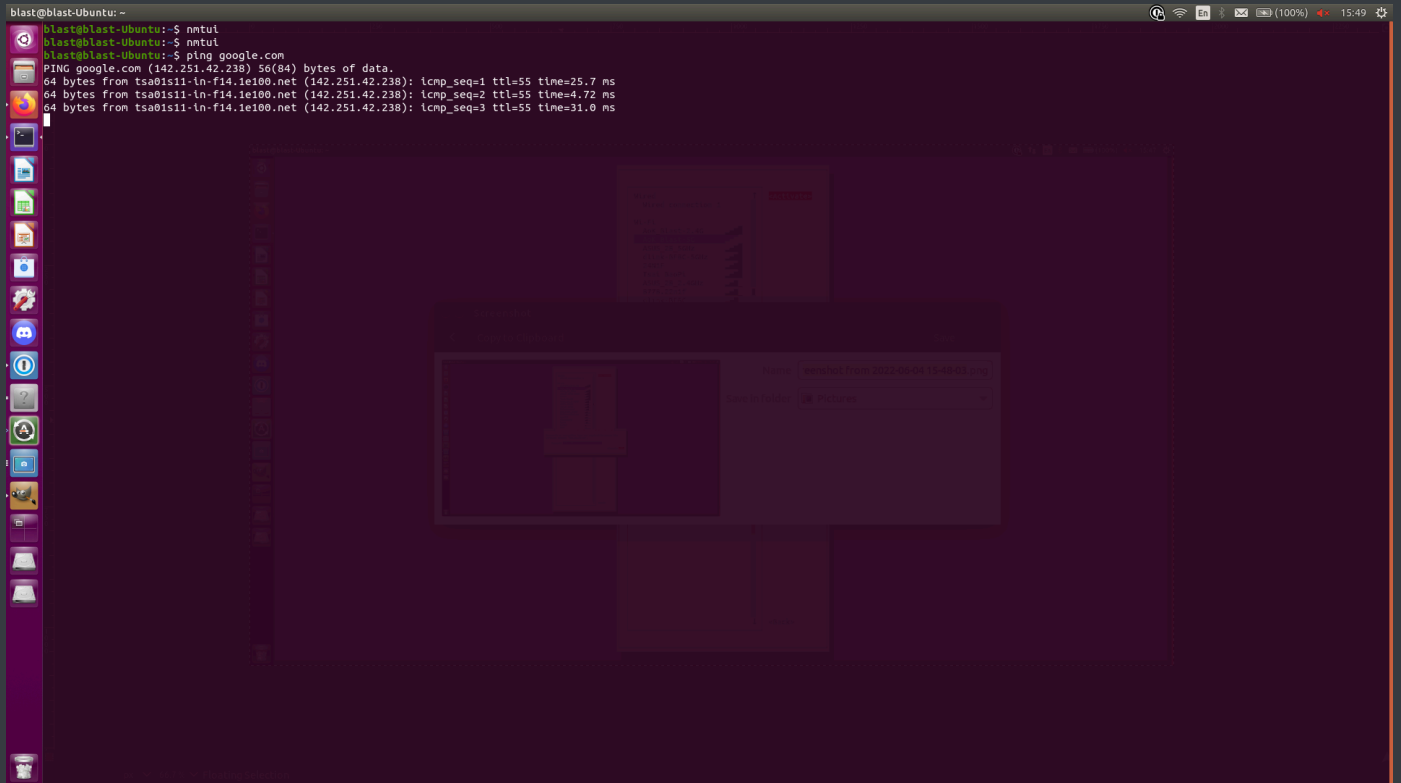
選擇要連線的SSID按啟用



輸入密碼後按啟用



退出後測試



AP channel

如果是三個機器的會，會分別設定在1,6,11，因為這三個頻道分別在
 2.412 ± 11 , 2.437 ± 11 , 2.462 ± 11 , 沒有相交的頻段，所以發生訊號干擾的機率較低

SA

static vs dynamic web server: <https://www.wix.com/blog/2021/11/static-vs-dynamic-web-website/>

Web Terminology

- 由於Nginx可以跑Epoll(非阻塞)，因此在有大量並行的情況下任然保持高吞吐量(高性能)，Apache背後還有基金會支撐，所以更偏向成熟(高穩定性)
 - 因為Apache非常高度模組化而且出來的早的關係，因此擴展性比Nginx好很多，甚至可以直接編譯Python Interpreter 到Apache上
 - Apache是早期產品(比較多Legacy Code)，因此佔用空間較大
- Static Web Server是靜態檔案，會將整個html的內容原封不動傳給你，Dynamic Web Server是由WebServer+其他軟體輔助(Ex. Database)，每次有請求的時候可能會從其他軟體裡面撈資訊把內容填進去，再把內容傳給你，因此Static的除非你改Code,不然每次Query的結果都會一樣，而Dynamic則是可能每次Query的結果都不一樣

3. a. Proxy是在進入請求到Web Server之前，把請求轉發到Proxy Server, 用Proxy Server的IP代為發送Request, 接收則是反過來一樣透過Proxy Server，算是一種在客戶端做的保護手段。
 - b.1. 隱藏Client真實IP。
 2. 讓不同網段的裝置可以像是在同一個網段(假設b,c在同一網段, b對外公開而c則不，a想連到c, 就可以先代理到b在到c)。
 3. Proxy也可以被當作快取Server, 當請求送到Proxy Server, Proxy Server可以選擇將快取內容送回去。
4. a. Reverse Proxy則是能夠隱藏真實的Web Server，當我們送到一個Reverse Proxy的Server的時候，這個Server不會直接回傳內容(他可能也沒有),而是將請求給轉發到真實的Web Server, 可以視為一種伺服器端的保護手段。
 - b. 1. 負載平衡，因為Reverse Proxy可以傳給不同Web Server。
 2. 隱藏真正的Server IP,。
 3. 也可以當作快取，同一般Proxy但是反向

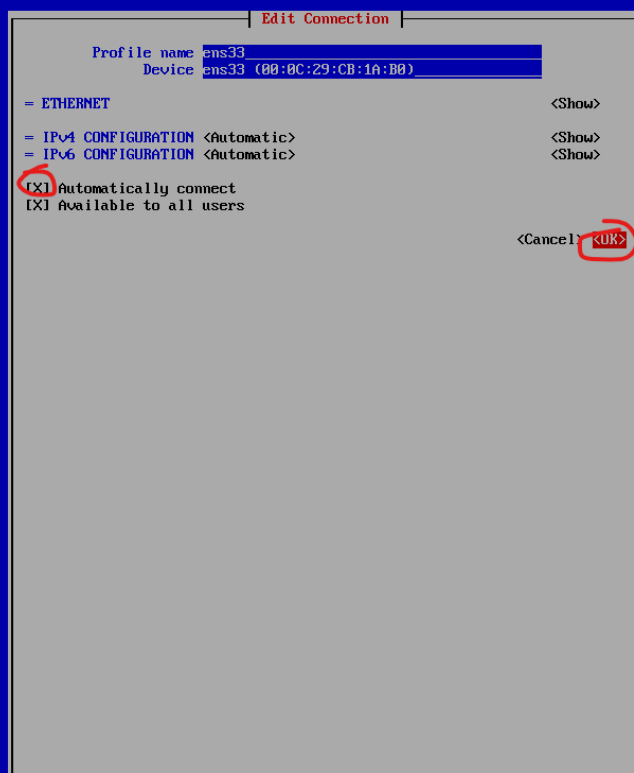
Web Server Configurations

apache install: <https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-centos-7>

404 not found: <https://www.ait.com/support/11456-create-a-custom-404-page-in-apache>

Files directive: <https://httpd.apache.org/docs/current/configuring.html>

nmtui



```
sudo yum install httpd -y
sudo systemctl start httpd
```

```
sudo firewall-cmd --permanent --add-port=http/tcp
sudo firewall-cmd --reload
```

```
sudo vim /var/www/html/index.html
```

```
# Input following text
```

```
# Hello! My name is 40947047s!
```

← → ↻ 192.168.171.137

Hello! My name is 40947047s!

```
sudo vim /etc/httpd/conf.d/userdir.conf
```

```
# Delete UserDir disabled
```

```
# Uncommon UserDir public_html
```

```
# change UserDir public_html to UserDir /home/*/public_html
```

```
# Change all public_html to htdocs(e.g. in vim,
```

```
%s/public_html/htdocs)
```

```
sudo setenforce 0 # turn off selinux because of the usage of home  
folder content
```

```
sudo mkdir -p /home/40947047s/htdocs
```

```
sudo vim /home/40947047s/htdocs/index.html
```

```
# Input following text
```

```
# Hello! My name is 40947047s!
```

```
sudo adduser 40947047s
```

```
sudo systemctl restart httpd
```



192.168.171.137/~40947047s/

Hello! My name is 40947047s!

```
sudo vim /var/www/html/404.html
# Input the following text
# Sorry, the page doesn't exist...

sudo vim /etc/httpd/conf/httpd.conf
# Input the following text
# Error Document 404 /404.html

sudo systemctl restart httpd
```



192.168.171.137/nowhere

Sorry, the page doesn't exist...


```
sudo vim /var/www/html/secret.html

# Input the following text
# Secret


sudo vim /etc/httpd/conf/httpd.conf
# Input the following text
# <Files "secret.html">
#     Require ip 192.168.28.0/255.255.255.0
# </Files>
sudo systemctl restart httpd
```

Reverse Proxy

Reverse Proxy setting: <https://blog.davidou.org/archives/1334>

```
# 使用到的module mod_proxy
sudo vim /etc/httpd/conf/httpd.conf
# Input the following text

# <VirtualHost *:80>
# ServerName 192.168.171.137
# ProxyRequests off
# <Proxy *>
# Order allow,deny
# Allow from all
# </Proxy>

# ProxyPass /hostA http://10.217.44.28          #setting proxy with
different directory
# ProxyPassReverse /hostA http://10.217.44.28 #setting proxy with
different directory
# ProxyPass /hostB http://10.217.44.6          #setting proxy with
different directory
```

