

ขั้นตอนการแก้ไข PermitRootLogin และทดสอบการเข้าใช้งาน (SSH)

ทำบน **SERVER**

Login ผ่าน PuTTY เป็น root (หรือผู้ใช้ที่มีสิทธิ์แก้ไขไฟล์)

ขั้นตอนที่ 1 : เปิดไฟล์ sshd_config ด้วย vi

vi /etc/ssh/sshd_config

เมื่อเปิดไฟล์แล้ว จะอยู่ใน **Normal Mode**

ขั้นตอนที่ 2 : ค้นหาบรรทัด PermitRootLogin

ใน Normal Mode ให้พิมพ์

/PermitRootLogin

แล้วกด **Enter**

หน้าจอจะเลื่อนไปที่บรรทัดลักษณะนี้

```
#PermitRootLogin prohibit-password
```

ขั้นตอนที่ 3 : แก้ไขบรรทัด (จาก Comment → ใช้งานจริง)

3.1 ลบเครื่องหมาย

- เลื่อนเคอร์เซอร์ไปที่ตัวอักษร #
- กดปุ่ม x

จะได้บรรทัดเป็น

```
PermitRootLogin prohibit-password
```

3.2 เข้าโหมดแก้ไข (Insert Mode)

กดปุ่ม

i

จากนั้นแก้คำว่า

prohibit-password

ให้เป็น

yes

บรรทัดสุดท้ายต้องเป็น

PermitRootLogin yes

ขั้นตอนที่ 4 : ออกจากโหมดแก้ไข

กดปุ่ม

Esc

ขั้นตอนที่ 5 : บันทึกไฟล์และออกจาก vi

พิมพ์

:wq

แล้วกด **Enter**

ขั้นตอนที่ 6 : Restart SSH Service

เพื่อให้ค่าที่แก้ไขมีผล

systemctl restart ssh

ตรวจสอบสถานะ

```
systemctl status ssh --no-pager
```

ต้องเห็นสถานะ

```
active (running)
```

ขั้นตอนที่ 7 : ทดสอบ Login ใหม่จาก Client (PuTTY)

1. ปิดหน้าต่าง PuTTY เดิม
2. เปิด PuTTY ใหม่
3. ใส่ IP Server:
4. 192.168.10.10
5. กด **Open**
6. Login ด้วย

```
login as: root
```

ใส่รหัสผ่าน → กด Enter

ขั้นตอนที่ 8 : ตรวจสอบว่า Login สำเร็จ

เมื่อเข้าได้ จะเห็น prompt เช่น

```
root@server:~#
```

ตรวจสอบผู้ใช้

```
whoami
```

ผลลัพธ์ต้องเป็น

```
root
```

ขั้นตอนการติดตั้ง Apache Web Server (Apache2)

1) ตรวจสอบสถานะเครื่องก่อนติดตั้ง (แนะนำ)

เพื่อยืนยันว่าอยู่ที่เครื่อง Server ถูกต้อง

```
whoami  
hostname  
ip a
```

ต้องแน่ใจว่า

- ผู้ใช้เป็น root
 - IP ตรงกับ Server (เช่น 192.168.10.10)
-

2) อัปเดตรายการแพ็คเกจ

```
apt update
```

เหตุผล

เพื่อให้ระบบดึงแพ็คเกจ Apache เวอร์ชันที่เหมาะสมและเสถียรที่สุด

3) ติดตั้ง Apache Web Server

```
apt install -y apache2
```

ระหว่างติดตั้ง

- ระบบจะสร้าง service ชื่อ apache2
 - เปิดพอร์ต 80 ให้อัตโนมัติ (ถ้าไม่มี firewall บล็อก)
-

4) ตรวจสอบสถานะ Apache Service

หลังติดตั้งเสร็จ

```
systemctl status apache2 --no-pager
```

ผลลัพธ์ที่ถูกต้อง ต้องเห็น

Active: active (running)

ถ้ายังไม่เริ่มทำงาน ให้สั่ง

```
systemctl start apache2
```

5) ตั้งค่าให้ Apache เริ่มทำงานอัตโนมัติเมื่อเปิดเครื่อง

```
systemctl enable apache2
```

6) ตรวจสอบเวอร์ชัน Apache

```
apache2 -v
```

ตัวอย่างผลลัพธ์

Server version: Apache/2.4.x (Ubuntu)

7) ตรวจสอบพอร์ตที่ Apache ใช้งาน

Apache จะฟังที่พอร์ต 80

```
ss -tlnp | grep apache2
```

ต้องเห็น

```
LISTEN 0 128 0.0.0.0:80
```

8) ทดสอบ Apache เบื้องต้น (หน้า Default)

8.1 ทดสอบจาก Server

curl http://localhost

หรือ

curl http://127.0.0.1

ถ้า Apache ทำงาน จะเห็น HTML ของหน้า Default

8.2 ทดสอบจาก Client (Browser)

เปิดเว็บเบราว์เซอร์ แล้วเข้า

http://192.168.10.10

ถ้าเห็นหน้า

"Apache2 Ubuntu Default Page"

แสดงว่า Apache ทำงานเรียบร้อย

9) โครงสร้างไฟล์สำคัญของ Apache (ควรรู้ใน LAB)

รายการ	ตำแหน่ง
ไฟล์หลัก	/etc/apache2/apache2.conf
Virtual Host	/etc/apache2/sites-available/
Enable Site	/etc/apache2/sites-enabled/
Web Root (Default)	/var/www/html/
Log	/var/log/apache2/

10) คำสั่งควบคุม Apache ที่ใช้บ่อย (สำหรับ LAB)

การทำงาน	คำสั่ง
Start	systemctl start apache2
Stop	systemctl stop apache2
Restart	systemctl restart apache2
Reload (ไม่ตัดการเชื่อมต่อ)	systemctl reload apache2
ตรวจ config	apache2ctl configtest

11) การตรวจสอบปัญหาเบื้องต้น

หาก Apache ไม่ทำงาน

11.1 ตรวจสอบ config

```
apache2ctl configtest
```

ต้องได้

Syntax OK

11.2 ดู log error

```
tail -n 50 /var/log/apache2/error.log
```

12) สรุปผลการติดตั้ง Apache

เมื่อทำครบขั้นตอนนี้ ถือว่า

- Apache Web Server ติดตั้งเรียบร้อยแล้ว
- เปิดให้บริการที่พอร์ต 80
- พร้อมสำหรับขั้นตอนถัดไป คือ
👉 การสร้าง **Virtual Host** สำหรับ **aaa / bbb / ccc / ddd.comed.net**

ขั้นตอนการติดตั้ง MySQL Server (mysql-server)

เงื่อนไข

- ทำบน **SERVER (Ubuntu Linux)**
- Login เป็น root
- ไม่ใช้ sudo
- ใช้ MySQL สำหรับรองรับงาน Web / LAB ในอนาคต (เช่น WordPress, Web App)

1) ตรวจสอบสถานะผู้ใช้งานและเครื่อง

ก่อนเริ่มติดตั้ง ให้ยืนยันว่าทำงานอยู่บน Server และเป็น root

whoami

hostname

ip a

ผลที่คาดหวัง

- ผู้ใช้เป็น root
- IP ตรงกับ Server (เช่น 192.168.10.10)

2) อัปเดตรายการแพ็คเกจ

apt update

เหตุผล

เพื่อให้ระบบติดตั้ง MySQL เวอร์ชันที่เสถียรจากคลังแพ็คเกจของ Ubuntu

3) ติดตั้ง MySQL Server

```
apt install -y mysql-server
```

ระหว่างติดตั้ง

- ระบบจะติดตั้ง MySQL Server และ dependency ที่จำเป็น
- Service ชื่อ mysql จะถูกสร้างให้อัตโนมัติ

4) ตรวจสอบสถานะ MySQL Service

หลังติดตั้งเสร็จ

```
systemctl status mysql --no-pager
```

ต้องเห็นสถานะ

Active: active (running)

ถ้ายังไม่ทำงาน ให้เริ่มบริการ

```
systemctl start mysql
```

5) ตั้งค่าให้ MySQL เริ่มอัตโนมัติเมื่อเปิดเครื่อง

```
systemctl enable mysql
```

6) ตรวจสอบเวอร์ชัน MySQL

```
mysql --version
```

ตัวอย่างผลลัพธ์

```
mysql Ver 8.0.xx for Linux on x86_64
```

7) ทดสอบการเข้าใช้งาน MySQL เบื้องต้น

เข้าสู่ MySQL ด้วยผู้ใช้ root

mysql

ถ้าเข้าได้ จะเห็น prompt

mysql>

ออกจาก MySQL

exit;

8) ตั้งค่าความปลอดภัยพื้นฐาน (แนะนำให้ทำ)

รันสคริปต์ปรับความปลอดภัยของ MySQL

mysql_secure_installation

ระบบจะถามคำถามเป็นลำดับ ให้ตอบดังนี้ (แนวทางสำหรับ LAB)

1. **Validate password component?**
→ N
 2. **Set root password?**
→ Y (ตั้งรหัสผ่านตามที่ผู้สอนกำหนด)
 3. **Remove anonymous users?**
→ Y
 4. **Disallow root login remotely?**
→ N (สำหรับ LAB ที่อาจต้องใช้ root จากเครื่องอื่น)
 5. **Remove test database and access to it?**
→ Y
 6. **Reload privilege tables now?**
→ Y
-

9) ทดสอบ Login ด้วยรหัสผ่าน

หลังตั้งรหัสผ่านแล้ว

```
mysql -u root -p
```

ใส่รหัสผ่าน → ต้องเข้าได้

10) ตรวจสอบพอร์ตที่ MySQL ใช้งาน

MySQL ใช้พอร์ต 3306

```
ss -tlnp | grep mysql
```

ต้องเห็น

```
LISTEN 0 151 127.0.0.1:3306
```

หมายเหตุ

ค่าเริ่มต้น MySQL จะฟังเฉพาะ localhost เพื่อความปลอดภัย

11) ตำแหน่งไฟล์สำคัญของ MySQL (ควรรู้)

รายการ	ตำแหน่ง
ไฟล์ config หลัก	/etc/mysql/mysql.conf.d/mysqld.cnf
Data directory	/var/lib/mysql/
Log	/var/log/mysql/
Socket	/var/run/mysqld/mysqld.sock

12) คำสั่งควบคุม MySQL ที่ใช้บ่อย

การทำงาน	คำสั่ง
Start	systemctl start mysql
Stop	systemctl stop mysql
Restart	systemctl restart mysql
Status	systemctl status mysql
Login	mysql -u root -p

13) ปัญหาที่พบบ่อยและการตรวจสอบ

13.1 MySQL ไม่เริ่มทำงาน

journalctl -xeu mysql

13.2 Login ไม่ได้

- ตรวจสอบรหัสผ่าน
- ตรวจสอบว่าใช้ socket login หรือ password login

14) สรุปผลการติดตั้ง MySQL

เมื่อทำครบขั้นตอนนี้ ถือว่า

- MySQL Server ติดตั้งเรียบร้อยแล้ว
- Service ทำงานปกติ
- พร้อมใช้งานร่วมกับ Apache และ Web Application
- พร้อมต่อยอดไปสู่ LAB ถัดไป (เช่น WordPress, phpMyAdmin)

ขั้นตอนการติดตั้ง phpMyAdmin

วัตถุประสงค์

เพื่อให้สามารถ จัดการฐานข้อมูล MySQL ผ่าน Web Browser
ใช้ร่วมกับ Apache Web Server ใน LAB (เช่น ตรวจสอบ DB, ใช้กับ
WordPress)

1) ตรวจสอบความพร้อมของระบบก่อนติดตั้ง

ให้ยืนยันว่าบริการที่จำเป็นทำงานอยู่แล้ว

```
systemctl status apache2 --no-pager  
systemctl status mysql --no-pager
```

ต้องเห็นสถานะ

active (running)

ตรวจสอบ PHP (ควรมีอย่างน้อย php-cli)

```
php -v
```

ถ้ายังไม่มี PHP ให้ติดตั้งก่อน (แนะนำ)

```
apt install -y php php-mysql php-mbstring php-zip php-gd php-  
json php-curl
```

2) อัปเดตรายการแพ็คเกจ

```
apt update
```

3) ติดตั้ง phpMyAdmin

```
apt install -y phpmyadmin
```

ระหว่างการติดตั้ง จะมีหน้าจอให้เลือกค่า (สำคัญมาก)
ให้เลือกตามลำดับดังนี้

4) ขั้นตอนตอบคำถามระหว่างติดตั้ง (Interactive Setup)

4.1 เลือก Web Server

หน้าจอ:

Configuring phpmyadmin
Please choose the web server that should be automatically
configured to run phpMyAdmin.

→ ใช้ปุ่ม **Spacebar** เลือก

[*] apache2

แล้วกด **Enter**

4.2 Configure database for phpMyAdmin

หน้าจอ:

Configure database for phpmyadmin with dbconfig-common?

→ เลือก

Yes

แล้วกด **Enter**

4.3 ตั้งรหัสผ่านสำหรับ phpMyAdmin

หน้าจอ:

MySQL application password for phpmyadmin:

➔ ใส่รหัสผ่าน (หรือกด Enter เพื่อให้ระบบสร้างให้)
แนะนำใน LAB:

- ตั้งรหัสผ่านง่ายแต่จำได้
- จดไว้ใช้ทดสอบ

5) เปิดใช้งาน PHP Extensions ที่ phpMyAdmin ต้องใช้

ตรวจสอบว่าโมดูล mbstring ถูกเปิดใช้งาน

phpenmod mbstring

Restart Apache

systemctl restart apache2

6) ตรวจสอบไฟล์ Config ของ phpMyAdmin

ไฟล์หลักจะอยู่ที่

/etc/phpmyadmin/config.inc.php

ไม่จำเป็นต้องแก้ไขในขั้นต้น

แต่สามารถตรวจสอบได้ว่าไฟล์มีอยู่

ls -l /etc/phpmyadmin/

7) ตรวจสอบว่า Apache รู้จัก phpMyAdmin

Apache จะสร้าง alias ให้อัตโนมัติ

ตรวจสอบไฟล์

```
ls /etc/apache2/conf-enabled/ | grep phpmyadmin
```

ถ้ายังไม่ถูก enable (กรณีผิดพลาด)

```
a2enconf phpmyadmin  
systemctl reload apache2
```

8) ทดสอบการเข้าใช้งาน phpMyAdmin ผ่าน Browser

8.1 เปิดจาก Server (ทดสอบเร็ว)

```
curl http://localhost/phpmyadmin | head
```

8.2 เปิดจาก Client (ใช้งานจริง)

เปิด Browser แล้วเข้า

```
http://192.168.10.10/phpmyadmin
```

หรือถ้าใช้โดเมน

```
http://aaa.comed.net/phpmyadmin
```

9) Login เข้า phpMyAdmin

หน้าจอ Login:

- **Username** : root
- **Password** : รหัสผ่าน MySQL ที่ตั้งไว้

ถ้าเข้าได้ จะเห็นหน้า Dashboard ของ phpMyAdmin

10) กรณี Login 'ไม่ได้' (พบบ่อยใน MySQL 8)

ถ้า root ใช้ plugin auth_socket

10.1 เข้า MySQL ด้วย shell

```
mysql
```

10.2 เปลี่ยน root ให้ใช้ password

```
ALTER USER 'root'@'localhost'  
IDENTIFIED WITH mysql_native_password  
BY 'รหัสผ่านที่ต้องการ';
```

```
FLUSH PRIVILEGES;  
EXIT;
```

Restart MySQL

```
systemctl restart mysql
```

11) ตำแหน่งไฟล์สำคัญของ phpMyAdmin

รายการ	ตำแหน่ง
phpMyAdmin config	/etc/phpmyadmin/
Apache alias	/etc/apache2/conf-enabled/phpmyadmin.conf
URL เรียกใช้งาน	/phpmyadmin

12) ปัญหาที่พบบ่อยและแนวทางแก้ไข

12.1 หน้าเว็บขาว / Error

ตรวจสอบ PHP

```
php -m | grep mbstring
```

12.2 Apache ไม่โหลด phpMyAdmin

```
a2enconf phpmyadmin  
systemctl reload apache2
```

12.3 ดู log

```
tail -n 50 /var/log/apache2/error.log
```

13) สรุปผลการติดตั้ง phpMyAdmin

เมื่อทำครบขั้นตอนนี้ ถือว่า

- phpMyAdmin ติดตั้งเรียบร้อยแล้ว
- ใช้งานผ่าน Web Browser ได้
- เชื่อมต่อ MySQL ได้ถูกต้อง
- พร้อมใช้จัดการ Database สำหรับ LAB ต่อไป

LAB ปฏิบัติการ ติดตั้ง DNS ด้วย BIND9 + ติดตั้ง Apache Virtual Host + Upload เว็บไซต์จาก Client

PART A: ติดตั้งและตั้งค่า DNS (BIND9) บน SERVER

A1) ติดตั้งแพ็คเกจ BIND9 + เครื่องมือทดสอบ

ทำบน SERVER (root)

```
apt update
apt install -y bind9 bind9utils dnsutils
systemctl enable --now bind9
```

ตรวจสอบสถานะ

```
systemctl status bind9 --no-pager
```

ต้องเป็น active (running)

A2) ตั้งค่า BIND9 ให้เป็น Authoritative DNS

แก้ไฟล์

```
nano /etc/bind/named.conf.options
```

ใส่/ปรับให้เป็น (แก้ IP ให้ตรงเครื่อง)

```
options {
    directory "/var/cache/bind";

    recursion no;
    allow-query { any; };

    listen-on { 192.168.10.10; 127.0.0.1; };
```

```
listen-on-v6 { none; };  
  
dnssec-validation auto;  
auth-nxdomain no;  
};
```

ตรวจสอบ Syntax

```
named-checkconf
```

ถ้าไม่ขึ้นอะไร = ผ่าน

A3) ประกาศ Zone comed.net

แก้ไขไฟล์

```
nano /etc/bind/named.conf.local
```

เพิ่มบรรทัดนี้

```
zone "comed.net" {  
    type master;  
    file "/etc/bind/zones/db.comed.net";  
};
```

สร้างโฟลเดอร์เก็บ zone

```
mkdir -p /etc/bind/zones
```

A4) สร้าง Zone File: db.comed.net

สร้าง/แก้ไขไฟล์

```
nano /etc/bind/zones/db.comed.net
```

วางเนื้อหานี้

\$TTL 86400

@ IN SOA ns1.comed.net. admin.comed.net. (
 2026010601 ; Serial (เพิ่มทุกครั้งเมื่อแก้)
 3600
 1800
 604800
 86400)

@	IN	NS	ns1.comed.net.
ns1	IN	A	192.168.10.10
aaa	IN	A	192.168.10.10
bbb	IN	A	192.168.10.10
ccc	IN	A	192.168.10.10
ddd	IN	A	192.168.10.10

A5) ตรวจสอบ Zone และ Restart BIND9

ตรวจสอบ zone

```
named-checkzone comed.net /etc/bind/zones/db.comed.net
```

ต้องขึ้น OK

Restart

```
systemctl restart bind9
systemctl status bind9 --no-pager
```

ถ้ามีปัญหาให้ดู log

```
journalctl -xeu bind9
```

A6) ตั้งค่า /etc/resolv.conf บน SERVER เพื่อทดสอบด้วย nslookup

แก้ไขไฟล์

```
nano /etc/resolv.conf
```

กำหนด

```
nameserver 192.168.10.10
```

A7) ตรวจสอบ DNS บน SERVER (ต้องทำให้ผ่านก่อน)

ทดสอบด้วย nslookup

```
nslookup aaa.comed.net
```

```
nslookup bbb.comed.net
```

```
nslookup ccc.comed.net
```

```
nslookup ddd.comed.net
```

ตรวจสอบ default server

```
nslookup
```

ต้องเห็น Default server: 192.168.10.10

ยืนยันด้วย dig

```
dig aaa.comed.net +short
```

ต้องได้ 192.168.10.10

PART B: ติดตั้ง Apache และทำ Virtual Host บน SERVER

B1) ติดตั้ง Apache

```
apt install -y apache2
systemctl enable --now apache2
systemctl status apache2 --no-pager
```

B2) สร้างโฟลเดอร์เว็บไซต์ตามโดเมน (ภายใต้ /var/www/html/)

```
mkdir -p /var/www/html/{aaa,bbb,ccc,ddd}.comed.net
```

ตรวจสอบ

```
ls -l /var/www/html
```

B3) สร้าง Virtual Host ของแต่ละโดเมน

B3.1 aaa.comed.net

```
nano /etc/apache2/sites-available/aaa.comed.net.conf
```

วาง

```
<VirtualHost *:80>
  ServerName aaa.comed.net
  DocumentRoot /var/www/html/aaa.comed.net

  <Directory /var/www/html/aaa.comed.net>
    AllowOverride All
    Require all granted
  </Directory>

  ErrorLog ${APACHE_LOG_DIR}/aaa_error.log
  CustomLog ${APACHE_LOG_DIR}/aaa_access.log combined
</VirtualHost>
```

B3.2 bbb.comed.net

```
nano /etc/apache2/sites-available/bbb.comed.net.conf
<VirtualHost *:80>
  ServerName bbb.comed.net
  DocumentRoot /var/www/html/bbb.comed.net

  <Directory /var/www/html/bbb.comed.net>
    AllowOverride All
    Require all granted
  </Directory>

  ErrorLog ${APACHE_LOG_DIR}/bbb_error.log
  CustomLog ${APACHE_LOG_DIR}/bbb_access.log combined
</VirtualHost>
```

B3.3 ccc.comed.net

```
nano /etc/apache2/sites-available/ccc.comed.net.conf
<VirtualHost *:80>
  ServerName ccc.comed.net
  DocumentRoot /var/www/html/ccc.comed.net

  <Directory /var/www/html/ccc.comed.net>
    AllowOverride All
    Require all granted
  </Directory>

  ErrorLog ${APACHE_LOG_DIR}/ccc_error.log
  CustomLog ${APACHE_LOG_DIR}/ccc_access.log combined
</VirtualHost>
```

B3.4 ddd.comed.net

```
nano /etc/apache2/sites-available/ddd.comed.net.conf
<VirtualHost *:80>
    ServerName ddd.comed.net
    DocumentRoot /var/www/html/ddd.comed.net

    <Directory /var/www/html/ddd.comed.net>
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/ddd_error.log
    CustomLog ${APACHE_LOG_DIR}/ddd_access.log combined
</VirtualHost>
```

B4) Enable vhost และ Reload Apache

```
a2ensite aaa.comed.net.conf
a2ensite bbb.comed.net.conf
a2ensite ccc.comed.net.conf
a2ensite ddd.comed.net.conf
```

```
a2dissite 000-default.conf
apache2ctl configtest
systemctl reload apache2
```

ตรวจสอบว่า vhost ถูกประกาศจริง

```
apache2ctl -S
```

PART C: ตั้งค่า DNS Resolver บน CLIENT และทดสอบด้วย nslookup

C1) ตั้ง /etc/resolv.conf บน CLIENT ให้ชี้ DNS มาที่ Server

ทำบน CLIENT (root)

```
nano /etc/resolv.conf
```

ใส่

```
nameserver 192.168.10.10
```

ตรวจสอบ

```
cat /etc/resolv.conf
```

C2) ทดสอบ nslookup จาก CLIENT (ต้องได้ IP ถูกต้อง)

```
nslookup aaa.comed.net
```

```
nslookup bbb.comed.net
```

```
nslookup ccc.comed.net
```

```
nslookup ddd.comed.net
```

PART D: Upload เว็บไซต์จาก CLIENT ไป SERVER

D1) โครงสร้างไฟล์เว็บที่เตรียมไว้บน Client

```
aaa/index.html -> มีข้อความ "AAA WEB OK"
```

```
bbb/index.html -> มีข้อความ "BBB WEB OK"
```

```
ccc/index.html -> มีข้อความ "CCC WEB OK"
```

```
ddd/index.html -> มีข้อความ "DDD WEB OK"
```

D2) Upload ด้วย scp (จาก CLIENT)

```
scp aaa/index.html
root@192.168.10.10:/var/www/html/aaa.comed.net/
scp bbb/index.html
root@192.168.10.10:/var/www/html/bbb.comed.net/
scp ccc/index.html
root@192.168.10.10:/var/www/html/ccc.comed.net/
scp ddd/index.html
root@192.168.10.10:/var/www/html/ddd.comed.net/
```

PART E: ตรวจสอบผลลัพธ์

E1) ตรวจสอบไฟล์บน SERVER

ทำบน SERVER

```
ls -l /var/www/html/aaa.comed.net
ls -l /var/www/html/bbb.comed.net
ls -l /var/www/html/ccc.comed.net
ls -l /var/www/html/ddd.comed.net
```

ต้องมี index.html ทุกโฟลเดอร์

E2) ทดสอบผ่าน Browser จาก CLIENT

- <http://aaa.comed.net> → ต้องแสดง **AAA WEB OK**
 - <http://bbb.comed.net> → ต้องแสดง **BBB WEB OK**
 - <http://ccc.comed.net> → ต้องแสดง **CCC WEB OK**
 - <http://ddd.comed.net> → ต้องแสดง **DDD WEB OK**
-

PART F: Troubleshooting แบบเร็ว (แยกเป็นชั้น)

1) DNS ไม่ตอบ / ตอบผิด

บน Server

```
named-checkconf
named-checkzone comed.net /etc/bind/zones/db.comed.net
systemctl status bind9 --no-pager
journalctl -xeu bind9
```

ตรวจ resolver (ทั้ง Server/Client)

```
cat /etc/resolv.conf
```

ต้องมี nameserver 192.168.10.10

2) DNS ตอบแล้ว แต่เว็บไม่ขึ้น

บน Server

```
apache2ctl -S
systemctl status apache2 --no-pager
```

ดู log

```
tail -n 50 /var/log/apache2/aaa_error.log
tail -n 50 /var/log/apache2/aaa_access.log
```
