การติดตั้ง FreeRadius บน Ubuntu Server

1. เตรียมระบบ Ubuntu

อัปเดตแพ็กเกจของระบบ (ควรทำเสมอเพื่อความปลอดภัยและเสถียรภาพ)

```
# apt update
# apt upgrade -y
```

(แนะนำ) ติดตั้งเครื่องมือเสริมที่อาจจำเป็น เช่น:

```
# apt install vim net-tools ufw -y
```

ตรวจสอบ Firewall (กรณีใช้ UFW)

เปิดใช้งาน UFW ถ้ายังไม่เคยเปิด

```
# ufw enable
```

อนุญาต SSH (กรณีรีโมตเข้ามา)

```
# ufw allow ssh
```

(ในภายหลัง) จะต้องอนุญาตพอร์ต UDP 1812 (Authentication) และ 1813 (Accounting) หากต้องใช้

```
# ufw allow 1812/udp
# ufw allow 1813/udp
```

2. ติดตั้ง FreeRADIUS

ติดตั้งแพ็กเกจหลัก

```
# apt install freeradius -y
```

แพ็กเกจหลักคือ freeradius ซึ่งจะติดตั้งตัว Daemon และไฟล์คอนฟิกพื้นฐาน

(แนะนำ) ติดตั้งเครื่องมือทดสอบ freeradius-utils

```
# apt install freeradius-utils -y
```

จะมีคำสั่ง radtest กับ radclient สำหรับการส่งคำขอ (Request) ทดสอบไปยัง RADIUS Server

เมื่อเสร็จแล้ว ไฟล์คอนฟิกหลักและโมดูลต่าง ๆ ของ FreeRADIUS จะอยู่ใน /etc/freeradius/3.0/ (หรือชื่อโฟลเดอร์ใกล้เคียงกัน ขึ้นกับเวอร์ชัน FreeRADIUS/Ubuntu)

3. ตรวจสอบและตั้งค่าเบื้องต้น

3.1 ตรวจสอบสถานะบริการ

หลังติดตั้ง ให้ตรวจสอบสถานะของ FreeRADIUS:

```
# systemctl status freeradius
```

กรณีมันยังไม่เริ่มทำงาน (inactive) สามารถสั่งให้เริ่มและให้ทำงานอัตโนมัติได้:

```
# systemctl enable --now freeradius
# systemctl start freeradius
```

หากคำสั่งเรียบร้อย จะได้สถานะ “active (running)”

3.2 แก้ไขการฟังพอร์ต (กรณีต้องการระบุ IP เฉพาะ)

ค่าเริ่มต้น FreeRADIUS จะฟัง (Listen) ทุกอินเทอร์เฟซ (0.0.0.0) บนพอร์ต 1812/udp (Auth) และ 1813/udp (Accounting) อยู่แล้ว สามารถตรวจสอบได้ที่ไฟล์:

/etc/freeradius/3.0/sites-enabled/default (สำหรับคำสั่งที่เกี่ยวกับการฟังพอร์ต)

```
# vim /etc/freeradius/3.0/sites-enabled/default
listen {
  type = auth
  ipaddr = 0.0.0.0       # หรือจะใส่ * ก็ได้
  port = 0               # หรือจะใส่ 1812 ก็ได้
}
listen {
  type = acct
  ipaddr = 0.0.0.0       # หรือจะใส่ * ก็ได้
  port = 0               # หรือจะใส่เป็น 1813 ก็ได้
}
```

หากต้องการให้ฟังเฉพาะ Public IP หรือ IP ใด IP หนึ่ง ก็ปรับค่า ipaddr เป็น IP ตามต้องการ

4. ตั้งค่า Client (clients.conf)

Clients ในที่นี้หมายถึงอุปกรณ์หรือบริการ (NAS, Access Point, VPN Server, หรืออะไรก็ตาม) ที่จะส่งคำขอ Authentication มาให้ RADIUS Server

ไฟล์คอนฟิกหลักอยู่ที่: /etc/freeradius/3.0/clients.conf

เปิดไฟล์ (ด้วย vim, nano หรือเครื่องมือที่ถนัด) เพื่อเพิ่มหรือแก้ไขการกำหนดค่า client เช่น

```
# vim /etc/freeradius/3.0/clients.conf
```

ตัวอย่างการกำหนด client:

```
client MyRouter {
  ipaddr  = 192.168.1.1       # IP ของอุปกรณ์/ระบบที่จะเชื่อมต่อ ต้องเป็นหมายเลข IP นี้เท่านั้น
  secret  = testing123        # Shared Secret ต้องตรงกันทั้งฝั่ง RADIUS และอุปกรณ์
  nastype = other             # ระบุประเภท เช่น cisco, other, etc. ไม่ใส่ก็ได้
}

client MyPublicClient {
  ipaddr  = 0.0.0.0/0         # IP Public ของระบบภายนอก หรือ จากหมายเลข IP อะไรก็ได้
  secret  = VerySecretKey2023
  nastype = other
}
```

ipaddr: IP ที่จะส่งคำขอ RADIUS จากฝั่ง Client

secret: Shared Secret ที่ต้องเซตเหมือนกันทั้งฝั่ง RADIUS และฝั่ง Client ควรใช้รหัสที่เดายากเพื่อความปลอดภัย

บันทึกไฟล์ แล้วรีสตาร์ต FreeRADIUS เพื่อให้คอนฟิกใหม่ทำงาน:

```
# systemctl restart freeradius
```

5. เพิ่มผู้ใช้ (Users) สำหรับ Authentication

มีหลายวิธีให้ FreeRADIUS รู้จัก “ชื่อผู้ใช้-รหัสผ่าน” สำหรับตรวจสอบสิทธิ์ (Authentication) เช่น

เก็บในไฟล์ /etc/freeradius/3.0/users (เรียบง่าย)

เก็บใน Database (MySQL/PostgreSQL) ผ่านโมดูล SQL

เชื่อมต่อกับ LDAP / Active Directory

ในตัวอย่างนี้ จะแสดงวิธีเก็บในไฟล์ users แบบง่ายที่สุด

5.1 แก้ไฟล์ /etc/freeradius/3.0/users

```
# vim /etc/freeradius/3.0/users
```

เพิ่มบรรทัดตัวอย่าง (โดยปกติในไฟล์จะมีตัวอย่างอยู่แล้ว):

```
pornchai     Cleartext-Password := "toonkaew"
             Reply-Message := "Hello, Pornchai. You are authenticated!"
```

คำอธิบาย: pornchai: ชื่อผู้ใช้ (Username)

Cleartext-Password := "toonkaew": รหัสผ่านเป็น "toonkaew" โดยระบุเป็น Cleartext-Password

Reply-Message: ข้อความตอบกลับ (Optional)

หากต้องการเพิ่มหลายคน ก็เพิ่มทีละบรรทัดในไฟล์นี้ได้

บันทึกไฟล์ แล้วรีสตาร์ต FreeRADIUS:

```
# systemctl restart freeradius
```

6. ทดสอบการทำงานด้วยคำสั่ง radtest

6.1 ติดตั้งเครื่องมือทดสอบ (หากยังไม่ได้ติดตั้ง)

```
# apt install freeradius-utils -y
```

6.2 ทดสอบจาก เครื่องเดียวกัน (Loopback)

ใช้คำสั่ง radtest เพื่อทดสอบส่ง RADIUS Request (Access-Request) มายังเซิร์ฟเวอร์

รูปแบบคำสั่ง radtest <Username> <Password> <Server> <Nas_port> <Secret> [Options...] ** แต่ละค่า เว้นว่าง 1 วรรค

ลำดับ	ความหมาย	ตัวอย่าง
Username	ชื่อผู้ใช้งาน (String)	pornchai
Password	รหัสผ่าน	toonkaew
Server	IP หรือ DNS ของ Radius Server	127.0.0.1
Nas_port	หมายเลข NAS-Port (Integer)	0
Secret	Shared Secret ตรงกับที่กำหนดใหน client แต่ละอันใน clients.conf	testing123
Options	ออปชันเพิ่มเติม (ไม่บังคับ)	

ตัวอย่าง

```
# radtest pornchai toonkaew 127.0.0.1 0 testing123
```

อธิบายพารามิเตอร์:

testuser testpass: User/Pass ที่เราตั้งในไฟล์ /etc/freeradius/3.0/users

127.0.0.1: IP ของ RADIUS Server (กรณีทดสอบบนเครื่องเดียวกัน)

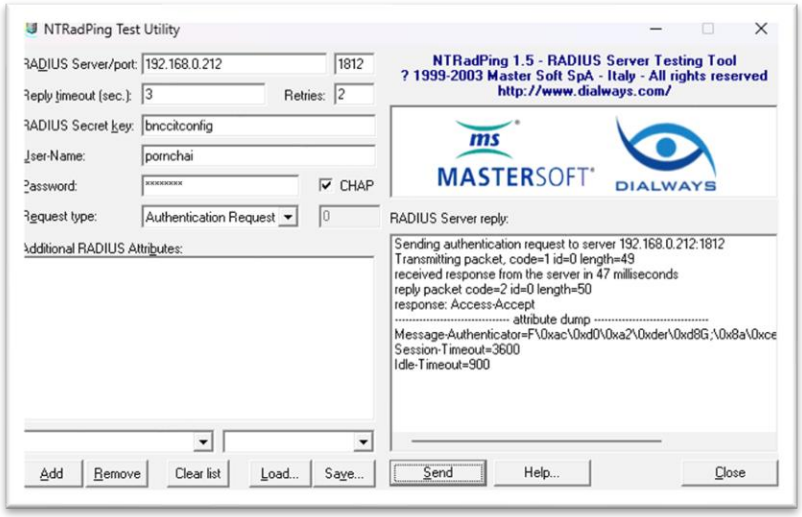
0: หมายเลข NAS-Port (ใส่ 0 ไปก่อน)

testing123: Shared Secret ที่กำหนดใน clients.conf ของ 127.0.0.1 (ถ้าไม่ได้ระบุ อาจเป็นค่า default “testing123”)

ถ้าทุกอย่างถูกต้อง จะได้รับผลลัพธ์ประมาณนี้:

```
# radtest pornchai toonkaew 127.0.0.1 0 testing123
Sent Access-Request Id 188 from 0.0.0.0:58107 to 127.0.0.1:1812 length 78
        User-Name = "pornchai"
        User-Password = "toonkaew"
        NAS-IP-Address = 192.168.0.212
        NAS-Port = 0
        Message-Authenticator = 0x00
        Cleartext-Password = "toonkaew"
Received Access-Accept Id 188 from 127.0.0.1:1812 to 127.0.0.1:58107 length 50
        Message-Authenticator = 0x419c658b30e53d1ee3d09cdc30ba46cd
        Session-Timeout = 3600
        Idle-Timeout = 900
```

หรือผ่านโปรแกรม NTRadPing Test Utility



แปลว่ายืนยันตัวตนผ่านเรียบร้อย

ถ้าได้รับ Access-Reject หรือไม่มีการตอบสนอง ให้ตรวจสอบว่า:

Shared Secret ตรงกันหรือไม่

IP client ถูกต้องใน clients.conf หรือไม่

ค่าผู้ใช้/รหัสผ่านถูกต้องหรือไม่

FreeRADIUS รันอยู่จริงหรือไม่ (sudo systemctl status freeradius)

6.3 ทดสอบจาก เครื่องภายนอก (ต่างเครื่อง, IP อื่น)

สมมติคุณมีเครื่องภายนอก (อาจเป็น Windows/Mac/Linux) และติดตั้งโปรแกรมส่ง RADIUS Request (เช่น radclient, NTRadPing บน Windows หรือใช้ radtest บน Linux เครื่องอื่น)

ปลายทางคือ IP ของ RADIUS Server ที่เป็น Public IP หรือ IP ภายใน (หากอยู่ในวง LAN เดียวกัน)

Shared Secret ต้องเหมือนกับค่าที่กำหนดใน clients.conf

ตัวอย่าง (บนเครื่องอื่นที่เป็น Ubuntu/Debian):

```
#radtest testuser testpass 127.0.0.1 0 testing123
```

หากคอนฟิกถูกต้อง และ Firewall เปิดพอร์ต UDP 1812 แล้ว ควรได้รับ Access-Accept

7. ตั้งค่า Firewall และการเปิดใช้งานบน Public IP

หากต้องการให้ RADIUS Server รับคำขอจากภายนอก (อินเทอร์เน็ต) หรือวงอื่น ๆ จะต้อง:

เปิดพอร์ต 1812/udp (และ 1813/udp ถ้าต้องการ Accounting) บน UFW หรือไฟร์วอลล์ระดับระบบปฏิบัติการ:

```
#ufw allow 1812/udp
#ufw allow 1813/udp
```

ตั้งค่า Router/Firewall ภายนอก (ถ้ามี) ให้ทำ NAT/Port Forwarding หรือใช้ Public IP ตรง ๆ แล้วส่งเข้ามาที่ RADIUS Server

กำหนด client ใน clients.conf ให้ตรงกับ IP Public หรือ IP ของอุปกรณ์ต้นทาง พร้อม Shared Secret

8. ติดตาม Log และจัดการ

8.1 ดู Log ของ FreeRADIUS

ไฟล์ Log หลักของ FreeRADIUS มักอยู่ที่:

/var/log/freeradius/radius.log หรือ /var/log/freeradius/radiusd.log (ขึ้นกับเวอร์ชันและการตั้งค่า)

สามารถดูแบบ Real-time ด้วย:

```
# tail -f /var/log/freeradius/radius.log
```

หรือ

```
# journalctl -u freeradius -f
```

(ดู Log ของ Systemd)

8.2 ตรวจสอบการทำงานเพิ่มเติม

ถ้าต้องการ Debug หรือดูรายละเอียดขั้นสูง สามารถหยุดบริการ freeradius แล้วสั่งรันโหมด Debug:

```
# systemctl stop freeradius
# freeradius -X
```

RADIUS จะรันหน้าจอ Foreground พร้อมพ่นข้อความ Log แบบละเอียด ซึ่งช่วยระบุปัญหาได้ดี

9. ตัวอย่างคอนฟิกไฟล์อย่างย่อ

```
# vim /etc/freeradius/3.0/clients.conf

client localhost {
  ipaddr  = 127.0.0.1
  secret  = testing123
  nas_type = other
}

client MyRouter {
  ipaddr  = 192.168.1.1
  secret  = fixedbnccitconfig
  nas_type = other
}

client MyPublicClient {
  ipaddr  = 0.0.0.0/0
  secret  = publicbnccitconfig
  nas_type = other
}
```

10 การจัดการผู้ใช้งาน โดยใช้ไฟล์ user

```
# vim /etc/freeradius/3.0/users
testuser    Cleartext-Password := "testpass"
            Reply-Message := "Hello, testuser. You are authenticated!"
adminuser   Cleartext-Password := "admin123"
            Reply-Message := "Welcome, admin!"
```

หลังแก้ไขไฟล์ใด ๆ ใน /etc/freeradius/3.0/ อย่าลืม

```
# systemctl restart freeradius
```

กรณีที่ต้องการกำหนดค่า login สำหรับ pfSense และ Mikrotik โดย

จำกัดให้ผู้ใช้งาน 1 คน Login ได้ไม่เกิน 3 อุปกรณ์ (Simultaneous-Use = 3)

ถ้าไม่มีการใช้งานเกิน 15 นาทีให้ตัดการเชื่อมต่อ (Idle-Timeout = 900 วินาที)

อยู่ในระบบ (Session) ได้สูงสุด 4 ชั่วโมงต่อการ Login 1 ครั้ง (Session-Timeout = 14400 วินาที)

จำกัดอัปโหลด 10 Mbps

จำกัดดาวน์โหลด 20 Mbps

จำกัดเวลาการ Login วันจันทร์–ศุกร์ 08:00–16:00

จำกัดเวลาการ Login วันเสาร์–อาทิตย์ 10:00–20:00

```
# vim /etc/freeradius/3.0/users
```

ตั้งค่าผู้ใช้พื้นฐาน (ไม่มีการจำกัดเวลา)

```
employee    Cleartext-Password := "password123"
            Simultaneous-Use := 3,
            Idle-Timeout := 900,
            Session-Timeout := 14400,
            WISPr-Bandwidth-Max-Up := 10000000,
            WISPr-Bandwidth-Max-Down := 20000000,
            Mikrotik-Rate-Limit := "10M/20M"
```

 # ให้ employee ใช้ได้ จันทร์–ศุกร์ 08:00–16:00

```
Employee    Cleartext-Password := "password123", Time == "Wk0800-1600"
            Simultaneous-Use := 3,
            Idle-Timeout := 900,
            Session-Timeout := 14400,
            WISPr-Bandwidth-Max-Up := 10000000,
            WISPr-Bandwidth-Max-Down := 20000000,
            Mikrotik-Rate-Limit := "10M/20M"
```

 # ให้ employee ใช้ได้ เฉพาะวันเสาร์ 10:00–20:00

```
employee    Cleartext-Password := "password123", Time == "Sa1000-2000"
            Simultaneous-Use := 3,
            Idle-Timeout := 900,
            Session-Timeout := 14400,
            WISPr-Bandwidth-Max-Up := 10000000,
            WISPr-Bandwidth-Max-Down := 20000000,
            Mikrotik-Rate-Limit := "10M/20M"
```

 # ให้ employee ใช้ได้ เฉพาะวันอาทิตย์ 10:00–20:00

```
employee    Cleartext-Password := "password123", Time == "Su1000-2000"
            Simultaneous-Use := 3,
            Idle-Timeout := 900,
            Session-Timeout := 14400,
            WISPr-Bandwidth-Max-Up := 10000000,
            WISPr-Bandwidth-Max-Down := 20000000,
            Mikrotik-Rate-Limit := "10M/20M"
```

ในเอกสาร FreeRADIUS ระบุว่าสามารถใช้ Time == "Wk0800-1600|Sa1000-2000|Su1000-2000" ได้ในบรรทัดเดียว แต่บางครั้งอาจเกิดปัญหา match ยาก ควรทดสอบดู หรือใช้วิธีเขียนหลายบรรทัดเพื่อความชัดเจน

---

11. คำแนะนำด้านความปลอดภัย

Shared Secret: ควรตั้งให้เดายาก ผสมอักษรตัวเล็ก/ใหญ่ ตัวเลข และเครื่องหมายพิเศษ

จำกัด IP ต้นทาง: ถ้าเปิด RADIUS สู่ Public Internet ควรกำหนดใน clients.conf เฉพาะ IP หรือ Subnet ที่ต้องการเท่านั้น อย่าเปิดกว้าง 0.0.0.0/0

Firewall: เปิดเฉพาะพอร์ตที่จำเป็น และอาจทำ ACL บน Router เพิ่มเติมเพื่อป้องกันการสแกน/โจมตีจาก IP แปลกปลอม

อัปเดตสม่ำเสมอ: คอยอัปเดต Patch ของ Ubuntu และ FreeRADIUS เพื่อลดช่องโหว่

การเข้ารหัส (EAP/PEAP): หากใช้ FreeRADIUS สำหรับ Wi-Fi (802.1X) ควรเปิดใช้ EAP/PEAP เพื่อเข้ารหัสข้อมูล ไม่ให้ผู้ใช้ส่งรหัสผ่านเป็น Plain text

2. การบริหารจัดการ User โดยใช้ Database

1.ติดตั้ง MariaDB (ตัวอย่างบน Ubuntu)

```
# apt update
# apt install mariadb-server mariadb-client freeradius-mysql -y
# systemctl enable --now mariadb
```

หรือถ้าต้องการ MySQL: (ให้เลือกติดตั้งอย่างใดอย่างหนึ่ง)

```
# apt update
# apt install mysql-server mysql-client freeradius-mysql -y
# systemctl enable --now mysql
```

2.ตั้งรหัสผ่าน root ของ DB และปรับ secure installation

```
# mysql_secure_installation
```

ทำตามขั้นตอน (ตั้งรหัสผ่าน, ลบ user/table ตัวอย่าง ฯลฯ)

3.สร้างฐานข้อมูลและผู้ใช้สำหรับ FreeRADIUS (ตัวอย่างคำสั่งใน MariaDB/MySQL)

```
# mysql -u root -p
> CREATE DATABASE radius_db;
> CREATE USER 'radius_user'@'localhost' IDENTIFIED BY 'radius_pass123';
> GRANT ALL PRIVILEGES ON radius_db.* TO 'radius_user'@'localhost';
> FLUSH PRIVILEGES;
> exit;
```

4.สร้างตารางตามสคีมา (Schema) ของ FreeRADIUS

ใน FreeRADIUS (แพ็กเกจ freeradius-mysql) มักมีไฟล์สคีมา SQL ให้ โดยปกติอาจอยู่ที่ /etc/freeradius/3.0/mods-config/sql/main/mysql/schema.sql หรือใกล้เคียง

รันคำสั่งเพื่อสร้างตารางใน radius_db

```
# mysql -u radius_user -p radius_db < /etc/freeradius/3.0/mods-config/sql/main/mysql/schema.sql
```

5.ตั้งค่าไฟล์ mods-available/sql แล้วลิงก์ไปที่ mods-enabled/sql

เปิดไฟล์ /etc/freeradius/3.0/mods-available/sql แก้ไขส่วน sql { ... } เช่น

```
# vim /etc/freeradius/3.0/mods-available/sql
dialect = "mysql"              #  Line 40
driver = "rlm_sql_mysql"       # Line 60
server = "localhost"           # Line 167
port = 3306                    # Line 168
login = "radius_user"          # Line 169
password = "radius_pass123"    # Line 170
radius_db = "radius_db"        # Line 185
```

จากนั้น สร้าง Link ไฟล์ sql ใน mods-enabled ให้อ้างอิงมายัง mods-available

```
# ln -s /etc/freeradius/3.0/mods-available/sql  /etc/freeradius/3.0/mods-enabled/sql
```

หรือ

```
# cd /etc/freeradius/3.0/mods-enabled/
# ln -s ../mods-available/sql  sql
```

6.ปรับไฟล์ sites-enabled/default ส่วน authorize {}, accounting {}, session {}, post-auth {} เพื่อเปิดใช้โมดูล sql ในบางครั้ง จะเป็น -sql ให้นำเครื่องหมาย - หน้า sql ออก หรือบางครั้ง เป็น #sql ให้เอาเครื่องหมาย # หน้า sql ออก

```
# vim  /etc/freeradius/3.0/sites-enabled/default
server default {
  listen {
                                          # พอร์ต / IP ที่รับคำขอ RADIUS Auth
  }
  listen {
                                          # พอร์ต / IP สำหรับ Radius Accounting
  }
  authorize {
    # ขั้นตอน authorize
    ...
    # -------- ตัวอย่างการเปิดใช้ sql --------
    sql                                   # Line 442
    ...
  }
  authenticate {
    ...
    # ส่วนใหญ่ไม่ต้องใส่ sql ใน block นี้
    ...
  }
  preacct {
    ...
    # ถ้าต้องการทำอะไรกับข้อมูลก่อน Accounting
    ...
  }
  accounting {
    # ขั้นตอนเก็บข้อมูล Accounting
    ...
    # -------- ตัวอย่างการเปิดใช้ sql --------
    sql                                   # Line 715
    ...
  }
  session {
    ...
    # บางครั้งใส่ sql เพื่อให้ตรวจสอบ session หรือ concurrency (ในบาง config จะให้ sql ทำงานที่ authorize แทน)
    sql                                   # Line 757
    ...
  }
  post-auth {
    ...
    # หลังยืนยันตัวตนเสร็จ ถ้าต้องการเขียน log ลง db
    sql                                   # Line 871
    ...
  }
}
```

*** ถ้าเราไม่ได้ต้องการเชื่อมต่อ MySQL ผ่าน SSL/TLS ก็สามารถคอมเมนต์บล็อก tls { ... } ทั้งหมด หรือเฉพาะบรรทัด ca_file ได้เลย ตัวอย่าง:

```
# vim /etc/freeradius/3.0/mods-enabled/sql
    ...
mysql {
    ...
    #tls {                                                    # Line 86 - 96
    #     ca_file = "/etc/ssl/certs/my_ca.crt"
    #     ca_path = "/etc/ssl/certs"
    #     certificate_file = "/etc/ssl/certs/client.crt"
    #     private_key_file = "/etc/ssl/private/client.key"
    #     # หรืออะไรก็ตามภายใน block นี้
    #}
    ...
}
```

7.รีสตาร์ต FreeRADIUS

```
# systemctl restart freeradius
```

8 เพิ่มข้อมูลผู้ใช้ และกลุ่มผู้ใช้

```
# mysql -u radius_user -p
> USE radius_db;
```

-- 1) สร้าง user "pornchai" ใน radcheck

```
>INSERT INTO radcheck (username, attribute, op, value) VALUES ('pornchai', 'Cleartext-Password', ':=', 'toonkaew');
```

-- 2) ผูก user "pornchai" เข้ากลุ่ม "Admin"

```
>INSERT INTO radusergroup (username, groupname, priority) VALUES ('pornchai', 'Admin', 0);
```

-- 3) กำหนดเงื่อนไขพื้นฐานในระดับกลุ่ม (Admin) โดยใช้ Check Attribute

```
>INSERT INTO radgroupcheck (groupname, attribute, op, value) VALUES ('Admin', 'Simultaneous-Use', ':=', '3');
```

-- 4) กำหนดการตอบกลับสำหรับกลุ่ม (Admin) โดยใช้ Reply Attribute

```
>INSERT INTO radgroupreply (groupname, attribute, op, value) VALUES ('Admin', 'Idle-Timeout', ':=', '900');
>INSERT INTO radgroupreply (groupname, attribute, op, value) VALUES ('Admin', 'Session-Timeout', ':=', '14400');
>INSERT INTO radgroupreply (groupname, attribute, op, value) VALUES ('Admin', 'WISPr-Bandwidth-Max-Up', ':=', '100000000');
>INSERT INTO radgroupreply (groupname, attribute, op, value) VALUES ('Admin', 'WISPr-Bandwidth-Max-Down', ':=', '200000000');
>INSERT INTO radgroupreply (groupname, attribute, op, value) VALUES ('Admin', 'Mikrotik-Rate-Limit', ':=', '100M/200M');
```

ความหมายของ Check Attribute ที่ใช้ในตาราง radcheck หรือ radgroupcheck ห้ามนำไปใส่ใน radreply หรือ radgroupreply เพราะเป็นการตรวจสอบ

Attribute	ความหมาย
Cleartext-Password	รหัสผ่าน
Simultaneous-Use	การเข้าใช้งานพร้อมกัน
Time	ช่วงเวลา Time == "[day-abbrev][start]-[end][| ...]" • day-abbrev: เป็นตัวย่อของวันต่าง ๆ เช่น Mo (Monday) , Tu (Tuesday) , We (Wednesday) , Th (Thursday) , Fr (Friday) , Sa (Saturday) , Su (Sunday) Wk (Weekdays = Mo-Fr) , Wd (Weekend = Sa,Su) (บางเวอร์ชันอาจไม่รองรับย่อ “Wd” แต่สามารถระบุ Sa,Su ได้เอง) • start-end: ระบุเวลาในรูปแบบ HHMM-HHMM (24 ชั่วโมง) เช่น 0800-1600 • สามารถใช้ เครื่องหมาย “|” (vertical bar) เพื่อ “OR” กันหลายช่วง ตัวอย่าง: Mo0800-1600|We0800-1600|Fr0800-1600 #ถ้าระบุ Time == "0800-1600" โดยไม่ใส่วัน หมายถึง “ทุกวัน 08:00–16:00” ***บางครั้งสามารถเขียน “Tu0800-1000,Tu1200-1400” เพื่อระบุหลายช่วงเวลาในวันเดียวกันได้ด้วยการคั่นด้วยเครื่องหมาย | หรือ , แล้วแต่เวอร์ชัน (ต้องทดสอบ)

ความหมายของ Reply Attribute ที่ใช้ในตาราง radreply หรือ radgroupreply ห้ามนำไปใส่ใน radcheck หรือ radgroupcheck เพราะเป็นการตอบกลับ

Attribute	ความหมาย
Idle-Timeout	เมื่อไม่ได้ใช้งาน ระบบจะตัดการเชื่อมต่อ ต้อง Login ใหม่
Session-Timeout	เมื่อใช้งานต่อครั้งเกิดเวลาที่กำหนด ระบบจะตัดการเชื่อมต่อ ต้อง Login ใหม่
WISPr-Redirection-URL	กำหนด URL เมื่อ login สำเร็จ เป็นมาตรฐาน WISPr
WISPr-Bandwidth-Max-Up	การจัดการ Bandwidth Upload ระบุเป็น bit/s ไม่มีหน่วย ระบุเป็นตัวเลข เป็นมาตรฐาน WISPr
WISPr-Bandwidth-Max-Down	การจัดการ Bandwidth Download ระบุเป็น bit/s ไม่มีหน่วย ระบุเป็นตัวเลข เป็นมาตรฐาน WISPr

Mikrotik-Rate-Limit	การจัดการ Bandwidth สำหรับ Mikrotik
	"10M/20M" หรือ "10M/20M 20M/40M 15M/30M 10 8"
	Mikrotik-Rate-Limit เป็น Vendor-Specific Attribute (VSA) ของ MikroTik เพื่อกำหนด Profile ความเร็ว (Rate) และพารามิเตอร์ต่าง ๆ ที่เกี่ยวกับ Bandwidth Shaping ให้กับผู้ใช้งาน (PPP, Hotspot หรือ Wireless) ผ่าน RADIUS
	โดยรูปแบบคำสั่งของ Mikrotik-Rate-Limit ที่ MikroTik กำหนดจะอยู่ในโครงสร้าง:
	`rx-rate[/tx-rate]`
	`[ rx-burst-rate[/tx-burst-rate]`
	`[ rx-burst-threshold[/tx-burst-threshold]`
	`[ rx-burst-time[/tx-burst-time]`
	`[ priority`
	`[ rx-min-rate[/tx-min-rate] ]`
	`]`
	`]`
	`]`
	`]`
	ซึ่งแต่ละส่วนหมายถึง
	rx-rate / tx-rate ความเร็วรับ (download) และส่ง (upload) ปกติ (Maximum Limit)
	rx-burst-rate / tx-burst-rate ความเร็วสูงสุดที่สามารถ “กระชาก” (Burst) ได้ชั่วขณะ (เกินกว่า rx/tx-rate ที่กำหนดไว้ปกติ)
	rx-burst-threshold / tx-burst-threshold ระดับปริมาณรับ/ส่งที่จะเป็นตัวตัดสินว่า “จะเข้าสู่โหมด Burst หรือไม่”
	rx-burst-time / tx-burst-time ระยะเวลา (วินาที) ที่อนุญาตให้วิ่งด้วย burst-rate ก่อนจะถูกลดลงมาตาม threshold
	priority ระดับความสำคัญของ traffic (1–8) ยิ่งเลขต่ำหมายถึงยิ่งสำคัญมาก (หรือในบางเอกสารบอก 8 เป็นค่าต่ำสุด)
	rx-min-rate / tx-min-rate ความเร็วขั้นต่ำที่จะไม่ถูกจำกัดต่ำกว่านี้ แม้จะโดน Queue หรือ Burst หมดแล้วก็ตาม
	หมายเหตุ: ไม่จำเป็นต้องใส่ทุกค่า หากไม่ใส่ (ละเว้น) ส่วนไหน ระบบจะใช้ค่า default หรือปล่อยว่างตามปรกติ

การบริหารจัดการ FreeRadius ด้วย Platform

ในการบริหารจัดการ FreeRadius จะมี Platform มากมายให้เลือก หรืออาจจะพัฒนาขึ้นมาด้วยตนเอง แต่ยังมี Platform ที่เป็นที่นิยม โดยในหน่วยนี้จะใช้ DaloRadius ในการบริหารจัดการ FreeRadius

1.ติดตั้ง Apache และ phpMyAdmin เพื่อความสะดวกในการใช้งาน โดยใช้คำสั่ง

```
# apt install phpmyadmin
```

หลังจากนั้นก็ทำตามที่โปรแกรมแนะนำตามขั้นตอน

2.ติดตั้ง Library ของ PHP ที่จำเป็นต้องใช้ โดยใช้คำสั่ง

```
# apt -y install libapache2-mod-php php-{gd,common,mail,mail-mime,mysql,pear,db,mbstring,xml,curl,zip}
```

3. แก้ไขไฟล์ /etc/freeradius/3.0/mods-enabled/sql ด้วยคำสั่ง

```
# vim /etc/freeradius/3.0/mods-enabled/sql
```

ค้นหาบรรทัดที่เขียนว่า `read_client = yes` โดยให้เอาเครื่องหมาย # ด้านหน้าออก

และ

ค้นหาบรรทัดที่เขียนว่า `client_table = "nas"` โดยให้เอาเครื่องหมาย # ด้านหน้าออก

4.เปลี่ยนกลุ่ม และเจ้าของไฟล์ ให้เป็นของ freerad:freerad ของไฟล์ และ Restart Service ของ FreeRadius ด้วยคำสั่ง

```
# chgrp -h freerad /etc/freeradius/3.0/mods-available/sql
# chown -R freerad:freerad /etc/freeradius/3.0/mods-enabled/sql
# systemctl restart freeradius.service
```

5. ติดตั้ง DaloRadius ด้วยการดึงมาจาก Github ด้วย

```
# cd \       (Enter 2 ครั้ง เพื่อกลับมายัง Home Directory ของตนเอง หากเป็น root อาจใช้คำสั่ง cd /root แทนได้)
# apt -y install git
# git clone https://github.com/lirantal/daloradius.git
```

6. นำโครงสร้างตาราง และข้อมูลของเบื้องต้นของ Daloradius เข้าฐานข้อมูล ด้วยคำสั่ง

```
# mysql -u radius_user -p radius_db < daloradius/contrib/db/fr3-mariadb-freeradius.sql
# mysql -u radius_user -p radius_db < daloradius/contrib/db/mariadb-daloradius.sql
```

7. ย้าย Directory ของ daloradius ที่อยู่ใน Home Directory ไปยัง DocumentRoot ของเครื่อง (/var/www/html/) ด้วยคำสั่ง

```
# mv daloradius /var/www/html/
```

8. ปรับแต่ง Config ของ Daloradius ด้วยการคัดลอกจากต้นฉบับ โดยใช้คำสั่ง

```
# cd /var/www/html/daloradius/app/common/includes/
# cp daloradius.conf.php.sample daloradius.conf.php
# chown www-data:www-data daloradius.conf.php
# vim daloradius.conf.php
```

และให้แก้ในส่วนของ configValues ได้แก่

```
$configValues['CONFIG_DB_HOST'] = 'localhost';
$configValues['CONFIG_DB_PORT'] = '3306';
$configValues['CONFIG_DB_USER'] = 'radius_user';
$configValues['CONFIG_DB_PASS'] = 'radius_pass123';
$configValues['CONFIG_DB_NAME'] = 'radius_db';
```

9. สร้าง Directory var ใน /var/www/html/daloradius/ พร้อมกับสร้าง Directory log กับ backup พร้อมกำหนดเจ้าของ ด้วยคำสั่ง

```
# cd /var/www/html/daloradius/
# mkdir -p var/{log,backup}
# chown -R www-data:www-data var
```

10. ตั้งค่า Port สำหรับใช้งาน ในกรณีนี้จะใช้ Port เบอร์ 8000

```
tee /etc/apache2/ports.conf<<EOF
Listen 80
Listen 8000

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
EOF
```

11. สร้าง ที่สำหรับเก็บ Logs ของ operators และ users

```
# mkdir -p /var/log/apache2/daloradius/{operators,users}
```

12. สร้าง Virtual Host สำหรับ Operator

```
tee /etc/apache2/sites-available/operators.conf<<EOF
<VirtualHost *:8000>
    ServerAdmin operators@localhost
    DocumentRoot /var/www/html/daloradius/app/operators

    <Directory /var/www/html/daloradius/app/operators>
        Options -Indexes +FollowSymLinks
        AllowOverride None
        Require all granted
    </Directory>

    <Directory /var/www/html/daloradius>
        Require all denied
    </Directory>

    ErrorLog \${APACHE_LOG_DIR}/daloradius/operators/error.log
    CustomLog \${APACHE_LOG_DIR}/daloradius/operators/access.log combined
</VirtualHost>
EOF
```

13. สร้าง Virtual Host สำหรับ User

```
tee /etc/apache2/sites-available/users.conf<<EOF
<VirtualHost *:80>
    ServerAdmin users@localhost
    DocumentRoot /var/www/html/daloradius/app/users

    <Directory /var/www/html/daloradius/app/users>
        Options -Indexes +FollowSymLinks
        AllowOverride None
        Require all granted
    </Directory>

    <Directory /var/www/html/daloradius>
        Require all denied
    </Directory>

    ErrorLog \${APACHE_LOG_DIR}/daloradius/users/error.log
    CustomLog \${APACHE_LOG_DIR}/daloradius/users/access.log combined
</VirtualHost>
EOF
```

14. ปรับแต่งการใช้งานของ Virtual Host

14.1 ยกเลิกการใช้งาน ค่า Default ด้วยคำสั่ง

```
# a2dissite 000-default.conf
```

14.2 เพิ่มการใช้งาน VirtualHost ของ operators และ users ด้วยคำสั่ง

```
# a2ensite users.conf operators.conf
```

14.3 ทำการ Restart Service ของ Apache ด้วยคำสั่ง

```
# systemctl restart apache2 freeradius
```

15. อนุญาตให้เข้าถึง ผ่าน Port 80 และ 8000 ด้วยคำสั่ง

```
# ufw allow http
# ufw allow 8000/tcp
```

16. ทดสอบเข้าใช้งาน ในส่วนของ Operators โดยการเปิด Web Browser แล้วพิมพ์ หมายเลขไอพี ตามด้วย Port 8000 ด้วยรูปแบบ http://<IP>:8000 เช่น

http://192.168.0.205:8000

โดยจะใช้ Username เป็น administrator

Password เป็น radius