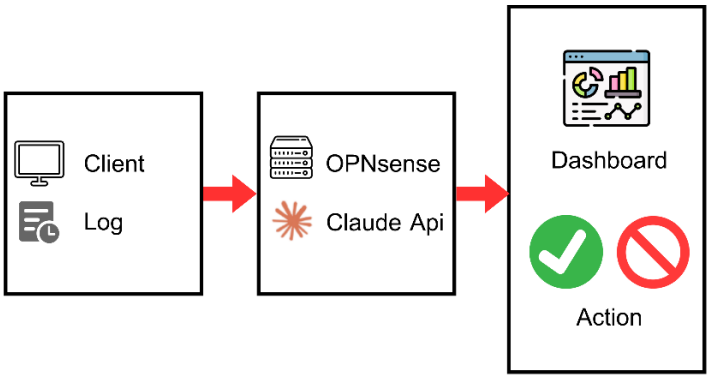
บทที่ 3

วิธีการดำเนินงาน

ในบทนี้นำเสนอขั้นตอนการพัฒนาระบบเครือข่ายสำหรับควบคุมการเข้าถึงอินเทอร์เน็ต โดยเน้นการผสานการทำงานของระบบเครือข่ายกับปัญญาประดิษฐ์ (Artificial Intelligence: AI) เพื่อยกระดับความมั่นคงปลอดภัยของระบบอย่างอัจฉริยะ โครงสร้างเครือข่ายถูกออกแบบโดยใช้ OPNsense เป็นศูนย์กลางในการจัดการ ทำหน้าที่เป็นทั้ง Router และ Firewall ในการควบคุมการรับ–ส่งข้อมูลระหว่างอินเทอร์เน็ตและเครือข่ายภายใน และนำปัญญาประดิษฐ์มาใช้ในการวิเคราะห์ข้อมูลทราฟฟิกและ Log ที่เกิดขึ้นภายในเครือข่าย เพื่อช่วยตรวจจับพฤติกรรมที่ผิดปกติ ประเมินความเสี่ยง และสนับสนุนการตัดสินใจเชิงรุกในการควบคุมการเข้าถึง เช่น การอนุญาต การจำกัดสิทธิ์ หรือการบล็อกการเชื่อมต่อโดยอัตโนมัติ

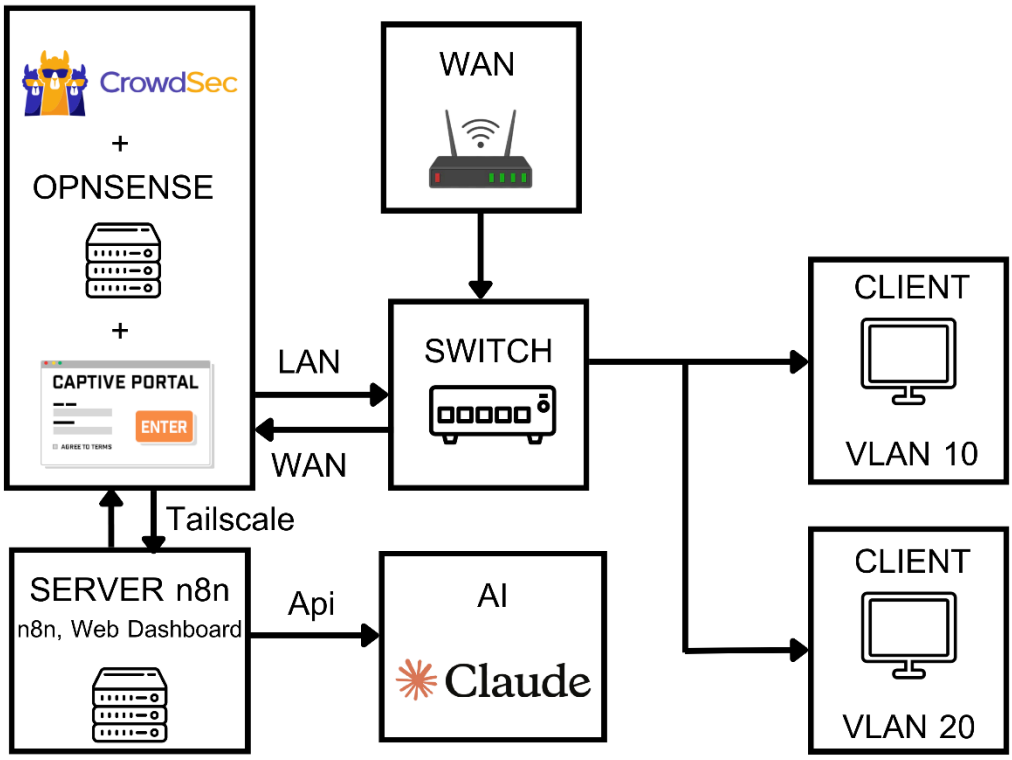
3.1 Block Diagram



รูปที่ 3-1 Block Diagram

จากรูปที่ 3-1 เป็น Block Diagram ในรูปแบบ Input–Process–Output โดยเริ่มจากส่วน Input คือข้อมูล Log และการใช้งานจากฝั่ง Client ภายในเครือข่าย จากนั้นเข้าสู่ส่วน Process ซึ่งประกอบด้วย OPNsense Firewall ที่ทำหน้าที่ควบคุมทราฟฟิก และการวิเคราะห์ข้อมูลด้วย Claude API (ปัญญาประดิษฐ์) เพื่อประเมินพฤติกรรมและความเสี่ยง สุดท้ายคือส่วน Output ซึ่งแสดงผลลัพธ์ผ่าน Dashboard เพื่อให้ผู้ดูแลระบบสามารถติดตาม ตรวจสอบ และใช้ประกอบการตัดสินใจด้านความปลอดภัยของเครือข่ายได้อย่างมีประสิทธิภาพ

3.2 Network Diagram



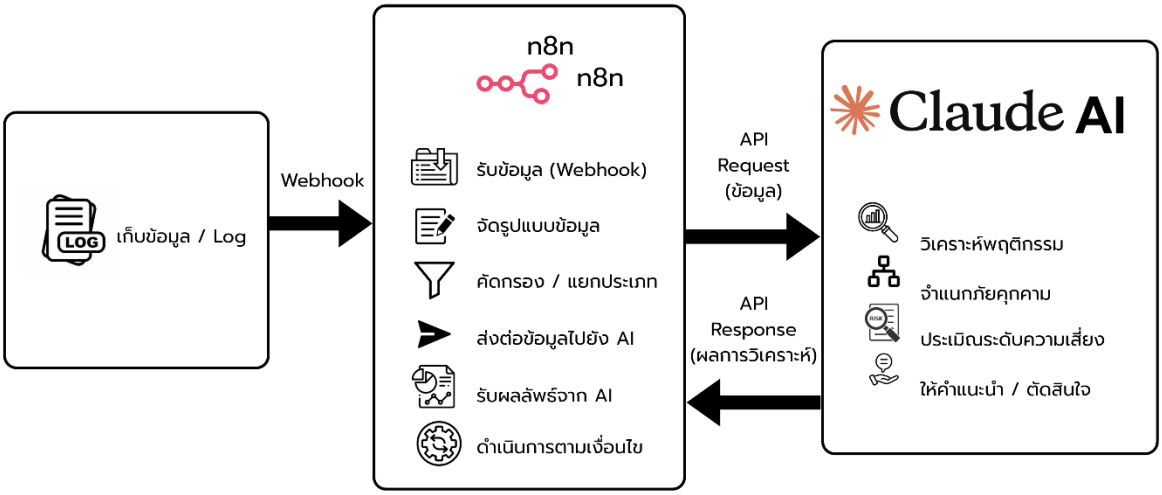
รูปที่ 3-2 Network Diagram แสดงผัง Network

จากรูปที่ 3-2 แสดงโครงสร้างระบบเครือข่ายที่พัฒนาขึ้น โดยผสานการทำงานระหว่างระบบควบคุมการเข้าถึงเครือข่ายและปัญญาประดิษฐ์ (Artificial Intelligence: AI) เพื่อเพิ่มประสิทธิภาพด้านความมั่นคงปลอดภัย ภายในระบบมี OPNsense ทำหน้าที่เป็น Firewall และศูนย์กลางในการควบคุมทราฟฟิก รวมถึงให้บริการ Captive Portal สำหรับการยืนยันตัวตนของผู้ใช้งาน การเชื่อมต่อเริ่มจากเครือข่ายภายนอก (WAN) ผ่าน Switch ซึ่งทำหน้าที่กระจายเครือข่ายไปยังเครื่อง Client ที่ถูกแบ่งออกเป็น VLAN 10 และ VLAN 20 เพื่อแยกกลุ่มผู้ใช้งานอย่างชัดเจน โดย VLAN 20 ถูกกำหนดให้ต้องผ่าน Captive Portal ก่อนจึงจะสามารถใช้งานเครือข่ายได้ ขณะที่ VLAN 10 สามารถเข้าถึงเครือข่ายได้ตามสิทธิ์ที่กำหนดโดยไม่ต้องผ่านกระบวนการดังกล่าว นอกจากนี้ภายในระบบยังมีการใช้ Tailscale เป็นช่องทางสำหรับรับ–ส่งข้อมูลระหว่าง OPNsense และเซิร์ฟเวอร์ที่ให้บริการ n8n และ Web Dashboard เพื่อรองรับการจัดการระบบและการเข้าถึงบริการจากระยะไกลอย่างปลอดภัย โดยมีปัญญาประดิษฐ์เป็นบทบาทสำคัญในการประมวลผลข้อมูลเชิงลึก จำแนกพฤติกรรมที่ผิดปกติ และสนับสนุนการตัดสินใจเชิงรุกของระบบ เช่น การอนุญาต การจำกัดสิทธิ์ หรือการบล็อกการเข้าถึงผ่านกลไกของ Firewall ส่งผลให้ระบบสามารถตอบสนองต่อภัยคุกคามได้อย่างรวดเร็วและมีประสิทธิภาพมากยิ่งขึ้น

3.3 การทำงานของส่วน Server, n8n และ ปัญญาประดิษฐ์ (Claude AI)

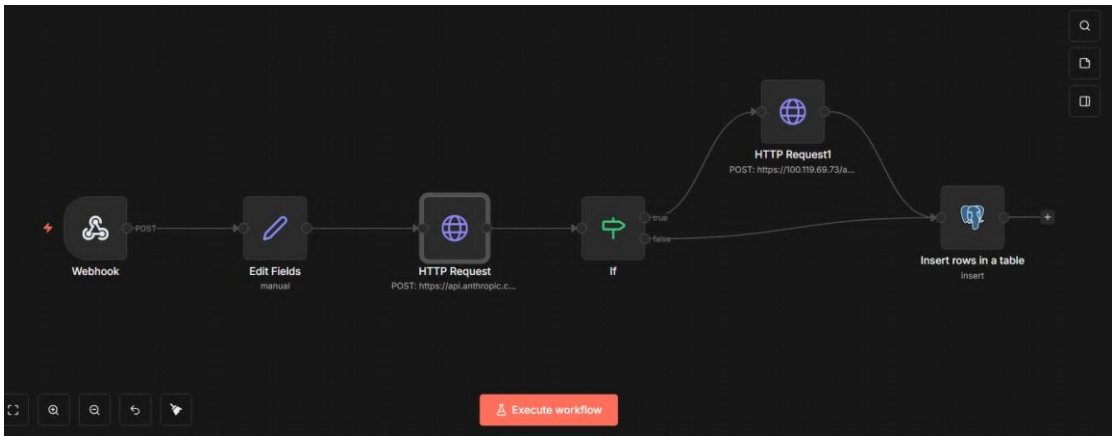
เมื่อเกิดเหตุการณ์ภายในเครือข่าย เช่น การใช้งานปกติหรือการตรวจพบพฤติกรรมที่น่าสงสัยจากระบบ Firewall หรือ IDS/IPS ข้อมูล Log และ Alert จะถูกส่งมายัง Server ซึ่งทำหน้าที่รวบรวมและจัดเก็บข้อมูล พร้อมทั้งแสดงผลผ่าน Web Dashboard

ภายใน Server มีการใช้ n8n เป็นระบบ Automation ในการจัดการ workflow โดย n8n จะรับข้อมูลผ่าน Webhook จาก CrowdSec ที่อยู่ใน OPNsense หรือเครื่องมือด้านความปลอดภัยที่ทำหน้าที่ตรวจจับและป้องกันภัยคุกคามโดยทำหน้าที่วิเคราะห์ log จาก OPNsense เพื่อตรวจจับพฤติกรรมผิดปกติ เช่น การแสกนพอร์ต เมื่อพบเหตุการณ์ที่เข้าข่ายภัยคุกคาม ผ่าน tailscale จากนั้น n8n จะทำหน้าที่จัดรูปแบบ และคัดกรองข้อมูลที่สำคัญ ก่อนส่งต่อไปยังปัญญาประดิษฐ์ผ่าน API ในส่วนของ ปัญญาประดิษฐ์ จะทำหน้าที่เป็นกลไกหลักในการวิเคราะห์ข้อมูลเชิงลึก โดยใช้ข้อมูล Log และพฤติกรรมเครือข่ายเพื่อวิเคราะห์รูปแบบการใช้งาน จำแนกเหตุการณ์ว่าเป็นปกติหรือภัยคุกคาม และ ประเมินระดับความเสี่ยง โดยผลลัพธ์จากปัญญาประดิษฐ์ จะถูกส่งกลับมายัง n8n เพื่อดำเนินการตามเงื่อนไขที่กำหนด เช่น กรณีความเสี่ยงต่ำ: อนุญาตการใช้งานและบันทึกลง Dashboard กรณีความเสี่ยงสูง: สั่งการไปยัง OPNsense ผ่าน API เพื่อดำเนินมาตรการตอบสนอง เช่น การบล็อก IP หรือกักกันอุปกรณ์



รูปที่ 3-3 การทำงานของส่วน Server, n8n และ AI (Claude)

ลำดับการทำงานของ n8n เริ่มต้นจากการรับข้อมูล Log จาก OPNsense ผ่าน webhook แล้วส่งไปคัดแยกและปรับรูปแบบข้อมูล ที่ node Edit Fields จากนั้นส่งข้อมูลตามรูปแบบ prompt ที่เขียนไว้ใน node http request ไปยัง Claude AI ผ่าน api ของ claude ai ผ่าน node http request แล้วรับผลจาก AI แล้วตัดสินใจ โดยแยกผลลัพธ์เป็น 2 กรณี True ⟶ เสี่ยง / อันตราย, False ⟶ ปกติ ถ้ากรณีผลลัพธ์เป็น True จะส่งคำสั่งไปยังระบบ เพื่อทำ Action เช่น Block IP จากนั้นบันทึกข้อมูลลงฐานข้อมูลผ่าน node Insert rows in a table



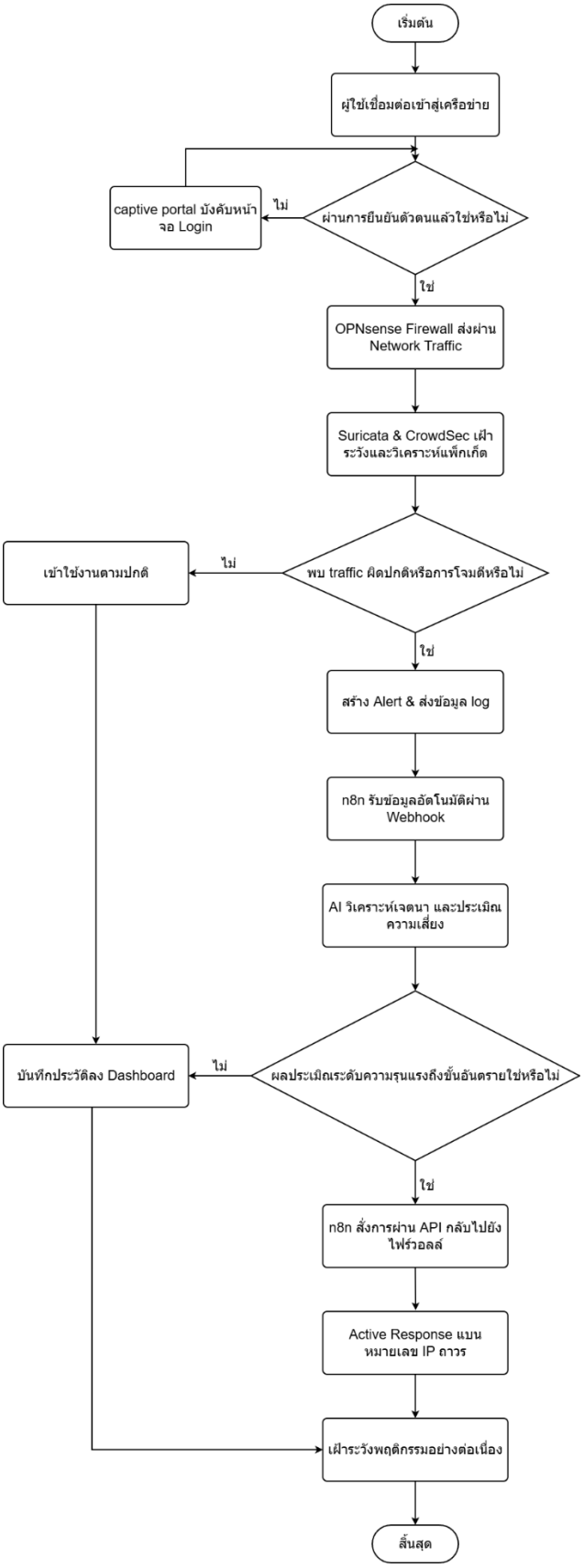
รูปที่ 3-4 ลำดับการทำงานของ n8n

โดย prompt หรือ รูปแบบข้อความที่ใช้จะอยู่ในรูปแบบ code JSON ตามกล่องข้อความด้านล่าง

```
{
  "model": "claude-sonnet-4-5-20250929",
  "max_tokens": 1500,
  "temperature": 0,
  "messages": [
    {
      "role": "user",
      "content": "คุณคือผู้เชี่ยวชาญ Cyber Security ระดับ Senior ของทีม SOC (Security Operations Center)\n\nข้อมูลการโจมตี:\n- IP คนร้าย: {{$json.attacker_ip}}\n- พอร์ตเป้าหมาย: {{$json.target_port}}\n- ประเภทการโจมตี: {{$json.attack_type}}\n- รายละเอียดเชิงลึก (Signature): {{$json.attack_signature}}\n- จำนวนครั้งที่โจมตี: {{$json.events_count}}\n- ข้อความแจ้งเตือนจากระบบ: {{$json.message}}\n\nหน้าที่ของคุณและกฎเกณฑ์การประเมิน (Strict Rules):\n1. วิเคราะห์การโจมตี (อธิบายว่าคืออะไร อันตรายแค่ไหน และแนะนำวิธีป้องกัน)\n2. ประเมินคะแนนความรุนแรง (Severity Score) เป็นตัวเลข 1-100 โดยอิงตามเกณฑ์บังคับดังต่อไปนี้:\n - Score 1-40: พฤติกรรมผิดปกติทั่วไปที่ความเสี่ยงต่ำ\n - Score 41-74: การสแกนเครือข่ายหรือสแกนพอร์ต (Reconnaissance) เช่น port-scan\n - Score 75-100: ความพยายามเจาะเข้าระบบ (Exploit), การเดารหัสผ่าน (Brute Force เช่น ssh-bf, ssh-slow-bf), หรือการโจมตีแอปพลิเคชัน\n3. กฎเหล็ก (CRITICAL): ห้ามลดคะแนนเพียงเพราะระบบ Firewall หรือ CrowdSec ได้ทำการบล็อกไปแล้ว ต้องประเมินจากเจตนาการโจมตีเป็นหลัก\n\nบังคับ: ให้ตอบกลับเป็น JSON เท่านั้น ห้ามมีข้อความอื่นใดนอกเหนือจาก JSON\n\nรูปแบบคำตอบ:\n{\n \"severity_score\": 0,\n  \"analysis\": \"\",\n  \"reason\": \"\"\n}"
    }
  ]
```

โดย prompt จะเรียกใช้งานปัญญาประดิษฐ์ผ่าน API โดยส่งข้อมูลเหตุการณ์การโจมตีจากระบบเครือข่ายเข้าไปให้ AI ทำการวิเคราะห์ในเชิง Cyber Security ภายใน prompt มีการระบุรายละเอียดของเหตุการณ์ เช่น IP ผู้โจมตี พอร์ต ประเภทการโจมตี พารามิเตอร์ model ใช้ระบุโมเดลปัญญาประดิษฐ์ที่ต้องการใช้งาน เช่น Claude เพื่อกำหนดความสามารถในการวิเคราะห์ข้อมูล ส่วน max_tokens คือการกำหนดจำนวนคำสูงสุดที่ AI สามารถตอบกลับได้ ซึ่งมีผลต่อความยาวของผลลัพธ์ ขณะที่ temperature ใช้ควบคุมความสุ่มของคำตอบ โดยค่าต่ำ (เช่น 0) จะทำให้คำตอบมีความแม่นยำและสม่ำเสมอมากขึ้น สำหรับ messages เป็นโครงสร้างข้อมูลที่ใช้ส่งคำสั่งและบริบทไปยัง AI โดยประกอบด้วยบทบาท (role) และเนื้อหา (content) เพื่อให้ AI เข้าใจสถานการณ์และตอบกลับได้ตรงตามที่ต้องการ และจำนวนครั้ง เพื่อให้ AI ใช้ประกอบการประเมินความเสี่ยง จากนั้น AI จะทำการวิเคราะห์พฤติกรรม ประเมินระดับความรุนแรง (Severity Score) ตามเกณฑ์ที่กำหนด และอธิบายเหตุผลในการให้คะแนน ผลลัพธ์จะถูกบังคับให้อยู่ในรูปแบบ JSON เพื่อให้ง่ายต่อการนำไปใช้งานต่อในระบบอัตโนมัติ เช่น n8n หรือการสั่งการ Firewall ดังนั้นโค้ดนี้จึงเป็นส่วนสำคัญในการเชื่อมต่อระหว่างข้อมูลเครือข่ายกับการตัดสินใจของปัญญาประดิษฐ์ในระบบรักษาความปลอดภัย

3.4 Flowchart ขั้นตอนการทำงานของระบบ

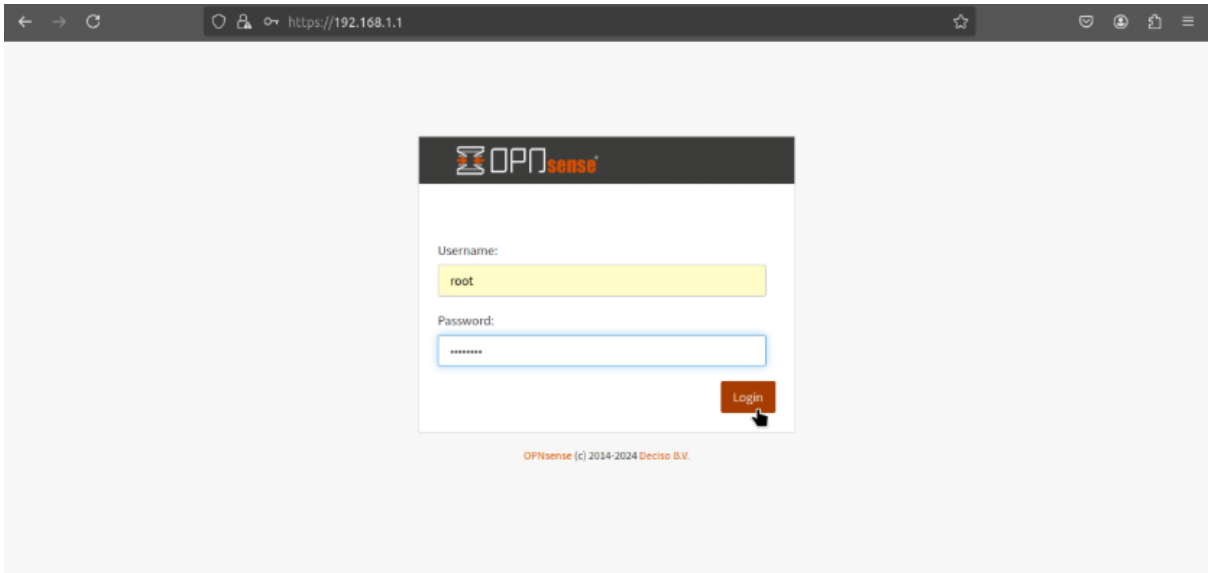


รูปที่ 3-5 Flowchart ขั้นตอนการทำงานของระบบ

จากรูปที่ 3-4 เป็นขั้นตอนการทำงานของระบบ โดยเริ่มจากผู้ใช้เชื่อมต่อเข้าสู่เครือข่าย จากนั้นต้องทำการยืนยันตัวตนผ่านระบบ Captive Portal ก่อน เมื่อยืนยันตัวตนแล้ว Firewall จะทำการตรวจสอบสิทธิ์การใช้งานและกำหนดบทบาท (Role) ของผู้ใช้ หลังจากนั้นระบบ IDS/IPS จะตรวจสอบพฤติกรรมการใช้งานเครือข่ายเพื่อค้นหาความผิดปกติ พร้อมทั้งส่งข้อมูล Logs และ Alerts ไปยัง NAC Controller เพื่อวิเคราะห์พฤติกรรมผ่าน RiskEngine หากระบบประเมินว่าระดับความเสี่ยงของผู้ใช้เกินค่าที่กำหนด (Threshold) ระบบจะสั่งให้ Firewall ปรับนโยบายเพื่อ Block หรือ Quarantine ผู้ใช้คนนั้น แต่หากระดับความเสี่ยงไม่เกินเกณฑ์ก็จะอนุญาตให้ใช้งานเครือข่ายได้ตามปกติ โดยสถานะการทำงานทั้งหมดจะถูกแสดงผลผ่าน Dashboard เพื่อให้ผู้ดูแลระบบสามารถติดตามและตรวจสอบสถานการณ์ของเครือข่ายได้อย่างต่อเนื่อง

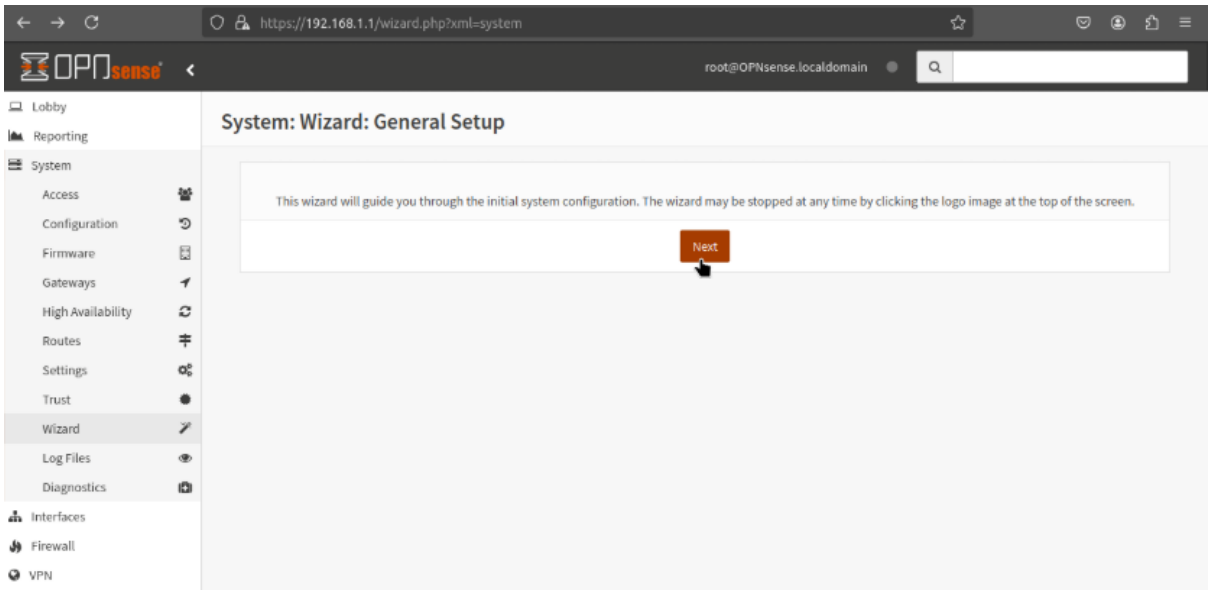
3.5 การตั้งค่า OPNsense และ VLAN

3.5.1 เข้า URL 192.168.1.1 แล้ว log in ด้วย username root และ password ที่ตั้งไว้



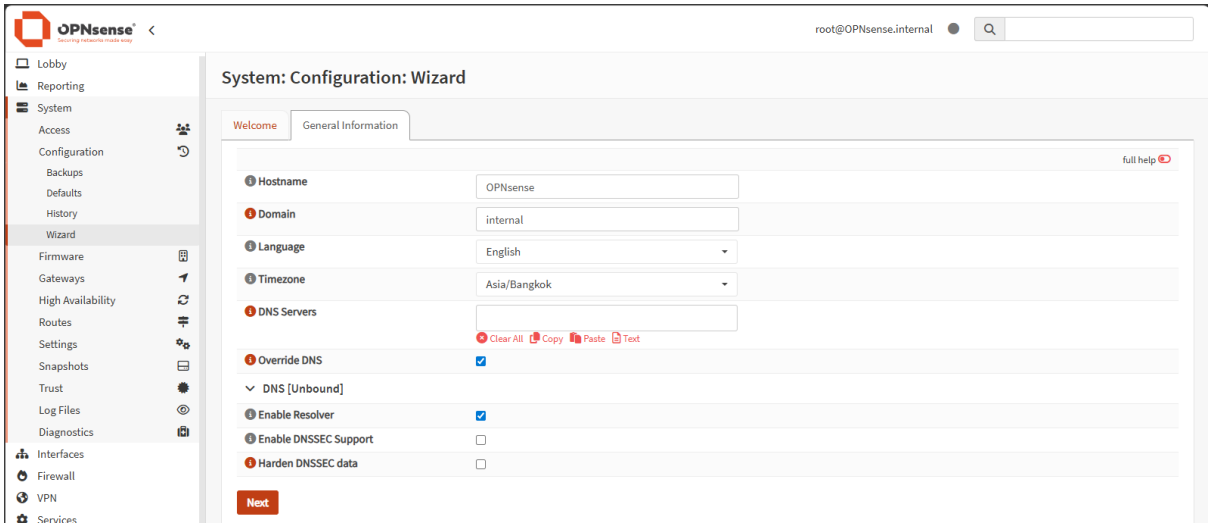
รูปที่ 3-6 log in ด้วย username root และ password ที่ตั้งไว้ก่อนหน้า

3.5.2 กดปุ่ม Next



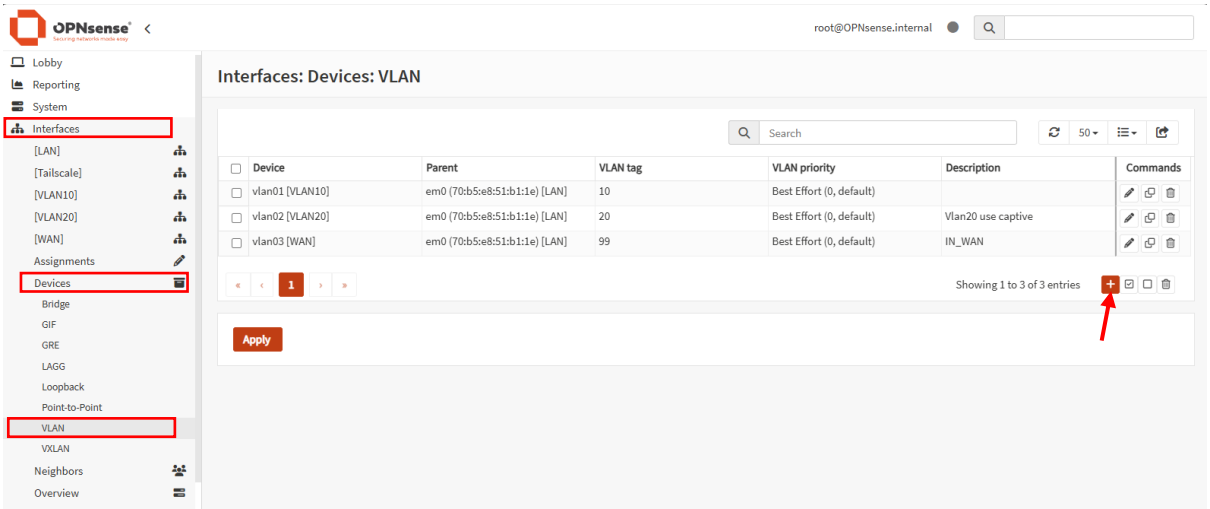
รูปที่ 3-7 กดปุ่ม Next

3.5.3 ตั้งค่าทั่วไป



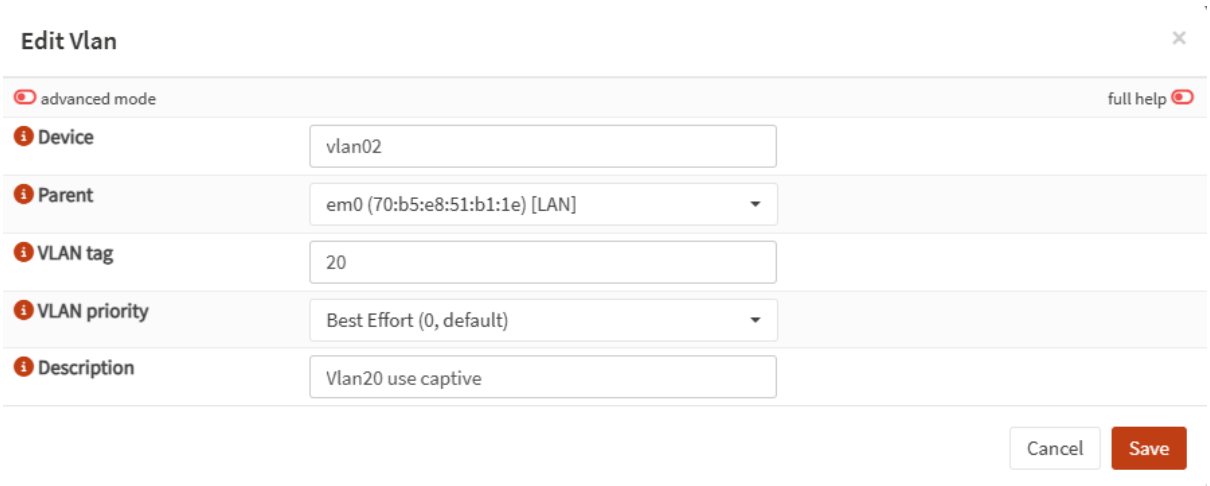
รูปที่ 3-8 การตั้งค่าทั่วไป

3.5.4 เข้าหน้าตั้งค่า VLAN โดยไปที่ Interface > Devices > VLAN ที่เมนูด้านซ้าย



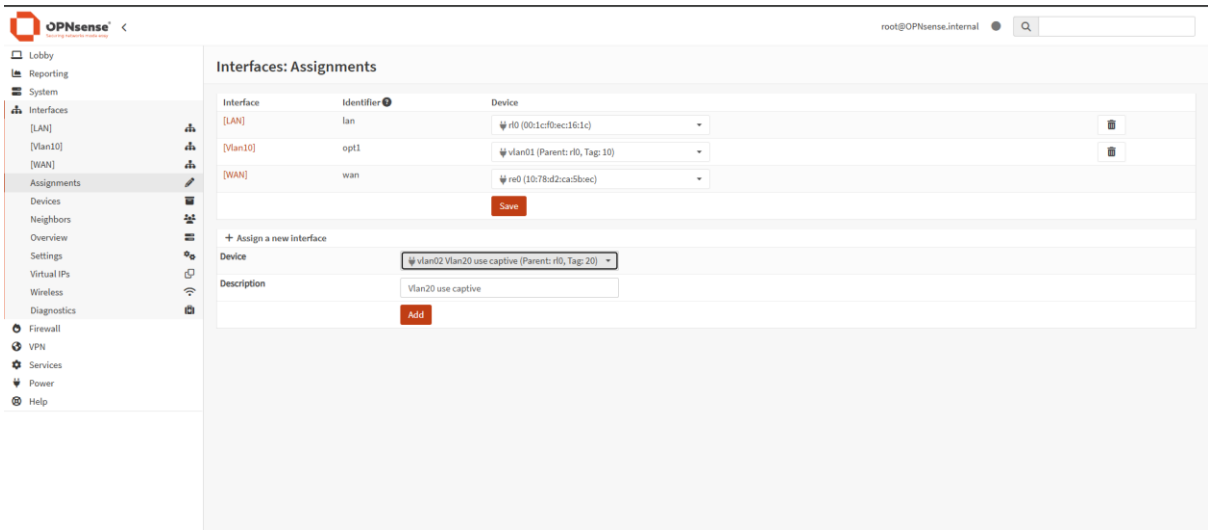
รูปที่ 3-9 เข้าหน้าตั้งค่า VLAN

3.5.4 ในการสร้าง VLAN ใหม่ ให้คลิกที่ปุ่มเครื่องหมาย + สีแดงทางด้านขวาของหน้าจอ หลังจากนั้นระบบจะแสดงหน้าต่าง Edit Vlan สำหรับกรอกข้อมูลและกำหนดค่าที่เกี่ยวข้องกับ VLAN



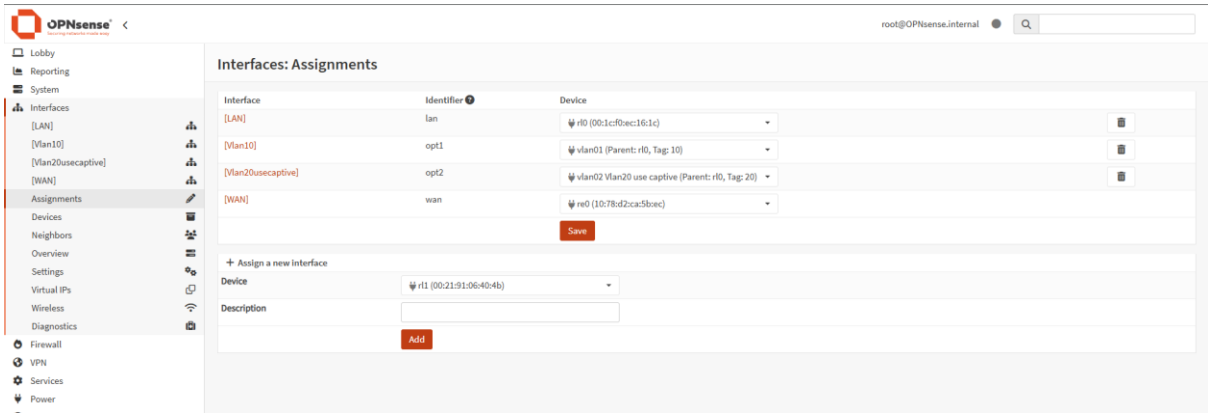
รูปที่ 3-10 แสดงหน้าต่าง Edit Vlan สำหรับกรอกข้อมูลและกำหนดค่า VLAN

3.5.6 เพิ่ม VLAN เข้าไปใน Assignments โดยไปที่ Interfaces > Assignments ในส่วน 'Assign a new interface' เลือก VLAN ที่สร้าง แล้วกด Add



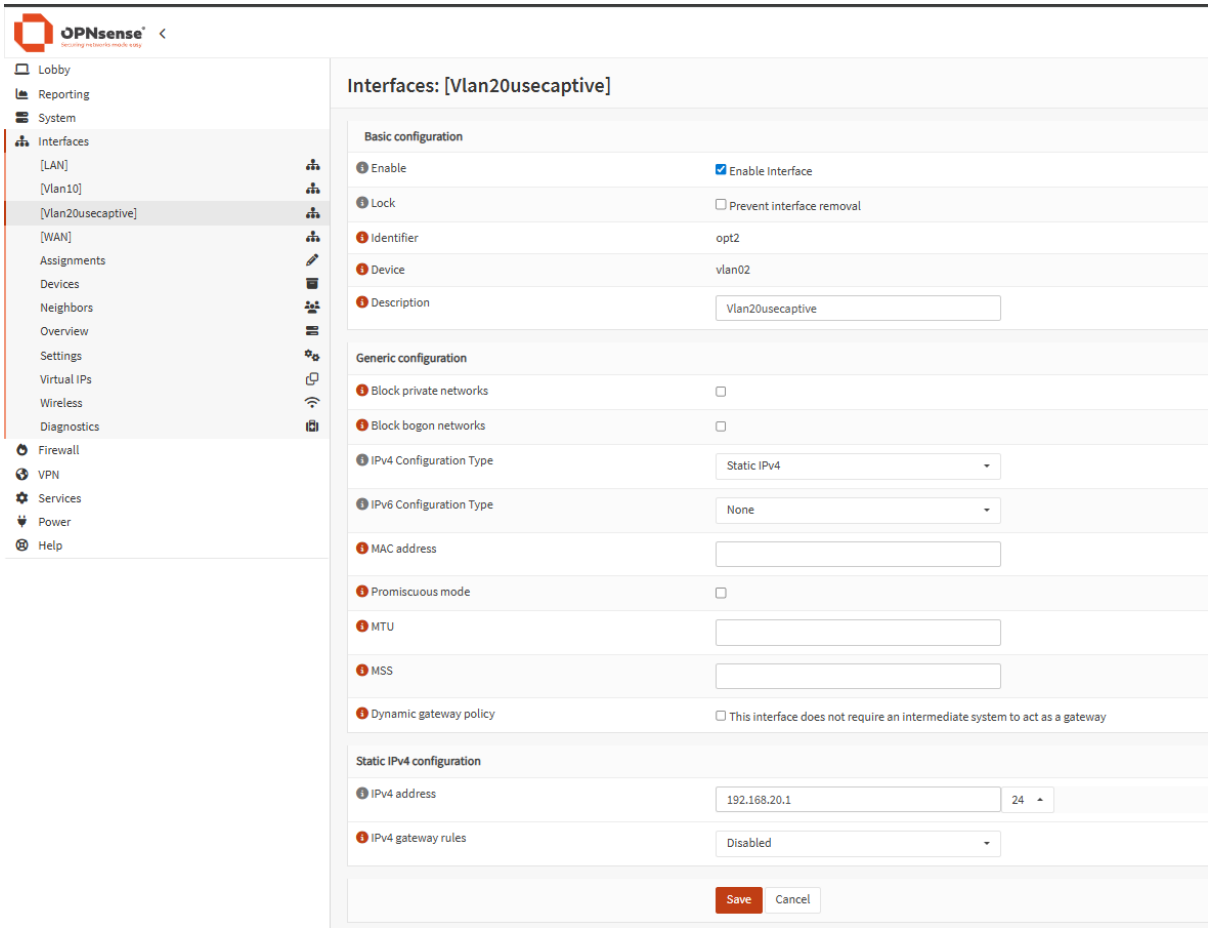
รูปที่ 3-11 เพิ่ม VLAN เข้าไปใน Assignments

3.5.7 ตรวจสอบว่า Interface ใหม่ถูกเพิ่มแล้ว โดยหลังจากที่ Add ไปแล้ว จะเห็น Interface ใหม่อยู่ในรายการ (ในรูปภาพจะเป็น OPT2 ที่ผูกกับ VLAN20) หลังจากนั้นให้กด Save เพื่อบันทึก



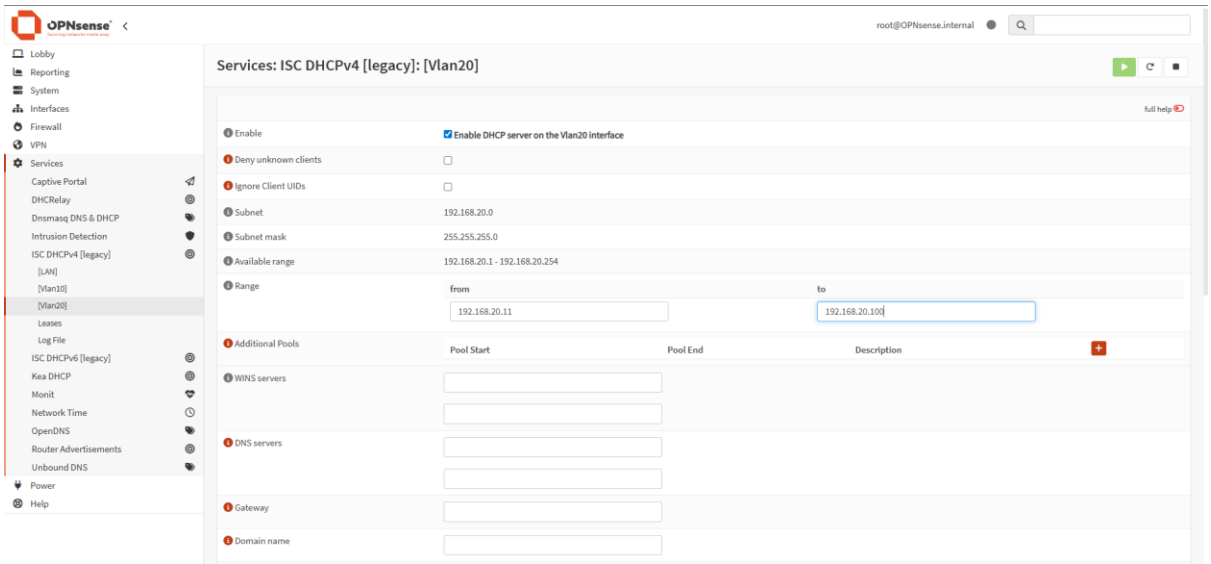
รูปที่ 3-12 ตรวจสอบว่า interface ใหม่ถูกเพิ่มแล้ว

3.5.8 เปิดใช้งาน Interface ใหม่ และตั้งค่า IP แบบ Static โดยคลิกที่ Interface ใหม่ (ในรูปภาพเป็น [Vlan20usecaptive]) ติ๊ก Enable Interface ตั้ง IPv4 Configuration Type = Static IPv4 จากนั้นกำหนด IPv4 address (ในรูปภาพใช้เป็น 192.168.20.1/24) แล้วกด Save



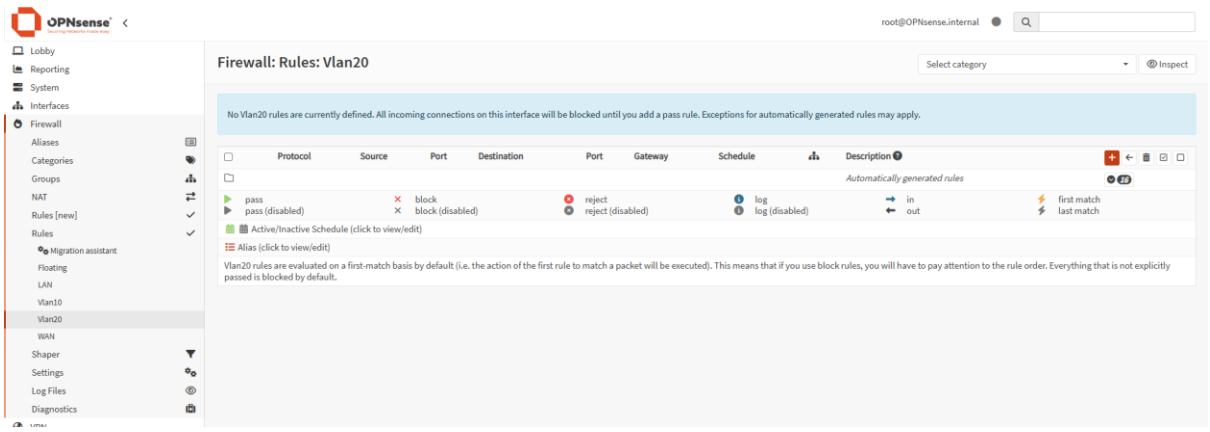
รูปที่ 3-13 เปิดใช้งาน Interface ใหม่ และตั้งค่า IP แบบ Static

3.5.9 เปิด DHCP Server สำหรับ VLAN ใหม่ โดยไปที่ Services > ISC DHCPv4 (legacy) > [Vlanใหม่] (หรือ DHCP ของ VLANใหม่) ติ๊ก Enable DHCP server on the Vlan interface และ ตั้ง Range เช่น 192.168.20.11 ถึง 192.168.20.100 แล้วกด Save



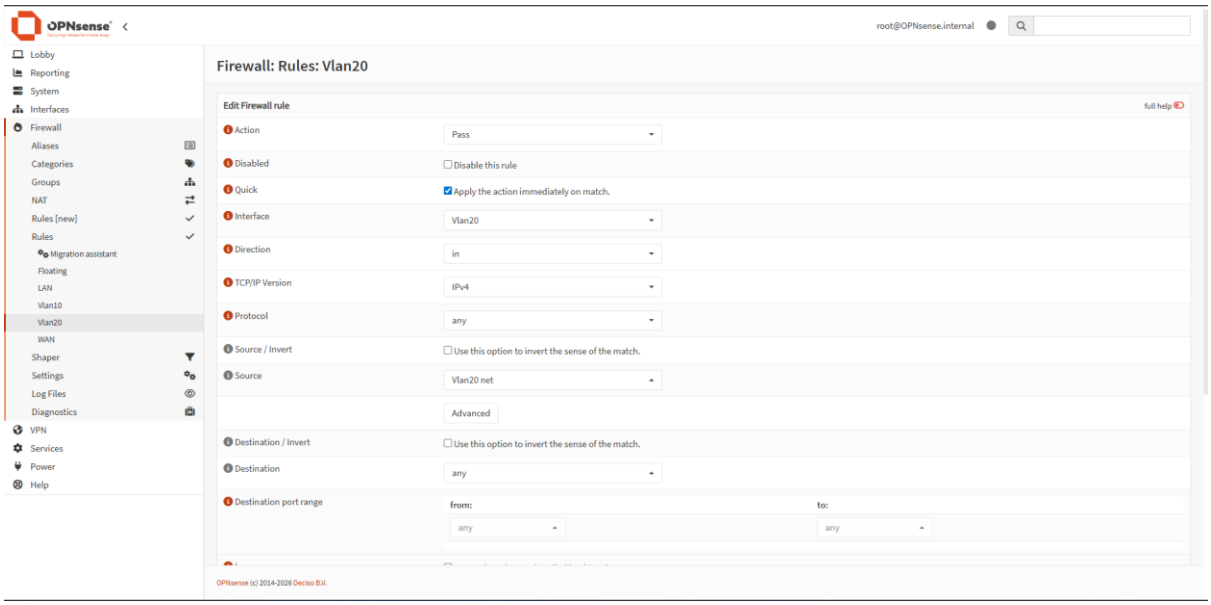
รูปที่ 3-14 เปิด DHCP Server สำหรับ VLAN ใหม่

3.5.10 สร้าง Firewall Rules บน Interface VLAN ใหม่ โดยไปที่ Firewall > Rules > Vlan ใหม่ หากยังไม่มี Rule ระบบจะ block ทั้งหมด ดังนั้นให้เพิ่ม Rule เพื่ออนุญาตทราฟฟิกที่ต้องการ



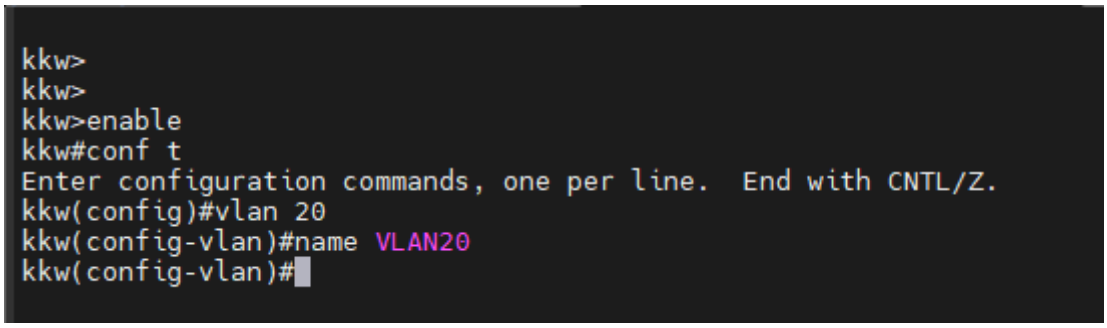
รูปที่ 3-15 สร้าง Firewall Rules บน Interface VLAN ใหม่

3.5.11 เพิ่ม Rule แบบ Pass สำหรับ VLAN ใหม่ โดยกดที่เครื่องหมาย + เพื่อเพิ่ม Rule ใหม่ จากนั้นตั้ง Action เป็น Pass, Interface เป็น Vlanใหม่, Direction เป็น in และตั้ง Source เป็น Vlanใหม่ net, Destination เป็น any แล้วกด Save



รูปที่ 3-16 เพิ่ม Rule แบบ Pass สำหรับ VLAN ใหม่

3.5.12 สร้าง VLAN ใหม่บนสวิตช์ โดยเข้าโหมด config ของสวิตช์ แล้วสร้าง VLAN ใหม่ขึ้นมา โดยตั้งชื่อเป็นชื่อเดียวกับ VLAN ที่ตั้งไว้ใน OPNsense



รูปที่ 3-17 สร้าง VLAN ใหม่บนสวิตช์

3.5.13 ตั้งค่า Trunk port ของสวิตช์ ที่ต่อไปยัง OPNsense โดยเลือกพอร์ต uplink (ตัวอย่าง Gi0/1) และ อนุญาต VLAN ใหม่ บน trunk (allowed vlan add 20)

```
kkw(config)#interface gigabitEthernet 0/1
kkw(config-if)#switchport trunk allowed vlan add 20
kkw(config-if)#
```

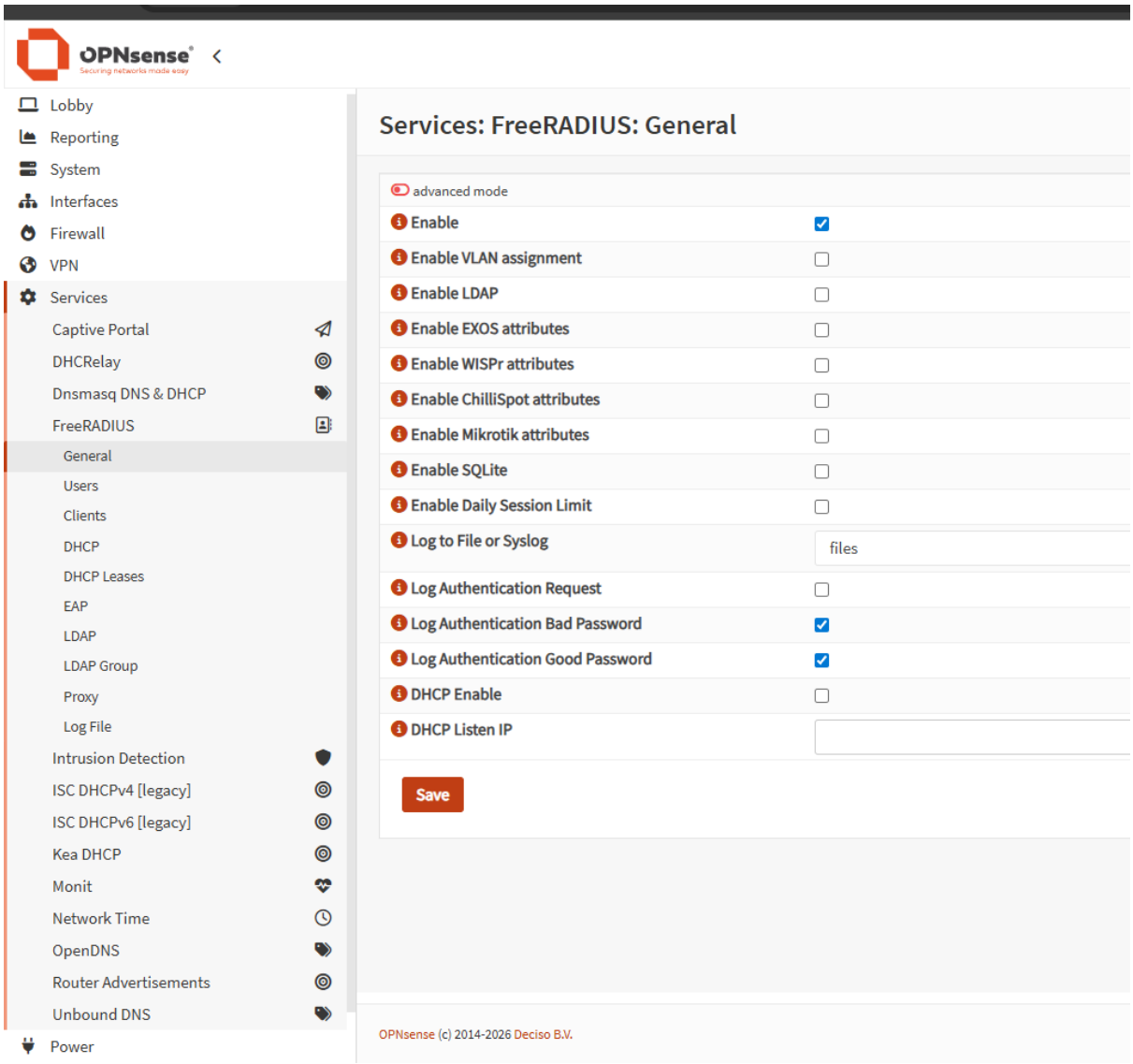
รูปที่ 3-18 ตั้งค่า Trunk port ของสวิตช์ ที่ต่อไปยัง OPNsense

3.5.14 ตั้งค่า Access port บนสวิตช์ สำหรับเครื่องลูกข่าย โดยเลือกพอร์ตที่ใช้ต่อเครื่อง (ตัวอย่าง Gi0/3) และ ตั้ง switchport mode access และกำหนด switchport access เป็น vlan ใหม่

```
kkw(config)#interface gigabitEthernet 0/3
kkw(config-if)#switchport mode access
kkw(config-if)#switchport access vlan 20
kkw(config-if)#
```

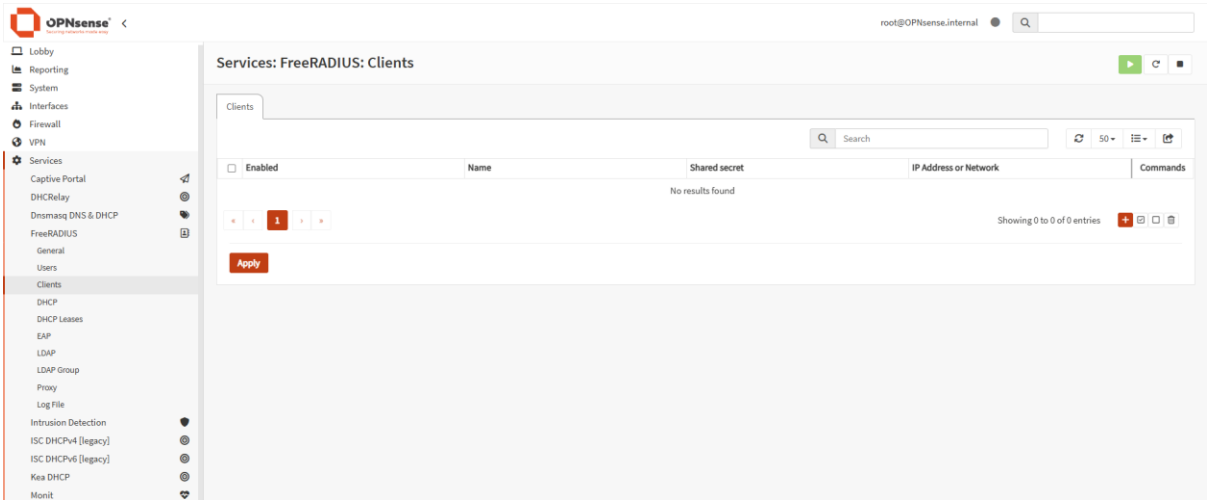
รูปที่ 3-19 ตั้งค่า Access port บนสวิตช์ สำหรับเครื่องลูกข่าย

3.5.15 เปิดใช้งาน FreeRADIUS ใน OPNsense โดยไปที่ Services > FreeRADIUS > General ติ๊ก Enable แล้วกด Save (สามารถเลือกค่าอื่น ๆ เช่น Logging เพื่อช่วย debug ได้ตามต้องการ)



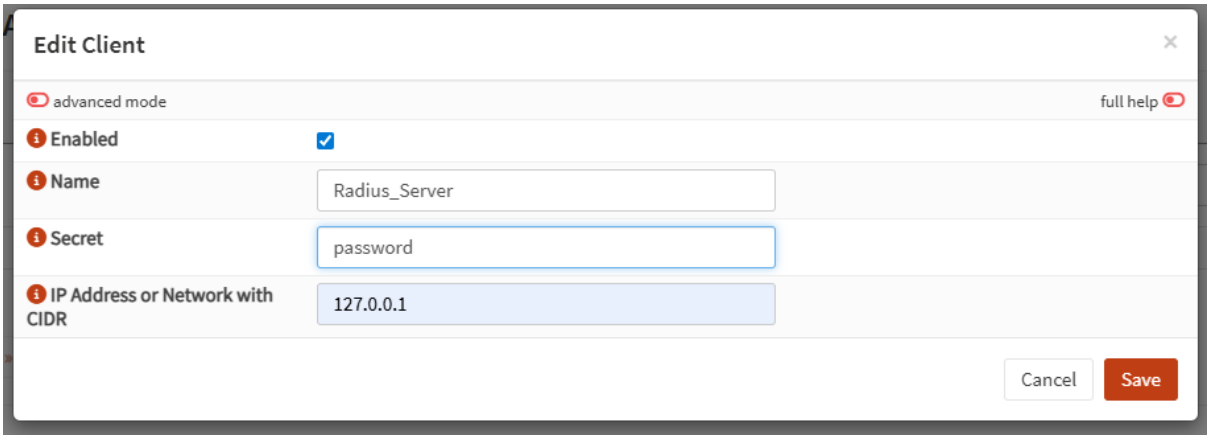
รูปที่ 3-20 เปิดใช้งาน FreeRADIUS ใน OPNsense

3.5.16 เข้าเมนู Clients ของ FreeRADIUS โดยไปที่ Services > FreeRADIUS > Clients (เริ่มต้นจะยังไม่มีรายการ Client ให้กดปุ่ม + เพื่อเพิ่ม)



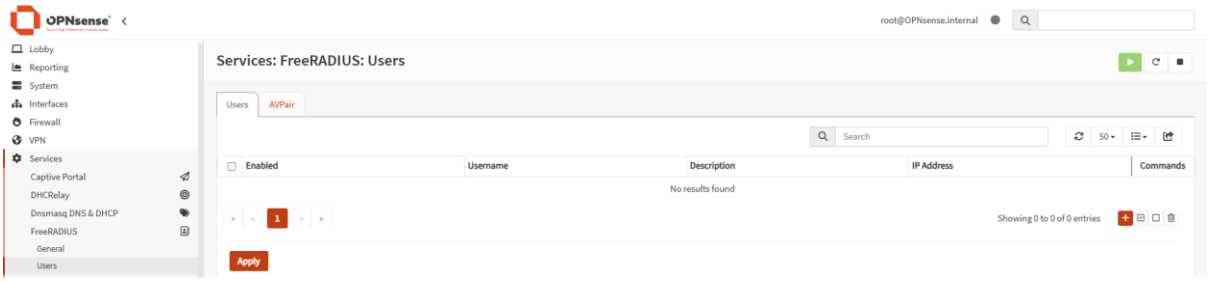
รูปที่ 3-21 เข้าเมนู Clients ของ FreeRADIUS

3.5.17 เพิ่ม RADIUS Client และกำหนด Shared Secret โดยตั้ง Name เช่น Radius_Server จากนั้นตั้ง Secret (Shared secret) เพื่อนำไปใช้ในฝั่ง 'Access Servers/Portal' แล้วกำหนด IP Address/Network ของอุปกรณ์ที่จะคุยกับ RADIUS (ตัวอย่าง 127.0.0.1 หากอยู่เครื่องเดียวกัน) แล้วกด Save



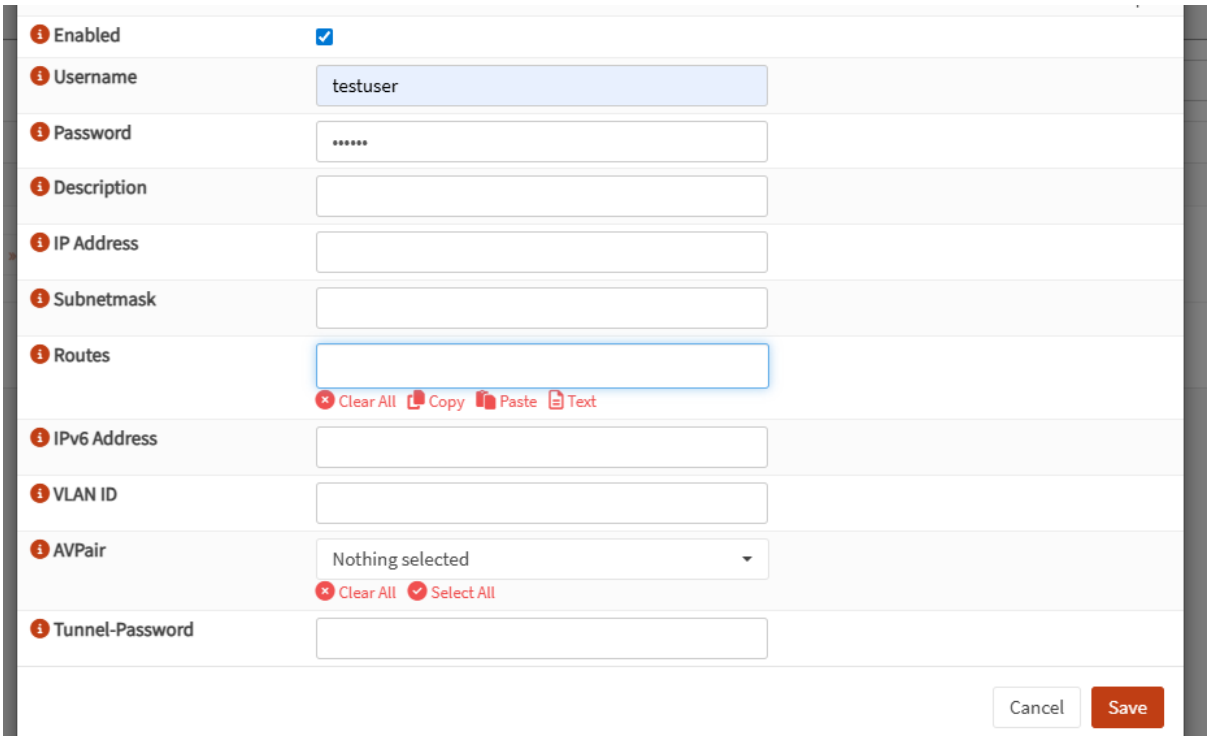
รูปที่ 3-22 เพิ่ม RADIUS Client และกำหนด Shared Secret

3.5.18 เข้าเมนู Users ของ FreeRADIUS โดยไปที่ Services > FreeRADIUS > Users แล้วกดปุ่มเครื่องหมาย + เพื่อเพิ่มผู้ใช้ที่จะใช้ล็อกอินผ่าน Captive Portal



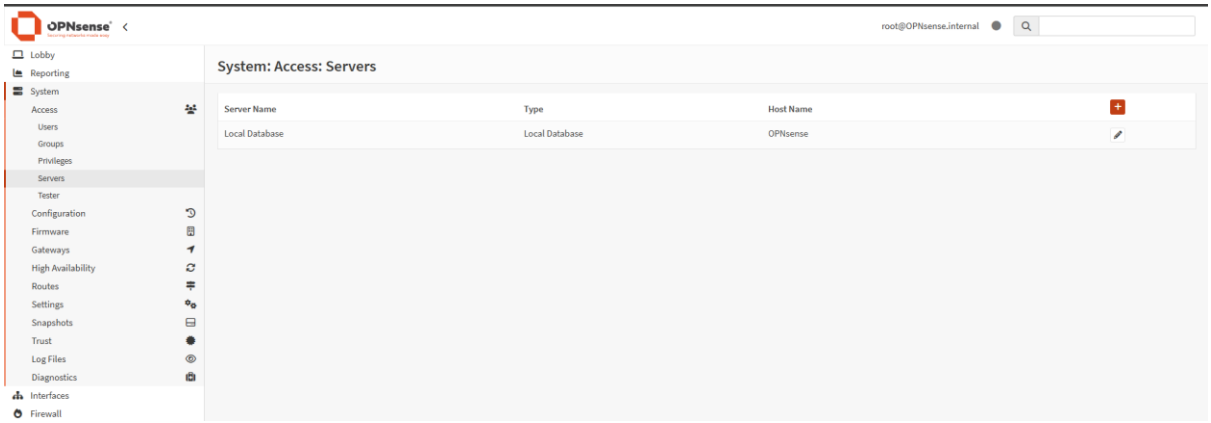
รูปที่ 3-23 เข้าเมนู Users ของ FreeRADIUS

3.5.19 เพิ่มผู้ใช้ FreeRADIUS (Username/Password) โดยกำหนด Username เช่น testuser และ กำหนด Password แล้วกด Save (สามารถเพิ่มหลายบัญชี หรือผูก LDAP ภายหลังได้)



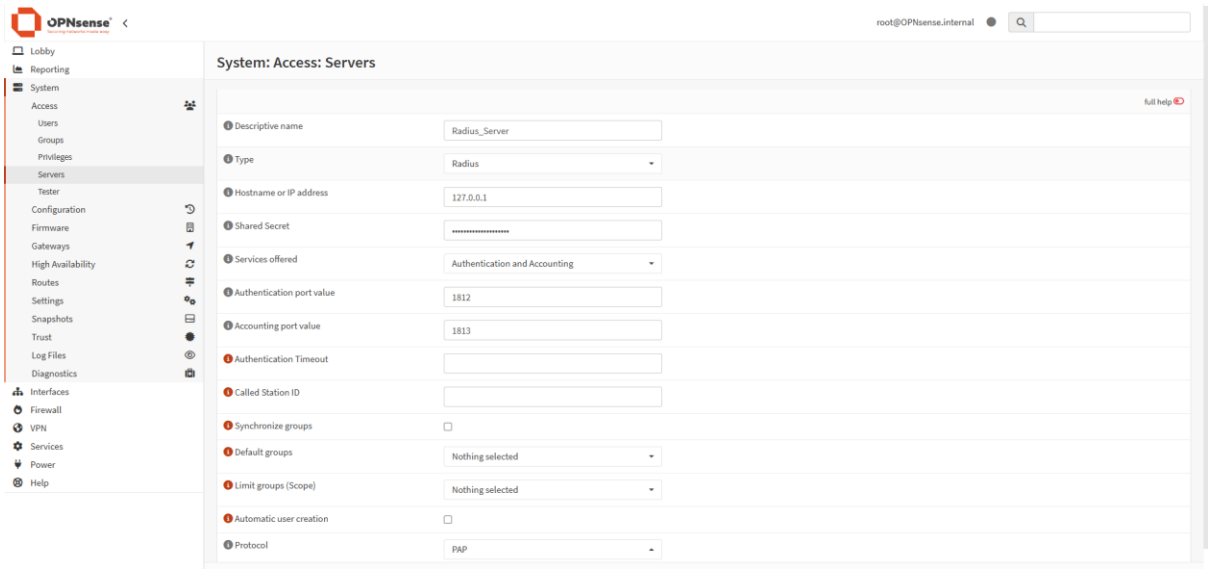
รูปที่ 3-24 เพิ่มผู้ใช้ FreeRADIUS

3.5.20 เพิ่ม RADIUS Server ใน Access Server โดยไปที่ System > Access > Servers แล้วกดปุ่มเครื่องหมาย + เพื่อเพิ่มเซิร์ฟเวอร์ชนิด RADIUS สำหรับการยืนยันตัวตน



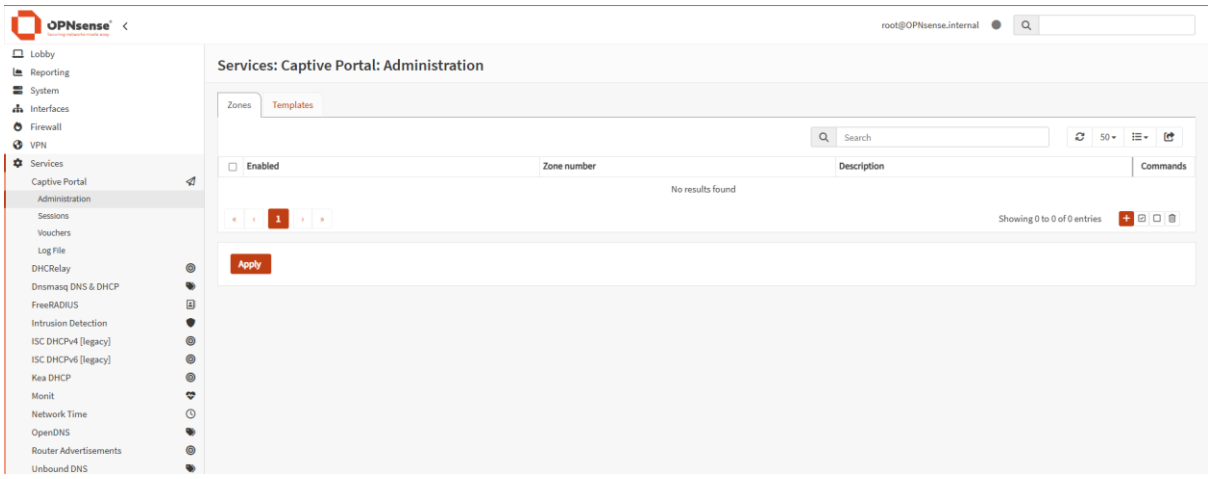
รูปที่ 3-25 เพิ่ม RADIUS Server ใน Access Server

3.5.21 ตั้งค่า Access Server ให้ชี้ไปที่ FreeRADIUS โดยตั้ง Descriptive name เช่น Radius_Server และ Type = Radius และ Hostname/IP = 127.0.0.1 (หรือ IP ของ RADIUS จริง) จากนั้นใส่ Shared Secret ให้ตรงกับที่ตั้งไว้ใน FreeRADIUS Clients และกำหนดเป็นพอร์ต: Authentication 1812 และ Accounting 1813 แล้วกด Save



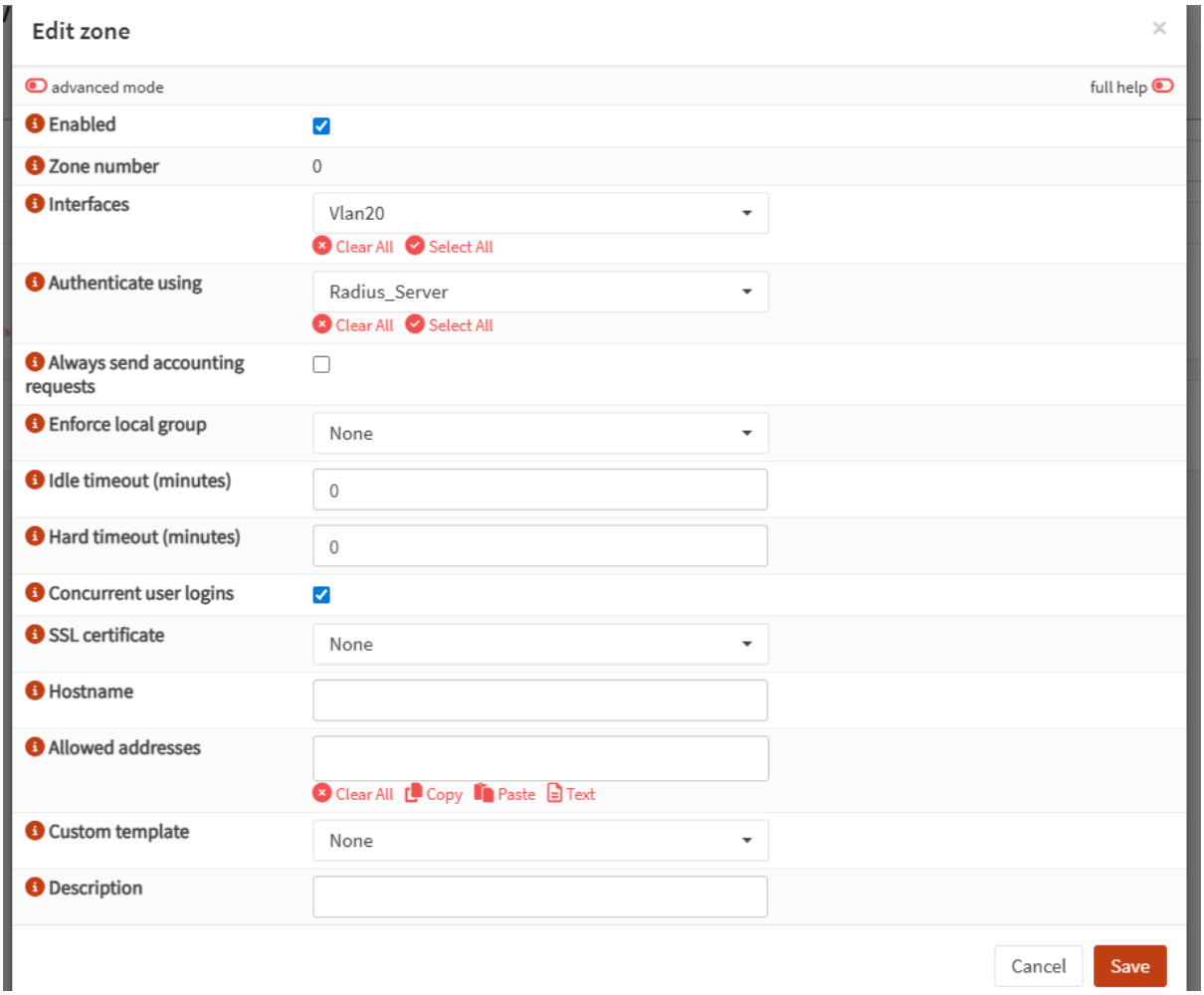
รูปที่ 3-26 ตั้งค่า Access Server ให้ชี้ไปที่ FreeRADIUS

3.5.22 สร้าง Captive Portal Zone โดยไปที่ Services > Captive Portal > Administration ในแท็บ Zones กด + เพื่อสร้าง Zone ใหม่



รูปที่ 3-27 สร้าง Captive Portal Zone

3.5.23 ผูก Zone กับ VLAN ใหม่ และเลือก Authentication เป็น RADIUS ติ๊ก Enabled และ เลือก Interface เป็น Vlan ใหม่ ใน 'Authenticate using' เลือก Radius_Server ที่สร้างไว้ แล้วกด Save

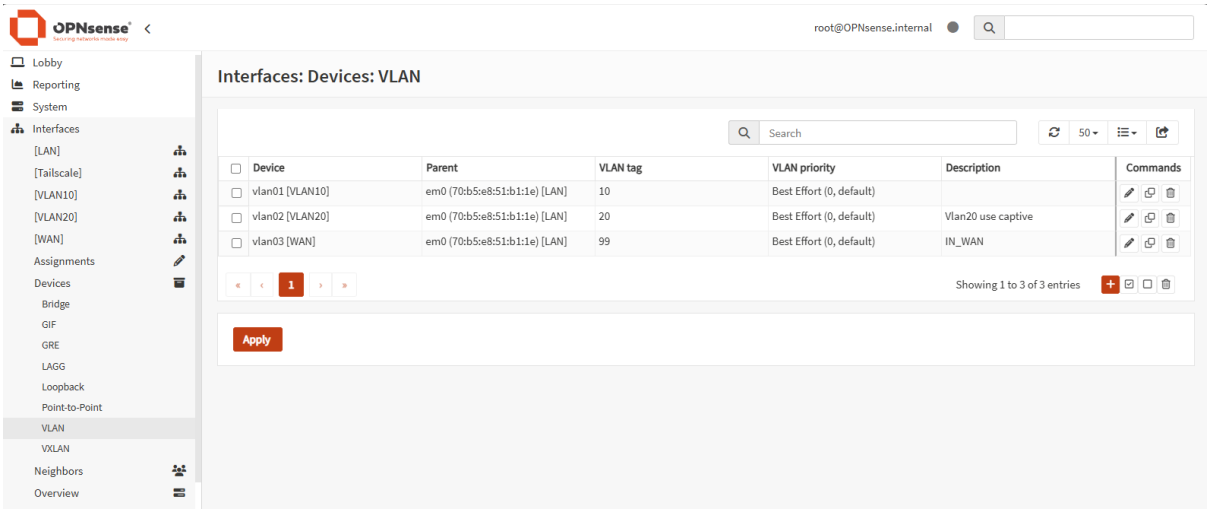


รูปที่ 3-28 ผูก Zone กับ VLAN ใหม่ และเลือก Authentication เป็น RADIUS

3.5.24 ทำตามขั้นตอนด้านล่างใน switch

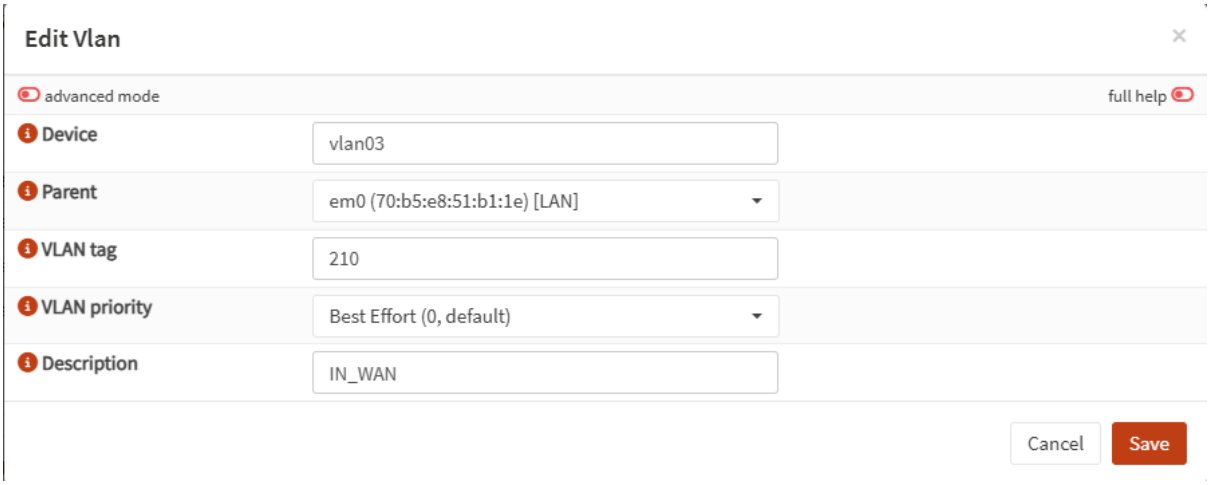
```
enable
configure terminal

! 1. สร้าง VLAN 210 (WAN)
vlan 210
 name IN_WAN
exit

! 2. พอร์ต Gig0/1 (ต่อเข้า OPNsense em0) ทำเป็น ท่อ Trunk ส่งทั้ง 2 VLAN
interface GigabitEthernet0/1
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport trunk allowed vlan 10,20,210
 exit

! 4. พอร์ต Gig0/24 (รับเน็ตจากมหาลัย) ให้อยู่ VLAN 210
interface GigabitEthernet0/24
```

3.5.25 ไปที่ Interfaces > Devices > VLAN ใน OPNsense แล้วกดเพิ่ม VLAN เพื่อเพิ่ม vlan 210



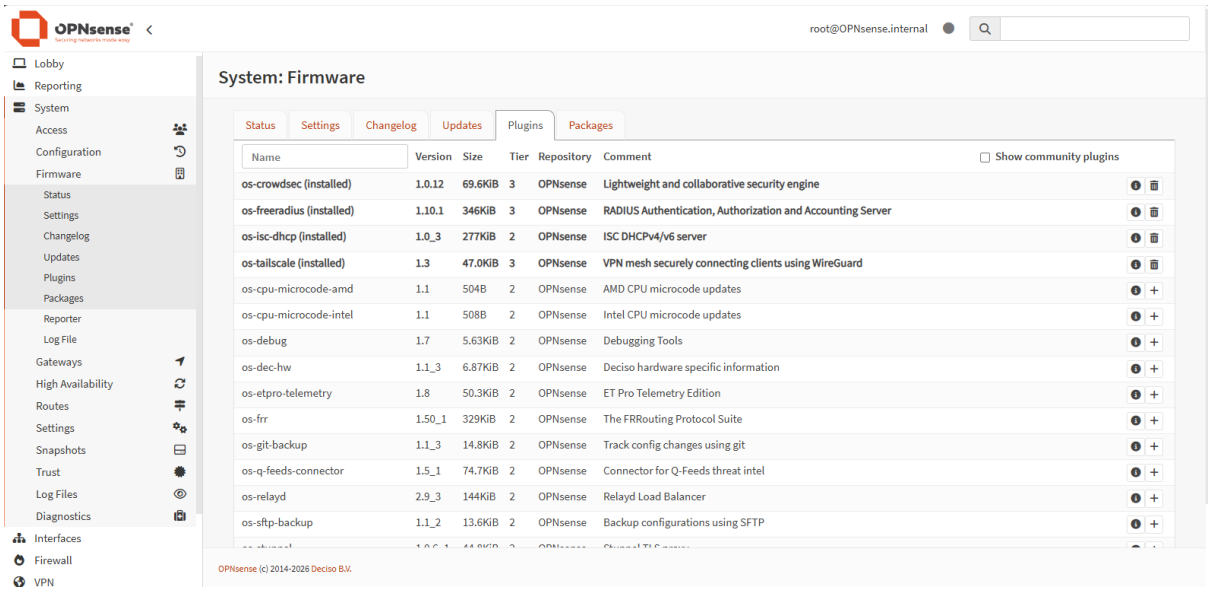
รูปที่ 3-29 กดเพิ่ม VLAN เพื่อเพิ่ม vlan 210

3.5.26 ตั้งค่า vlan 210 ตามภาพด้านล่าง แล้วกดปุ่ม save



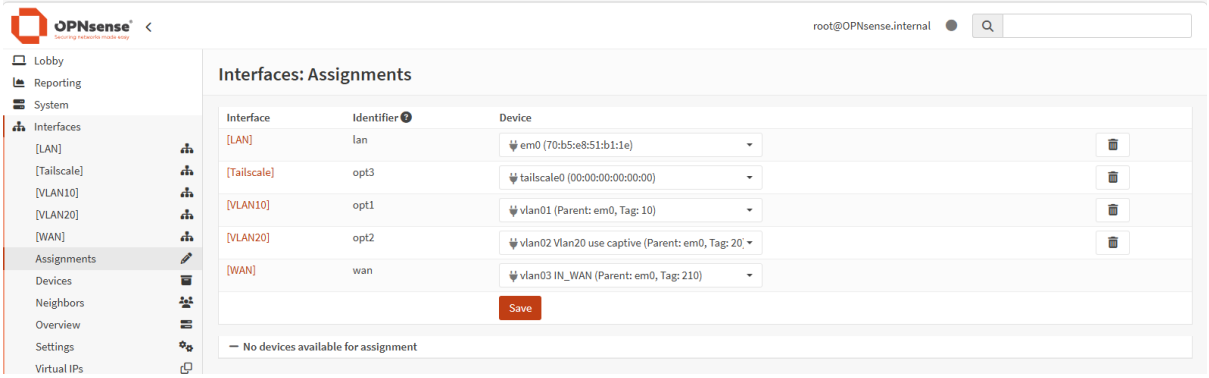
รูปที่ 3-30 ตั้งค่า vlan 210 แล้วกดปุ่ม save

3.5.27 ติดตั้ง Tailscale plugin โดยเข้าไปที่ System > Firmware > Plugins จากนั้นหา Plugin ที่ชื่อว่า os-tailscale แล้วกดปุ่มเครื่องหมาย + เพื่อติดตั้ง



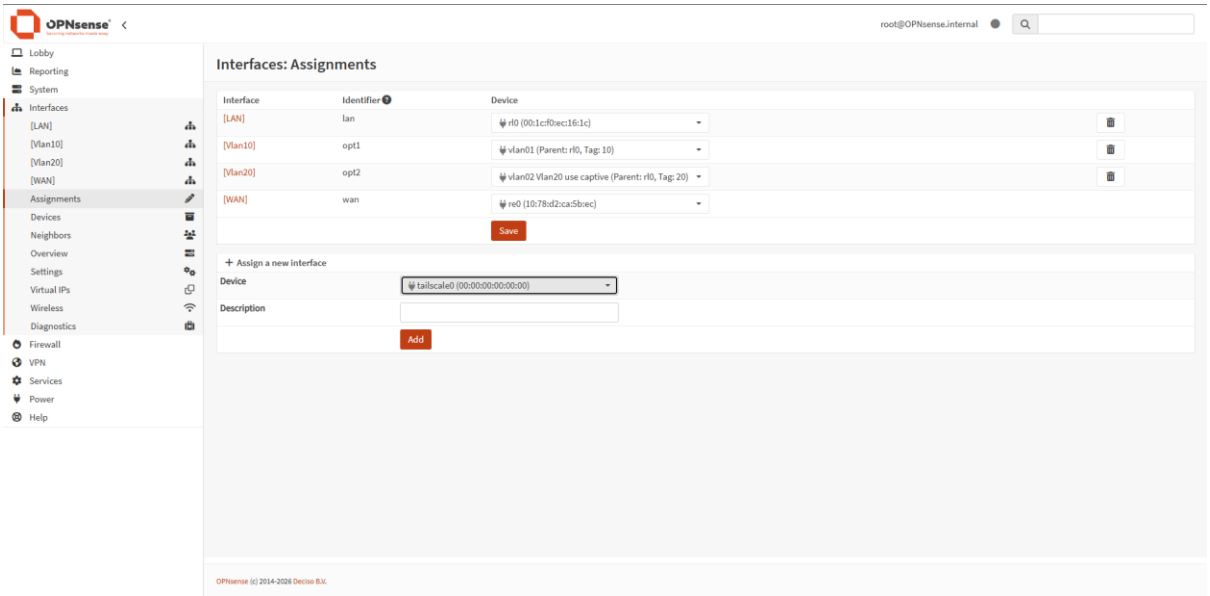
รูปที่ 3-31 การติดตั้ง Tailscale Plugin

3.5.28 ไปที่ Interfaces > Assignments แล้ว assign WAN ตามรูปด้านล่าง แล้วกดปุ่ม save



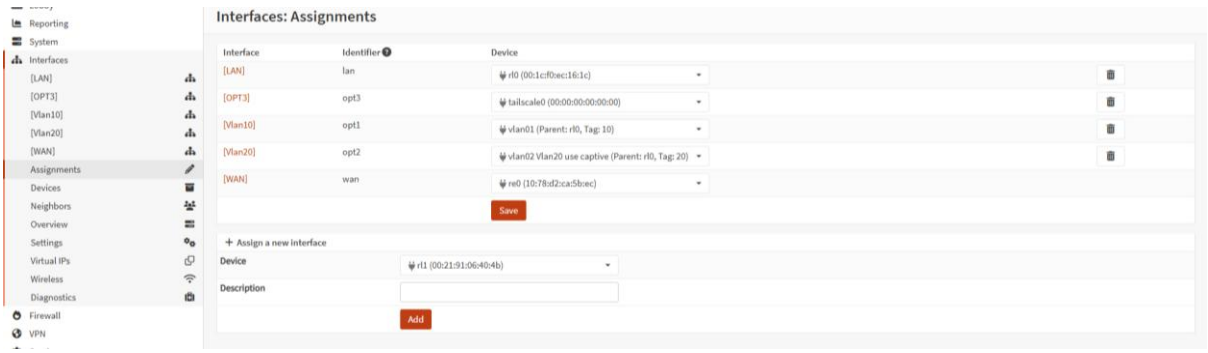
รูปที่ 3-32 assign WAN แล้วกดปุ่ม save

3.5.29 เพิ่ม tailscale เข้าไปใน Assignments โดยไปที่ Interfaces > Assignments ในส่วน 'Assign a new interface' เลือก talescale แล้วกด Add



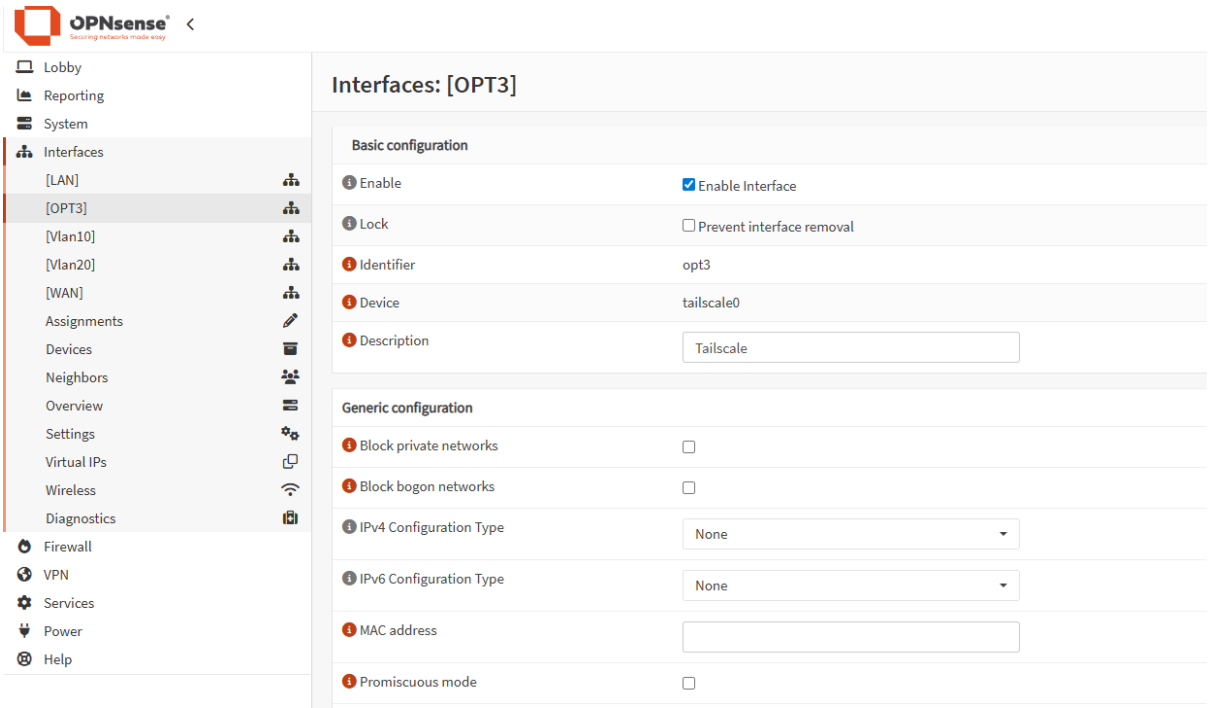
รูปที่ 3-33 เพิ่ม tailscale เข้าไปใน Assignments

3.5.30 ตรวจสอบว่า tailscale ที่เป็น Interface ใหม่ถูกเพิ่มแล้ว หลังจากนั้นให้กด Save เพื่อบันทึก



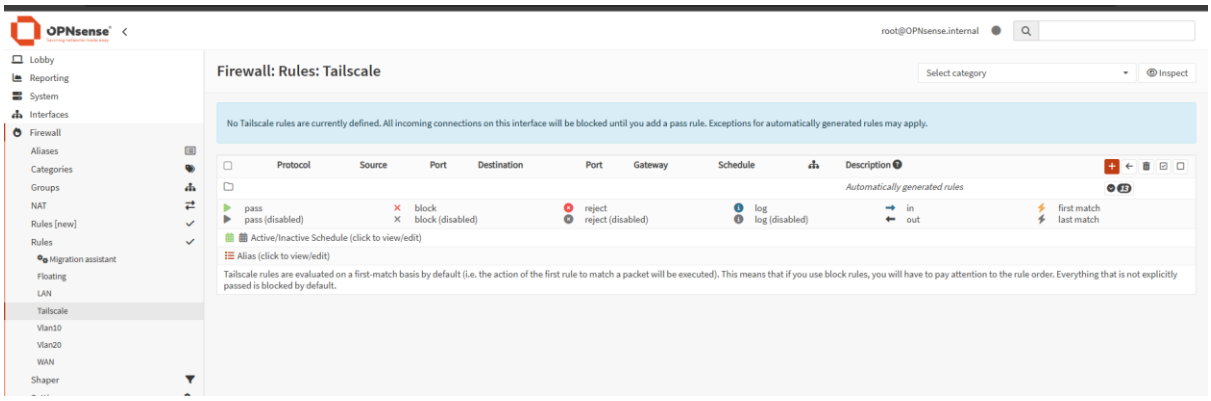
รูปที่ 3-34 ตรวจสอบว่า tailscale ที่เป็น Interface ใหม่ถูกเพิ่มแล้วกด save

3.5.31 เข้าไปเปิดใช้งาน Interface ของ talescale โดยคลิกเข้าไปที่ tailscale ใน interfaces แล้วกดติ๊ก Enable Interface จากนั้นตั้ง Description เป็น Tailscale แล้วกดปุ่ม save



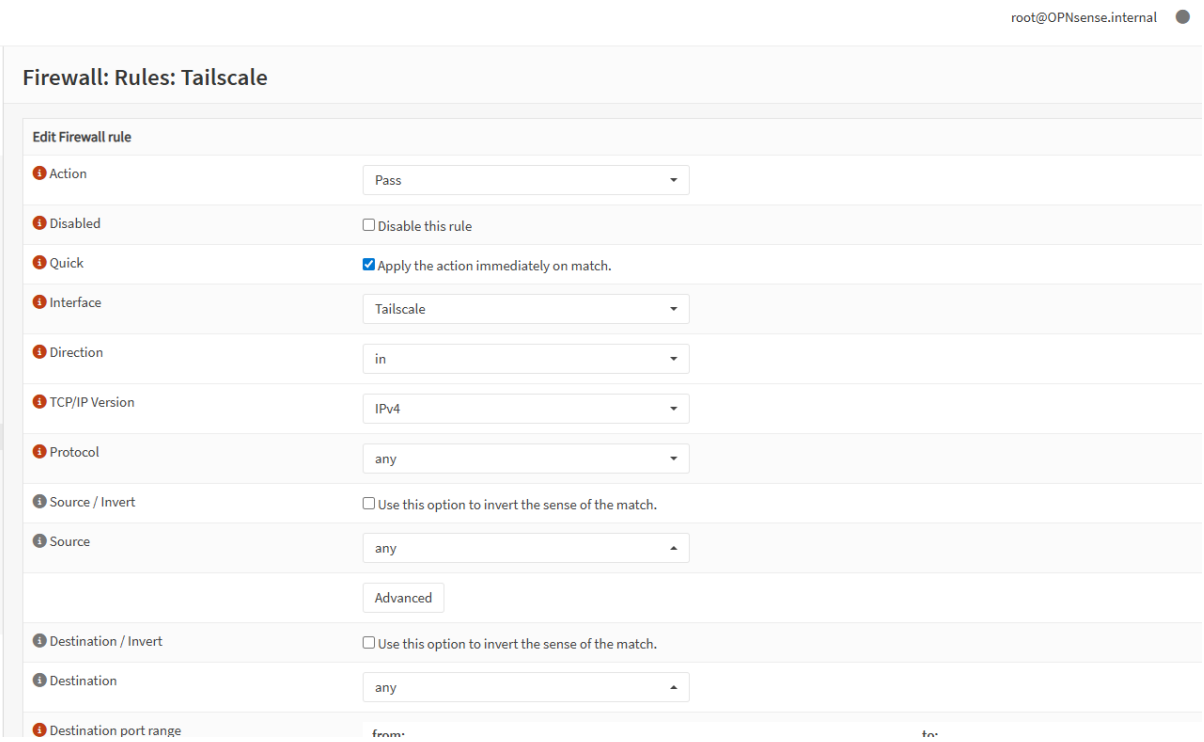
รูปที่ 3-35 เปิดใช้งาน interface ของ talescale

3.5.32 เข้าไปหน้า Firewall Rules บน Interface talescale โดยไปที่ Firewall > Rules > Tailscale



รูปที่ 3-36 แสดงหน้า Firewall Rules บน Interface talescale

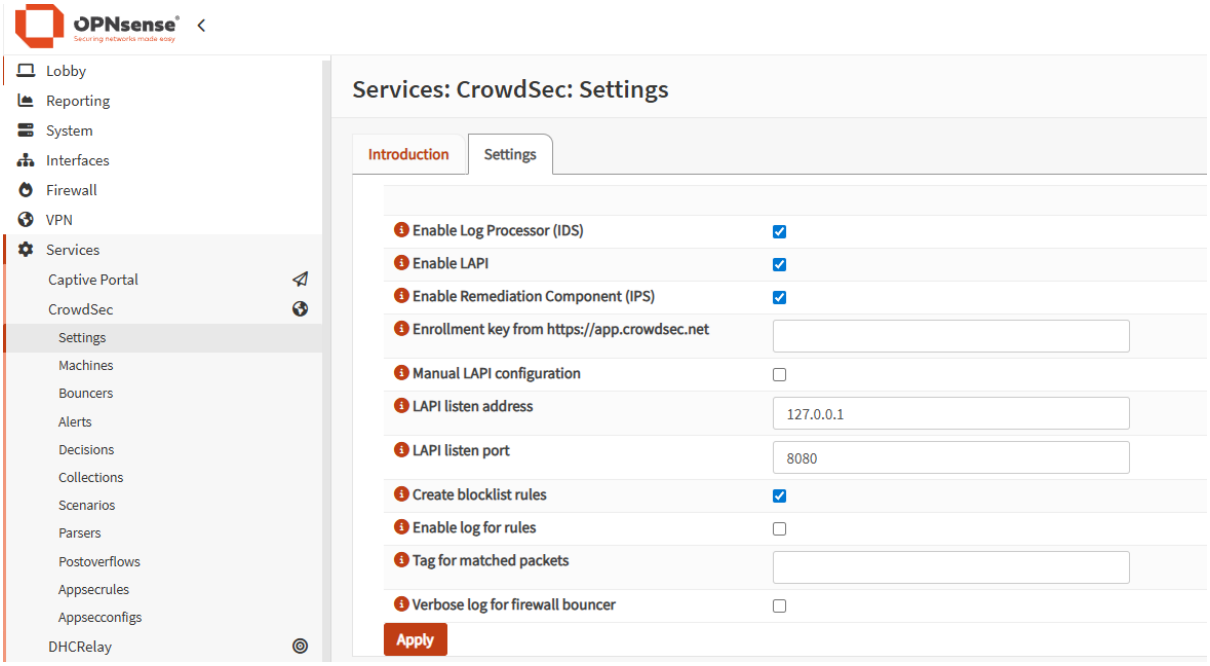
3.5.33 เพิ่ม Rule แบบ Pass สำหรับ talescale โดยกดที่เครื่องหมาย + เพื่อเพิ่ม Rule ใหม่ จากนั้นตั้ง Action เป็น Pass, Interface เป็น Talescale, Direction เป็น in และตั้ง Source เป็น any, Destination เป็น any แล้วกด Save



รูปที่ 3-37 เพิ่ม Rule แบบ Pass สำหรับ talescale

3.6 การติดตั้ง CrowdSec

3.6.1 กำหนด address และ port ของ CorwdSec โดยเข้าไปที่ Services > CrowdSec > Settings กำหนด LAPI listen address เป็น 127.0.0.1 และกำหนด LAPI listen port เป็น 8080 แล้วกด Apply



รูปที่ 3-38 กำหนด address และ port ของ CorwdSec บนเว็บไซต์ของ OPNsense

3.6.2 ฝั่งเครื่องเซิร์ฟเวอร์ OPNsense หลังจาก log in เสร็จ ให้เลือก option 8

```
*** OPNsense.internal: OPNsense 26.1.2 (amd64) ***

 LAN (rl0)       -> v4: 192.168.1.1/24
 Tailscale (tailscale0) -> v4: 100.119.69.73/32
                        v6: fd7a:115c:a1e0::8401:459b/48
 Vlan10 (vlan01) -> v4: 192.168.10.1/24
 Vlan20 (vlan02) -> v4: 192.168.20.1/24
 WAN (re0)       -> v4/DHCP4: 192.168.0.100/24

 HTTPS: SHA256 F1 44 D0 FA 9D 15 33 F9 6A 47 25 51 92 37 F9 CF
               EE BE 96 CD CE 21 50 F3 35 A5 76 EB D1 F2 1E 30
 SSH:   SHA256 yfOoe0EL6ddCrLd5hZDuEuDOuc4C/ek7U/l5LZzoKdk (ECDSA)
 SSH:   SHA256 0qoqOrwkt/c1we19J3Cw32H6dVo/iyxuAji9rABpQCk (ED25519)
 SSH:   SHA256 0bi8yrF1/xOfzCZThWegraRHnW3gU8AXRTNqluBKoBI (RSA)

  0) Logout                              7) Ping host
  1) Assign interfaces                   8) Shell
  2) Set interface IP address            9) pfTop
  3) Reset the root password            10) Firewall log
  4) Reset to factory defaults          11) Reload all services
  5) Power off system                   12) Update from console
  6) Reboot system                      13) Restore a backup

Enter an option: 8
```

รูปที่ 3-39 เลือก option 8 หลังจาก log in บนเครื่องเซิร์ฟเวอร์ OPNsense

3.6.3 install CrowdSec บนเครื่องเซิร์ฟเวอร์ โดยพิมพ์คำสั่ง cscli collections install crowdsecurity/suricata

```
root@OPNsense:~ # cscli collections install crowdsecurity/suricata
Action plan:
 download
  collections: crowdsecurity/suricata (0.2)
  contexts: crowdsecurity/suricata_base (0.2)
  scenarios: crowdsecurity/suricata-alerts (0.5)
  parsers: crowdsecurity/suricata-logs (0.6)
 enable
  collections: crowdsecurity/suricata
  contexts: crowdsecurity/suricata_base
  scenarios: crowdsecurity/suricata-alerts
  parsers: crowdsecurity/suricata-logs

downloading parsers:crowdsecurity/suricata-logs
downloading scenarios:crowdsecurity/suricata-alerts
downloading contexts:crowdsecurity/suricata_base
downloading collections:crowdsecurity/suricata
enabling parsers:crowdsecurity/suricata-logs
enabling scenarios:crowdsecurity/suricata-alerts
enabling contexts:crowdsecurity/suricata_base
enabling collections:crowdsecurity/suricata
```

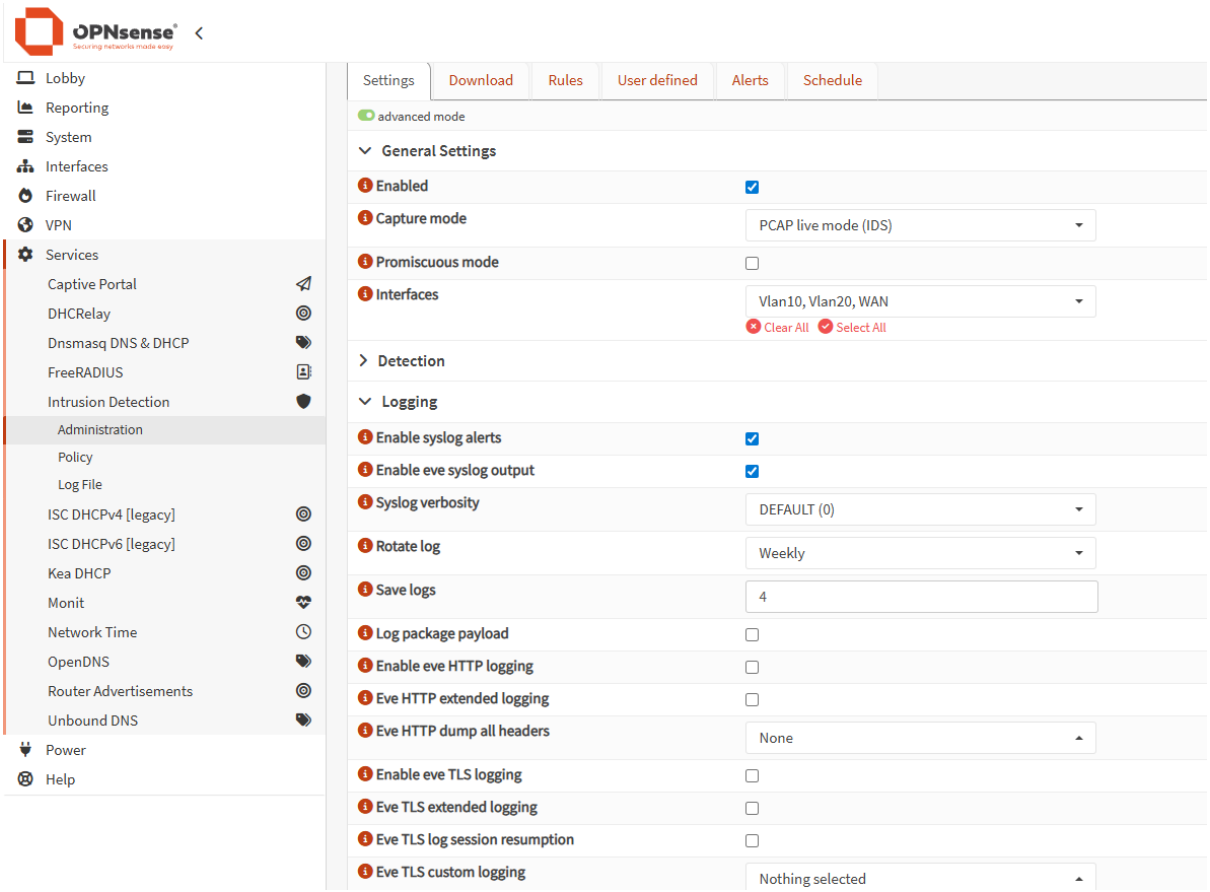
รูปที่ 3-40 install CrowdSec บนเครื่องเซิร์ฟเวอร์

3.6.4 พิมพ์คำสั่ง nano /usr/local/etc/crowdsec/acquis.yaml แล้วแก้ไขไฟล์ log ตามภาพ

```
  GNU nano 8.7
filenames:
  - /var/log/nginx/*.log
  - ./tests/nginx/nginx.log
#this is not a syslog log, indicate which kind of logs it is
labels:
  type: nginx
---
filenames:
 - /var/log/auth.log
 - /var/log/syslog
labels:
  type: syslog
---
filenames:
 - /var/log/httpd-access.log
 - /var/log/httpd-error.log
labels:
  type: apache2
---
filenames:
 - /var/log/suricata/eve.json
labels:
  type: suricata-evelogs
```

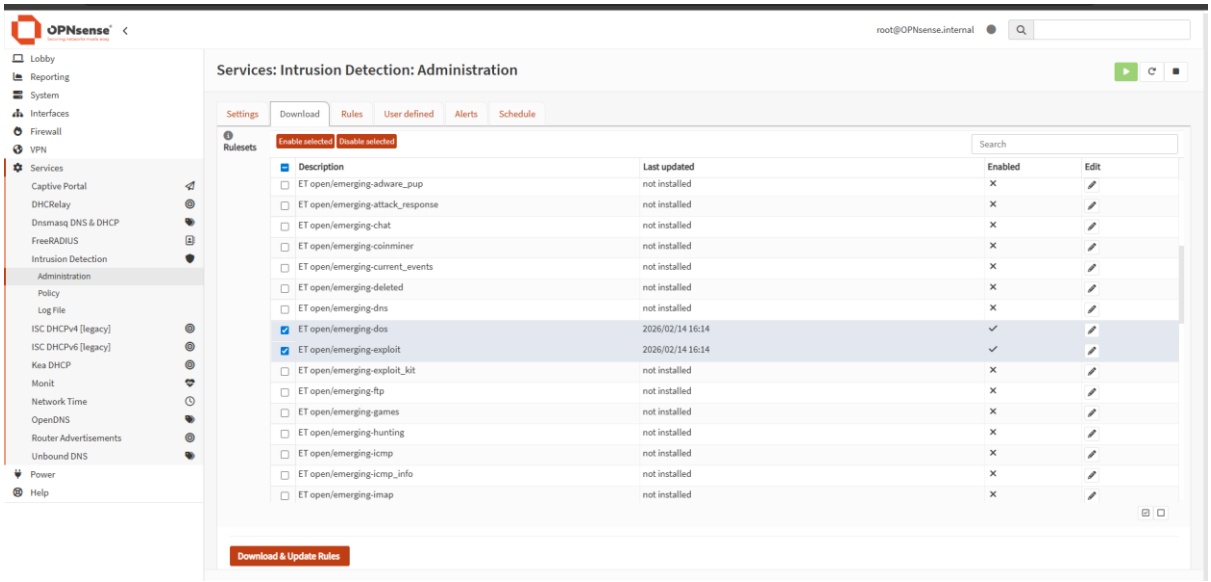
รูปที่ 3-41 แก้ไขไฟล์ log

3.6.5 เปิดแจ้งเตือน log โดยไปที่ Services > Intrusion Detection > Administration แล้วตั้งค่าตามภาพ



รูปที่ 3-42 เปิดแจ้งเตือน log

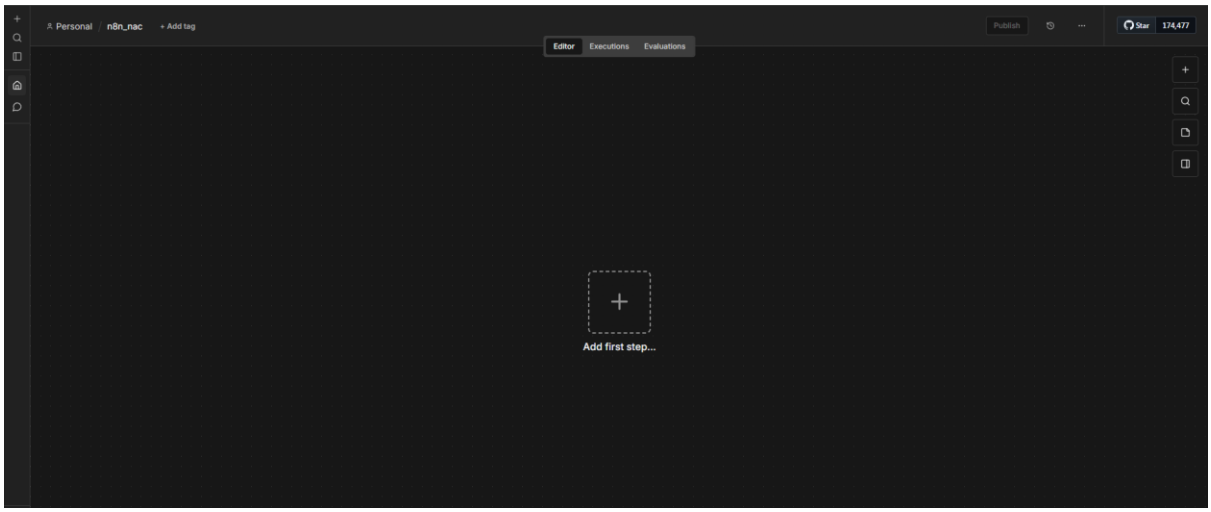
3.6.6 ไปที่แท็บ Download แล้วเลือก ET open/emerging-dos และ ET open/emerging-exploit จากนั้นกดปุ่ม Download & Update Rules



รูปที่ 3-43 ดาวน์โหลด Rules ET open/emerging-dos และ ET open/emerging-exploit

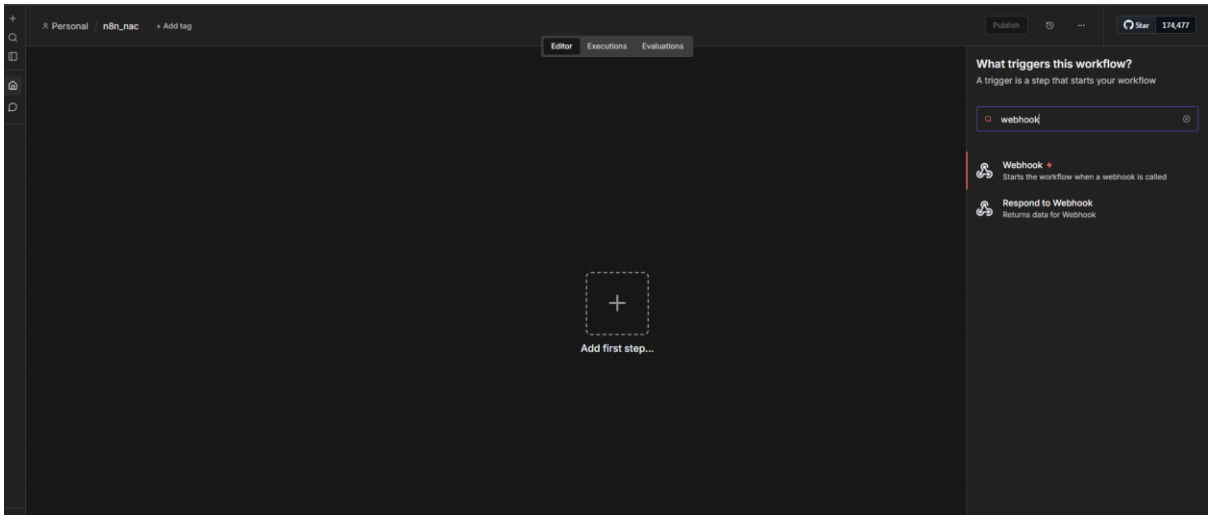
3.7 ตั้งค่าการเชื่อมต่อ จาก CrowdSec ไป n8n

3.7.1 กดปุ่มเครื่องหมาย + ตรงกลาง เพื่อสร้าง node



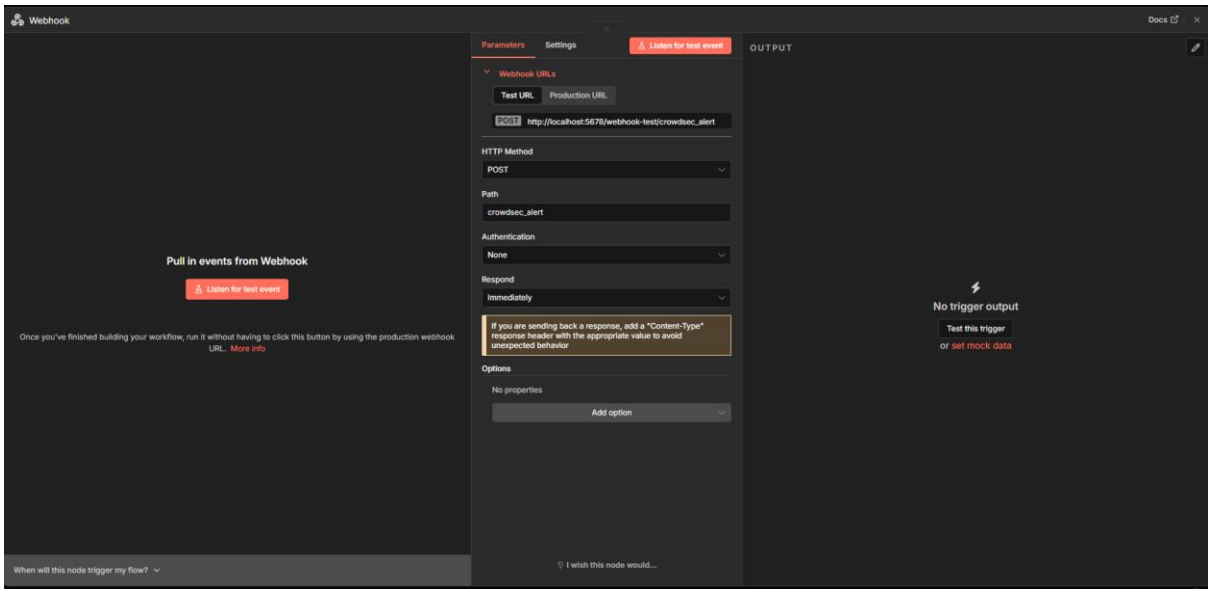
รูปที่ 3-44 สร้าง Trigger ใน n8n

3.7.2 ค้นหาโดยพิมพ์คำว่า webhook แล้วเลือก node ที่เขียนว่า webhook



รูปที่ 3-45 เลือก node เป็น webhook

3.7.3 กำหนด Parameters ตามภาพด้านล่าง



รูปที่ 3-46 กำหนด Parameters

3.7.4 รันคำสั่ง rm /usr/local/etc/crowdsec/notifications/http.yaml และ nano /usr/local/etc/crowdsec/notifications/http.yaml บน server OPNsense

```
root@OPNsense:~ # rm /usr/local/etc/crowdsec/notifications/http.yaml
root@OPNsense:~ # nano /usr/local/etc/crowdsec/notifications/http.yaml
```

รูปที่ 3-47 รันคำสั่ง rm และ nano บน server OPNsense

3.7.5 ตั้งค่าตามภาพด้านล่าง เพื่อให้ CrowdSec ส่ง alert ไปที่ Webhook ของ n8n ในรูปแบบ JSON

```
  GNU nano 8.7                                                    /us
type: http
name: http_default
log_level: info
format: |
  {{ . | toJson }}
url: http://100.118.179.21:5678/webhook/crowdsec_alert
method: POST
headers:
  Content-Type: application/json
```

รูปที่ 3-48 ตั้งค่า เพื่อให้ CrowdSec ส่ง alert ไปที่ Webhook ของ n8n ในรูปแบบ JSON

3.7.6 จากนั้นพิมพ์คำสั่ง nano /usr/local/etc/crowdsec/profiles.yaml เพื่อแก้ไขไฟล์ profiles.yaml

```
root@OPNsense:~ # nano /usr/local/etc/crowdsec/profiles.yaml
```

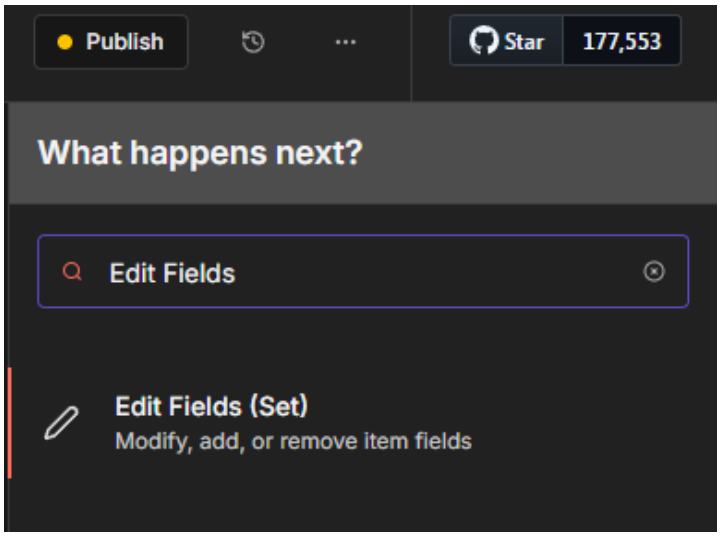
รูปที่ 3-49 พิมพ์คำสั่ง nano /usr/local/etc/crowdsec/profiles.yaml เพื่อแก้ไขไฟล์ profiles.yaml

3.7.7 ตั้งค่า Profile.yaml ตามภาพด้านล่าง เพื่อให้ส่งแจ้งเตือน

```
  GNU nano 8.7                                                    /usr/local/etc/crowdsec/profiles.yaml
name: default_ip_remediation
#debug: true
filters:
 - Alert.Remediation == true && Alert.GetScope() == "Ip"
decisions:
 - type: ban
   duration: 4h
#duration_expr: Sprintf('%dh', (GetDecisionsCount(Alert.GetValue()) + 1) * 4)
notifications:
#   - slack_default  # Set the webhook in /etc/crowdsec/notifications/slack.yaml before enabling this.
#   - splunk_default # Set the splunk url and token in /etc/crowdsec/notifications/splunk.yaml before enabling this.
 - http_default   # Set the required http parameters in /etc/crowdsec/notifications/http.yaml before enabling this.
#   - email_default  # Set the required email parameters in /etc/crowdsec/notifications/email.yaml before enabling this.
on_success: break
---
name: default_range_remediation
#debug: true
filters:
 - Alert.Remediation == true && Alert.GetScope() == "Range"
decisions:
 - type: ban
   duration: 4h
#duration_expr: Sprintf('%dh', (GetDecisionsCount(Alert.GetValue()) + 1) * 4)
# notifications:
#   - slack_default  # Set the webhook in /etc/crowdsec/notifications/slack.yaml before enabling this.
#   - splunk_default # Set the splunk url and token in /etc/crowdsec/notifications/splunk.yaml before enabling this.
#   - http_default   # Set the required http parameters in /etc/crowdsec/notifications/http.yaml before enabling this.
#   - email_default  # Set the required email parameters in /etc/crowdsec/notifications/email.yaml before enabling this.
on_success: break
```

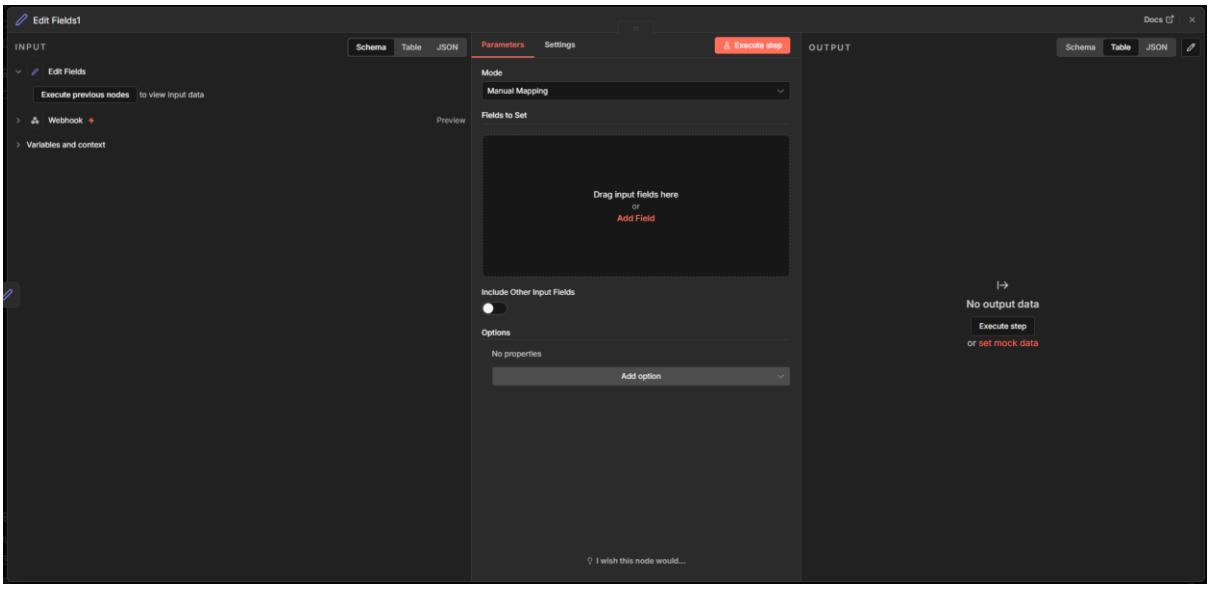
รูปที่ 3-50 ตั้งค่า Profile.yaml เพื่อให้ส่งแจ้งเตือน

3.7.8 เพิ่ม node Edit Fields โดยกดปุ่มเครื่องหมาย + แล้วค้นหา Edit Fields เลือก Edit Fields (Set)



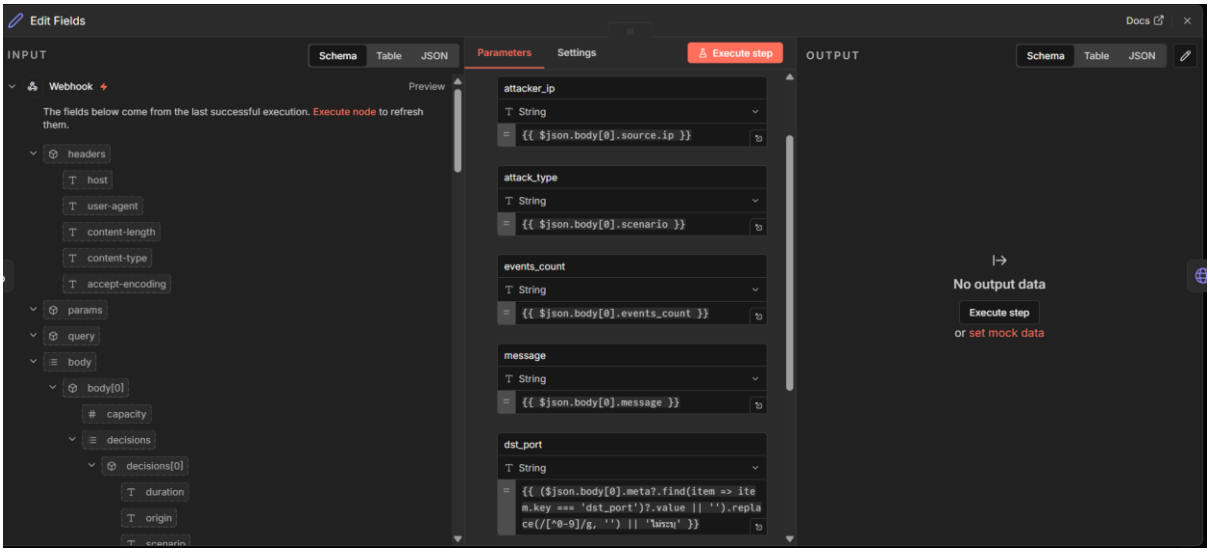
รูปที่ 3-51 เพิ่ม node Edit Fields

3.7.9 กดปุ่ม add field



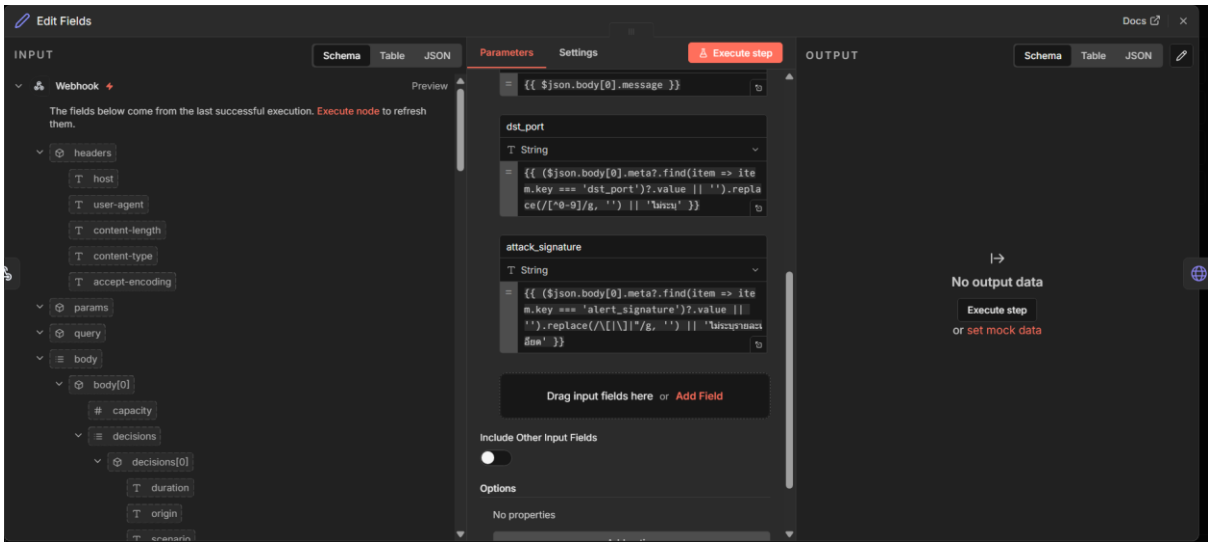
รูปที่ 3-52 กดปุ่ม add field

3.7.10 ตั้งค่า Edit Field ตามภาพด้านล่าง



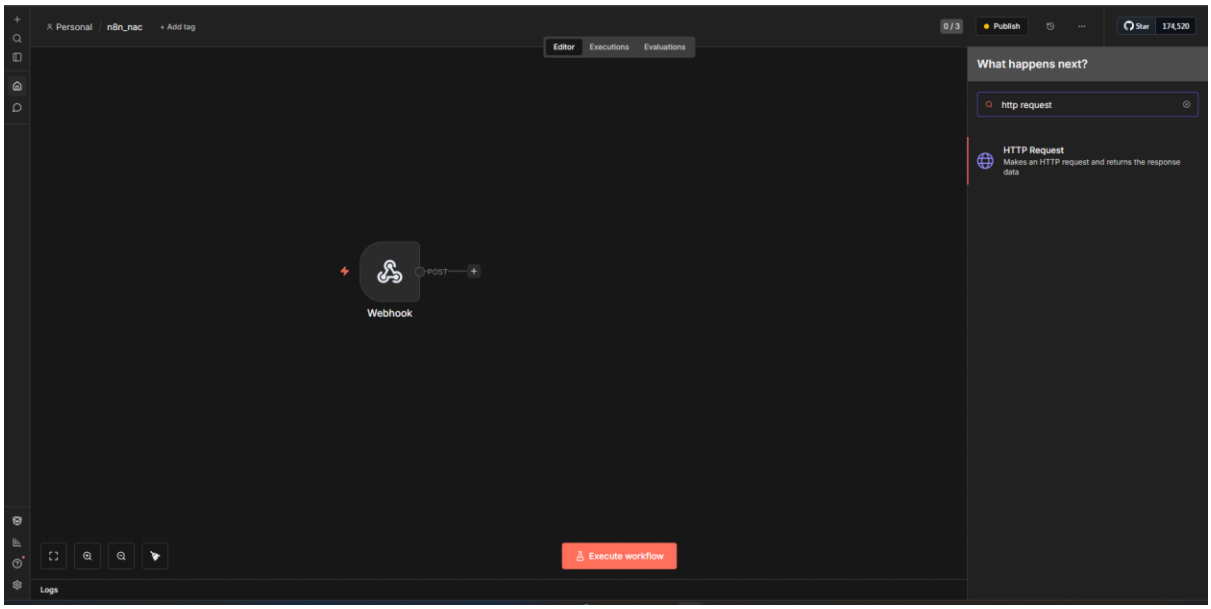
รูปที่ 3-53 ตั้งค่า Edit Field ตามภาพด้านล่าง

3.7.11 ตั้งค่า Edit Field ตามภาพด้านล่าง (ต่อ) แล้วกดปิด node



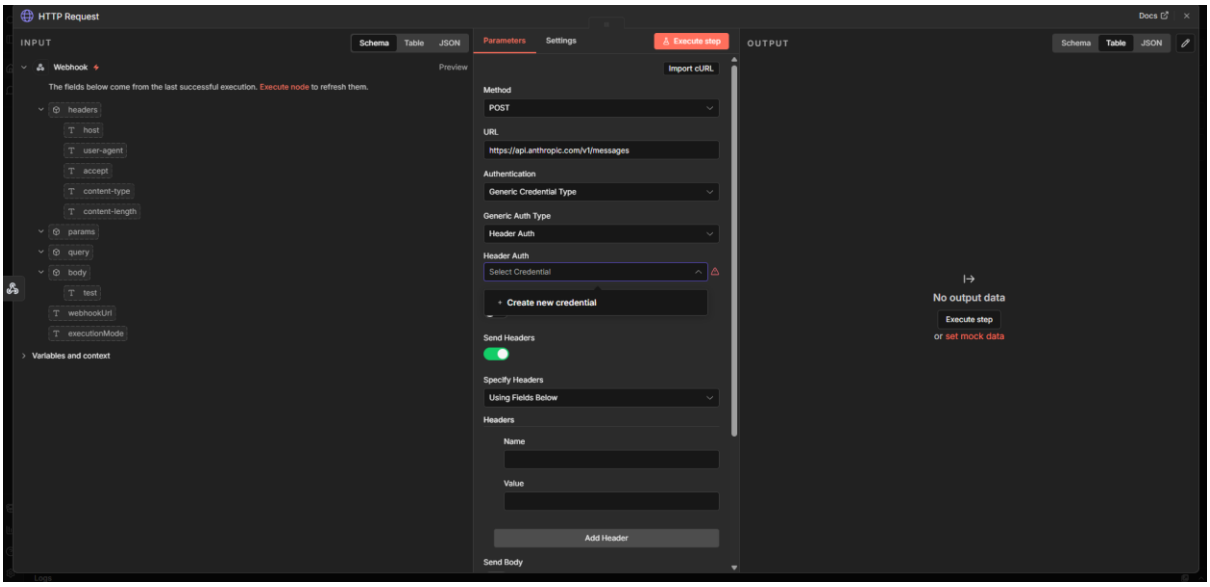
รูปที่ 3-54 ตั้งค่า Edit Field ตามภาพด้านล่าง (ต่อ)

3.7.12 สร้าง node ใหม่ โดยใช้เป็น http request



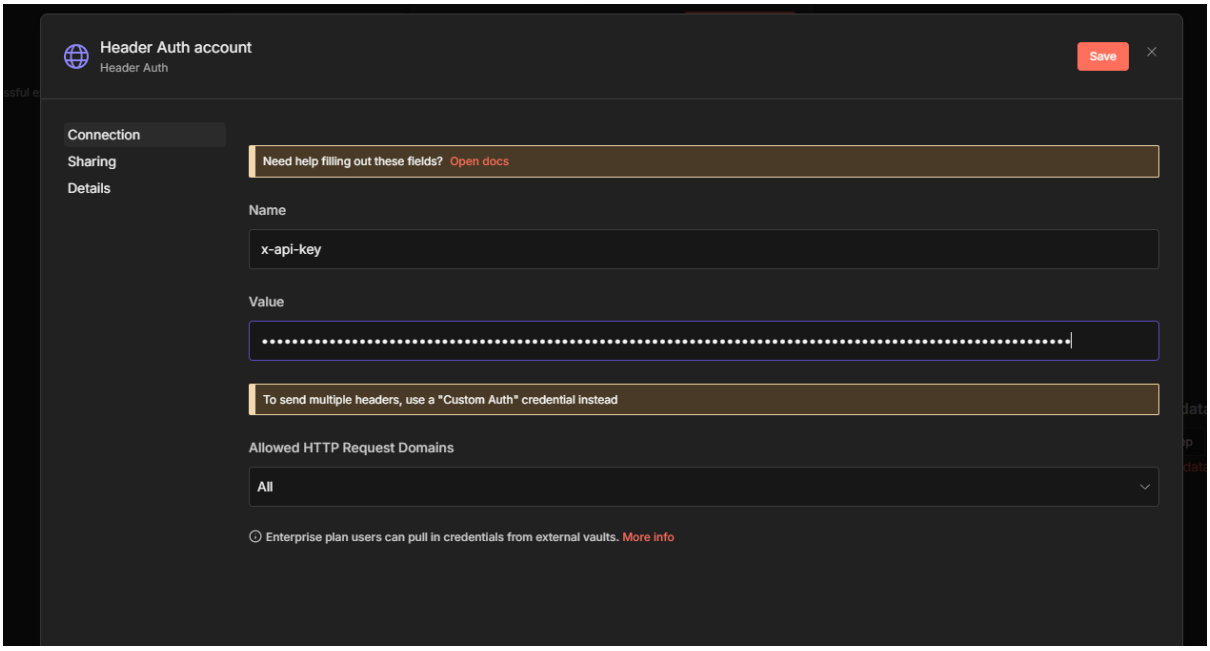
รูปที่ 3-55 สร้าง node http request

3.7.13 ตั้งค่า node http request แล้วกดปุ่ม Create new credential



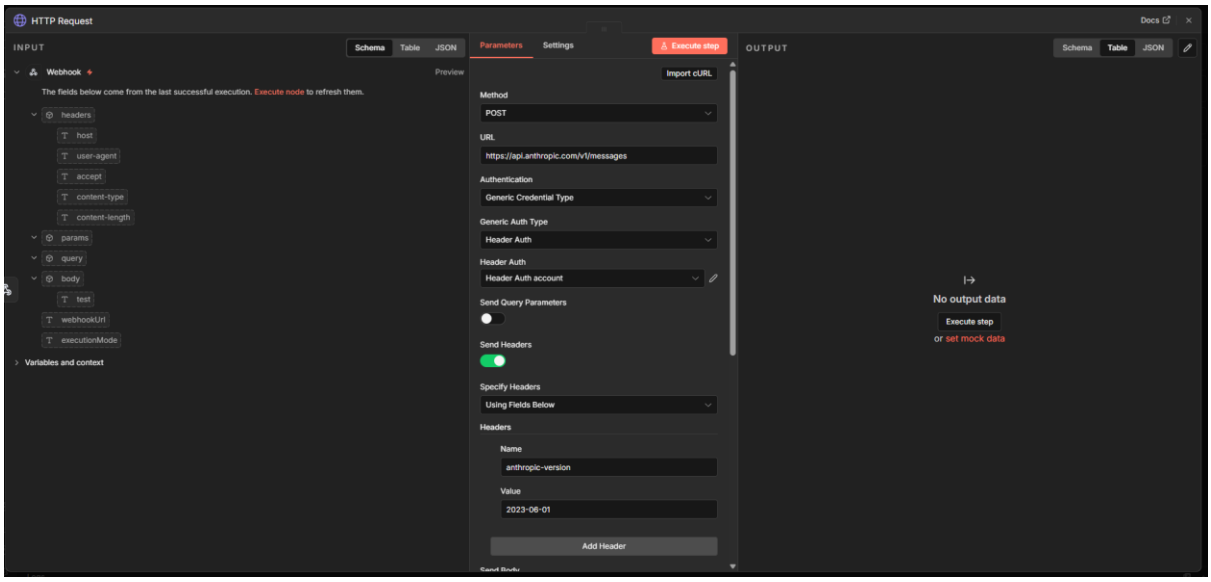
รูปที่ 3-56 ตั้งค่า node http request

3.7.14 นำ key ของ claude api มาใส่ใน value



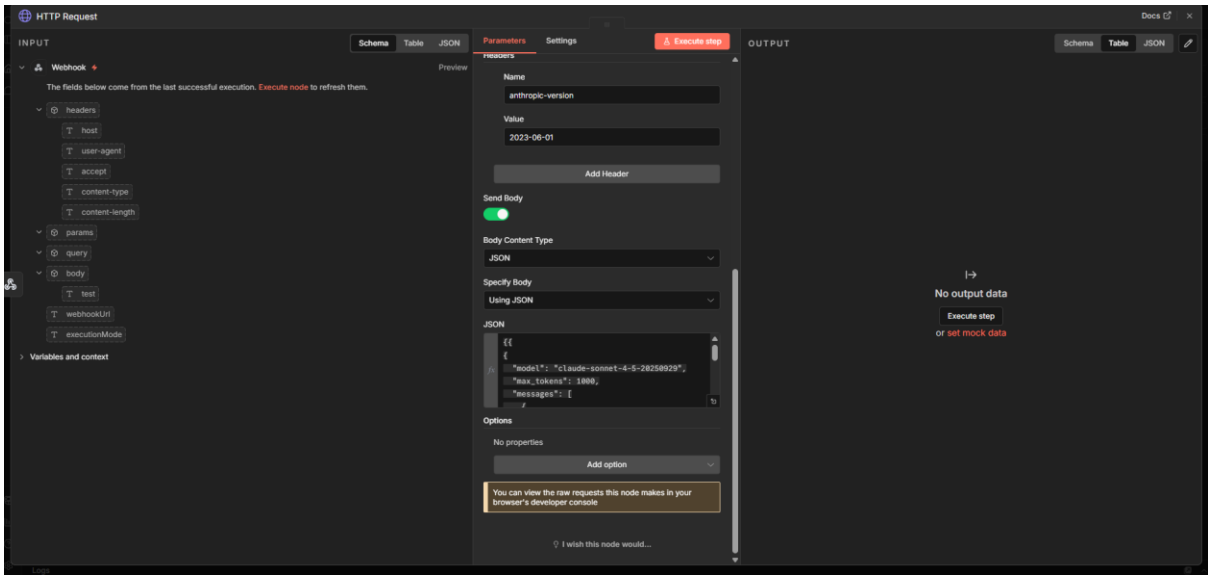
รูปที่ 3-57 นำ key ของ claude api มาใส่ใน value

3.7.15 ตั้งค่า node http request (ต่อ)



รูปที่ 3-58 ตั้งค่า node http request (ต่อ)

3.7.16 ตั้งค่า node http request (ต่อ)

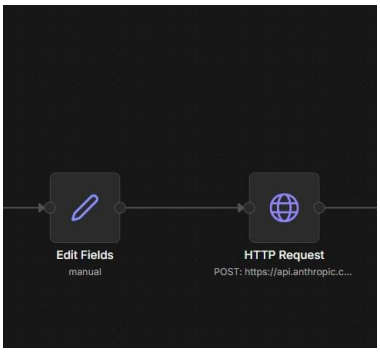


รูปที่ 3-59 ตั้งค่า node http request (ต่อ)

Code JSON ที่ใช้ใน 3.7.16

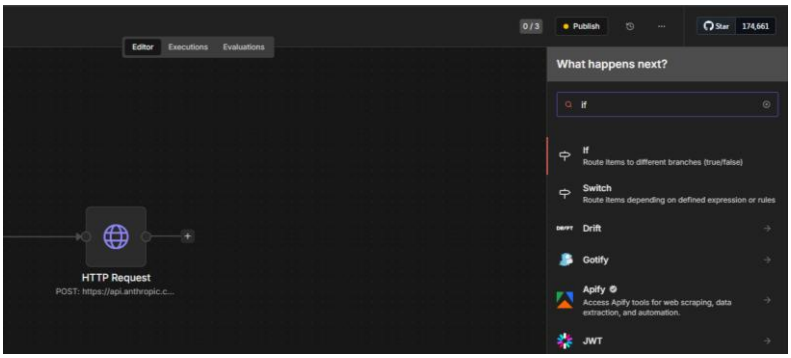
```
{ "model": "claude-sonnet-4-5-20250929", "max_tokens": 1500, "messages": [ { "role": "user", "content": "คุณคือผู้เชี่ยวชาญ Cyber Security ระดับ Senior ของทีม SOC (Security Operations Center)\n\nข้อมูลการโจมตี:\n- IP คนร้าย: {{$json.attacker_ip}}\n- พอร์ตเป้าหมาย: {{$json.target_port}}\n- ประเภทการโจมตี: {{$json.attack_type}}\n- รายละเอียดเชิงลึก (Signature): {{$json.attack_signature}}\n- จำนวนครั้งที่โจมตี: {{$json.events_count}}\n- ข้อความแจ้งเตือนจากระบบ: {{$json.message}}\n\nหน้าที่ของคุณและกฎเกณฑ์การประเมิน (Strict Rules):\n1. วิเคราะห์การโจมตี (อธิบายว่าคืออะไร อันตรายแค่ไหน แนะนำวิธีป้องกัน)\n2. ประเมินคะแนนความรุนแรง (Severity Score) เป็นตัวเลข 1-100 โดยอิงตามเกณฑ์บังคับดังต่อไปนี้:\n - Score 1-40: พฤติกรรมผิดปกติทั่วไปที่ความเสี่ยงต่ำ\n - Score 41-74: การสแกนเครือข่ายหรือสแกนพอร์ต (Reconnaissance) เช่น port-scan\n - Score 75-100: ความพยายามเจาะเข้าระบบ (Exploit), การเดารหัสผ่าน (Brute Force ทั้งแบบ Fast และ Slow เช่น ssh-bf, ssh-slow-bf), หรือการโจมตีแอปพลิเคชัน\n3. กฎเหล็ก (CRITICAL): ห้ามหักคะแนนหรือมองว่าปลอดภัยเพียงเพราะระบบด่านหน้า (Firewall/CrowdSec) ทำการบล็อกชั่วคราวไปแล้ว ให้ประเมินคะแนนจาก 'เจตนามุ่งร้ายดั้งเดิม' ของรูปแบบการโจมตีนั้นๆ อย่างเคร่งครัด เพราะคะแนนนี้จะถูกใช้เป็นเกณฑ์ในการสั่งลงโทษแบบถาวร (Permanent Ban)\n\nบังคับ: ให้ตอบกลับมาเป็นรูปแบบ JSON ที่ถูกต้องสมบูรณ์เท่านั้น ห้ามพิมพ์ข้อความอารัมภบทหรือคำลงท้ายใดๆ นอกเหนือจากวงเล็บปีกกา { } โดยใช้โครงสร้างดังนี้:\n{\n \"severity_score\": ตัวเลขตั้งแต่ 1 ถึง 100,\n \"analysis\": \"คำอธิบายการโจมตี การประเมินความรุนแรง และคำแนะนำ (ตอบเป็นภาษาไทย แบบละเอียดและเป็นทางการ)\",\n \"reason\": \"เหตุผลที่ให้คะแนนระดับนี้ (ตอบเป็นภาษาไทย)\"\n}" } ] }
```

3.7.17 เชื่อมต่อจาก Edit Fields เข้า http request



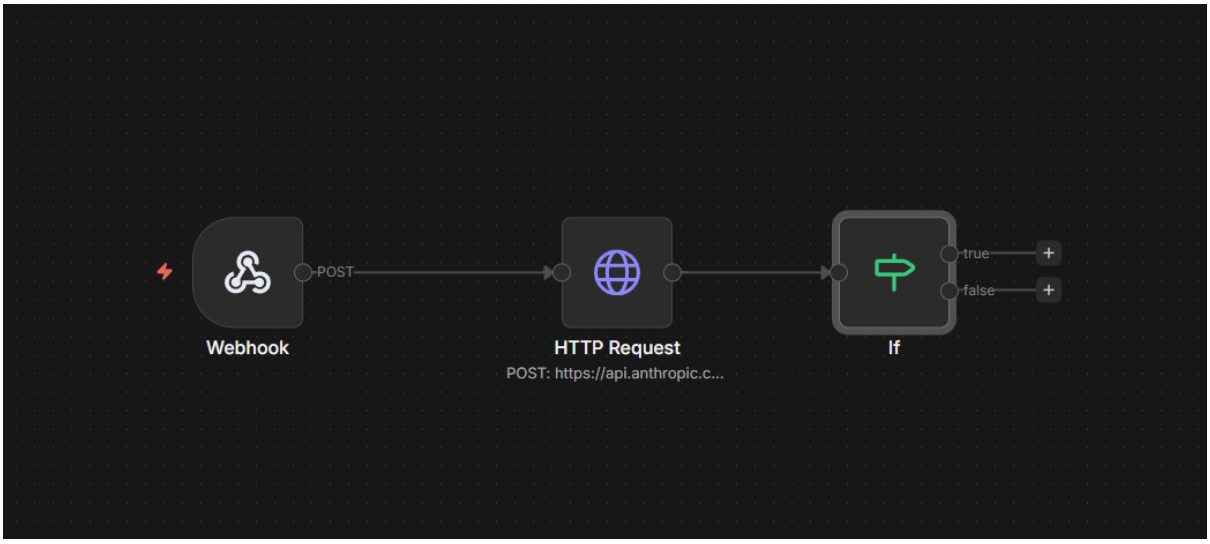
รูปที่ 3-60 เชื่อมต่อจาก Edit Fields เข้า http request

3.7.18 เพิ่ม node if



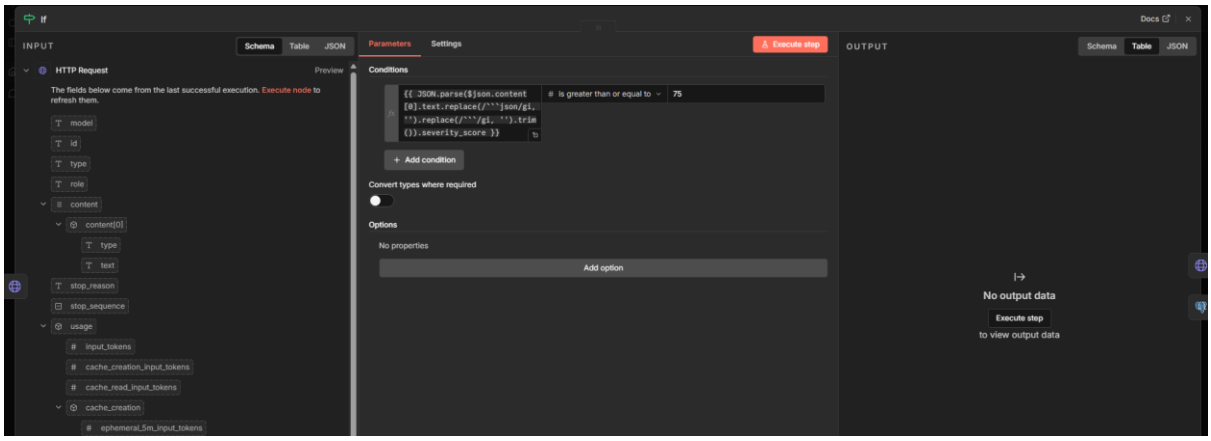
รูปที่ 3-61 เพิ่ม node if

3.7.19 เชื่อมต่อจาก http request เข้า if



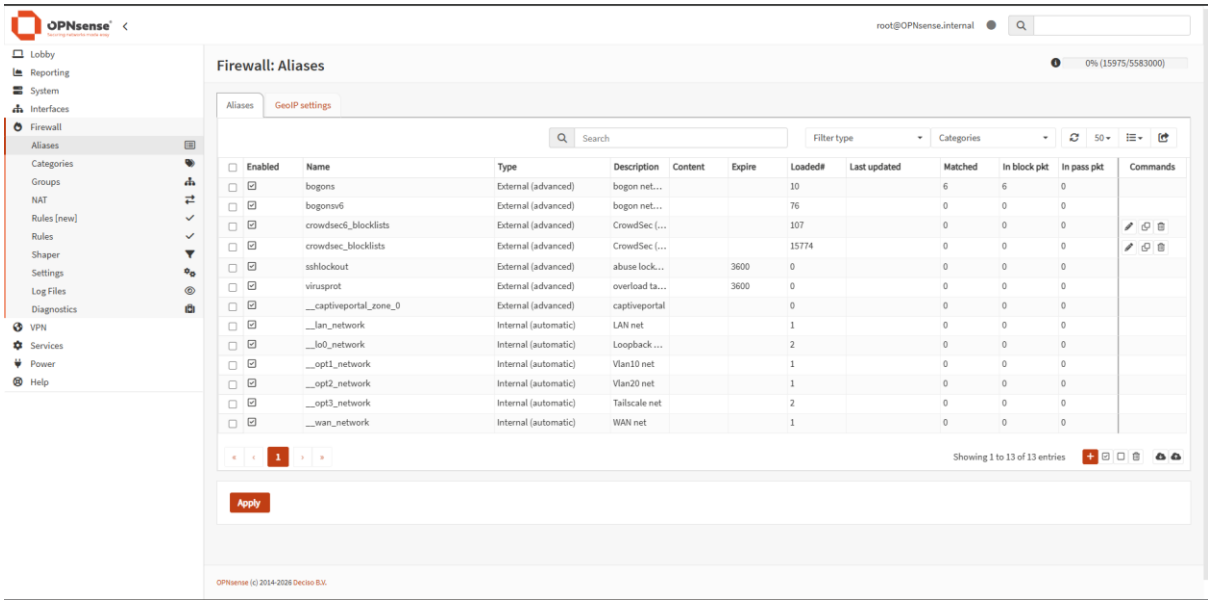
รูปที่ 3-62 เชื่อมต่อจาก http request เข้า if

3.7.20 ตั้งค่า เงื่อนไข ของ node if



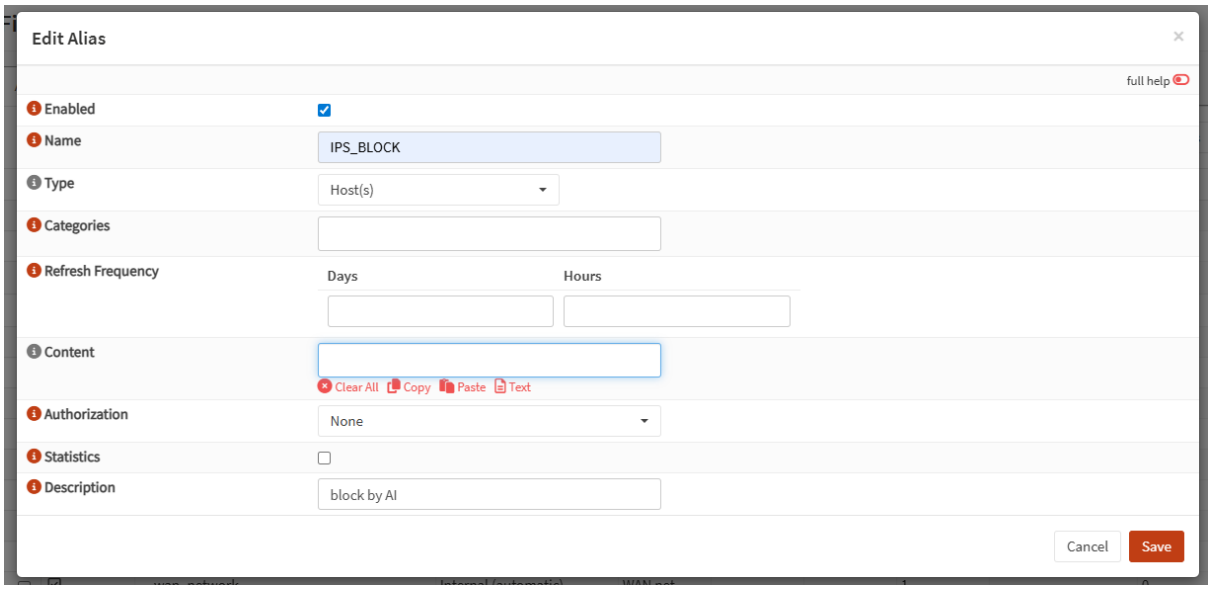
รูปที่ 3-63 ตั้งค่า เงื่อนไข ของ node if

3.7.21 ไปที่ Firewall > Aliases แล้วกดปุ่มเครื่องหมาย +



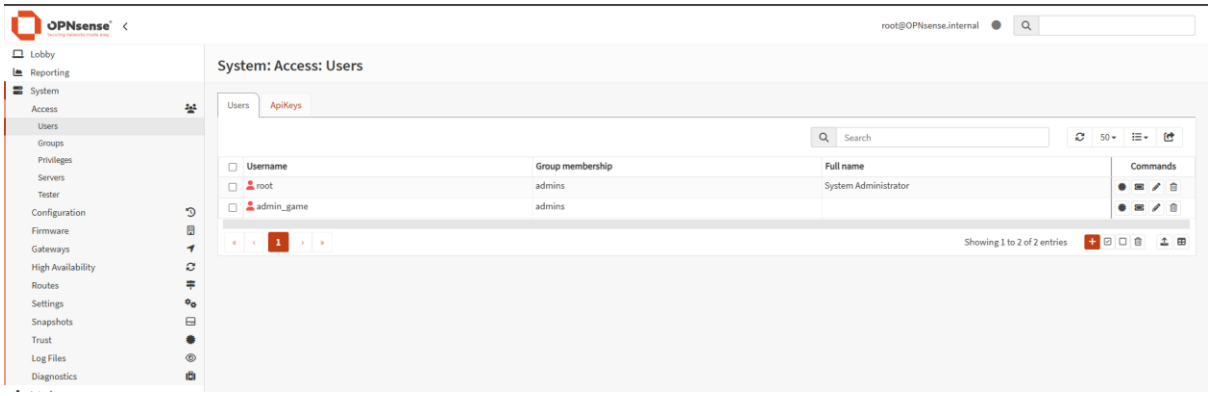
รูปที่ 3-64 ไปที่ Firewall > Aliases

3.7.22 ตั้งชื่อ IPS_block แล้วตั้งค่าตามรูปด้านล่าง



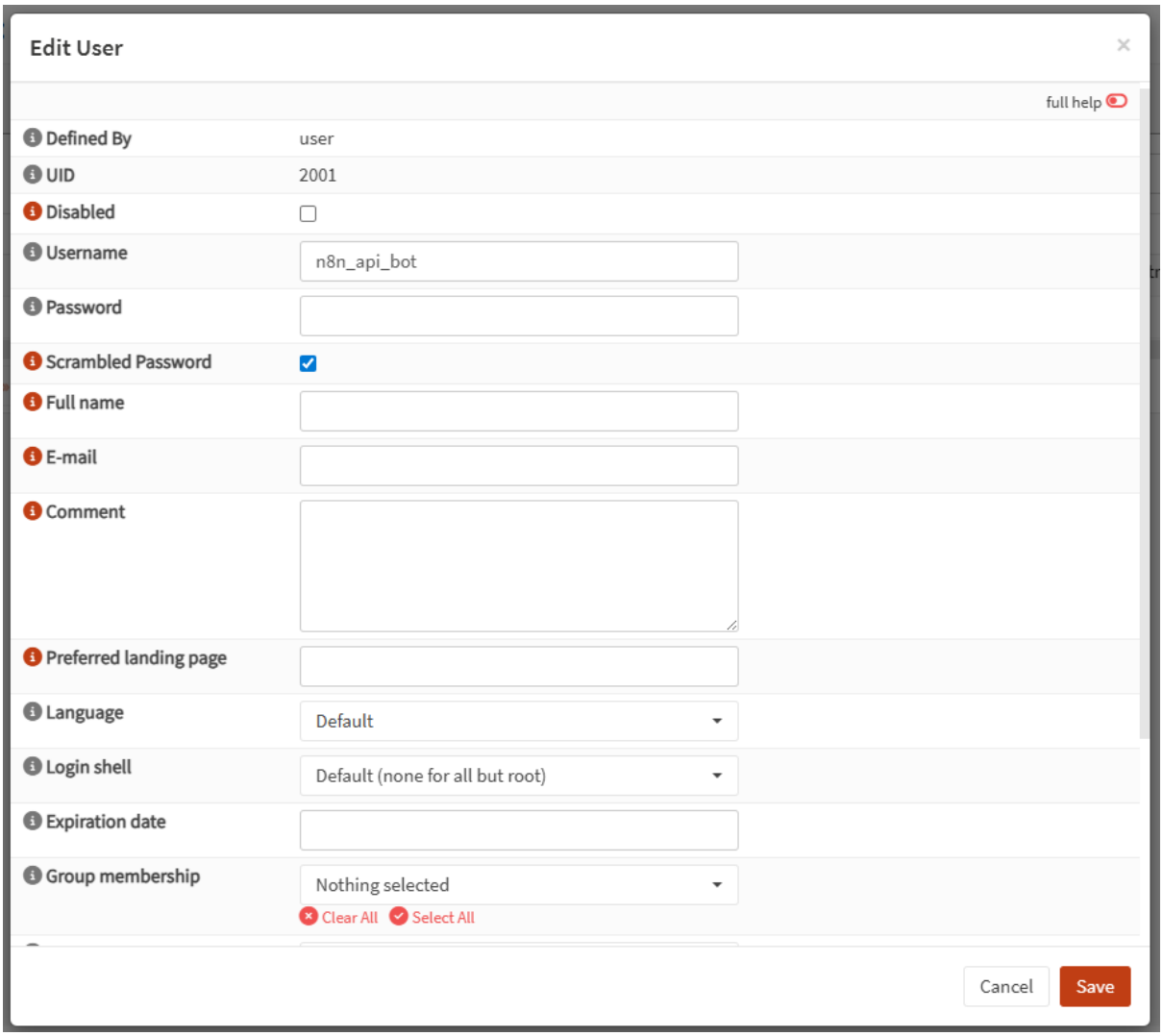
รูปที่ 3-65 ตั้งชื่อ IPS_block แล้วตั้งค่าตามรูปด้านล่าง

3.7.23 ไปที่ System > Access > Users แล้วกดปุ่มเครื่องหมายบวก เพื่อสร้าง User



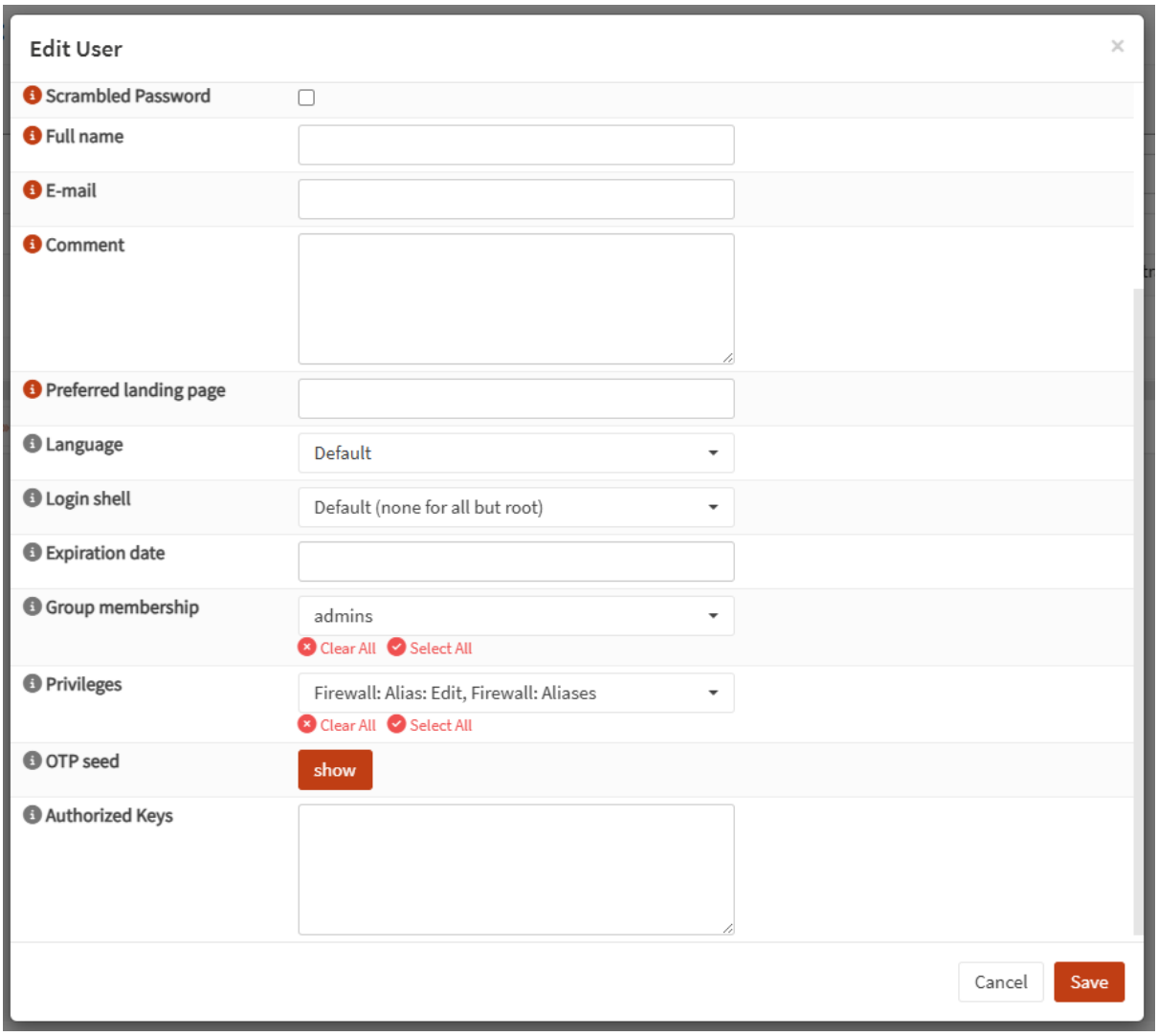
รูปที่ 3-66 ไปที่ System > Access > Users แล้วกดปุ่มเครื่องหมายบวก เพื่อสร้าง User

3.7.24 ตั้งค่า User ตามรูปด้านล่าง



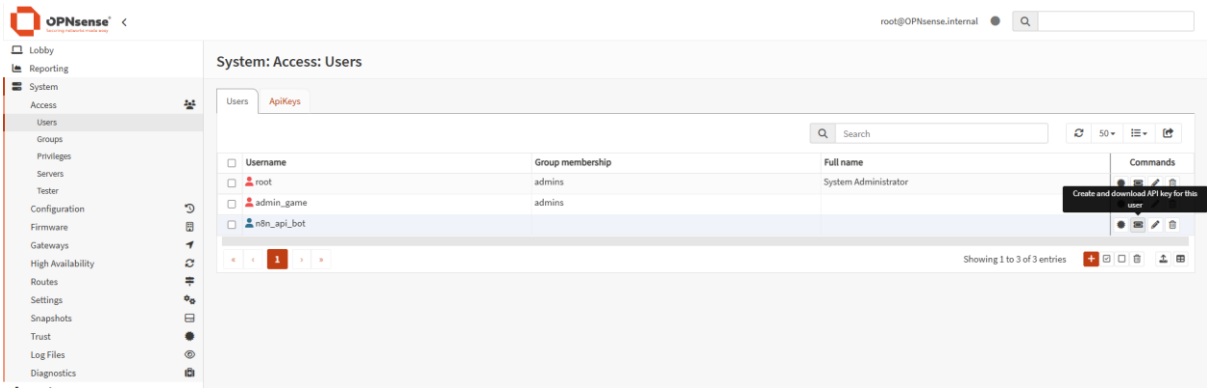
รูปที่ 3-67 ตั้งค่า User

3.7.25 ตั้งค่า User ตามรูปด้านล่าง (ต่อ) แล้วกดปุ่ม save



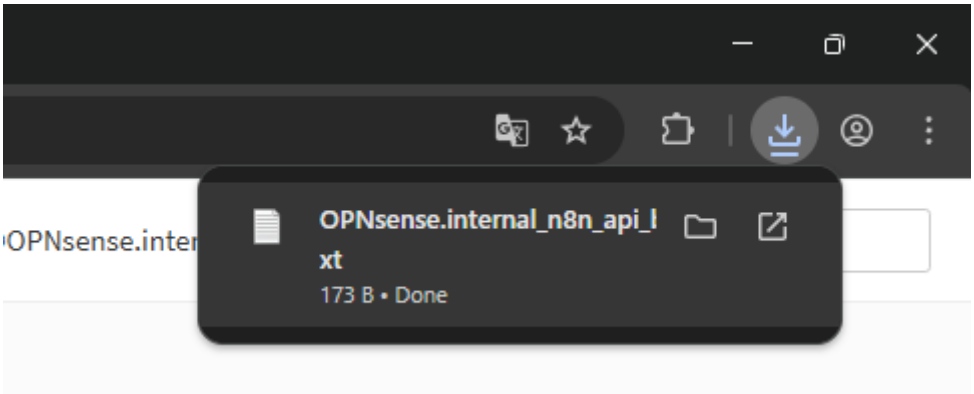
รูปที่ 3-68 ตั้งค่า User ตามรูปด้านล่าง (ต่อ) แล้วกดปุ่ม save

3.7.26 กดปุ่ม create and download API key for this user



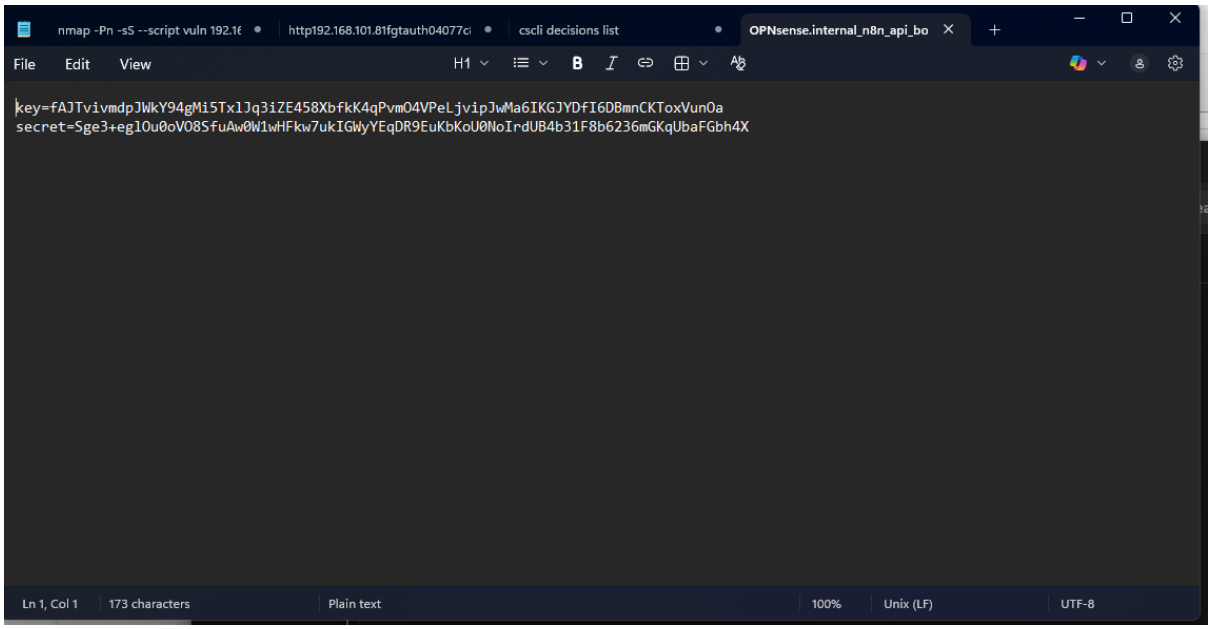
รูปที่ 3-69 กดปุ่ม create and download API key for this user

3.7.27 เปิดไฟล์ key ที่ดาวน์โหลด



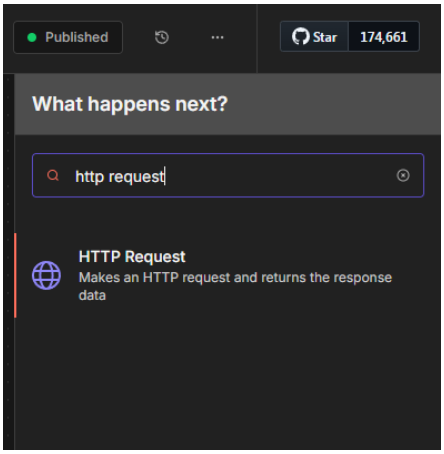
รูปที่ 3-70 เปิดไฟล์ key ที่ดาวน์โหลด

3.7.28 copy key ไฟล์ที่ดาวน์โหลด



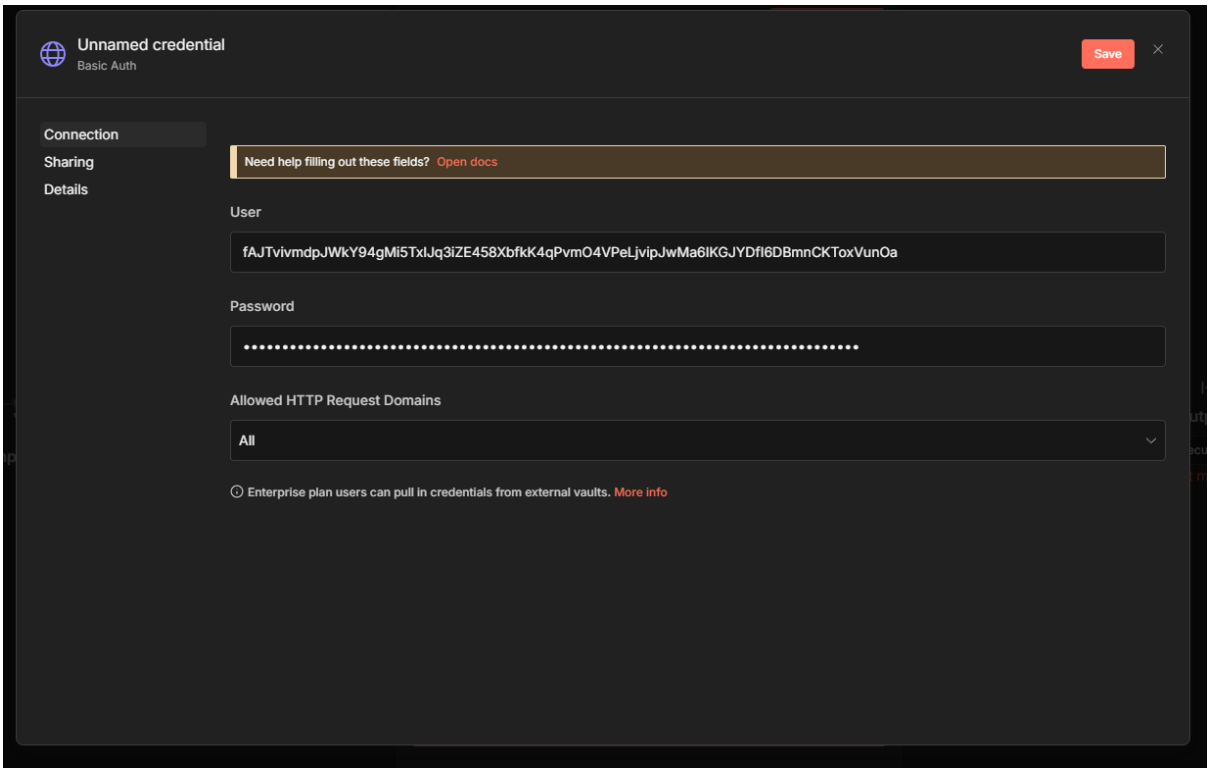
รูปที่ 3-71 copy key ไฟล์ที่ดาวน์โหลด

3.7.29 สร้าง node http reqest ใหม่ใน n8n



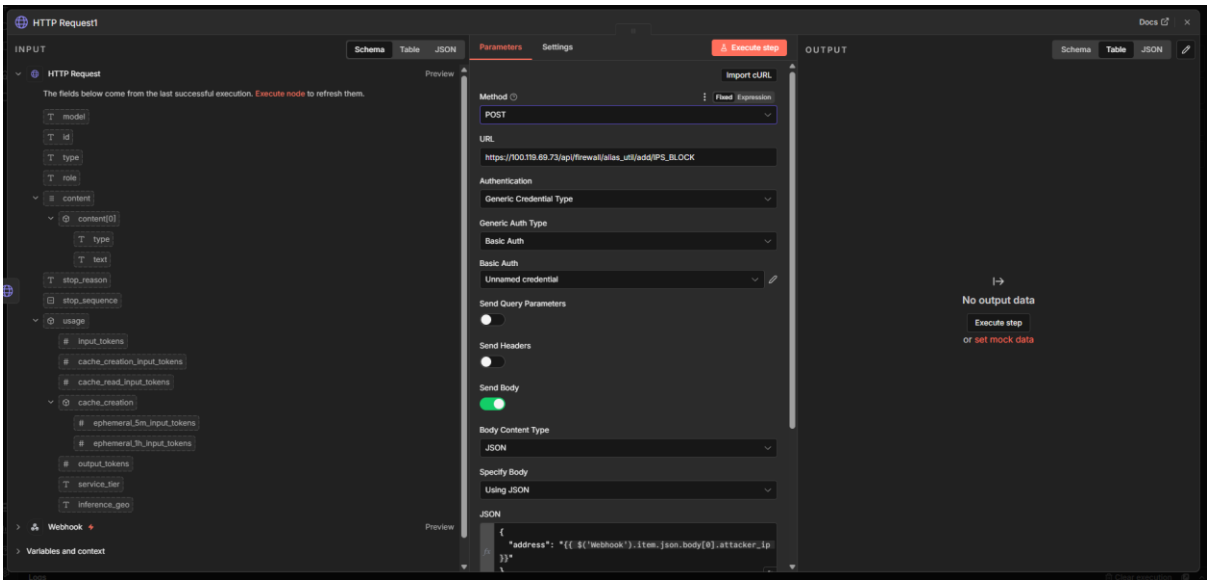
รูปที่ 3-72 สร้าง node http reqest ใหม่ใน n8n

3.7.30 นำ key มาใส่ใน http request เหมือน node ก่อนหน้า



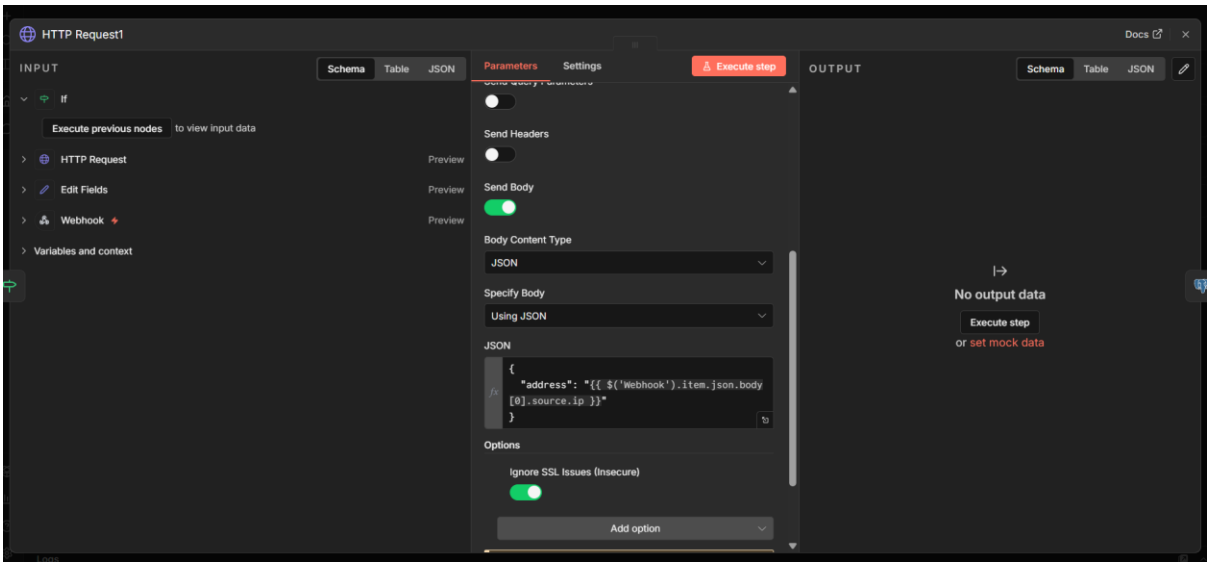
รูปที่ 3-73 นำ key มาใส่ใน http request เหมือน node ก่อนหน้า

3.7.31 ตั้งค่า node http request อันใหม่ตามรูปด้านล่าง



รูปที่ 3-74 ตั้งค่า node http request อันใหม่

3.7.32 ตั้งค่า node http request อันใหม่ตามรูปด้านล่าง (ต่อ)



รูปที่ 3-75 ตั้งค่า node http request อันใหม่ (ต่อ)

3.7.33 ใช้คำสั่ง nano docker-compose.yml เพื่อแก้ไขไฟล์ configuration ของระบบ n8n

```
n8n@n8n:~/n8n-nac$ nano docker-compose.yml
```

รูปที่ 3-76 ใช้คำสั่ง nano docker-compose.yml เพื่อแก้ไขไฟล์

3.7.34 แก้ไขไฟล์ docker-compose.yml

```
  GNU nano 7.2                                                    docker-compo
services:
  n8n:
    image: n8nio/n8n:latest
    restart: always
    ports:
      - "5678:5678"
    environment:
      - GENERIC_TIMEZONE=Asia/Bangkok
      - TZ=Asia/Bangkok
      - N8N_HOST=${DOMAIN_NAME:-localhost}
      - N8N_PORT=5678
      - N8N_PROTOCOL=http
      - WEBHOOK_URL=http://${DOMAIN_NAME:-localhost}:5678/
      - DB_TYPE=postgresdb
      - DB_POSTGRESDB_HOST=postgres
      - DB_POSTGRESDB_PORT=5432
      - DB_POSTGRESDB_DATABASE=n8n
      - DB_POSTGRESDB_USER=n8n
      - DB_POSTGRESDB_PASSWORD=admingamejk04_n8n
      - N8N_SECURE_COOKIE=false
    links:
      - postgres
    depends_on:
      - postgres
    volumes:
      - n8n_data:/home/node/.n8n

  postgres:
    image: postgres:16
    restart: always
    environment:
      - POSTGRES_USER=n8n
      - POSTGRES_PASSWORD=admingamejk04_n8n
      - POSTGRES_DB=n8n
    ports:
      - "5432:5432"
    volumes:
      - postgres_data:/var/lib/postgresql/data

volumes:
  n8n_data:
  postgres_data:
```

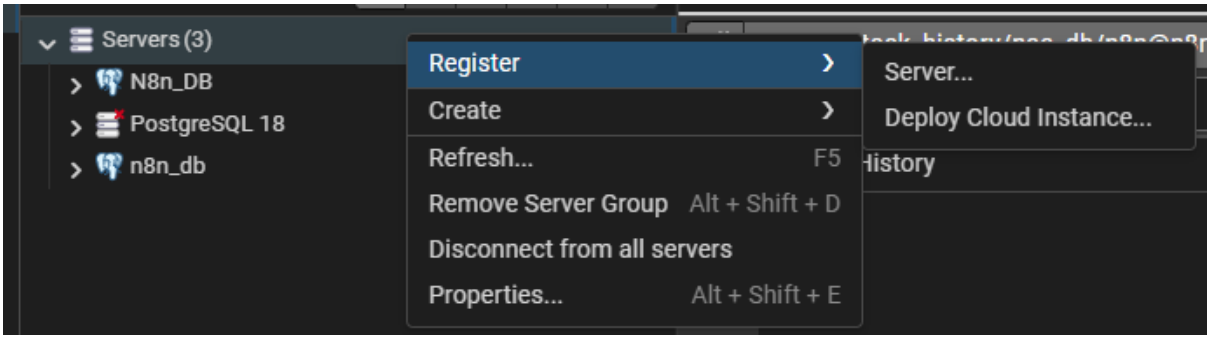
รูปที่ 3-77 แก้ไขไฟล์ docker-compose.yml

3.7.35 หลังจากแก้ไขไฟล์ docker-compose.yml เรียบร้อยแล้ว ให้ใช้คำสั่ง docker-compose down เพื่อหยุด container เดิม และใช้คำสั่ง docker-compose up -d เพื่อเริ่มต้น container ใหม่ เพื่อให้ค่าที่แก้ไขมีผล

```
n8n@n8n:~/n8n-nac$ sudo docker-compose down
[+] down 3/3
 ✔ Container n8n-nac-n8n-1      Removed
 ✔ Container n8n-nac-postgres-1 Removed
 ✔ Network n8n-nac_default      Removed
n8n@n8n:~/n8n-nac$ sudo docker-compose up -d
[+] up 3/3
 ✔ Network n8n-nac_default      Created
 ✔ Container n8n-nac-postgres-1 Created
 ✔ Container n8n-nac-n8n-1      Created
```

รูปที่ 3-78 ใช้คำสั่ง docker-compose down และ docker-compose up -d

3.7.36 สร้างฐานข้อมูลที่เอาไว้สำหรับเก็บ log ใน postgres

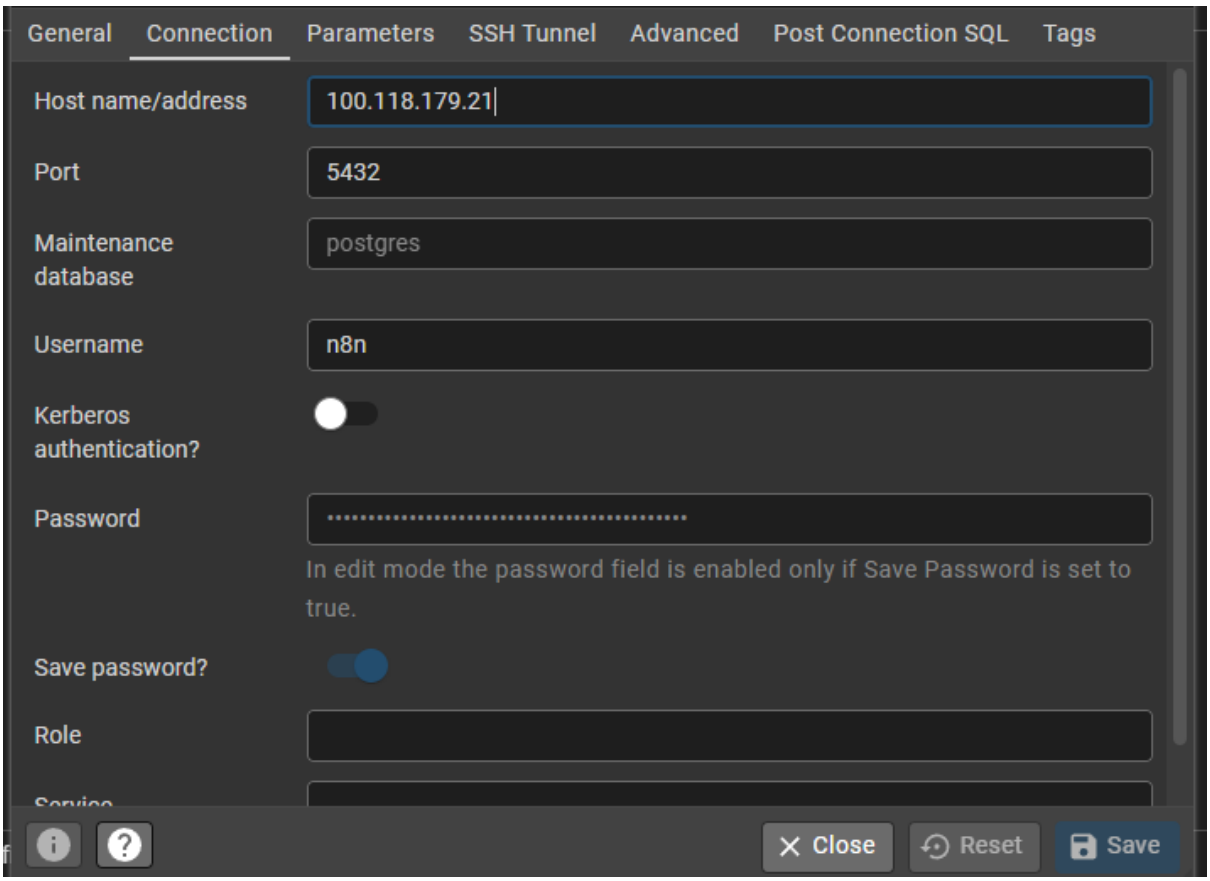


รูปที่ 3-79 สร้างฐานข้อมูลที่เอาไว้สำหรับเก็บ log

3.7.37 ตั้งชื่อฐานข้อมูล

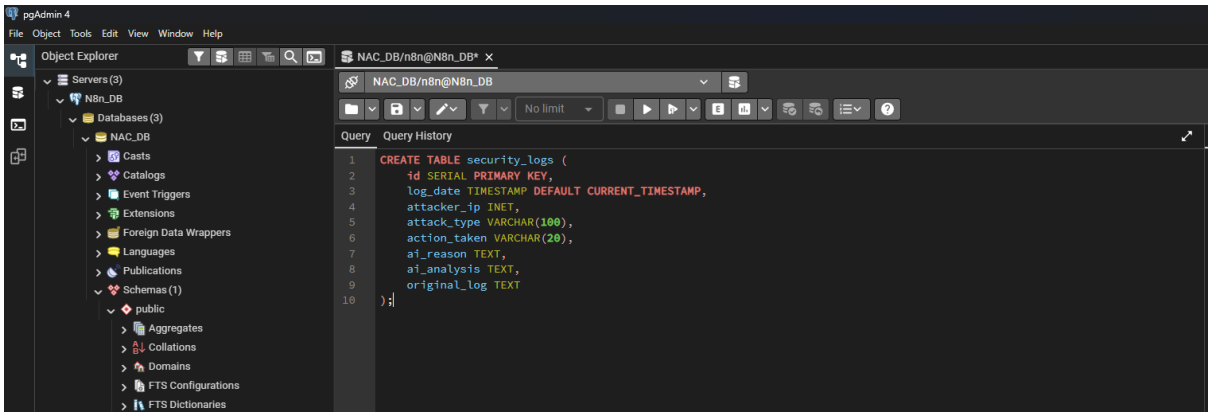
รูปที่ 3-80 ตั้งชื่อฐานข้อมูล

3.7.38 ไปที่หน้า connection แล้วกำหนด address port และ password แล้วกด save



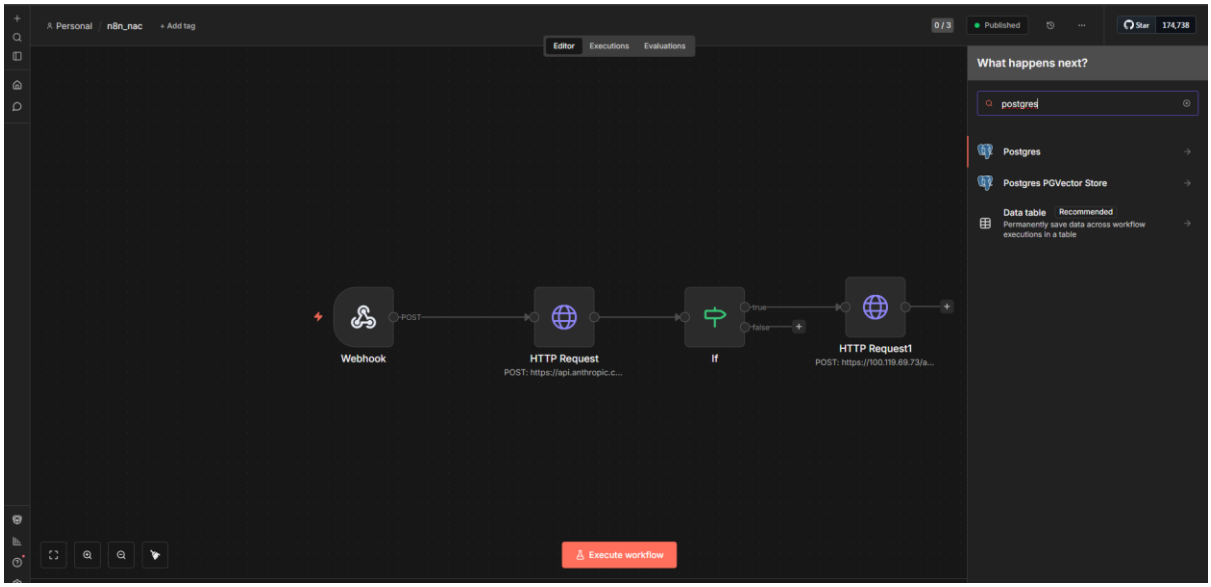
รูปที่ 3-81 ไปที่หน้า connection แล้วกำหนด address port และ password

3.7.39 สร้างตารางแล้วรันคำสั่ง sql ตามรูปที่ 3-80



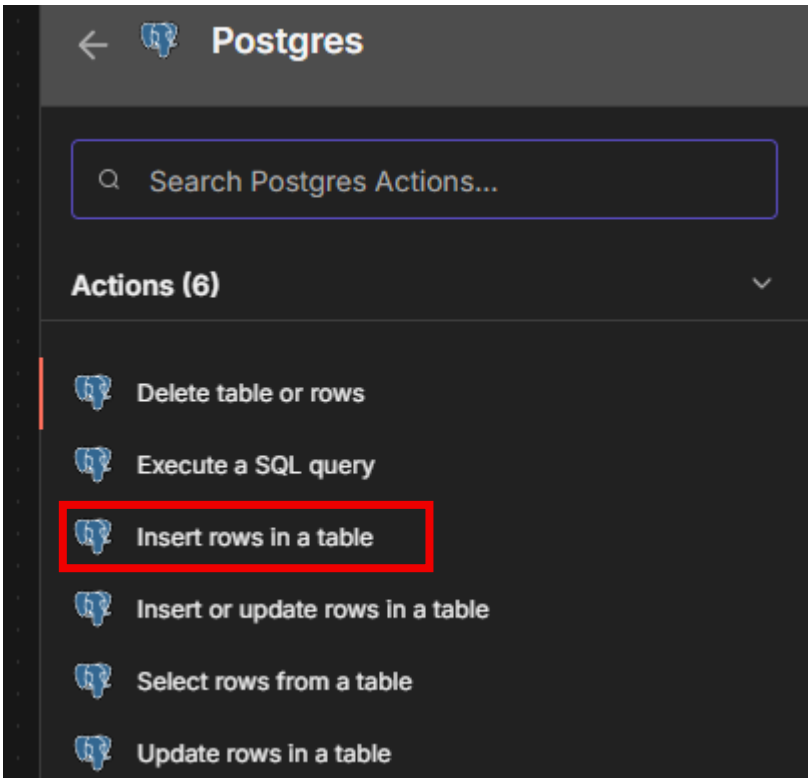
รูปที่ 3-82 สร้างตารางแล้วรันคำสั่ง sql

3.7.40 สร้าง node ใหม่ใน n8n โดยค้นหาด้วยคำว่า postgres



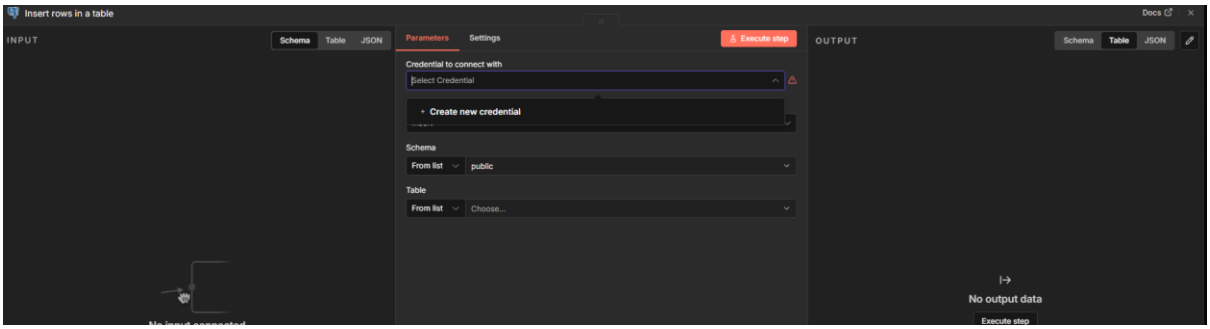
รูปที่ 3-83 สร้าง node ใหม่ใน n8n โดยค้นหาด้วยคำว่า postgres

3.7.41 เลือก insert rows in a table



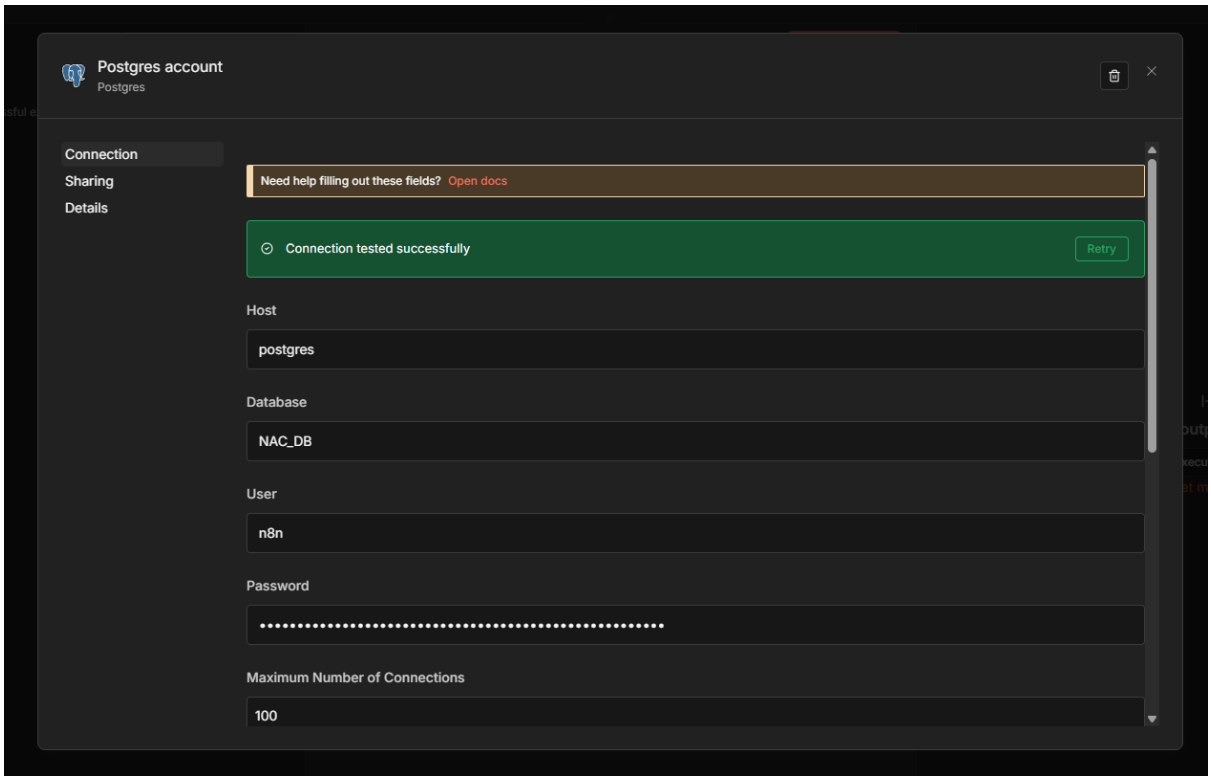
รูปที่ 3-84 เลือก insert rows in a table

3.7.42 กดปุ่ม Create new credential ใน node postgres



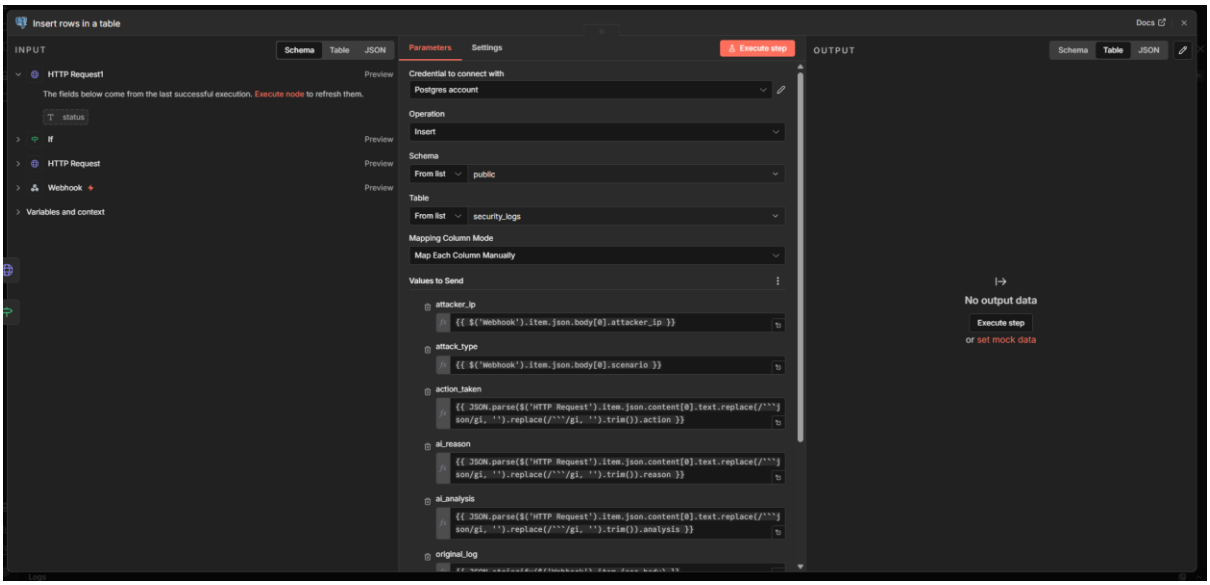
รูปที่ 3-85 กดปุ่ม Create new credential ใน node postgres

3.7.43 ตั้งค่า connection ใน node postgres



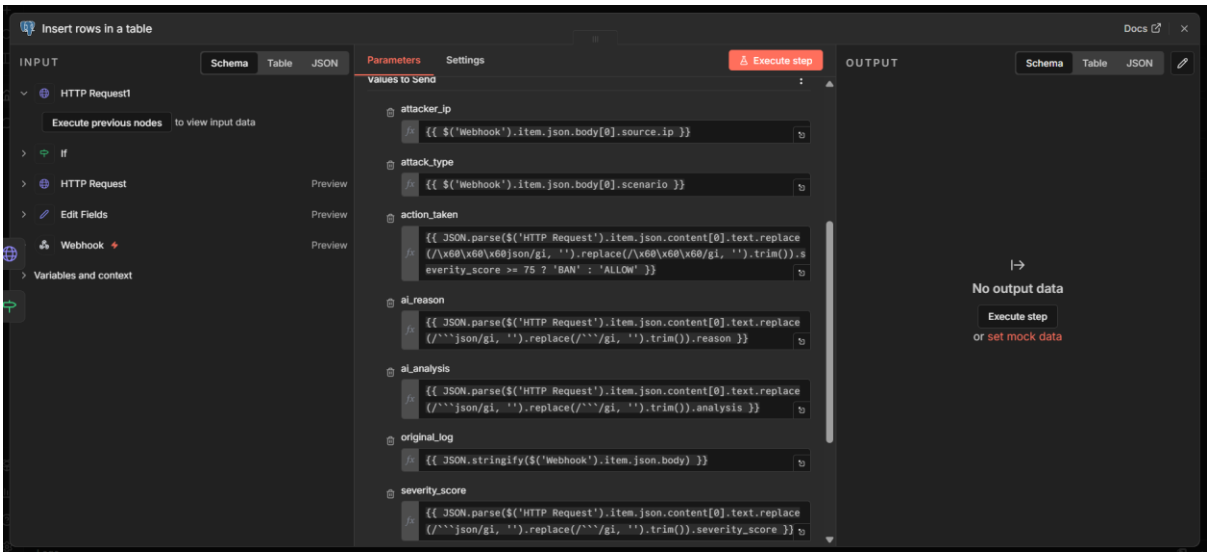
รูปที่ 3-86 ตั้งค่า connection ใน node postgres

3.7.44 เขียน code ใน parameters ของ node postgres ตามรูปด้านล่าง



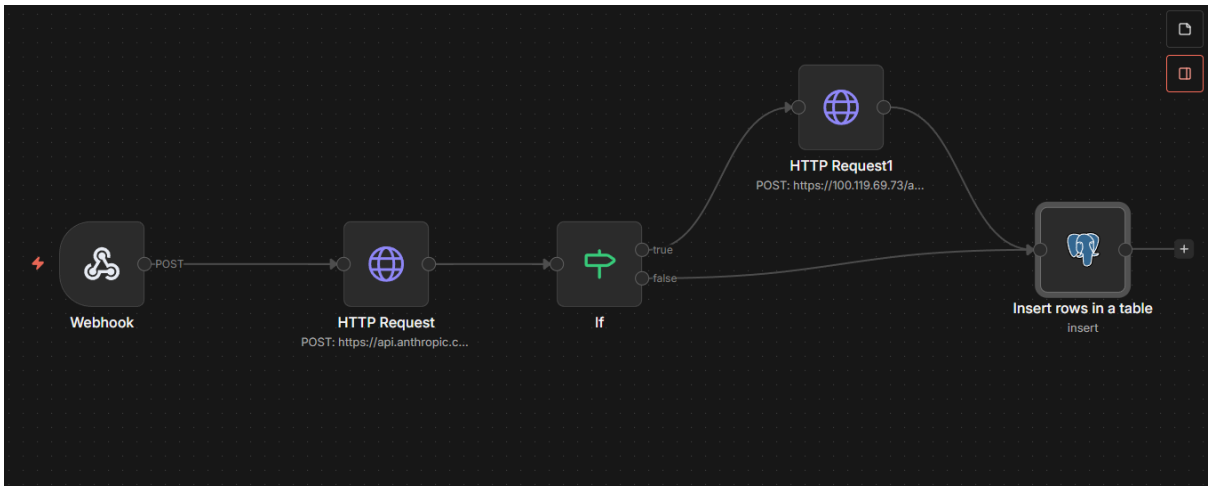
รูปที่ 3-87 เขียน code ใน parameters ของ node postgres

3.7.45 เขียน code ใน parameters ของ node postgres ตามรูปด้านล่าง (ต่อ)



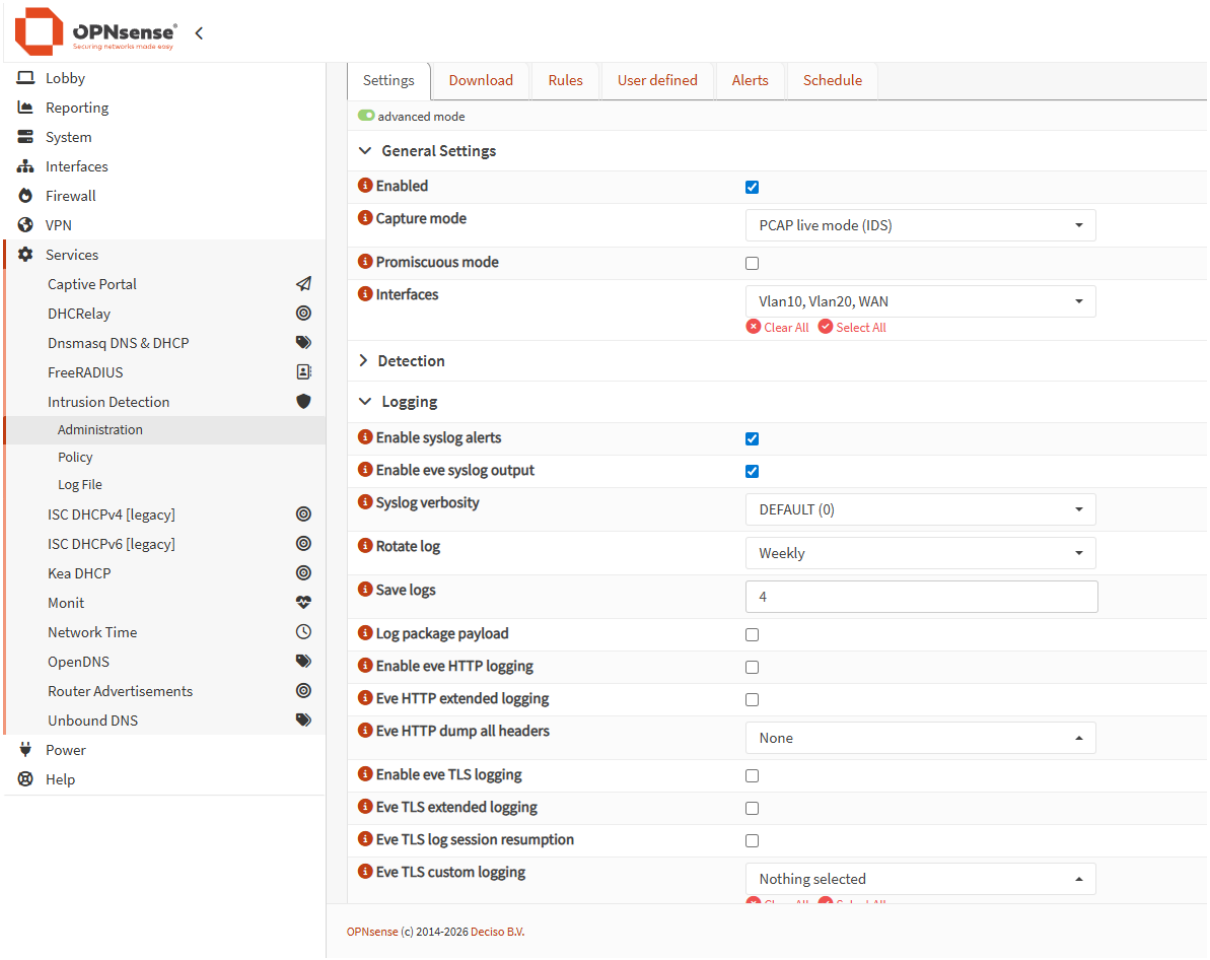
รูปที่ 3-88 เขียน code ใน parameters ของ node postgres (ต่อ)

3.7.46 เชื่อมต่อ node postgres กับ node if และ http request



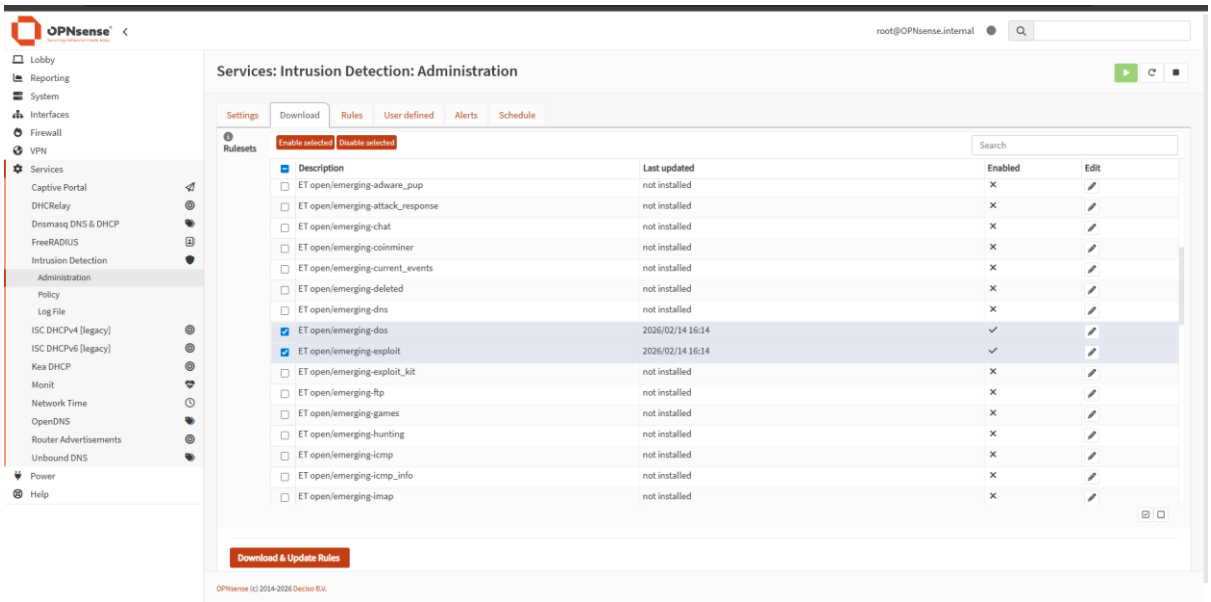
รูปที่ 3-89 เชื่อมต่อ node postgres กับ node if และ http request

3.7.47 ไปที่ Service > Intrusion Detection > Administration ใน OPNsense แล้วติ๊กตามภาพด้านล่าง เพื่อเปิดแจ้งเตือน log



รูปที่ 3-90 เปิดแจ้งเตือน log ใน OPNsense

3.7.48 ไปที่แท็บ Download แล้วเลือกดาวน์โหลด Ruleset ตามภาพด้านล่างแล้วกด Download & Update Rule



รูปที่ 3-91 ดาวน์โหลด Ruleset