กราบเรียนกรรมการและอาจารย์ทุกท่านครับ กระผมนายศุภวิชญ์ แซ่ลิ่ม วันนี้ผมจะนำเสนอโครงงานเรื่อง 'เว็บแอปพลิเคชันกำหนดค่าอุปกรณ์และตรวจสอบช่องโหว่ของการกำหนดค่าอุปกรณ์เครือข่าย CISCO' โดยมีอาจารย์ ดร.วิรชัย คงศิริวัฒนา เป็นอาจารย์ที่ปรึกษาในโครงงานนี้ครับ

โดยโครงงานนี้มุ่งเน้นการพัฒนาเว็บแอปพลิเคชันที่จะช่วยให้ผู้ดูแลระบบเครือข่ายสามารถจัดการและตรวจสอบการตั้งค่าอุปกรณ์เครือข่ายของ Cisco ซึ่งจะช่วยเพิ่มประสิทธิภาพและความปลอดภัยมากขึ้น

ในส่วนของความเป็นมาและความสำคัญ ผมขอนำเสนอวิธีการตั้งค่าอุปกรณ์เครือข่ายแบบดั้งเดิมก่อนครับ โดยปกติการตั้งค่าอุปกรณ์เครือข่ายนั้นเราสามารถทำได้ 3 วิธีหลักๆ คือ

1. ผ่านพอร์ต Console ซึ่งเป็นการเชื่อมต่อโดยตรงด้วยสาย Console เหมาะสำหรับการตั้งค่าเริ่มต้นและการเข้าถึงในกรณีฉุกเฉิน
2. ผ่าน Secure Shell

และ 3. ผ่าน Telnet ซึ่งเป็นวิธีดั้งเดิมที่ไม่ปลอดภัยเท่า SSH

โดยวิธีการเหล่านี้จะใช้ได้กับอุปกรณ์เครือข่ายทั้ง Router, Multilayer Switch และ Management Switch ซึ่งเป็นอุปกรณ์พื้นฐานที่สำคัญในการสร้างระบบเครือข่าย

นอกจากวิธีการตั้งค่าที่ได้กล่าวไปแล้ว สิ่งที่เป็นความท้าทายสำหรับผู้ดูแลระบบคือการจดจำคำสั่งในการตั้งค่าอุปกรณ์ เนื่องจากในองค์กรหนึ่งๆ อาจมีอุปกรณ์เครือข่ายจากหลากหลายผู้ผลิต ซึ่งแต่ละยี่ห้อมีรูปแบบคำสั่งและไวยากรณ์ที่แตกต่างกัน

จากภาพตัวอย่าง แสดงให้เห็นความผิดพลาดที่เกิดขึ้นในการพยายามตั้งค่า Interface และ Routing ซึ่งเป็นสถานการณ์ที่พบได้บ่อย เมื่อผู้ดูแลระบบต้องสลับไปมาระหว่างการจัดการอุปกรณ์ต่างยี่ห้อ ทำให้เกิดความสับสนในการใช้คำสั่ง ส่งผลให้เสียเวลาในการตรวจสอบเอกสารคู่มือและแก้ไขข้อผิดพลาด

ในการตั้งค่าอุปกรณ์เครือข่ายแบบดั้งเดิมนั้น ผู้ดูแลระบบจำเป็นต้องกำหนดค่าอุปกรณ์ทีละเครื่อง ซึ่งเป็นกระบวนการที่ใช้เวลามาก จากภาพจะเห็นว่าการตั้งค่าต้องทำตามลำดับทีละอุปกรณ์

ปัญหาสำคัญของกระบวนการนี้คือ:

* ต้องตั้งค่าอุปกรณ์แต่ละตัวแยกกัน
* การเปลี่ยนแปลงค่าต่างๆ ต้องป้อนคำสั่งด้วยตนเองทีละเครื่อง
* และยิ่งมีจำนวนอุปกรณ์มากขึ้น ก็จะยิ่งใช้เวลาในการตั้งค่ามากขึ้นตามไปด้วย

นอกจากปัญหาเรื่องการตั้งค่าที่ใช้เวลานานแล้ว การตั้งค่าอุปกรณ์เครือข่ายแบบดั้งเดิมยังอาจนำไปสู่ปัญหาด้านความปลอดภัยที่สำคัญ 3 ประการ คือ

ประการแรก การตั้งค่าที่ไม่เหมาะสมของบริการต่างๆ เช่น การเปิดให้บริการที่ไม่จำเป็น หรือการตั้งค่าที่ไม่ถูกต้อง อาจนำไปสู่ช่องโหว่ด้านความปลอดภัย

ประการที่สอง คือภัยคุกคามด้านความปลอดภัย เมื่อผู้ไม่ประสงค์ดีอาจใช้ประโยชน์จากช่องโหว่เหล่านี้ในการบุกรุกระบบ

ซึ่งนำมาเกิดประการสุดท้าย คือความเสี่ยงในการเข้าถึงข้อมูลสำคัญขององค์กรโดยไม่ได้รับอนุญาต รวมถึงการเข้าถึงอุปกรณ์เครือข่ายหรือระบบเครือข่ายโดยไม่ได้รับสิทธิ์

จากปัญหาที่กล่าวมาข้างต้น โครงงานนี้จึงมีวัตถุประสงค์หลัก 3 ประการด้วยกัน

ประการแรก การพัฒนาเว็บแอปพลิเคชันสำหรับการตั้งค่าอุปกรณ์เครือข่าย เพื่อให้ผู้ดูแลระบบสามารถจัดการอุปกรณ์ผ่านส่วนต่อประสานกราฟิกที่ใช้งานง่าย โดยไม่จำเป็นต้องใช้คำสั่งผ่าน CLI และช่วยลดข้อผิดพลาดในการตั้งค่า

ประการที่สอง การพัฒนาระบบที่สามารถจัดการอุปกรณ์หลายตัวพร้อมกัน ช่วยประหยัดเวลาด้วยการตั้งค่าแบบกลุ่ม และมีการตรวจสอบสถานะแบบเรียลไทม์

ประการสุดท้าย การเพิ่มความปลอดภัยให้กับระบบเครือข่าย ด้วยการตรวจสอบความถูกต้องของการตั้งค่า การตรวจสอบการกำหนดค่า และระบบแนะนำการตั้งค่าที่เหมาะสมโดยอัตโนมัติ

ทั้งหมดนี้จะช่วยให้การจัดการอุปกรณ์เครือข่ายเป็นไปอย่างอัตโนมัติ ประหยัดเวลา และมีความปลอดภัยมากขึ้น

ส่วนแรกของขอบเขตโครงงาน จะเป็นการพัฒนาระบบหลักสำหรับการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายใหม่ผ่านการเชื่อมต่อ Serial Port โดยมีฟังก์ชันการทำงานพื้นฐานดังนี้

ระบบจะประกอบด้วยส่วนต่อประสานเว็บ กระบวนการตั้งค่าอัตโนมัติ และการผสานรวมกับระบบเครือข่าย ซึ่งผู้ใช้สามารถกำหนดค่าต่างๆ ได้ดังนี้:

* กำหนดและเชื่อมต่อผ่านพอร์ต Console
* ตั้งค่าชื่ออุปกรณ์ (Hostname)
* กำหนดรหัสผ่านสำหรับสิทธิ์พิเศษ (Privilege Password)
* ตั้งค่าชื่อผู้ใช้และรหัสผ่านสำหรับ SSH
* กำหนดชื่อโดเมน
* และตั้งค่า Interface โดยสามารถเลือกได้ระหว่าง Static IP หรือ DHCP

ส่วนที่สองของขอบเขตโครงงาน เป็นการพัฒนาระบบที่รองรับการตั้งค่าอุปกรณ์เครือข่ายหลายตัวพร้อมกัน โดยใช้เทคนิค Threading ซึ่งช่วยให้สามารถทำงานแบบขนาน

ตัวอย่างที่เห็นในภาพ เป็นการตั้งค่า VLAN และ SNMP ให้กับอุปกรณ์ทั้ง 3 ตัวพร้อมกัน ซึ่งมีประโยชน์หลักๆ ดังนี้:

1. ด้านประสิทธิภาพเวลา: ช่วยลดเวลาในการตั้งค่าโดยรวม เนื่องจากสามารถกำหนดค่าหลายอุปกรณ์ได้พร้อมกัน
2. ด้านความสอดคล้อง: ทำให้การตั้งค่าเป็นแบบเดียวกันทั้งหมดในทุกอุปกรณ์
3. ด้านความยืดหยุ่น: สามารถจัดการอุปกรณ์จำนวนมากได้อย่างมีประสิทธิภาพ

ส่วนที่สามของขอบเขตโครงงาน เป็นส่วนของการจัดการข้อมูลอุปกรณ์เครือข่าย โดยผู้ใช้สามารถเพิ่มรายละเอียดของอุปกรณ์และจัดเก็บลงในฐานข้อมูลได้

ข้อมูลที่ผู้ใช้สามารถป้อนเข้าสู่ระบบประกอบด้วย:

* ชื่ออุปกรณ์ (Device Name)
* ที่อยู่ไอพี (IP Address)
* ชื่อผู้ใช้ (Username)
* รหัสผ่าน (Password)
* รหัสผ่านสำหรับโหมด Privilege

จากภาพด้านขวาแสดงให้เห็นตัวอย่างข้อมูลที่ถูกจัดเก็บในฐานข้อมูล ซึ่งผู้ดูแลระบบสามารถดูรายละเอียดของอุปกรณ์ทั้งหมดที่เพิ่มเข้าสู่ระบบได้

ในส่วนของขอบเขตที่ 4 เป็นส่วนของการจัดการอุปกรณ์ ที่ผู้ใช้สามารถดูรายละเอียดของอุปกรณ์ทั้งหมดที่เพิ่มเข้าสู่ระบบฐานข้อมูลและมีส่วนของ Actions ที่ผู้ใช้สามารถแก้ไขหรือลบข้อมูลของอุปกรณ์นั้นๆ ได้

ส่วนแรก การตั้งค่าพื้นฐานของอุปกรณ์ ประกอบด้วยการกำหนดชื่ออุปกรณ์ (Hostname) การตั้งค่ารหัสผ่านแบบเข้ารหัส (Secret Password) และการกำหนดข้อความต้อนรับ (Banner) เมื่อผู้ใช้เข้าสู่ระบบ

สอง การจัดการ Interface เครือข่าย ซึ่งรองรับทั้งโปรโตคอล IPv4 และ IPv6 โดยผู้ใช้สามารถเลือกรูปแบบการกำหนดค่า IP แบบ Manual หรือ DHCP พร้อมทั้งสามารถควบคุมสถานะการทำงาน (Up/Down) จัดการที่อยู่ IP และปรับโหมด Duplex ระหว่าง Half หรือ Full ได้ตามความต้องการ

ส่วนที่สาม การจัดการ VLAN ที่มีฟังก์ชันครบถ้วน ทั้งการสร้าง การเปลี่ยนชื่อ การควบคุมสถานะ (Enable/Disable) และการลบ VLAN นอกจากนี้ยังสามารถกำหนดโหมดการทำงานของ Interface ให้เป็นแบบ Access หรือ Trunk Mode ได้ตามความเหมาะสมของการใช้งาน

สี่การจัดการการตั้งค่าจะมีรายละเอียดต่างๆดังนี้

ส่วนแรก การจัดการการเข้าถึงผ่าน Line Configuration ประกอบด้วย:

* การตั้งค่า Console Line ที่รองรับการกำหนดรหัสผ่าน, การตั้งค่าระยะเวลาหมดเวลาการเชื่อมต่อ, การควบคุมการบันทึกข้อมูลแบบซิงโครนัส และการเลือกวิธีตรวจสอบสิทธิ์ผ่าน Local User หรือ Console Password
* การตั้งค่า VTY Line ที่มีความสามารถคล้ายคลึงกับ Console Line แต่เพิ่มเติมการเลือกโปรโตคอลที่ยอมรับได้ทั้ง SSH, TELNET, ALL หรือ NONE

ส่วนที่สอง การจัดการโปรโตคอลพื้นฐาน ประกอบด้วย:

* การตั้งค่า DHCP Server ที่ครอบคลุมการกำหนดชื่อ Pool, ขนาดเครือข่าย, IP Exclude, Default Gateway, DNS Server และชื่อโดเมน
* การตั้งค่า NTP ที่รองรับการกำหนดที่อยู่เซิร์ฟเวอร์และเขตเวลา

ส่วนที่สาม การจัดการโปรโตคอลเพิ่มเติม ได้แก่:

* การตั้งค่า SNMP ที่สามารถกำหนด Community String ทั้งแบบ read-only และ read-write รวมถึงข้อมูล Location และ Contact
* การควบคุมการทำงานของโปรโตคอล Discovery ทั้ง CDP และ LLDP

ระบบได้เพิ่มความสามารถในการตั้งค่าโปรโตคอลเครือข่ายที่สำคัญสองประเภท ได้แก่ Spanning Tree Protocols และ Aggregation Protocols โดยมีรายละเอียดดังนี้

ส่วนแรก การจัดการ Spanning Tree Protocols ประกอบด้วยความสามารถหลักดังนี้:

* รองรับการเลือกใช้งานโปรโตคอลได้สองรูปแบบ คือ Per-VLAN Spanning Tree Mode (PVST) และ Rapid Per-VLAN Spanning Tree Mode (Rapid-PVST)
* สามารถกำหนดบทบาท Root Bridge แบบ Primary หรือ Secondary ให้กับอุปกรณ์ในแต่ละ VLAN ID ที่ต้องการ
* มีระบบรักษาความปลอดภัยผ่านการควบคุม Portfast Interface และระบบ BPDU-Guard ที่ทำงานอัตโนมัติบน Interface ที่เปิดใช้งาน Portfast

ส่วนที่สอง การจัดการ Aggregation Protocols รองรับการทำงานสองโปรโตคอลหลัก:

* Port Aggregation Protocol (PAgP) ที่สามารถกำหนด Interface, หมายเลข Channel-group และเลือกโหมดการทำงานได้ระหว่าง Desirable หรือ Auto Mode
* Link Aggregation Control Protocol (LACP) ที่มีความสามารถในการกำหนด Interface, หมายเลข Channel-group และเลือกโหมดการทำงานได้ระหว่าง Active หรือ Passive Mode
* ทั้งสองโปรโตคอลรองรับการลบ Port Group ที่ไม่ต้องการใช้งานได้

ขอบเขต 6 ระบบรองรับการกำหนดค่าโปรโตคอลการหาเส้นทาง (Routing Protocols) ที่ครอบคลุมทั้งแบบ Static Route และ Dynamic Route โดยแต่ละโปรโตคอลมีความสามารถที่เป็นเอกลักษณ์และมีประโยชน์ในการจัดการเครือข่ายที่แตกต่างกัน

สำหรับ Static Route ผู้ดูแลระบบสามารถกำหนดเส้นทางแบบคงที่ได้สูงสุด 4 เครือข่ายปลายทาง พร้อมทั้งรองรับการกำหนด Default Route สำหรับจัดการเส้นทางที่ไม่มีการระบุไว้โดยเฉพาะ

ในส่วนของ Dynamic Route ระบบรองรับโปรโตคอลที่นิยมใช้งาน 3 ประเภท ได้แก่:

* OSPF ที่สามารถกำหนดค่า Process ID และ Router ID เฉพาะตัว โดยรองรับการกำหนดเครือข่ายได้ครั้งละ 3 เครือข่าย
* RIPv2 ที่มีความยืดหยุ่นในการจัดการผ่านฟังก์ชัน Auto-Summary และสามารถควบคุมการทำงานของโปรโตคอลได้
* EIGRP ที่ใช้ AS Number ในการจัดกลุ่มเครือข่าย พร้อมความสามารถในการกำหนด Router ID และรองรับการกำหนดเครือข่ายได้ครั้งละ 3 เครือข่าย

ทั้งนี้ ทุกโปรโตคอลมีคุณสมบัติพื้นฐานร่วมกันคือ สามารถลบการกำหนดค่าเส้นทางที่ไม่ต้องการใช้งานได้ ทำให้การบริหารจัดการเส้นทางในเครือข่ายมีความยืดหยุ่นและมีประสิทธิภาพ

ระบบมีความสามารถในการจัดการอุปกรณ์และการรักษาความปลอดภัยที่ครอบคลุมสามด้านหลัก ได้แก่ การดูและส่งออกการตั้งค่า การควบคุมอุปกรณ์ และการตรวจสอบความปลอดภัย

ด้านการจัดการการตั้งค่า ระบบสามารถแสดงการกำหนดค่าต่างๆ ของอุปกรณ์ตามที่ผู้ใช้ต้องการเลือกดู พร้อมทั้งมีฟังก์ชันในการส่งออกการตั้งค่าเพื่อการสำรองข้อมูลหรือนำไปใช้งานต่อ

ในส่วนของการควบคุมอุปกรณ์ ระบบรองรับการคืนค่าโรงงาน (Factory Reset) ซึ่งจะส่งผลให้ข้อมูลการตั้งค่าของอุปกรณ์นั้นถูกลบออกจากฐานข้อมูลโดยอัตโนมัติ เพื่อป้องกันความสับสนในการจัดการ

ส่วนการรักษาความปลอดภัย ระบบได้รวมความสามารถในการตรวจสอบช่องโหว่แบบอัตโนมัติที่ครอบคลุมประเด็นสำคัญห้าด้าน ได้แก่ การตรวจสอบความแข็งแรงของรหัสผ่าน การเข้ารหัสรหัสผ่าน การตรวจสอบสถานะของพอร์ตที่ไม่ได้ใช้งาน การตั้งค่าการหมดเวลาการเชื่อมต่อ และการตรวจสอบความปลอดภัยของโปรโตคอลสำหรับการเข้าถึงระยะไกล

ด้วยความสามารถเหล่านี้ ผู้ดูแลระบบจึงสามารถจัดการอุปกรณ์ พร้อมทั้งมั่นใจในความปลอดภัยของระบบเครือข่าย

ระบบสามารถรวบรวมและแสดงข้อมูลสำคัญของอุปกรณ์เครือข่ายผ่านโปรโตคอล SNMP โดยแบ่งการแสดงผลเป็นสามส่วนหลักเพื่อความสะดวกในการตรวจสอบและติดตาม

ส่วนแรกเป็นการแสดงข้อมูลพื้นฐานที่ระบุตัวตนของอุปกรณ์ ซึ่งประกอบด้วยประเภทของอุปกรณ์ (Router หรือ Switch) ชื่ออุปกรณ์ และคำอธิบายเพิ่มเติมที่ช่วยให้ผู้ดูแลระบบสามารถทราบถึงบทบาทและหน้าที่ของอุปกรณ์แต่ละตัว

ส่วนที่สองแสดงข้อมูลระบบที่จำเป็นต่อการบริหารจัดการ ได้แก่ ระยะเวลาการทำงานของอุปกรณ์ที่ช่วยในการติดตามความเสถียร ข้อมูลการติดต่อสำหรับการประสานงาน และสถานที่ติดตั้งที่ช่วยในการบำรุงรักษาและแก้ไขปัญหา

ส่วนสุดท้ายเป็นการแสดงรายละเอียดของ Interface ทั้งหมดที่มีในอุปกรณ์

การพัฒนาระบบนี้ได้นำเทคโนโลยีและเครื่องมือที่มาใช้ โดยแบ่งออกเป็นสามส่วนหลัก:

* ส่วนแรก เป็นสภาพแวดล้อมการพัฒนา (Development Environment) ที่ใช้ Visual Studio Code เป็นเครื่องมือหลักในการเขียนโค้ด ร่วมกับ Flask Web Framework ซึ่งเป็นเฟรมเวิร์กที่เหมาะสำหรับการพัฒนาเว็บแอปพลิเคชันด้วยภาษา Python
* ส่วนที่สอง เป็นเครื่องมือจัดการเครือข่าย (Network Management) ที่ประกอบด้วย Netmiko Library สำหรับการเชื่อมต่อและจัดการอุปกรณ์เครือข่าย และ EVE-NG Emulator ที่ใช้จำลองสภาพแวดล้อมเครือข่ายเสมือน นอกจากนี้ยังใช้ Net-SNMP ซึ่งเป็นไลบรารีสำหรับการเก็บข้อมูลและควบคุมอุปกรณ์ผ่านโปรโตคอล SNMP รวมถึง Serial-Python สำหรับการสื่อสารผ่านพอร์ตอนุกรม
* ส่วนที่สาม เป็นเทคโนโลยีขั้นสูง (Advanced Technologies) ที่ช่วยเพิ่มความสามารถของระบบ ได้แก่ Python Threading สำหรับการทำงานหลายงานพร้อมกัน การใช้ NLP ในการวิเคราะห์และตรวจสอบความปลอดภัย และการจัดเก็บข้อมูลด้วย MongoDB ซึ่งเป็นฐานข้อมูลที่เหมาะกับการจัดการข้อมูลที่มีโครงสร้างยืดหยุ่น

งานวิจัยนี้มุ่งเน้นการพัฒนาเว็บแอปพลิเคชันสำหรับจัดการและตรวจสอบความปลอดภัยในการตั้งค่าของอุปกรณ์เครือข่าย Cisco โดยแบ่งการศึกษาออกเป็นแปดส่วนหลัก ที่ครอบคลุมตั้งแต่การพัฒนาระบบ การจัดการเครือข่าย ไปจนถึงการวิเคราะห์ความปลอดภัย

ส่วนการพัฒนาหลัก (Core Development) ใช้ภาษา Python ในการเขียนโปรแกรม ร่วมกับ Flask Web Framework สำหรับพัฒนาส่วนติดต่อผู้ใช้ และการพัฒนา API

ส่วนการตั้งค่าเครือข่าย (Network Config) รองรับการตั้งค่าทั้ง Switch และ Router โดยใช้ Netmiko เป็นเครื่องมือหลักในการดำเนินการ

ส่วนการวิเคราะห์ความปลอดภัย (Security Analysis) ใช้เทคนิค NLP วิเคราะห์ความแข็งแรงของรหัสผ่าน และประเมินความเสี่ยงของระบบ

ส่วนการผสานระบบ (System Integration) จัดการการเชื่อมต่อ API จัดการข้อผิดพลาด และควบคุมกระบวนการย่อยต่างๆ

ส่วนการจัดการข้อมูล (Data Management) ใช้ MongoDB จัดเก็บข้อมูล วิเคราะห์การตั้งค่า และจัดการพื้นที่จัดเก็บ

ส่วนโปรโตคอล (Protocols) รองรับการสื่อสารผ่าน SNMP, Net-SNMP, การสื่อสารแบบอนุกรม และ SSH

ส่วนการประมวลผล (Processing) จัดการการทำงานแบบ Threading และจัดตารางการทำงานของระบบ

ส่วนการเข้าถึงอุปกรณ์ (Device Access) ควบคุมการสื่อสารผ่านพอร์ตอนุกรม การสื่อสารกับอุปกรณ์ และการจัดการการเข้าถึง

Context Diagram แสดงภาพรวมการทำงานของระบบ โดยแสดงความสัมพันธ์ระหว่างผู้ใช้งาน (User) และระบบเว็บแอปพลิเคชัน ซึ่งประกอบด้วยการทำงานหลักสองส่วน: ส่วนการตั้งค่าและจัดการอุปกรณ์ที่ผู้ใช้สามารถดำเนินการได้ และส่วนการแสดงผลข้อมูลและการวิเคราะห์ความเสี่ยงที่ระบบส่งกลับไปยังผู้ใช้

ระบบใช้ MongoDB ในการจัดเก็บข้อมูลอุปกรณ์เครือข่าย โดยโครงสร้างเอกสาร (Document) ประกอบด้วยสามส่วนหลัก: ส่วนข้อมูลพื้นฐาน เก็บค่า ID เฉพาะของอุปกรณ์, ชื่ออุปกรณ์ และเวลาที่สร้างข้อมูล ส่วน device\_info เก็บรายละเอียดการตั้งค่าของอุปกรณ์ เช่น ประเภทอุปกรณ์, IP address, ข้อมูลการเข้าถึง และการตั้งค่า session log

ส่วน analysis เก็บข้อมูลการวิเคราะห์และการแจ้งเตือน พร้อมเวลาที่ทำการอัพเดตล่าสุด

ในส่วนต่อไปผมจะนำเสนอผลการพัฒนาระบบและการทดสอบการทำงานของเว็บแอปพลิเคชัน โดยจะแสดงให้เห็นการทำงานของฟังก์ชันต่างๆ ที่ได้พัฒนาตามขอบเขตที่ได้กล่าวไปแล้ว ซึ่งประกอบด้วย:

1. ส่วนติดต่อผู้ใช้หลักของระบบ ที่ออกแบบให้ใช้งานได้ง่ายและเป็นมิตรกับผู้ใช้
2. ระบบการตั้งค่าอุปกรณ์เบื้องต้นผ่าน Serial Port
3. การจัดการอุปกรณ์เครือข่ายแบบกลุ่มที่รองรับการทำงานพร้อมกันหลายอุปกรณ์
4. ระบบตรวจสอบความปลอดภัยอัตโนมัติที่ช่วยวิเคราะห์และแจ้งเตือนความเสี่ยง
5. ส่วนแสดงผลข้อมูลและสถานะของอุปกรณ์ผ่านโปรโตคอล SNMP

ส่วนแรกที่จะแสดงคือหน้าจอสำหรับการตั้งค่าเริ่มต้นของอุปกรณ์ ซึ่งประกอบด้วยส่วนสำคัญ 6 ส่วน ได้แก่:

* การเชื่อมต่อผ่านพอร์ต Console
* การตั้งค่าชื่ออุปกรณ์และโดเมน
* การกำหนดรหัสผ่านและการยืนยันตัวตน
* การเลือก Interface ที่ต้องการกำหนดค่า
* การตั้งค่า IP Address แบบ Manual หรือ DHCP
* และการตั้งค่าเครือข่ายเพิ่มเติม

ต่อไปผมจะสาธิตการใช้งานระบบให้ดูครับ

ถัดมาเป็นส่วนของการเพิ่มข้อมูลอุปกรณ์เข้าสู่ระบบ โดยผู้ใช้ต้องระบุข้อมูลพื้นฐาน 3 ส่วนคือ ชื่ออุปกรณ์ ข้อมูลการเข้าถึงอุปกรณ์ และข้อมูลยืนยันตัวตนสำหรับ SSH ต่อไปผมจะสาธิตการใช้งานครับ

หน้านี้จะตรงตามขอบเขตในส่วนของการจัดการข้อมูลอุปกรณ์ในระบบ ที่ผู้ใช้สามารถดูรายละเอียด แก้ไข และลบข้อมูลได้ นอกจากนี้ผมได้เพิ่มฟังก์ชันเสริมเพื่อให้ระบบมีประสิทธิภาพมากขึ้น ได้แก่ ระบบค้นหาอุปกรณ์ การทดสอบการเชื่อมต่อด้วย Ping และการเข้าถึงอุปกรณ์ผ่าน SSH ต่อไปผมจะสาธิตการใช้งานครับ

ในส่วนของ Configuration จะมี sub-menu ย่อยหลายส่วน โดยส่วนแรกที่จะแสดงคือ Basic Settings ซึ่งผู้ใช้สามารถเลือกอุปกรณ์ที่ต้องการตั้งค่าได้ทั้งแบบเดี่ยวหรือหลายอุปกรณ์พร้อมกัน โดยมีฟังก์ชันการตั้งค่าพื้นฐาน 3 ส่วนคือ การตั้งชื่ออุปกรณ์ การกำหนดรหัสผ่าน และการตั้งค่าข้อความต้อนรับ นอกจากนี้ยังมีส่วนของการเพิ่มผู้ใช้และรหัสผ่านเพื่อเพิ่มเข้าสู่อุปกรณ์ ต่อไปผมจะสาธิตการใช้งานครับ

ถัดมาเป็นส่วนของการตั้งค่า Network Interface โดยแบ่งเป็น 3 ส่วนหลักคือ การจัดการ IPv4 ที่รองรับทั้งการกำหนดค่าแบบ Manual และ DHCP พร้อมฟังก์ชันเปิด-ปิดและลบ IP address การจัดการ IPv6 ที่มีความสามารถคล้ายกับ IPv4 และสุดท้ายคือการกำหนด Duplex Mode ที่สามารถเลือกได้ทั้งแบบ Auto-Negotiation, Full และ Half duplex ต่อไปผมจะสาธิตการใช้งานครับ

ถัดมาเป็นส่วนของการจัดการ VLAN โดยระบบรองรับการทำงานหลายส่วน ได้แก่ การสร้าง VLAN ใหม่ การเปลี่ยนชื่อ VLAN การเปิด-ปิดการทำงาน การลบ VLAN และการกำหนดพอร์ตให้เป็นแบบ Access หรือ Trunk โดยผู้ใช้สามารถจัดการอุปกรณ์หลายตัวพร้อมกันได้ ต่อไปผมจะสาธิตการใช้งานครับ

ถัดมาเป็นส่วนของ Management Settings ที่แบ่งการตั้งค่าเป็น 5 ส่วนหลักคือ:

1. การตั้งค่า VTY Line ที่รองรับการเข้าถึงผ่าน SSH และ TELNET พร้อมระบบยืนยันตัวตน
2. การตั้งค่า Console Line ที่สามารถกำหนดรหัสผ่านและเวลาหมดเซสชัน
3. การจัดการ DHCP Server ที่รองรับการกำหนดค่า Pool, Network, Default Router และ DNS Server
4. การตั้งค่า NTP เพื่อซิงโครไนซ์เวลากับเซิร์ฟเวอร์และกำหนดโซนเวลา
5. การตั้งค่า SNMP และโปรโตคอล Discovery อย่าง CDP และ LLDP

ลองไปดูโปรแกรมครับ

ถัดมาเป็นส่วนของการตั้งค่า Spanning Tree Protocol โดยระบบรองรับการทำงาน 3 ส่วนหลักคือ

1. การเลือกโหมด STP ระหว่าง PVST หรือ Rapid-PVST
2. การกำหนด Root Bridge ทั้งแบบ Primary และ Secondary สำหรับแต่ละ VLAN
3. การตั้งค่า PortFast สำหรับพอร์ตที่เชื่อมต่อกับอุปกรณ์ปลายทาง พร้อมระบบ BPDU-Guard

เมนูสุดท้าเกี่ยวกับระบบ Aggregation Protocols ที่ถูกพัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพในการรวมลิงค์เครือข่าย ระบบนี้รองรับการทำงานของทั้ง Port Aggregation Protocol และ Link Aggregation Control Protocol หรือ LACP โดยมีการออกแบบส่วนติดต่อผู้ใช้ให้สามารถจัดการได้ง่ายผ่านเว็บแอปพลิเคชัน

สำหรับ PAgP นั้น ผู้ใช้สามารถเลือกการทำงานได้ทั้งในโหมด Desirable และ Auto พร้อมกำหนดค่า Interface และ Channel Group ได้ตามต้องการ ซึ่งเหมาะสำหรับการเชื่อมต่อระหว่างอุปกรณ์ Cisco ส่วน LACP ที่เป็นมาตรฐานเปิด ก็รองรับการทำงานทั้งในโหมด Active และ Passive ทำให้สามารถใช้งานร่วมกับอุปกรณ์จากผู้ผลิตที่หลากหลายได้ ต่อไปผมจะสาธิตการใช้งานฟังก์ชันต่างๆ ให้ดูครับ

ถัดมาผมจะนำเสนอในส่วนของการจัดการเส้นทางเครือข่าย (Routing Configuration) ซึ่งเป็นส่วนสำคัญของระบบ โดยแบ่งออกเป็นโปรโตคอลย่อยๆ ได้แก่ Static Route, OSPF, RIPv2 และ EIGRP เพื่อรองรับการทำงานที่หลากหลาย

ในส่วนของ Static Route ที่เห็นในภาพนี้ ผมได้พัฒนาให้มีความยืดหยุ่นมากกว่าที่กำหนดไว้ในขอบเขตเดิม โดยจากที่ระบุว่าจะรองรับการกำหนดเส้นทางสูงสุด 4 เส้นทางต่อการตั้งค่าหนึ่งครั้ง แต่จากการทดสอบจริงพบว่าระบบสามารถรองรับได้มากถึง 10 เส้นทาง และยังทำงานได้อย่างมีประสิทธิภาพ

ระบบนี้รองรับการทำงานที่หลากหลาย ทั้งการกำหนดเส้นทางไปยังเครือข่ายปลายทาง การตั้งค่า Default Route และการลบเส้นทางที่ไม่ต้องการ โดยผู้ใช้สามารถระบุ Destination Network และ Exit Interface หรือ Next Hop IP ได้ตามต้องการ นอกจากนี้ยังรองรับการจัดการอุปกรณ์หลายตัวพร้อมกันเพื่อความสะดวกในการใช้งาน ต่อไปผมจะสาธิตการทำงานให้ดูครับ

ถัดมาเป็นส่วนของโปรโตคอล OSPF ซึ่งเป็นอีกหนึ่งฟีเจอร์สำคัญในการจัดการเส้นทางเครือข่าย โดยระบบสามารถรองรับการกำหนดค่าที่ยืดหยุ่น เช่นเดียวกับ Static Route ที่ได้กล่าวไปแล้ว ระบบถูกออกแบบให้จัดการ OSPF Networks ได้มากกว่าที่กำหนดในขอบเขตเดิม

ฟังก์ชันหลักประกอบด้วยการกำหนด Process ID, Router ID และการจัดการเครือข่ายใน Area ต่างๆ โดยผู้ใช้สามารถเพิ่มและลบเครือข่ายได้ตามต้องการ นอกจากนี้ยังมีฟีเจอร์เสริมอย่างการลบ Process ID และเครื่องมือจัดการเส้นทางที่ช่วยให้การบริหารระบบทำได้สะดวกยิ่งขึ้น ต่อไปผมจะสาธิตการทำงานให้ดูครับ

ถัดไปเป็นส่วนของ RIPv2 ที่ออกแบบมาให้ใช้งานง่าย โดยมีส่วนประกอบหลักคือการเพิ่มและลบเครือข่าย และการควบคุมการทำงานของโปรโตคอล ระบบรองรับการจัดการอุปกรณ์หลายตัวพร้อมกัน มีฟังก์ชัน Auto-Summary สรุปเส้นทางเครือข่ายโดยอัตโนมัติ และปุ่มเปิด-ปิดการทำงานของ RIP

ถัดมาในส่วนของ EIGRP ผมได้พัฒนาให้มีการตั้งค่าระบบที่ครอบคลุม ประกอบด้วยการกำหนดค่า AS Number และ Router ID พร้อมระบบจัดการเครือข่ายที่สามารถเพิ่มและลบเครือข่ายได้ตามต้องการ รวมถึงฟังก์ชันลบ AS Number เมื่อต้องการยกเลิกการใช้งาน EIGRP ทั้งหมด ต่อไปผมจะสาธิตการทำงานให้ดูครับ

ถัดมาเป็นส่วนของการคืนค่าโรงงานและจัดการการตั้งค่าอุปกรณ์ ซึ่งมีฟังก์ชันการทำงานหลัก 3 ส่วน:

1. Erase Configuration - สามารถลบการตั้งค่าทั้งหมดของอุปกรณ์และคืนค่ากลับสู่ค่าเริ่มต้น โดยระบบจะทำการ reload อุปกรณ์โดยอัตโนมัติหลังจากลบการตั้งค่าเสร็จสิ้น
2. Reload Device - สามารถรีสตาร์ทอุปกรณ์เพื่อให้การตั้งค่าใหม่มีผล โดยไม่ต้องลบการตั้งค่าเดิม
3. Save Configuration - บันทึกการตั้งค่าปัจจุบันลงใน running-config เพื่อให้การตั้งค่ายังคงอยู่แม้อุปกรณ์จะรีสตาร์ท

หน้าจอแสดงรายการอุปกรณ์ทั้งหมดในระบบ พร้อมชื่อและ IP Address ผู้ใช้สามารถค้นหาอุปกรณ์ได้ผ่านช่อง Search และจัดเรียงข้อมูลตามคอลัมน์ที่ต้องการ ระบบมีการแจ้งเตือนเพื่อยืนยันก่อนดำเนินการทุกครั้ง เพื่อป้องกันการทำงานผิดพลาด

ต่อไปผมจะสาธิตการใช้งานฟังก์ชันต่างๆ ให้ดูครับ

ถัดมาเป็นส่วนของการตรวจสอบการตั้งค่าอุปกรณ์ ที่แบ่งออกเป็น 4 หมวดหมู่:

1. Device Information - แสดงข้อมูลพื้นฐานของอุปกรณ์ เช่น เวอร์ชัน, เวลาระบบ, startup-config และข้อมูลฮาร์ดแวร์
2. Network Configuration - แสดงการตั้งค่าเครือข่าย ทั้ง running-config, interface, VLAN, MAC address และ DHCP
3. Routing and Protocols - แสดงข้อมูลเกี่ยวกับ routing table, โปรโตคอลต่างๆ เช่น OSPF, EIGRP, และ spanning-tree
4. Performance & Diagnostics - แสดงข้อมูลประสิทธิภาพและการวินิจฉัย เช่น log, การใช้งาน CPU และสถิติต่างๆ

ระบบมีฟังก์ชันการส่งออกข้อมูลเป็นไฟล์ text และแสดงเวลาที่ดึงข้อมูลล่าสุด

ถัดมาเป็นส่วนของการตรวจสอบความปลอดภัยของระบบ โดยมีการตรวจเช็คพื้นฐานสำคัญ 4 ด้าน:

1. Password Strength Analysis - ตรวจสอบความแข็งแรงของรหัสผ่านและการเข้ารหัส
2. Configuration Verification - ตรวจสอบการตั้งค่าที่อาจก่อให้เกิดความเสี่ยง เช่น พอร์ตที่ไม่ได้ใช้งาน timeout settings และโปรโตคอลที่ไม่ปลอดภัย
3. Protocol Security Review - ตรวจสอบการใช้งานโปรโตคอลและบริการต่างๆ ว่ามีความปลอดภัย
4. Real-time Monitoring - แจ้งเตือนปัญหาแบบ real-time พร้อมตัวเลือกการแก้ไขอัตโนมัติ

ระบบสามารถส่งออกรายงานเป็นไฟล์ PDF เพื่อใช้ในการวิเคราะห์ได้อยู่ ขอแสดงการทำงานของระบบ

ถัดมาผมจะอธิบายกระบวนการตรวจสอบความปลอดภัยที่ทำงานอยู่เบื้องหลัง โดยระบบประกอบด้วย 4 ส่วนหลัก:

1. Scheduler - ตัวจัดการเวลาที่คอยกระตุ้นให้เกิดการตรวจสอบทุกๆ 10 วินาที
2. SecurityChecker - กลไกวิเคราะห์ความปลอดภัย ทำหน้าที่เชื่อมต่อกับอุปกรณ์และตรวจสอบการตั้งค่าต่างๆ
3. Devices – อุปกรณ์เครือข่ายทั้งหมดในระบบจะได้รับการตรวจสอบ
4. MongoDB - เก็บผลการวิเคราะห์และประวัติการตรวจสอบฐานข้อมูลที่ลงในฐานข้อมูล

ระบบจะทำงานต่อเนื่องโดยอัตโนมัติ เริ่มจาก Scheduler สั่งให้เริ่มกระบวนการ จากนั้น SecurityChecker จะดึงรายการอุปกรณ์จาก MongoDB มาวิเคราะห์ทีละตัว และบันทึกผลกลับไปยังฐานข้อมูล ก่อนจะรอจนถึงรอบการตรวจสอบครั้งถัดไป ทำให้มั่นใจได้ว่าระบบมีการเฝ้าระวังความปลอดภัยอย่างต่อเนื่อง

เมื่อ user กดปุ่ม Fix ของอุปกรณ์นั้นๆ ระบบจะดำเนินการดังนี้

1. Input Validation - ตรวจสอบข้อมูลอุปกรณ์ เช่น IP Address, ข้อมูลเข้าถึง และการตั้งค่า timeout
2. Device Search - ค้นหาอุปกรณ์ในฐานข้อมูล วิเคราะห์คำเตือน และตรวจสอบสถานะพอร์ต
3. Device Connection - เชื่อมต่อผ่าน SSH ตรวจสอบโหมดการทำงาน และดึงการตั้งค่า
4. Security Setup - ปรับแต่งความปลอดภัยอัตโนมัติ ตั้งแต่การกำหนด timeout, เปิดใช้ SSH-only, จัดการบริการ และสั่งปิดพอร์ตทีไม่ได้ใช้งาน

หน้า Device Details (SNMP) ใช้สำหรับตรวจสอบข้อมูลอุปกรณ์เครือข่ายทั้งหมดในระบบ โดยแสดงประเภทของอุปกรณ์ว่าเป็น Router หรือ Switch พร้อมชื่อและคำอธิบายของอุปกรณ์ ระยะเวลาการทำงาน ข้อมูลการติดต่อผู้ดูแล สถานที่ติดตั้ง และรายชื่อ Interfaces ทั้งหมดที่มีในอุปกรณ์นั้นๆ

และส่วนสุดท้าย เป็นการสรุปผลการวิจัยและข้อเสนอแนะ สำหรับโครงงานนี้

“แอปพลิเคชันเว็บได้ทำให้การจัดการอุปกรณ์เครือข่ายของ CISCO โดยทำให้การตั้งค่าที่ซับซ้อนง่ายขึ้น ลดข้อผิดพลาดในการกำหนดค่า และเสริมความปลอดภัยผ่านระบบตรวจจับช่องโหว่อัตโนมัติ”

"ระบบสามารถเพิ่มประสิทธิภาพได้อย่างดีด้วยความสามารถในการกำหนดค่าหลายอุปกรณ์พร้อมกัน ฟีเจอร์ตรวจสอบแบบเรียลไทม์ และการตรวจสอบช่องโหว่ด้านความปลอดภัยโดยอัตโนมัติ ซึ่งช่วยลดความจำเป็นในการใช้ Command-Line Interface แบบเดิม"

"ระหว่างการใช้พัฒนาผมเจอข้อจำกัดในการ reboot อุปกรณ์ระยะไกล โดยในเวอร์ชัน 15.2 (SW) ไม่สามารถ reboot system ด้วย ssh ได้ และการเริ่มต้นอุปกรณ์ใหม่ ระยะเวลาการบูตที่แตกต่างกันระหว่างอุปกรณ์ใช้เวลาไม่เท่ากัน ทำให้เพิ่มเวลาในการพัฒนา และปัญหาความเข้ากันได้ของ SNMP OID ระหว่างสภาพแวดล้อมจริงและเสมือนไม่เหมืนอกัน”

* 1. **การแก้ปัญหา**
     1. การรองรับคำสั่งในต่างเวอร์ชัน

เพื่อแก้ไขปัญหาความแตกต่างของคำสั่งในอุปกรณ์ Cisco ที่ไม่รองรับการรีสตาร์ทจากระยะไกลผ่าน Telnet การตั้งค่าการรีบูตอุปกรณ์จากระยะไกลจึงต้องใช้คำสั่ง config-register 0x2102 ที่อุปกรณ์ ซึ่งเป็นการกำหนดค่าให้สามารถรีบูตอุปกรณ์ได้อย่างถูกต้องเมื่อใช้คำสั่ง reload ผ่านการเชื่อมต่อระยะไกล

* + 1. การจัดการความแตกต่างในกระบวนการบูตของอุปกรณ์

ในการแก้ไขปัญหาประสิทธิภาพของการบูตที่แตกต่างกันระหว่างอุปกรณ์ การพัฒนาโค้ดได้เพิ่มการใช้เวลาหน่วง (sleep) ระหว่างการส่งคำสั่งผ่าน Serial Script โดยมีการปรับค่า ตามความเร็วในการบูตของอุปกรณ์แต่ละตัว การหน่วงเวลาช่วยให้ระบบสามารถรอให้คำสั่งได้รับการประมวลผลและดำเนินการได้ตามลำดับ

* + 1. การจัดการความแตกต่างของ SNMP OID

ในกรณีที่พบความแตกต่างของ SNMP OID ระหว่างอุปกรณ์จริงและอุปกรณ์ใน EVE-NG ระบบได้พัฒนาฟังก์ชันตรวจสอบ OID ของอุปกรณ์โดยอัตโนมัติ เพื่อให้สามารถดึงข้อมูล SNMP ได้อย่างถูกต้องจากอุปกรณ์ในทุกสภาพแวดล้อม โดยการเชื่อมต่อกับ SNMP และตรวจสอบค่าผลลัพธ์ในรูปแบบที่แตกต่างกันให้เหมาะสมกับอุปกรณ์ที่ใช้

* + 1. การเชื่อมต่อหลายอุปกรณ์พร้อมกัน

เพื่อจัดการกับการเชื่อมต่อหลายอุปกรณ์พร้อมกัน ระบบได้พัฒนาโค้ดเพื่อจัดลำดับคำสั่งและประมวลผลข้อมูลที่ได้รับกลับมาอย่างมีประสิทธิภาพ โดยการใช้เธรด (Thread) เพื่อให้ระบบสามารถจัดการกับคำสั่งและข้อมูลจากอุปกรณ์หลายตัวในเวลาเดียวกันได้ ลดการรอคอยและ เพิ่มประสิทธิภาพในการทำงาน

* + 1. การจัดการการดึงค่าการตั้งค่าจากอุปกรณ์

เพื่อแก้ไขปัญหาความล่าช้าในการดึงค่าการตั้งค่าจากอุปกรณ์ที่อาจเกิด timeout เมื่อคำสั่ง show running-config หรือคำสั่งอื่น ๆ ต้องใช้เวลานานในการตอบกลับ ระบบได้เพิ่ม การจัดการเวลารอคอย (timeout) เพื่อให้มั่นใจว่าได้รับข้อมูลครบถ้วนจากอุปกรณ์ และลดปัญหาการสูญหายของข้อมูล

* 1. **ข้อเสนอแนะ**
     1. ปรับปรุงระบบยืนยันตัวตน ควรพิจารณาเพิ่มระบบยืนยันตัวตนแบบ Multi-Factor Authentication (MFA) เพื่อลดความเสี่ยงในการเข้าถึงโดยไม่ได้รับอนุญาต
     2. ควรเพิ่มฟังก์ชันการแสดงผลในรูปแบบกราฟิกหรือแดชบอร์ด (Dashboard) เพื่อให้ผู้ใช้สามารถดูข้อมูลที่เกี่ยวข้องกับสถานะของอุปกรณ์หรือเครือข่ายได้อย่างรวดเร็วและเข้าใจง่าย เช่น การแสดงสถานะของ VLAN หรือ Interface ที่ใช้งานอยู่
     3. รองรับการเชื่อมต่อแบบ Secure API การเปลี่ยนระบบการสื่อสารระหว่างส่วนต่าง ๆ ให้รองรับ API ที่เข้ารหัส (เช่น HTTPS) จะช่วยลดช่องโหว่จากการดักจับข้อมูล
     4. ขยายความสามารถของระบบให้รองรับอุปกรณ์จากผู้ผลิตรายอื่น เพื่อเพิ่มความยืดหยุ่นและการนำไปใช้ในองค์กรต่าง ๆ