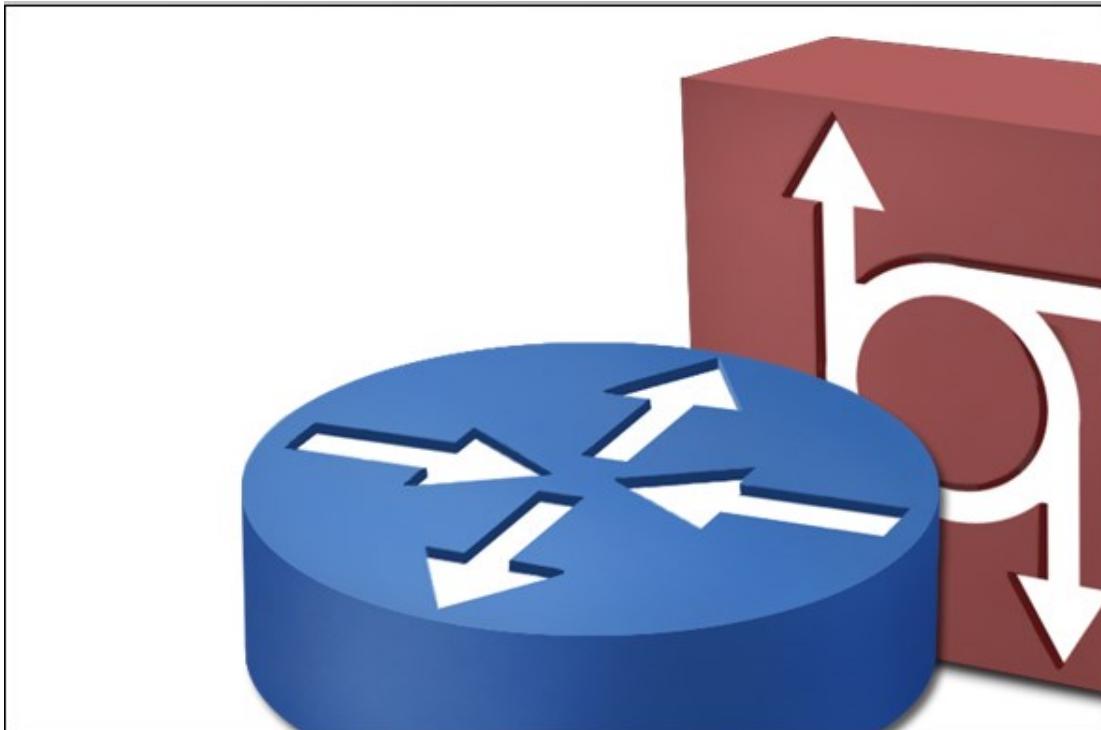


บทที่ 1

มาตรฐานการเชื่อมต่อระบบเครือข่าย (Network Connectivity Standards)



- OSI model layers
- TCP/IP
- Device communications

แนวคิด

ปัจจุบันระบบสื่อสารและเครือข่ายเชื่อมโยงเข้ากันอย่าง слับซับช้อนแบบใหม่ๆ ผลการเชื่อมโยงและเจริญก้าวหน้าของเทคโนโลยีทางด้านเครือข่าย ทำให้มีอิทธิพลต่อการดำเนินชีวิตของเราในบทนี้จะกล่าวถึง โครงสร้างพื้นฐานในการสื่อสาร และมาตรฐานการสื่อสารข้อมูล ที่ถูกกำหนดโดยองค์กรมาตรฐานสากลหรือไอเอสโอดี (ISO)

วัตถุประสงค์

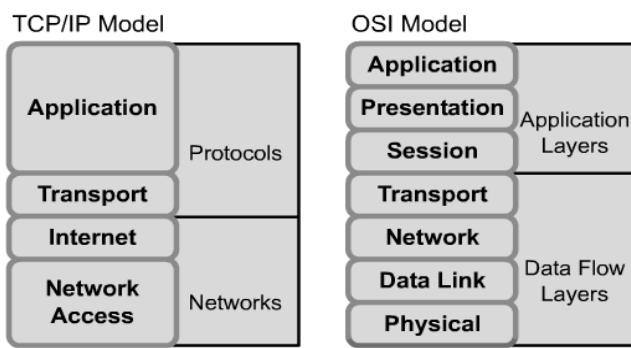
1. เพื่อให้เข้าใจความหมายและความรู้พื้นฐานที่เกี่ยวข้องกับระบบเครือข่าย
2. เพื่อให้เข้าใจลักษณะการสื่อสาร โครงสร้างพื้นฐานในการสื่อสาร และมาตรฐานการสื่อสารข้อมูล

1. มาตรฐานการเชื่อมต่อระบบเครือข่ายแบบโอลีอส์ไอ (Open System Interconnection: OSI)

มาตรฐานการเชื่อมต่อแบบโอลีอส์ไอได้แบ่งการสื่อสารออกเป็นชั้นย่อย ๆ จำนวน 7 ชั้น หรือ 7 เลเยอร์ [1, 19, 21] เหตุผลที่ทำให้ต้องมีการแบ่งออกเป็น 7 ชั้น เพื่อให้ง่ายต่อการกำหนดมาตรฐานการเชื่อมต่อและการอ้างอิงในแต่ละชั้น ซึ่งการกำหนดมาตรฐานดังกล่าวจะช่วยลดปัญหาในการสื่อสาร และการบริหารจัดการระบบเครือข่ายได้ง่ายขึ้น เช่น ผู้ผลิตอุปกรณ์เครือข่ายชนิดต่าง ๆ สามารถพัฒนาผลิตภัณฑ์ที่ตนเองมีความถนัดหรือเชี่ยวชาญได้อย่างเต็มที่ สำหรับการเชื่อมต่อกับอุปกรณ์อื่น ๆ ในลำดับชั้นเดียวกัน หรือชั้นอื่น ๆ ผู้ผลิตแต่ละรายจะต้องปฏิบัติตามข้อกำหนดการเชื่อมต่อระหว่างอุปกรณ์หรือระหว่างชั้นการสื่อสารโดยการอ้างอิงกับมาตรฐานโอลีอส์ไออย่างเคร่งครัด นอกจากนี้มาตรฐานโอลีอส์ไอยังช่วยให้ผู้พัฒนาซอฟต์แวร์ระบบสื่อสาร สามารถพัฒนาระบบโดยไม่จำเป็นต้องเริ่มต้นจากศูนย์เสมอไป หรือไม่มีความจำเป็นต้องพัฒนาซอฟต์แวร์สื่อสารให้ครบถ้วนองค์ประกอบตั้งแต่ลำดับชั้นที่ 1 ถึง 7 นั่นเอง

1.1 การเปรียบเทียบมาตรฐานการเชื่อมต่อโอลีอส์ไอ (OSI Model) และทีซีพี/ไอพี (TCP/IP Model)

รูปแบบการเชื่อมต่อระบบเครือข่ายที่ได้รับความนิยมในปัจจุบันมี 2 แบบ คือ โอลีอส์ไอ (OSI) และทีซีพี/ไอพี (TCP/IP) [1, 9, 23, 31] แสดงดังรูปที่ 1.1 สำหรับโอลีอส์ไอนั้นจะแบ่งออกเป็น 7 ชั้น โดยรายละเอียดในแต่ละชั้นจะอธิบายในหัวข้อถัดไป และแบบทีซีพี/ไอพี จะมีจำนวนชั้นน้อยกว่าแบบโอลีอส์ไอ คือ มีทั้งหมด 4 ชั้น โดยทั้งสองรูปแบบมีความแตกต่างกัน คือ แบบทีซีพี/ไอพีจะยุบรวมบางชั้นเข้าไว้ด้วยกันเพื่อความยืดหยุ่นในการใช้งาน (รูปที่ 1.2) รูปแบบการเชื่อมต่อแบบทีซีพี/ไอพี มีอิทธิพลต่อการเชื่อมต่อระบบเครือข่ายในปัจจุบันมากกว่าโอลีอส์ไอ เนื่องทีซีพี/ไอพีถูกนำมาใช้งานจริงในปัจจุบัน

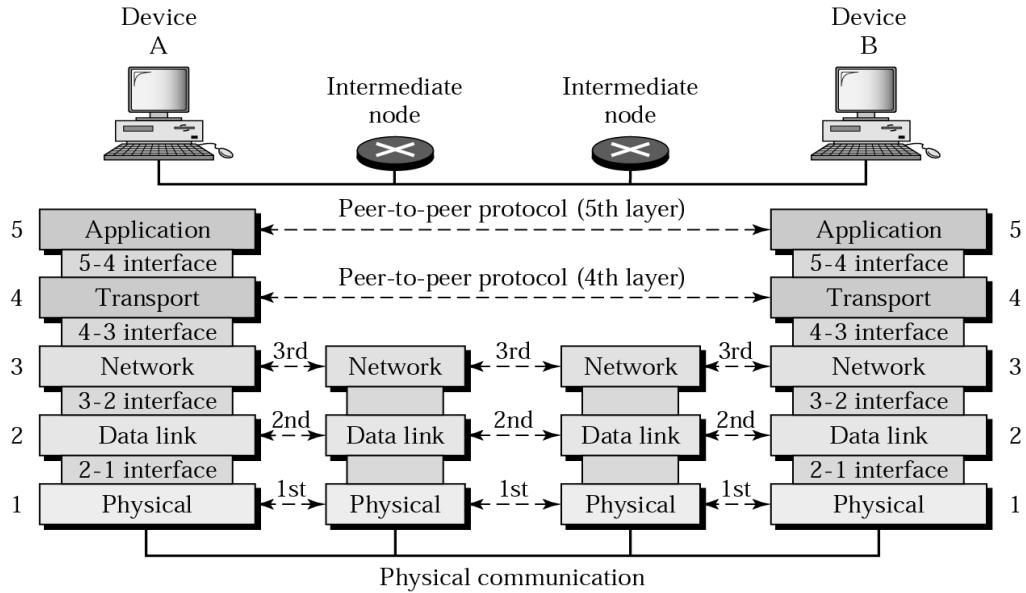


รูปที่ 1.1 โอลีอส์ไอและทีซีพี/ไอพี

โดยหลักการแล้วแต่ละชั้นจะสื่อสารกับชั้นในระดับชั้นเดียวกันที่อยู่ต่างอุปกรณ์กัน (Peer-to-peer) ดังรูปที่ 1.3 เช่น การสื่อสารในระดับชั้นที่ 5 ของอุปกรณ์ A จะสามารถสื่อสารกับอุปกรณ์ B ในระดับชั้นที่ 5 เท่านั้น แบบจำลองโวเอสไวน์ เป็นต้นแบบของสถาปัตยกรรมการสื่อสารที่เกิดขึ้นมา ก่อน จำนวนจึงเริ่มออกแบบและสร้างอุปกรณ์เครือข่ายในแต่ละชั้นตามมาในภายหลัง อุปกรณ์ที่ถูก สร้างขึ้นจะต้องสอดคล้องกับมาตรฐานชั้นใดชั้นหนึ่งของสถาปัตยกรรมนี้อย่างสมบูรณ์ อย่างไรก็ตาม ในโลกแห่งความเป็นจริงนั้นมีเทคโนโลยีจำนวนไม่น้อยที่ถูกพัฒนาขึ้นก่อนที่จะมีแบบจำลองนี้ และใน บางกรณีเทคโนโลยีที่เกิดขึ้นภายหลังจากนี้ บางอย่างก็ไม่ได้ดำเนินตามแบบจำลองนี้อย่างสมบูรณ์ เสมอไป แต่ผลที่ได้รับจากแบบจำลองโวเอสไวน์ ก็ช่วยให้เกิดแรงผลักดันเพื่อนำไปสู่การพัฒนา ระบบสื่อสารที่สามารถทำงานร่วมกันได้ของผู้ผลิตอุปกรณ์รายต่าง ๆ ได้เป็นอย่างดี จากรูปที่ 1.2 แสดงการเปรียบเทียบระหว่างมาตรฐานโวเอสไอและทีซีพี/ไอพี จากรูปแสดงให้เห็นว่าทีซีพี/ไอพี จะ ครอบคลุมชั้นการให้บริการระหว่างในชั้นกายภาพ (Physical) และดาต้าลิงค์ (Data link) เข้าด้วยกัน และเรียกชั้นที่รวมเข้ากันใหม่นี้ว่าชั้นเน็ตเวิร์ค (Network) สืบเนื่องจากตัวแบบทีซีพี/ไอพี เล็งเห็นว่า การทำงานระหว่างชั้นกายภาพและชั้นดาต้าลิงค์ มีความจำเป็นต้องทำงานร่วมกันอย่างใกล้ชิด ดังนั้น ไม่ควรแยกชั้นทั้งสองออกจากกัน สำหรับชั้นเน็ตเวิร์ค (ชั้นที่ 3) ของทีซีพี/ไอพี จะถูกเปลี่ยนชื่อใหม่ จากชั้นเน็ตเวิร์คเป็นชั้นอินเทอร์เน็ต (Internet) แทน ส่วนชั้นที่ 5, 6 และ 7 ของโวเอสไอ จะถูกมอง ว่าเป็นชั้นเดียวกันในตัวแบบของทีซีพี/ไอพี

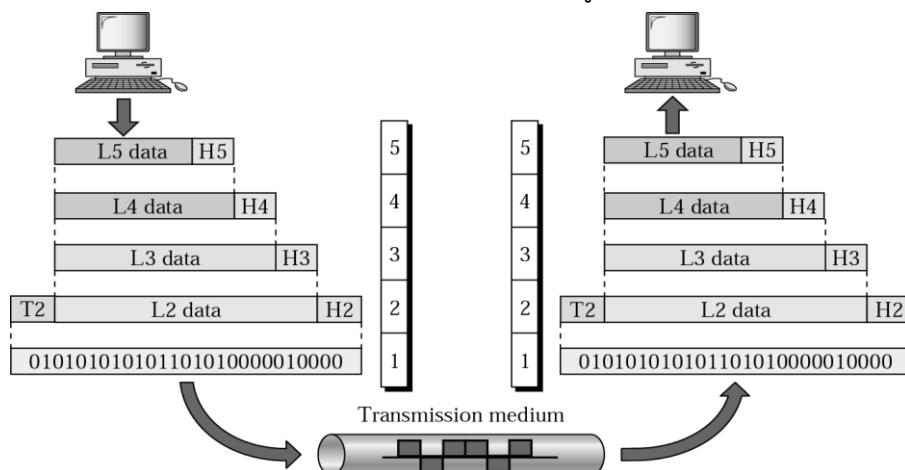
OSI		TCP/IP
Application	Application	
Presentation		
Session		
Transport	Transport	TCP, UDP
Network	Internet	IP
Data Link	Network Interface	Ethernet, Token Ring, Frame Relay, etc.
Physical		

รูปที่ 1.2 การเปรียบเทียบการจัดลำดับชั้นของโวเอสไอและทีซีพี/ไอพี



รูปที่ 1.3 ตัวแบบโอลีโอจะสื่อสารกันในชั้นที่ต้องกันเท่านั้น [1]

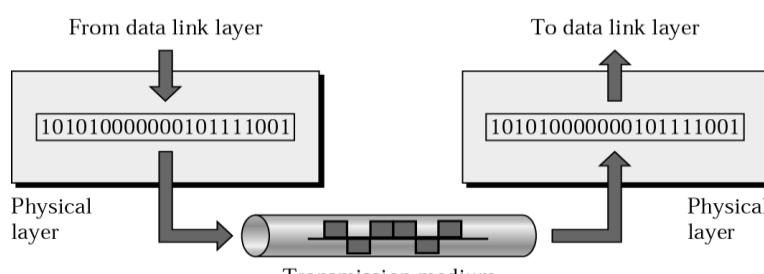
ข้อมูลในลำดับชั้นสื่อสารที่สูงกว่าจะถูกห่อหุ้ม (Data encapsulation) ด้วยข้อมูลของชั้นสื่อสารที่อยู่ในลำดับที่ต่ำกว่า จากตัวอย่างในรูปที่ 1.4 ข้อมูลของชั้นสื่อสารระดับที่ 5 (Application) ของผู้ส่ง ซึ่งประกอบไปด้วยข้อมูลส่วนหัว (Header: H5) และเนื้อข้อมูล (Payload: L5 data) จะถูกห่อหุ้มด้วยข้อมูลในลำดับชั้นสื่อสารที่ 4 คือ $H5 + L5 \text{ data} = L4 \text{ data}$ ซึ่งข้อมูล L4 data ก็จะเป็นเนื้อข้อมูล (Payload) ในลำดับชั้นการสื่อสารที่ 4 นั่นเอง วิธีการห่อหุ้มข้อมูลจะมีลักษณะเหมือนกันทุกชั้น ในทางตรงกันข้าม ผู้รับจะต้องทำการถอดข้อมูลที่ถูกห่อหุ้มในลักษณะตรงกันข้ามกับทางผู้ส่ง เช่น ข้อมูลสื่อสารในระดับชั้นที่ 1 ซึ่งเป็นสัญญาณดิจิทอลจะถูกแปลงข้อมูลให้อยู่ในรูปของเฟรมข้อมูลในระดับชั้นที่ 2 ก่อน จากนั้นผู้รับจะดำเนินการถอดข้อมูลส่วนหัวในชั้นที่ 2 ออก ก่อนส่งต่อไปเพื่อถอดข้อมูลในระดับชั้นที่ 3 ต่อไปเรื่อย ๆ จนกว่าจะเหลือเฉพาะข้อมูลที่ใช้สื่อสารจริง ๆ เท่านั้น



รูปที่ 1.4 ข้อมูลผู้ส่งจะถูกห่อหุ้มตามลำดับชั้นส่วนผู้รับจะถอดออกตามลำดับชั้น [1]

ชั้นกายภาพ (Physical Layer)

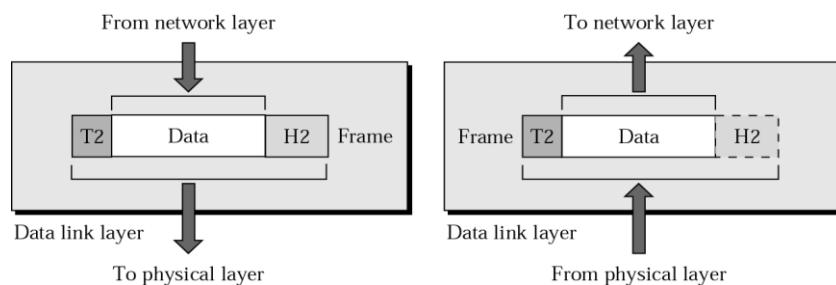
ชั้นกายภาพเป็นการสื่อสารระดับล่างสุดของตัวแบบโอลีโอ [3, 15, 25] ซึ่งมีหน้าที่ให้บริการเกี่ยวกับการส่งข้อมูลในระดับบิต ระหว่างอุปกรณ์ 2 ชนิด เช่น อุปกรณ์ทางฝั่งของผู้ใช้งาน กับอุปกรณ์ของผู้ระบบเครือข่าย เป็นต้น ข้อมูลในระดับชั้นกายภาพจะถูกประกอบเข้าเป็นชุดข้อมูล เพื่อส่งต่อให้กับลำดับชั้นที่สูงกว่า (ชั้นดาต้าลิงค์) เรียกว่าเฟรมข้อมูล นอกจากนั้นชั้นกายภาพยังมีหน้าที่รับผิดชอบดูแลในรายละเอียดการส่งข้อมูลกับฮาร์ดแวร์จริง เช่น การควบคุมการเดินเร็วๆ การส่งสัญญาณผ่านสายสัญญาณแบบต่าง ๆ เป็นต้น เมื่อกล่าวโดยสรุปแล้วชั้นกายภาพจะจัดการเกี่ยวกับสัญญาณทางไฟฟ้า สัญญาณเสียง หรือสัญญาณแสงที่จำเป็นต่อการสื่อสารโดยตรง ดังรูปที่ 1.5



รูปที่ 1.5 แสดงการทำงานของชั้นกายภาพ [1]

ชั้นดาต้าลิงค์ (Data link layer)

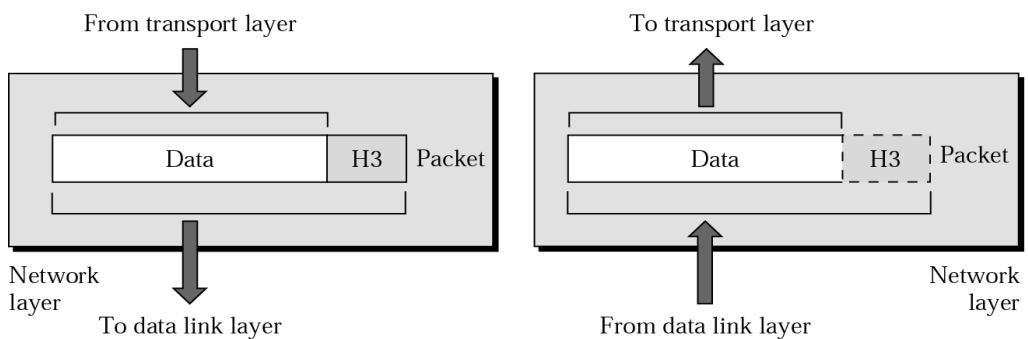
ดาต้าลิงค์มีหน้าที่ดังนี้ คือ จัดเตรียมข้อมูลเกี่ยวกับการสื่อสารให้กับชั้นเน็ตเวิร์ก (รูปที่ 1.6) การควบคุมลำดับและอัตราการรับส่งข้อมูล (Flow Control) และบริหารจัดการชื่อที่อยู่ทางกายภาพ ของอุปกรณ์ในระบบเครือข่าย (Media Access Control Address: MAC) [15-17] เป็นต้น ข้อมูลที่ใช้รับส่งในระดับชั้นดาต้าลิงค์เรียกว่า เฟรม (Frame) รูปแบบการรับส่งข้อมูลในชั้นนี้จะมีลักษณะเป็นแบบ Hop-to-Hop ดาต้าลิงค์จะใช้วิธีการตรวจสอบความถูกต้องของข้อมูลด้วยวิธีการที่เรียกว่า Checksum คือ การคำนวณผลรวมของข้อมูลทั้งหมดและบันทึกลงในส่วนท้ายของเฟรมข้อมูล ปลายทางที่ได้รับเฟรมข้อมูลจะอาศัย Checksum เพื่อคำนวณว่าเฟรมข้อมูลที่ส่งมาดังกล่าวครบถ้วน หรือไม่ ถ้าไม่ครบก็จะแจ้งไปยังต้นทางว่าให้ส่งเฟรมข้อมูลมาใหม่อีกจนกว่าจะครบถ้วน สำหรับเทคโนโลยีที่ใช้ในชั้นดาต้าลิงค์มีหลายแบบ เช่น Ethernet, Token Ring หรือ FDDI เป็นต้น



รูปที่ 1.6 แสดงการทำงานของชั้นดาต้าลิงค์ [1]

ชั้นเน็ตเวิร์ค (Network layer)

ชั้นเน็ตเวิร์กมีหน้าที่ดังนี้ คือ ควบคุมการสื่อสารระหว่างจุดต่อจุดบนเครือข่าย คำนวณหาเส้นทางที่ดีที่สุดหรือสั้นที่สุด และการบริหารจัดการที่อยู่เสมือนของอุปกรณ์บนเครือข่าย (IP Address) [1, 32] เป็นต้น ข้อมูลที่ใช้สื่อสารในระดับชั้นเน็ตเวิร์กเรียกว่า แพ็คเก็ต (Packet) โดยแพ็คเก็ตนั้นถูกออกแบบมาเพื่อสนับสนุนการรับส่งข้อมูลข้ามเครือข่ายได้ง่ายขึ้น เนื่องจากในสถานการณ์จริงระบบเครือข่ายจะมีขนาดของแบบดิจิตที่ไม่เท่ากัน ดังนั้นแต่ละแพ็คเก็ตจะต้องมีขนาดน้อยกว่าหรือเท่ากับแบบดิจิตที่เล็กที่สุดในระบบเครือข่ายที่สื่อสารกัน ชั้นเน็ตเวิร์กยังมีหน้าที่หลักสำคัญอีกประการ คือ การจัดเตรียมข้อมูลสำหรับชั้นสื่อสารท่านสปอร์ต (Transport Layer) และชั้นสื่อสารดาต้าลิงก์ด้วย โปรโตคอลที่นิยมใช้งานในชั้นเน็ตเวิร์ก เช่น IP และ IPX เป็นต้น การทำงานของชั้นเน็ตเวิร์กแสดงดังรูปที่ 1.7

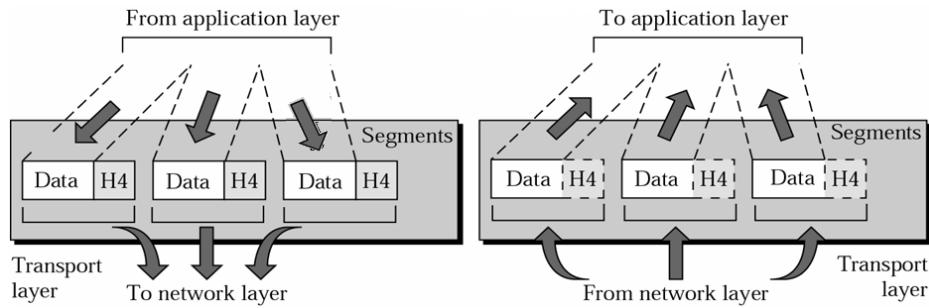


รูปที่ 1.7 แสดงการทำงานของชั้นสื่อสารเน็ตเวิร์ก [1]

ชั้นท่านสปอร์ต (Transport layer)

หน้าที่ควบคุมการสื่อสารข้อมูลบนเครือข่ายให้เป็นไปด้วยความราบรื่นและข้อมูลครบถ้วนรวมไปถึงการกำหนดปริมาณการรับส่งข้อมูลให้เหมาะสมกับสภาพแวดล้อมการเชื่อมต่อเครือข่ายในขณะนั้น ๆ ด้วย ข้อมูลที่ใช้รับส่งในชั้นท่านสปอร์ตเรียกว่า เซกเมนต์ (Segment) [1, 19] การรับส่งข้อมูลเป็นแบบ Process-to-Process คือ การสื่อสารระหว่างโปรแกรมประยุกต์ใด ๆ ที่ทำงานอยู่ต่างสถานที่กัน หรือกล่าวอีกนัยหนึ่งคือ อุปกรณ์บนเครือข่ายสามารถเชื่อมต่อได้พร้อม ๆ กันมากกว่า 1 ช่องทางนั้นเอง โดยอาศัยหมายเลขพอร์ต (Port number) ในการเชื่อมต่อ แสดงในรูปที่ 1.8 ชั้นสื่อสารท่านสปอร์ตมีความสามารถในการตรวจสอบความครบถ้วนของข้อมูล โดยอาศัยโปรโตคอลชื่อว่าทีซีพี (Transmission Control Protocol: TCP) และทำงานร่วมกับโปรโตคอลไอพี

ในชั้นเน็ตเวิร์กอย่างใกล้ชิด ดังนั้นโปรโตคอลทั้งสองจะถูกจับคู่กันเสมอในตัวแบบทีซีพี/ไอพีนั่นเอง



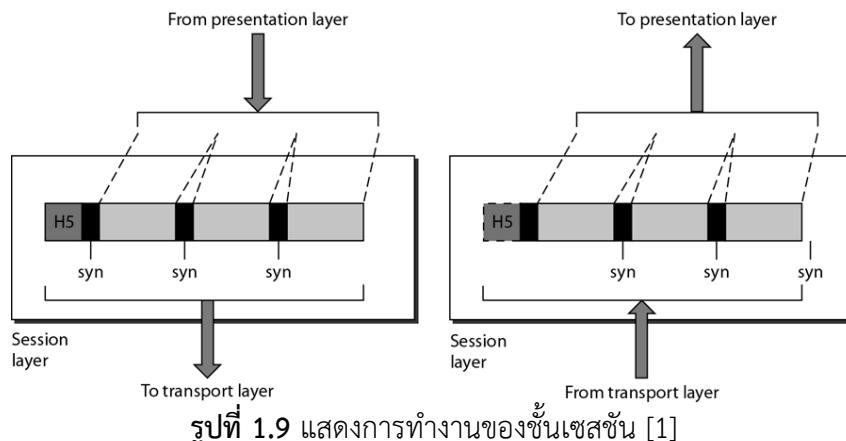
รูปที่ 1.8 แสดงการทำงานของชั้นทرانสปอร์ต [1]

ชั้นเซสชัน (Session layer)

เป็นชั้นสื่อสารที่จัดการเรื่องของการสร้าง "การเชื่อมต่อแต่ละครั้ง" ให้กับอุปกรณ์บนระบบเครือข่ายทั้งสองฝ่าย (ฝ่ายรับและฝ่ายส่ง) กล่าวคือ ชั้นเซสชันจะให้บริการแก่โปรแกรมประยุกต์ โดยทำหน้าที่ตั้งแต่เริ่มการเชื่อมต่อ ดูแลช่องสัญญาณการสื่อสารข้อมูลในแต่ละครั้ง ไปจนถึงยกเลิกการเชื่อมต่อเมื่อสิ้นสุดการสื่อสาร [1, 21] (ดังรูปที่ 1.9) ข้อมูลที่ใช้สื่อสารในระดับชั้นเซสชันเรียกว่า ข้อความ (Message) นอกจากนั้นยังให้บริการด้านอื่น ๆ เช่น

- กำหนดเงื่อนไขการรับส่งข้อมูลชนิดผลักกันรับส่ง (Half-Duplex) และแบบทั้งสองทิศทางพร้อม ๆ กัน (Full-Duplex)
- กำหนดรูปแบบการสื่อสารระยะไกล เช่น กำหนดช่วงเวลาในการสื่อสารใหม่เมื่อเกิดการสื่อสารที่ผิดพลาดขึ้น

- รายงานข้อผิดพลาดเกี่ยวกับการสื่อสารให้กับชั้นโปรแกรมประยุกต์ให้ทราบ

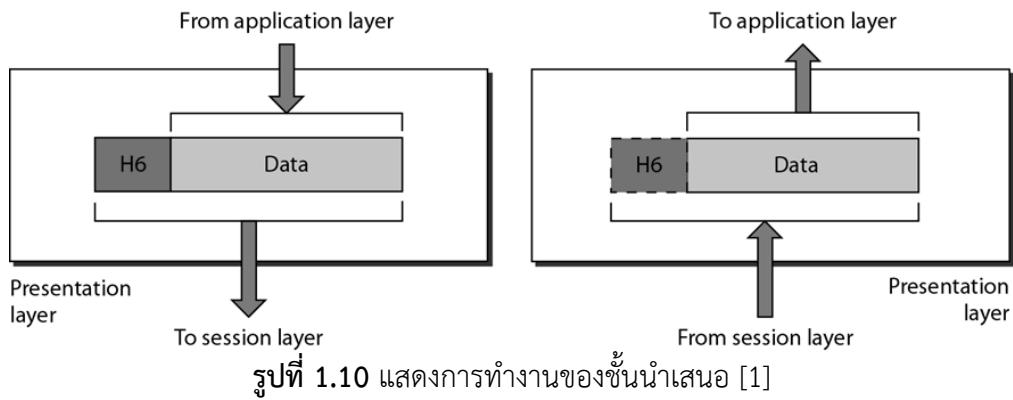


รูปที่ 1.9 แสดงการทำงานของชั้นเซสชัน [1]

ชั้นนำเสนอ (Presentation Layer)

หน้าที่หลัก คือ จัดรูปแบบและนำเสนอข้อมูลระหว่างการสื่อสาร ให้เป็นไปตามที่ต้องการโดยมีการกำหนดรูปแบบการรับส่งข้อมูลสำหรับใช้ในการแลกเปลี่ยน [1, 33, 39] ทั้งนี้ยังรวมไปถึง

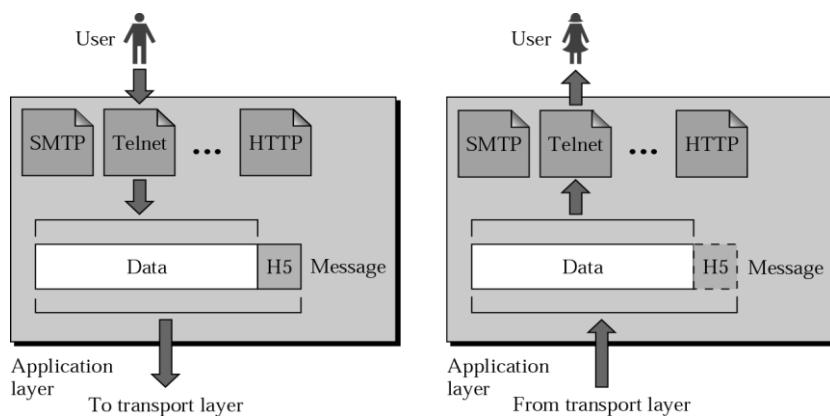
การแปลงข้อมูลให้อยู่ในรูปแบบรหัสมาตรฐาน เช่น ASCII หรือ EBCDIC ลดขนาดข้อมูล (Data compression) การเข้ารหัสหรือถอดรหัสข้อมูล (Data encryption/decryption) เพื่อความปลอดภัยในการสื่อสาร เป็นต้น ดังรูปที่ 1.10 ข้อมูลที่ใช้สื่อสารในระดับชั้นนำเสนอเรียกว่า ข้อความ (Message)



รูปที่ 1.10 แสดงการทำงานของชั้นนำเสนอ [1]

ชั้นแอพพลิเคชันหรือประยุกต์ (Application layer)

เป็นชั้นบนสุดของตัวแบบโอลีโอส์ไอ มีหน้าที่อำนวยความสะดวกในการติดต่อสื่อสารระหว่างโปรแกรมประยุกต์กับผู้ใช้งานให้เป็นไปตามที่ผู้ใช้ต้องการ [1, 33] ตัวอย่าง โปรแกรมประยุกต์ เช่น การรับส่งจดหมายอิเล็กทรอนิกส์ (E-mail) การโอนย้ายแฟ้มข้อมูลข้ามเครือข่าย (File transfer) การขอเข้าใช้ระบบคอมพิวเตอร์ในเครือข่าย (Host Terminal) การจัดแฟ้มข้อมูลในลักษณะต่าง ๆ เป็นต้น ดังรูปที่ 1.11 การใช้บริการในระดับโปรแกรมประยุกต์ โดยปกติจะใช้การพิมพ์คำสั่งต่าง ๆ ผ่านทางระบบปฏิบัติการ (Command line interface) หรือการใช้งานด้วยกราฟฟิก (Graphic user interface)



รูปที่ 1.11 แสดงการทำงานของชั้นประยุกต์

จากตารางที่ 1.1 แสดงหน้าที่ในแต่ละชั้นของตัวแบบโอลีโอโดยสรุป ซึ่งตัวแบบทั้ง 2 ชนิด (โอลีโอและทีซีพี/โอลีพี) จะถูกใช้ในการอ้างอิงสำหรับการออกแบบโครงสร้างเครือข่ายในบทต่อ ๆ ไปเสมอ

ตารางที่ 1.1 สรุปการทำงานของแต่ละเลเยอร์

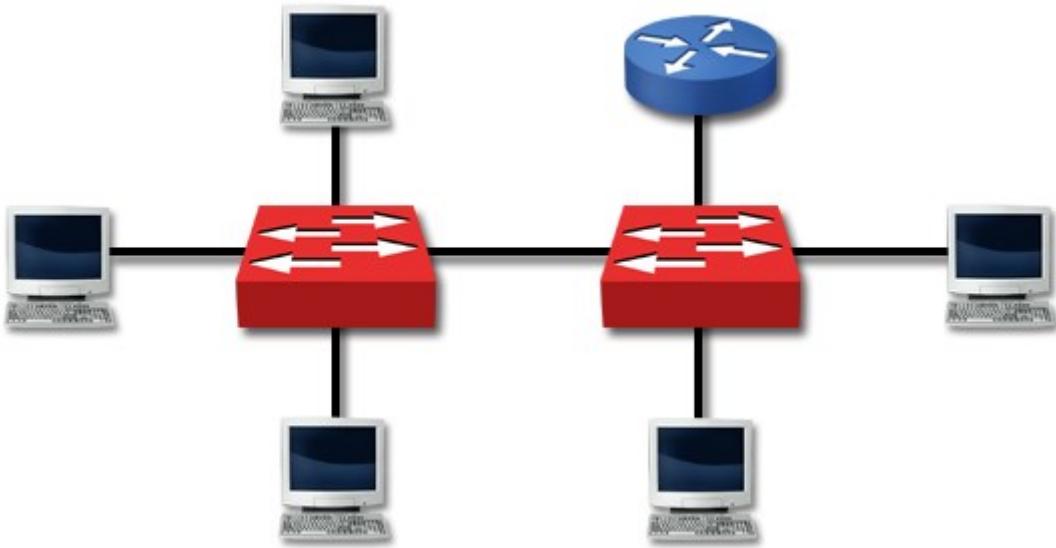
OSI Layer	Functions
7 Application (Message)	ควบคุมการตอบสนองกับผู้ใช้งานและจัดเตรียมบริการต่าง ๆ ให้กับผู้ใช้งาน
6 Presentation (Message)	ควบคุมการแสดงผลข้อมูลและจัดเตรียมกลไกเพื่อรับการเข้ารหัส การถอดรหัส การบีบอัดข้อมูล ข้อมูลจะถูกพิจารณาเป็นข้อความ
5 Session (Message)	ดูแลเกี่ยวกับจัดการแลกเปลี่ยนข้อมูล เช่น การสถาปนาการเชื่อมต่อและการยกเลิกการเชื่อมต่อ ข้อมูลจะถูกพิจารณาเป็นข้อความ
4 Transport (Segment)	การส่งข้อมูลเป็นแบบโพรเซสกับโพรเซส ควบคุมการทำงานให้โปรแกรมประยุกต์สามารถทำงานได้พร้อม ๆ กัน ข้อมูลจะถูกพิจารณาเป็นเซ็กเมนต์
3 Network (Packet)	เป็นการส่งข้อมูลแบบต้นทางไปยังปลายทาง กำหนดเส้นทางโดยใช้ที่อยู่เสมือนจริง (Addressing) ค้นหาเส้นทางที่ดีที่สุดสำหรับการสื่อสาร และพิจารณาข้อมูลในระดับแพ็กเก็ต
2 Data Link (Frame)	เป็นการเชื่อมต่อแบบจุดต่อจุด ควบคุมการให้เลขอข้อมูล ตรวจสอบข้อผิดพลาดของข้อมูล และพิจารณาข้อมูลในระดับเฟรม
1 Physical (Bit)	ไม่สนใจความถูกผิดของข้อมูล จะทำหน้าที่แปลงและรับส่งข้อมูล เป็น 0 และ 1 พิจารณาข้อมูลในระดับบิต

แบบฝึกหัดท้ายบท

1. โครงสร้างการทำงานของโมเดลแบบ OSI กับโมเดล TCP/IP มีลักษณะเหมือนหรือแตกต่างกันอย่างไร
2. การสื่อสารแบบ peer-to-peer มีลักษณะเป็นอย่างไร
3. ในโครงสร้างการทำงานของโมเดลแบบ OSI มีทั้งหมดกี่ชั้นและแต่ละชั้นทำหน้าที่อย่างไร
4. ข้อมูลที่ใช้สื่อสารในแต่ละชั้นของโมเดลแบบ OSI แตกต่างกัน มีชื่อเรียกว่าอะไรบ้างและแต่ละชนิดมีความแตกต่างกันอย่างไร

บทที่ 2

การออกแบบระบบเครือข่ายท้องถิ่น (Local Area Network Design)



- Ethernet
- Bridging and switching
- Routing
- LAN segmentation
- Using show commands

แนวคิด

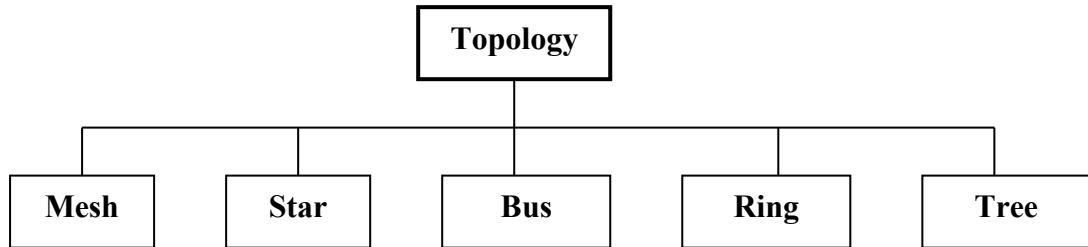
ก่อนการติดตั้งอุปกรณ์ระบบเครือข่ายจริงจำเป็นอย่างยิ่งที่จะต้องทำความเข้าใจเกี่ยวกับขั้นตอนการออกแบบ การวิเคราะห์โครงสร้างของระบบเครือข่ายรวมถึงเทคโนโลยีที่มีอยู่ในปัจจุบันได้อย่างเหมาะสม ซึ่งความรู้และความเข้าใจดังกล่าวจะนำไปสู่การสร้างระบบเครือข่ายที่เหมาะสมต่อองค์กรของตัวเองให้มากที่สุด

วัตถุประสงค์

1. เพื่อให้ทราบถึงขั้นตอนการวิเคราะห์และออกแบบระบบเครือข่ายในระดับท้องถิ่น
2. เพื่อให้ทราบถึงขั้นตอนการดำเนินงานสำหรับติดตั้งระบบเครือข่ายระดับท้องถิ่น

1. ประเภทของการเชื่อมต่อเครือข่าย (Categories of Topology)

ปัจจุบันการเชื่อมต่อที่นิยมและใช้งานมีอยู่ 5 ประเภท คือ Mesh, Star, Bus, Ring, Tree [1, 5, 18] ดังรูปที่ 2.1 ซึ่งการเชื่อมต่อแต่ละประเภทมีข้อดีข้อเสียต่างกันดังนี้



รูปที่ 2.1 ประเภทของการเชื่อมต่อเครือข่ายทั้งถิ่น

1.1 การเชื่อมต่อแบบ Mesh Topology

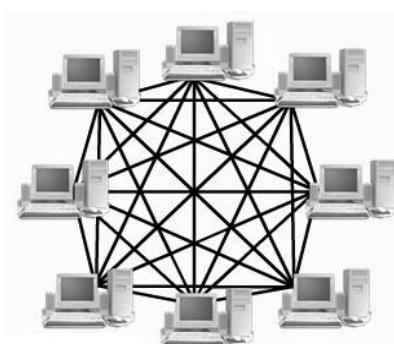
การเชื่อมต่อแบบนี้จะเชื่อมสายสัญญาณทุกเส้นถึงกันทั้งหมด ดังรูปที่ 2.2

ข้อดี

1. ถ้าเครื่องใดเครื่องหนึ่งไม่สามารถใช้งานได้จะไม่ส่งผลกระทบกับเครื่องอื่น ๆ
2. เมื่อต้องการส่งข้อมูลไม่จำเป็นต้องรอสามารถส่งข้อมูลได้ทันที
3. มีความน่าเชื่อถือสูง

ข้อเสีย

1. สิ้นเปลืองสายสัญญาณที่ใช้เชื่อมต่อเป็นอย่างมาก
2. ไม่สะดวกเมื่อต้องการย้ายสถานที่ตั้งของเครื่องใหม่
3. สิ้นเปลืองพอร์ตสำหรับใช้เชื่อมต่อ เช่น ใช้เน็ตเวิร์คการ์ดมากกว่า 1 ใน 4
4. ถ้าจำนวนสายสัญญาณมาก ๆ จะไม่สะดวกในการจัดให้เป็นระเบียบ



รูปที่ 2.2 การเชื่อมต่อแบบ Mesh topology

1.2 การเชื่อมต่อแบบ Star Topology

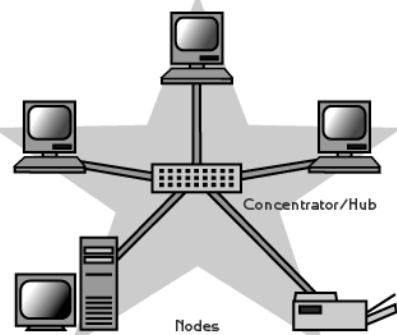
ลักษณะการเชื่อมต่อโครงสร้างแบบสตาร์ มีลักษณะเป็นแบบดาวกระจาย คือ จะมีอุปกรณ์ เช่น อับ หรือสวิตช์เป็นศูนย์กลาง ซึ่งการเชื่อมต่อลักษณะนี้มีประโยชน์คือ กรณีมีสายสัญญาณเส้นใดเส้นหนึ่งหลุดหรือเสียก็จะไม่มีผลกระทบต่อการทำงานของระบบ นอกจากนี้ถ้าหากมีการเพิ่มเครื่องคอมพิวเตอร์เข้าไปอีกในเครือข่ายก็สามารถทำงานได้ทันที การเชื่อมต่อแบบนี้เป็นที่นิยมมากในปัจจุบัน เนื่องจากอุปกรณ์ที่ใช้เป็นศูนย์กลาง เช่น อับหรือสวิตช์ มีราคาถูกลงอย่างมาก ในขณะที่ประสิทธิภาพการทำงานเพิ่มสูงขึ้นเรื่อยๆ จนในปัจจุบันอุปกรณ์ดังกล่าวมีความเร็วในระดับกิกะบิตแล้ว ดังรูปที่ 2.3

ข้อดี

1. ถ้าเครื่องใดเครื่องหนึ่งไม่สามารถใช้งานได้จะไม่ส่งผลกระทบกับเครื่องอื่น ๆ
2. การเชื่อมต่อทำได้ง่ายและสะดวก
3. จำนวนเส้นของสัญญาณใช้เท่ากับจำนวนของเครื่องที่ทำการเชื่อมต่อ (น้อยกว่า Mesh)
4. ปรับปรุงได้ง่ายและอุปกรณ์มีราคาถูก

ข้อเสีย

1. ถ้าจุดที่รวมศูนย์ เช่น อับ เสียหายจะส่งผลกระทบต่อทุก ๆ เครื่องที่เชื่อมต่อด้วย
2. การส่งข้อมูลต้องผลัดกันส่ง ถ้าสัญญาณไม่ว่างจะต้องเสียเวลาในการรออย
3. เมื่อปริมาณข้อมูลเพิ่มขึ้นถึงระดับหนึ่งจะทำให้เกิดคอกขวด

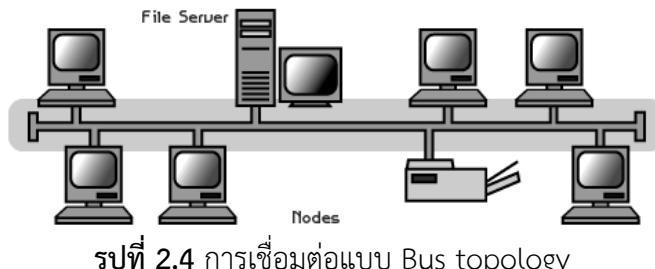


รูปที่ 2.3 การเชื่อมต่อแบบ Star topology

1.3 การเชื่อมต่อแบบ Bus Topology

ลักษณะการเชื่อมต่อแบบบัสจะมีลักษณะเป็นแบบอนุกรม โดยใช้สายเคเบิลหรือสายสัญญาณเพียงเส้นเดียว เชื่อมต่อกันไปตลอดเส้นทาง ทำให้โครงสร้างแบบนี้มีจุดอ่อนก็ คือ เมื่อคอมพิวเตอร์ตัวใดตัวหนึ่งเกิดปัญหา ก็จะทำให้คอมพิวเตอร์ทั้งระบบประสบปัญหาไปด้วย ข้อดีของโครงสร้างแบบนี้ก็คือ ไม่จำเป็นต้องมีอุปกรณ์ที่ทำหน้าที่เป็นศูนย์กลางควบคุมการทำงานอย่างเช่น อับหรือสวิตช์ ใช้สายสัญญาณเพียงเส้นเดียวที่เพียงพอ โครงสร้างแบบนี้เหมาะสมกับเครือข่ายที่มีขนาดเล็กและมีจำนวนเครื่องคอมพิวเตอร์ในปริมาณไม่มาก ในปัจจุบันไม่นิยมใช้กันแล้ว เนื่องจาก

ไม่ได้มีการพัฒนาความสามารถเพิ่มเติม ส่วนความเร็วในการรับส่งข้อมูลได้ถึง 10 เมกะบิตต่อวินาที (Mbps) ดังรูปที่ 2.4



รูปที่ 2.4 การเชื่อมต่อแบบ Bus topology

ข้อดี

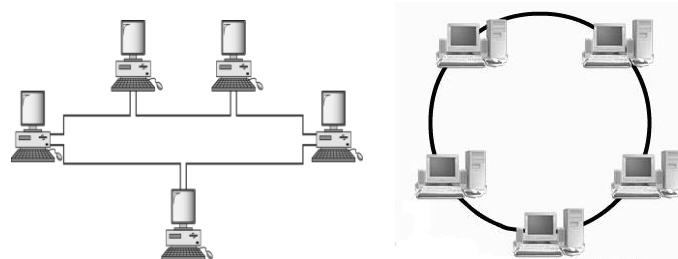
1. ปริมาณการส่งข้อมูลทำได้สูง เพราะสายสัญญาณหลักเป็นสายประเภทป้องกันสัญญาณ
รบกวน
2. ใช้สายนำสัญญาณไม่มาก

ข้อเสีย

1. ข้อมูลที่ส่งและรับจะผ่านไปยังทุก ๆ เครื่อง ซึ่งทำให้ประสิทธิภาพโดยรวมเสียไป
2. ต้องใช้อุปกรณ์เชื่อมต่อค่อนข้างมาก ทำให้โอกาสเกิดความผิดพลาดได้ง่ายกว่าแบบอื่น
3. เมื่อเครื่องใดเครื่องหนึ่งเสียหายหรือการเชื่อมต่อไม่สมบูรณ์จะส่งผลกระทบต่อเครื่องอื่น
ๆ ทั้งหมด
4. อุปกรณ์ค่อนข้างมีราคาแพง เชื่อมต้อยาก และไม่เป็นที่นิยมในปัจจุบัน

1.4 การเชื่อมต่อแบบ Ring Topology

ลักษณะการเชื่อมต่อจะเป็นแบบวงแหวน ดังรูปที่ 2.5 การส่งข้อมูลจะเป็นแบบทิศทางเดียว ซึ่งถ้าข้อมูลที่ส่งออกไปแล้วไม่ตรงกับคอมพิวเตอร์เครื่องรับตามที่เครื่องต้นทางระบุมา ข้อมูลจะถูกส่ง ต่อไปยังเครื่องถัดไป จนกว่าจะถึงเครื่องปลายทางที่ระบุไว้ จุดอ่อนของการเชื่อมต่อ มีลักษณะคล้าย ๆ แบบบัส เมื่อสายนำสัญญาณขาดจุดใดจุดหนึ่งระบบจะหยุดทำงานทันที ปัจจุบันการเชื่อมต่อแบบริง ยังมีการใช้งานอยู่บ้าง โดยส่วนใหญ่ใช้เชื่อมต่อแบบวงแหวน 2 เส้น เพื่อใช้เป็นเส้นทางสำรองและ นิยมเชื่อมต่อเพื่อทำหน้าที่เป็นเครือข่ายหลักของระบบ (Backbone) ด้วย



รูปที่ 2.5 การเชื่อมต่อแบบ Ring topology

ข้อดี

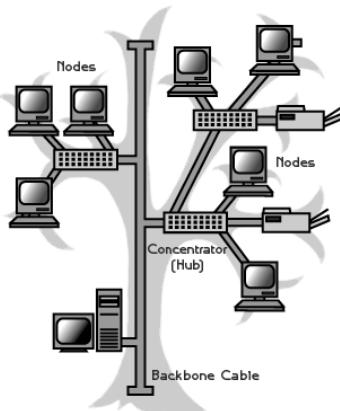
1. ใช้สายนำสัญญาณไม่มาก
2. การส่งข้อมูลจะไม่ชนกันเนื่องจากจะใช้ Token ควบคุมจังหวะของการส่งข้อมูลแบบเป็นลำดับ เครื่องที่ได้ Token เท่านั้นจึงจะส่งข้อมูลได้

ข้อเสีย

1. ถ้าเครื่องคอมพิวเตอร์เครื่องหนึ่งส่ายนำสัญญาณในระบบเกิดปัญหาจะทำให้ระบบไม่สามารถทำงานต่อไปได้
2. ความรวดเร็วในการส่งข้อมูลไม่มีประสิทธิภาพ เพราะต้องได้รับ Token ก่อนจึงสามารถส่งข้อมูลออกได้

1.5 การเชื่อมต่อแบบ Tree Topology

โครงสร้างการเชื่อมต่อแบบทรีเป็นแบบสุดท้ายของโครงสร้างเครือข่ายที่ได้รับความนิยม โดยลักษณะโครงข่ายแบบนี้ คือ การนำเครือข่ายย่อย ๆ ที่มีโครงข่ายตามแบบที่กล่าวไว้ข้างต้นทั้ง 4 แบบมารวมกันหรือเชื่อมต่อกันให้มีขนาดใหญ่ขึ้น เช่น เครือข่ายที่ผู้ผลิตจากภารกิจเครือข่ายที่มีโครงสร้างแบบบัสและแบบสตาร์มาผสมกัน ดังรูปที่ 2.6



รูปที่ 2.6 การเชื่อมต่อแบบ Tree topology

ข้อดี

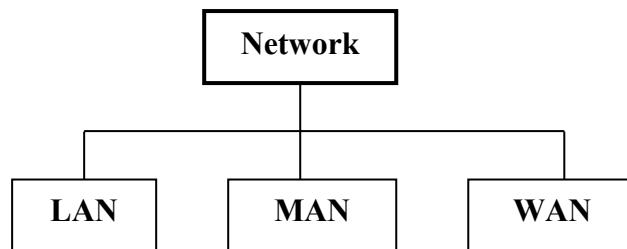
1. ผสมผสานการเชื่อมต่อเครือข่ายหลาย ๆ ประเภทเข้าด้วยกันเพื่อให้เหมาะสมกับสภาพแวดล้อมจริงที่ทำงานอยู่
2. ความเร็วในการส่งข้อมูลใกล้เคียงกับความเป็นจริง คือ โครงข่ายหลักจะมีขนาดของแบบวิดีโอมาก ส่วนเชื่อมต่อกับผู้ใช้ก็ขึ้นอยู่กับโภพโลยีที่ใช้งาน
3. การปรับปรุงโครงข่ายสามารถทำได้ง่าย

ข้อเสีย

1. โครงข่ายหลักส่วนมากต้องรองรับความเร็วที่สูง ดังนั้นราคากำลังเชื่อมต่อโครงสร้างเครือข่ายจึงสูงและต้องใช้ความชำนาญในการติดตั้งด้วย
2. เมื่อปริมาณข้อมูลสูงถึงจุดที่โครงข่ายหลักไม่สามารถรองรับได้ เครือข่ายจะเกิดปัญหาของขาดของเครือข่าย (Bottleneck network)
3. กรณีของโครงข่ายหลักเสียหายจะทำให้ระบบหั้งหมดหยุดทำงานทันที แต่สามารถแก้ไขด้วยการสร้างสายนำสัญญาณสำรองไว้อีกชุดหนึ่ง
4. เมื่อโทโพลีย์ที่ใช้เชื่อมต่อมีความหลากหลายมาก จะทำให้การบริหารจัดการเครือข่ายยุ่งยากเพิ่มขึ้น

2. ประเภทของระบบเครือข่าย (Categories of Network)

โดยหลัก ๆ แล้วแบ่งออกเป็น 3 ประเภท คือ Local Area Network (LAN), Metropolitan Area Network (MAN), Wide Area Network (WAN) [1, 17, 18] ดังรูปที่ 2.7 โดยขึ้นอยู่กับขนาดของเครือข่าย เช่น เครือข่ายที่มีความเร็วสูงและมีบริเวณไม่กว้างจะเป็นชนิด LAN ส่วนเครือข่ายที่ต้องสื่อสารกันระหว่างเมืองใหญ่ ๆ จะเป็นลักษณะของ MAN ส่วน WAN จะใช้เชื่อมต่อผ่าน ISP หรือระหว่างประเทศซึ่งจะมีความเร็วในการส่งข้อมูลที่ต่ำที่สุด

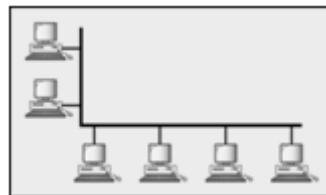


รูปที่ 2.7 ประเภทของระบบเครือข่าย

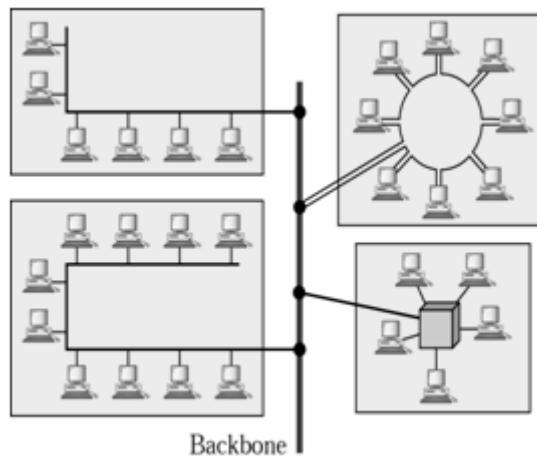
2.1 เครือข่ายท้องถิ่น (LAN)

เครือข่ายท้องถิ่นมีขอบเขตการเชื่อมต่อแคบ เช่น ภายในอาคาร ออฟฟิศ สำนักงาน หรืออาคารที่อยู่ติด ๆ กัน ระยะทางไม่ควรเกิน 2,000 ฟุต เครือข่ายท้องถิ่นได้รับความนิยมมากในการเชื่อมต่ออุปกรณ์สำนักงานเข้าด้วยกัน เช่น คอมพิวเตอร์ตั้งโต๊ะ คอมพิวเตอร์โน๊ตบุ๊ค เครื่องพิมพ์งาน และอุปกรณ์ในสำนักงาน เป็นต้น โดยอาศัยโทโพลีย์ในการเชื่อมต่อ เช่น บัส ริง สถาาร์ หรือหลาย ๆ แบบทำงานร่วมกัน ดังรูปที่ 2.8 และ 2.9 สำหรับความเร็วในการสื่อสารข้อมูลในเครือข่ายแบบท้องถิ่นสูงมาก ปัจจุบันสามารถสื่อสารด้วยความเร็วระดับ 40 กิกะบิตต่อวินาที (Gbps) สำหรับเครือข่ายชนิดใช้สายนำสัญญาณ และความเร็ว 1 กิกะบิตสำหรับเครือข่ายชนิดไร้สาย

(Wireless) เครือข่ายห้องถินที่รวมอุปกรณ์สื่อสารของผู้ใช้ทั้งหมดเข้าไว้ด้วยกันรวมถึงเครือข่ายย่อย ๆ โดยมีโครงข่ายหลักที่ผสานเครือข่ายย่อย ๆ เข้าไว้ด้วยกันเรียกว่า แกนหลักเครือข่าย (Backbone network) การสื่อสารข้อมูลภายในเครือข่าย LAN สามารถสื่อสารได้ทันทีตราบใดที่สายน้ำสัญญาณว่าง แต่เมื่อมีอุปกรณ์ใดในเครือข่ายต้องการสื่อสารข้ามเครือข่าย ข้อมูลจำเป็นต้องถูกส่งออกไปทางตำแหน่งทางเข้าออกของระบบเครือข่าย ตำแหน่งดังกล่าว เรียกว่าเกตเวย์ (Gateway) ของเครือข่าย



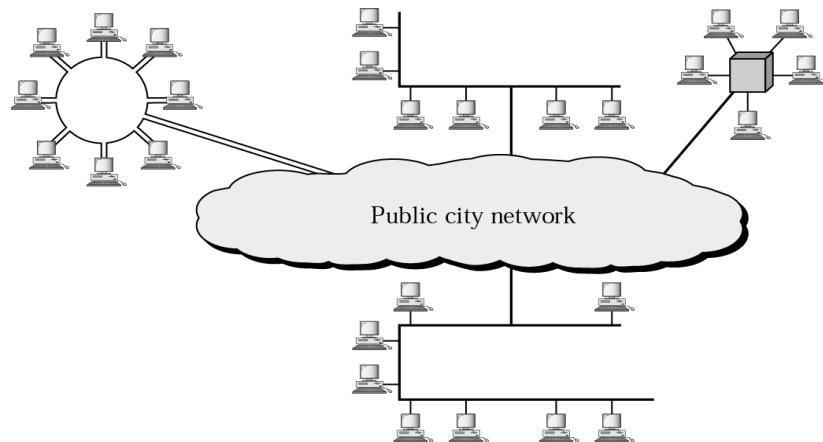
รูปที่ 2.8 การเชื่อมต่อ LAN เพียงโ拓โพโลยีเดียว



รูปที่ 2.9 การเชื่อมต่อ LAN หลาย ๆ topology เข้าด้วยกัน

2.2 เครือข่ายเมือง (MAN)

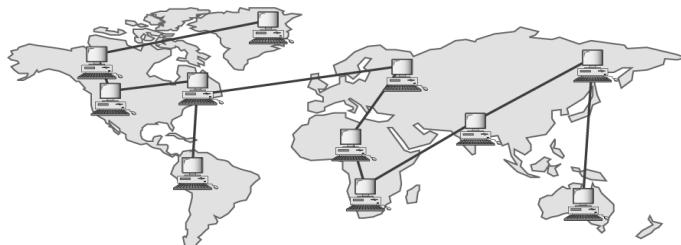
โดยพื้นฐานแล้วระบบเครือข่ายเมือง (MAN) มีลักษณะคล้ายกับระบบเครือข่ายห้องถิน แต่มีอาณาเขตที่กว้างใหญ่กว่า คือ มีขอบเขตตั้งแต่ภายในเมืองเดียวกันหรือหลาย ๆ เมืองที่อยู่ติดกันก็ได้ เช่น ระบบเครือข่ายที่เชื่อมต่อภัยในจังหวัด เป็นต้น โดยปกติการเชื่อมต่อเครือข่ายดังกล่าวจะอาศัยระบบบริการเครือข่ายสาธารณะ เช่น โครงข่ายโทรศัพท์ เป็นต้น จึงเป็นเครือข่ายที่ใช้กับองค์กรที่มีสำนักงานห่างไกลและต้องการเชื่อมสำนักงานเหล่านั้นเข้าด้วยกัน เช่น ธนาคาร เครือข่ายสถาบันการศึกษา บริษัทเอกชน เป็นต้น เครือข่ายระดับเมืองจะเชื่อมโยงเครือข่ายในระยะทางที่ไกลมาก ดังนั้นความเร็วในการสื่อสารจึงไม่สูงเท่ากับเครือข่ายห้องถิน เนื่องจากมีสัญญาณรบกวนมาก เทคโนโลยีที่ใช้เชื่อมเครือข่ายเมืองมีความหลากหลาย เช่น ผ่านระบบสัญญาณดาวเทียม เส้นใยแก้วนำแสง คลื่นไมโครเวฟ คลื่นวิทยุ สายเคเบิล เป็นต้น ดังรูปที่ 2.10



รูปที่ 2.10 การเชื่อมต่อแบบเครือข่ายเมือง (MAN)

2.3 เครือข่ายบริเวณกว้าง (WAN)

เป็นระบบที่มีขอบเขตการใช้งานกว้างไกลกว่าระบบเมือง ซึ่งอาจกล่าวได้ว่าเป็นระบบที่ไร้ขอบเขต เช่น ระบบการสื่อสารข้อมูลผ่านดาวเทียมของสถานีโทรทัศน์ต่าง ๆ แต่การที่จะเชื่อมต่อเครือข่ายที่มีระยะทางห่างกันมาก ๆ ให้เป็นเครือข่ายเดียวกันทั้งหมดนั้น จะเป็นต้องอาศัยเครือข่ายสาธารณะ (Public Networks) ที่ให้บริการการสื่อสาร โดยเชื่อมต่อกับโนดเดิมผ่านเครือข่ายโทรศัพท์สาธารณะ (Public Switching Telephone Network: PSTN) ซึ่งมีทั้งลักษณะที่ต้องมีการเชื่อมต่อ ก่อน (Dial-up) หรือเชื่อมต่อแบบสายตัว เช่น สายเช่า (Lease Line) ดังรูปที่ 2.11 ปริมาณข้อมูลที่สื่อสารบนระบบเครือข่ายบริเวณกว้างมีความเร็วต่ำกว่าระดับเมืองและมีต้นทุนในการเชื่อมต่อสูง



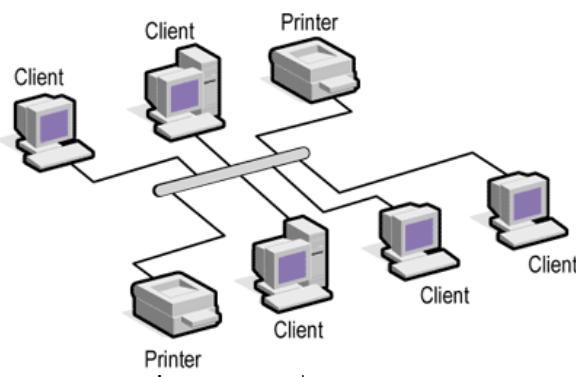
รูปที่ 2.11 การเชื่อมต่อเครือข่ายแบบ WAN

3. วิธีการเชื่อมเครือข่ายแบบท้องถิ่น

ในหัวข้อนี้จะกล่าวถึงวิธีการเชื่อมต่อระบบเครือข่ายท้องถิ่นในทางปฏิบัติ โดยปกตินิยมเรียกว่า "การเชื่อมต่อแลน" ซึ่งรูปแบบการเชื่อมต่อเครือข่ายในปัจจุบันมีอยู่ 3 ประเภท คือ การเชื่อมต่อแบบบัส วงแหวน และแบบสตาร์ ในทางปฏิบัติการเชื่อมต่อเครือข่ายทั้ง 3 วิธี จะใช้อุปกรณ์และวิธีการที่แตกต่างกัน โดยพิจารณาเลือกจากข้อดีข้อเสียหรือจากความเหมาะสมสมกับงานแต่ละประเภท ปัจจุบันพบว่าการเชื่อมต่อแบบสตาร์นิยมใช้งานมากที่สุด เพราะสามารถตรวจสอบหาข้อผิดพลาดได้ง่าย ในตัวอย่างเกือบทั้งหมดในเอกสารคำสอนนี้จะกล่าวถึงการเชื่อมต่อแบบสตาร์เป็นหลัก สำหรับการเชื่อมต่ออีก 2 แบบ จะมีการใช้งานอยู่บ้างแต่ไม่มากนัก โดยมีรายละเอียดดังต่อไปนี้

3.1 การติดตั้งเครือข่ายแบบบัส

วิธีการเชื่อมต่อแบบนี้มีลักษณะเหมือนกับการสร้างถนนสายหลักแล้วมีซอยแยกจากถนนหลักแต่ก็จะไปเรื่อยตามจำนวนผู้ใช้งาน โดยเริ่มต้นจากการวางสายสัญญาณเป็นแกนหลักเรียกว่า บัสหลักหรือแบคโบน (Backbone) จากนั้นเชื่อมสายสัญญาณรองจากแกนหลักไปยังเครื่องคอมพิวเตอร์ตามจุดต่าง ๆ โดยที่ปลายของสายสัญญาณหลักทั้งสองข้างจะมีเทอร์มิเนเตอร์ (Terminator) ปิดอยู่ เพื่อให้สัญญาณไฟฟ้าคงจร [22, 23] ดังรูปที่ 2.12



รูปที่ 2.12 การเชื่อมต่อแบบบัส

สายสัญญาณที่ใช้ในการเชื่อมต่อแบบบัสมีอยู่สองชนิด คือ

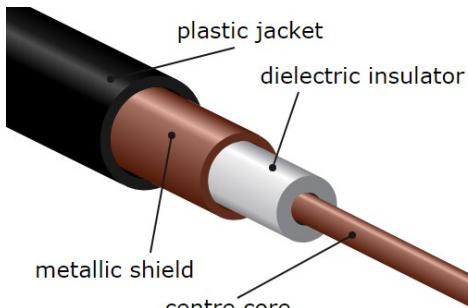
- 1) สายโคแอกซ์แบบบาง (Thin Coaxial Cable) เป็นสายที่มีขนาดเล็ก เส้นผ่านศูนย์กลาง ประมาณ 0.64 เซนติเมตร (รูปที่ 2.13) เนื่องจากสายประเภทนี้มีขนาดเล็กและมีความยืดหยุ่นสูง จึงสามารถใช้ได้กับการติดตั้งเครือข่ายเกือบทุกประเภท สายประเภทนี้สามารถนำสัญญาณไฟฟ้าได้ไกลถึง 185 เมตร สายโคแอกซ์แบบบางอยู่ในประเภท RG-58 ซึ่งจะมีความต้านทานในสาย (Impedance) ที่ 50 โอม สายประเภทนี้จะมีแกนกลางอยู่ 2 ลักษณะคือแบบที่เป็นสายห้องแดงเส้นเดียวและแบบที่เป็นไอล์โลหะหลายเส้น



รูปที่ 2.13 สายโคแอกซ์แบบบาง

- 2) สายโคแอกซ์แบบหนา (Thick Coaxial Cable) เป็นสายที่ค่อนข้างแข็งและขนาดใหญ่กว่า สายโคแอกซ์แบบบาง โดยมีเส้นผ่านศูนย์กลางประมาณ 1.27 เซนติเมตร (รูปที่ 2.14) สายชนิดนี้เป็นสายนำสัญญาณประเภทแรกที่ใช้งานกับเครือข่ายแบบอีเทอร์เน็ต (Ethernet) ส่วน

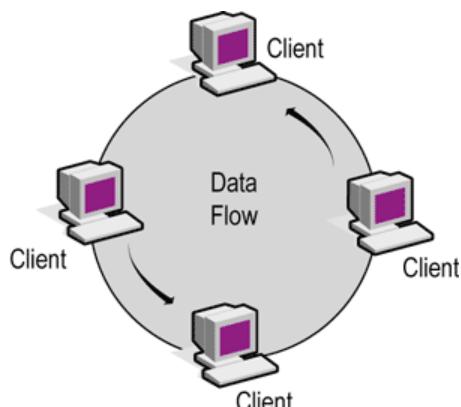
แกนกลางที่เป็นสายทองแดงจะมีขนาดใหญ่ ดังนั้นสายโคแอกซ์แบบหนานี้จึงสามารถนำสัญญาณได้ไกลกว่าแบบบาง โดยปกติประมาณ 500 เมตร ด้วยความสามารถนี้สายโคแอกซ์แบบหนานี้นิยมใช้ในการเชื่อมต่อเส้นทางหลักของข้อมูล หรือแบ็คโอนของเครือข่ายสมัยเริ่มแรก ๆ แต่ปัจจุบันได้ยกเลิกใช้สายโคแอกซ์แล้ว สายสัญญาณที่นิยมใช้ทำเป็นแบ็คโอนคือ สายใยแก้วนำแสง ซึ่งจะได้กล่าวในรายละเอียดในส่วนต่อไป



รูปที่ 2.14 สายโคแอกซ์แบบหนา

โคแอกซ์แบบบางนิยมติดตั้งภายในอาคาร ส่วนสายชนิดหนานี้ใช้ติดตั้งระหว่างอาคารหรือเชื่อมระหว่างชั้นต่าง ๆ ข้อดีของการเดินสายแบบบัสนี้ คือ ติดตั้งง่าย อุปกรณ์ราคาถูก ไม่ต้องใช้อุปกรณ์รวมสัญญาณ (ฮับหรือสวิตซ์) ระยะติดตั้งใกล้ และสามารถติดตั้งอุปกรณ์ทวนสัญญา (Repeater) เพื่อเพิ่มระยะการติดตั้งสายได้ แต่มีข้อเสียคือ ถ้าเกิดข้อผิดพลาดในสายสัญญาณ หรือเกิดการชำรุดที่จุดหนึ่งจุดใดบนบัส จะทำให้เครือข่ายทั้งระบบไม่สามารถใช้งานได้ และการตรวจสอบหาจุดเสียทำได้ยาก ภายหลังจึงไม่นิยมติดตั้งสายแบบบัสมากนัก

3.2 การติดตั้งเครือข่ายแบบวงแหวนหรือริง



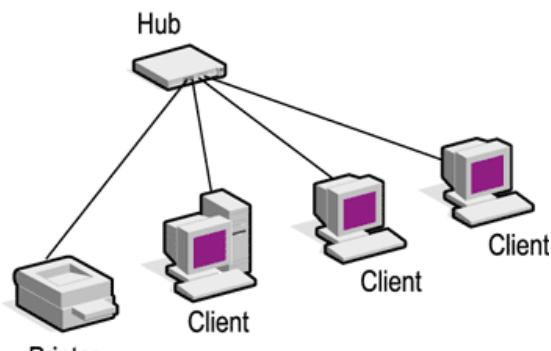
รูปที่ 2.15 การเชื่อมต่อแบบวงแหวน

การติดตั้งเครือข่ายแบบวงแหวนนี้ มีลักษณะทางกายภาพเป็นวงกลม [22, 34] โดยเชื่อมจากเครื่องแรกไปยังเครื่องสุดท้ายและวนกลับมายังเครื่องแรกอีกรัง ดังรูปที่ 2.15 วิธีการเดินสายสัญญาณแบบวงแหวนนี้ ปราศในระบบเครือข่ายมานานแล้ว แต่ปัจจุบันไม่เป็นที่นิยมนักเนื่องจากมีข้อด้อยหลายประการ (ตามที่กล่าวมาแล้วในหัวข้อประเภทของการเชื่อมต่อเครือข่าย) เช่น จุดใดจุดหนึ่งในวงแหวนเสียหายจะทำให้เครื่องอื่น ๆ ไม่สามารถส่งข้อมูลได้ และอุปกรณ์มีราคา

ค่อนข้างสูง แต่อย่างไรก็ตาม การเชื่อมต่อด้วยวิธีการนี้นิยมออกแบบให้เป็นเครือข่ายสำรอง โดยการออกแบบให้เป็นลักษณะวงแหวนซ้อนกันสองเส้นทาง

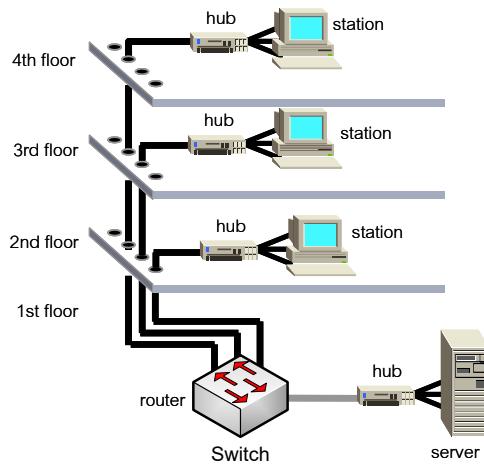
3.3 การติดตั้งเครือข่ายแบบสถาําร์

การเชื่อมต่อเครือข่ายในลักษณะสถาําร์นั้นสามารถพบรหัสโดยทั่วไป โดยรูปแบบการเชื่อมต่อนั้นจะมีลักษณะคล้ายรูปด้านล่าง ตัวอย่างเช่น เครื่องคอมพิวเตอร์ลูกข่ายแต่ละเครื่องจะใช้สายสัญญาณเชื่อมไปที่อับหรือสวิตช์ซึ่งเป็นศูนย์กลางในการเชื่อมต่อ [22, 34] ดังรูปที่ 2.15 โดยจำนวนของเครื่องลูกข่ายจะขึ้นอยู่กับจำนวนพอร์ตของอับหรือสวิตช์ ซึ่งโดยปกติจะมีให้เลือกใช้งานตั้งแต่ 8, 16, 24, 32 และ 48 พอร์ต เป็นต้น



รูปที่ 2.15 การเชื่อมต่อเครือข่ายแบบสถาําร์

ลักษณะการเชื่อมสายสัญญาณแต่ละเครื่องไปยังอับหรือสวิตช์แบบสถาําร์มีข้อดี คือ ถ้าสายสัญญาณเส้นหนึ่งเส้นได้ขาด ก็จะไม่ส่งผลกระทบต่อเครื่องคอมพิวเตอร์อื่น ๆ ทำให้การบำรุงรักษาและแก้ไขปัญหาทำได้ง่าย ปัจจุบันการเชื่อมต่อในลักษณะนี้สามารถรับส่งข้อมูลด้วยความเร็วตั้งแต่ 10 เมกะบิตต่อวินาที ไปจนถึง 40 กิกะบิตต่อวินาที ในขณะที่ราคาของอุปกรณ์ถูกลงเรื่อย ๆ ทำให้ได้รับความนิยมอย่างสูง การเชื่อมต่อแบบสถาําร์นี้สามารถวางตำแหน่งของเครื่องคอมพิวเตอร์และเชื่อมสายสัญญาณอย่างไรก็ได้ โดยไม่จำเป็นต้องวางให้เรียงตามลำดับอย่างการเดินสายแบบบัสหรือแบบวงแหวน สายสัญญาณแบบสถาําร์แต่ละเส้นมีความยาวได้ไม่เกิน 100 เมตร แต่ในทางปฏิบัติความยาวของสายสัญญาณไม่ควรเกิน 85 เมตร สำหรับตัวอย่างการเชื่อมต่อเครือข่ายแบบสถาําร์ในสำนักงานมีลักษณะดังรูปที่ 2.16 ซึ่งผู้ออกแบบเครือข่ายนิยมวางอับหรือสวิตช์ไว้ที่มุมใดมุมหนึ่งของห้องแล้วเชื่อมสายสัญญาณขึ้นมาจากพื้นอาคาร หรือใต้หลังคาเพื่อความเรียบร้อยและสวยงาม

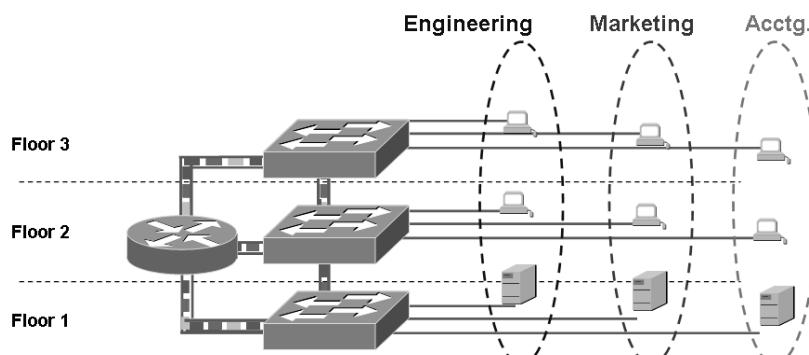


รูปที่ 2.16 แสดงตัวอย่างการเชื่อมโยงเครือข่ายแบบสตาร์ภายในอาคาร

ก่อนการเชื่อมต่อเครือข่ายจริงทุกครั้งจำเป็นต้องออกแบบเครือข่ายในแผนผังหรือในซอฟต์แวร์จำลองเครือข่ายเสียก่อน เพื่อเป็นการประหยัดเวลาการทำงาน ป้องกันการทำงานที่ซ้ำซ้อน มองภาพรวมเครือข่ายได้อย่างครบถ้วน และป้องกันความผิดพลาดที่คาดไม่ถึง เป็นต้น

4. เครือข่ายเสมือนจริง (Virtual Area Network: VLAN)

เครือข่ายเสมือนจริง คือ การแบ่งแยกระบบเครือข่ายคอมพิวเตอร์ออกเป็นส่วน ๆ หรือเป็นกลุ่มย่อย ๆ ด้วยซอฟต์แวร์ โดยไม่มีความจำเป็นต้องปรับแต่งหรือเพิ่มเติมอุปกรณ์ที่ใช้เชื่อมต่อทางกายภาพเลย คอมพิวเตอร์ที่อยู่ภายใต้เครือข่ายเสมือนจริงเดียวกันสามารถสื่อสารกันได้โดย [2, 23, 34] สำหรับคอมพิวเตอร์ที่อยู่ต่างเครือข่ายเสมือนจริง จะไม่สามารถสื่อสารกันได้ ถ้าต้องการสื่อสารไปยังเครือข่ายเสมือนจริงอื่น ๆ จะต้องสื่อสารผ่านเกตเวย์ (Network gateway) ระหว่างเครือข่ายเสมือนจริงเท่านั้น จุดประสงค์หลักของการสร้างเครือข่ายเสมือนจริง เพื่อจำกัดการเข้าถึงข้อมูลของเครื่องคอมพิวเตอร์ที่อยู่ต่างกลุ่ม หรือต่างเครือข่าย ทั้งนี้เพื่อความปลอดภัยของเครือข่าย รวมทั้งสามารถเพิ่มประสิทธิภาพการทำงานของเครือข่ายด้วย ในทางปฏิบัติเครือข่ายหนึ่ง ๆ อาจประกอบด้วยอุปกรณ์กระจายสัญญาณ (Switching) ได้หลาย ๆ ตัว และในอุปกรณ์กระจายสัญญาณแต่ละตัว อาจจะมีเครือข่ายเสมือนจริงประภูมิอยู่ได้มากกว่า 1 เครือข่าย โดยทั่วไปอุปกรณ์กระจายสัญญาณ 1 ตัว สามารถสร้างเครือข่ายเสมือนจริงได้มากถึง 1,024 เครือข่าย ดังรูปที่ 2.17



รูปที่ 2.17 แสดงลักษณะของการสร้างเครือข่ายเสมือนจริง

จากรูปที่ 2.17 เครือข่ายเสมือนจริง ซึ่งว่า Engineering ปรากฏอยู่ในอุปกรณ์ขยายสัญญาณที่ติดตั้งไว้บนชั้นที่ 1 (Floor 1), 2 และ 3 ตามลำดับ เช่นเดียวกันกับเครือข่ายเสมือนจริง Marketing และ Accounting ซึ่งแสดงให้เห็นว่า วิศวกรที่ทำงานอยู่ต่างสถานที่กันสามารถทำงานเสมือนอยู่ในสภาพแวดล้อมเดียวกันได้ คุณสมบัติของเครือข่ายเสมือนจริงจะไม่อนุญาตให้ผู้ใช้งานของแต่ละเครือข่ายเสมือนจริงสื่อสารกันได้โดยตรง เช่น ฝ่ายวิศวกรไม่สามารถเข้าถึงข้อมูลของฝ่ายบัญชีและการตลาดได้ ถ้าต้องให้เครือข่ายเสมือนจริงแต่ละเครือข่ายสามารถสื่อสารระหว่างกันได้ จะเป็นต้องอาศัยตัวกลางที่ทำหน้าที่เชื่อมต่อเครือข่ายเสมือนจริงเข้าด้วยกัน เรียกว่า เกตเวย์ (Gateway) จากรูปที่ 2.17 อุปกรณ์เกตเวย์ คือ เร��เตอร์ ที่ติดตั้งบนชั้นที่ 2)

ข้อดีของการใช้เครือข่ายเสมือนจริง

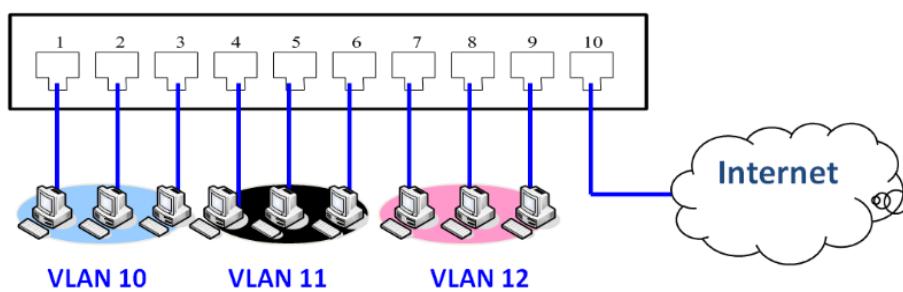
1. สามารถป้องกันปัญหาการบอร์ดคลาส (Broadcast) ข้อมูลให้กระจายไปทั่วทั้งเครือข่าย
2. สามารถจำกัดข้อมูลการจราจร (Traffic) ให้อยู่ในบริเวณที่สามารถควบคุมได้
3. การรักษาความปลอดภัย เนื่องจากการสร้างเครือข่ายเสมือนจริงทำให้อุปกรณ์ที่อยู่ต่างเครือข่ายไม่สามารถสื่อสารกันได้ (เมื่อสื่อสารกันไม่ได้ ก็ไม่สามารถโจมตีกันได้)
3. สามารถกำหนดขอบเขตการแพร่กระจายข้อมูลเฉพาะกลุ่มได้ (Multicast)
4. สามารถเพิ่มประสิทธิภาพการทำงานของเครือข่าย เนื่องจากเครือข่ายเสมือนจริงสามารถลดปัญหาของบอร์ดคลาส ส่งผลให้สื่อนำสัญญาณว่า ดังนั้นข้อมูลจึงสามารถสื่อสารกันได้เพิ่มมากขึ้น

4.1 ชนิดของเครือข่ายเสมือนจริง

เครือข่ายเสมือนจริงแบ่งออกได้หลายประเภทขึ้นอยู่กับชนิดของอุปกรณ์กระจายสัญญาณ ลักษณะของงาน และการจัดโครงแบบของเครือข่าย (Configuration) เป็นต้น สำหรับรูปแบบที่ไปที่นิยมใช้สำหรับจำแนกรูปแบบเครือข่ายเสมือนจริง ดังต่อไปนี้

1) เครือข่ายเสมือนจริงชนิด Port-Based

เป็นการจัดแบ่งเครือข่ายเสมือนจริง โดยอาศัยหมายเลขพอร์ตเป็นหลัก คือ กำหนดว่าในอุปกรณ์กระจายสัญญาณแต่ละตัวมีเครือข่ายเสมือนจริงจำนวนเท่าใด มีชื่ออะไรบ้าง และต้องการให้พอร์ตใดเป็นสมาชิกของเครือข่ายเสมือนจริงใด เป็นต้น ดังรูปที่ 2.18 [25, 34]



รูปที่ 2.18 แสดงเครือข่ายเสมือนจริงชนิด Port-Based

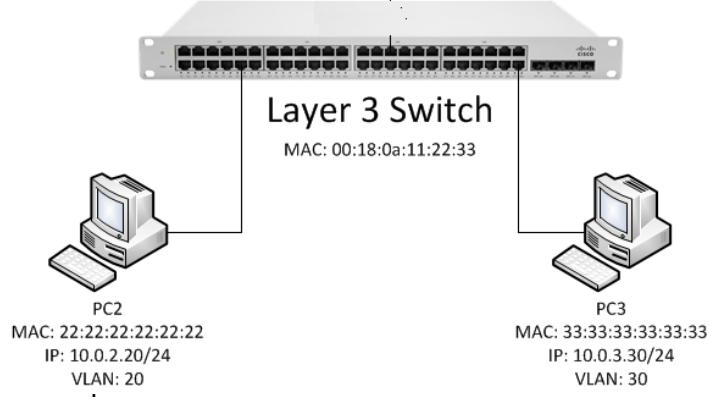
ขั้นตอนในการติดตั้งเครือข่ายเสมือนจริงชนิด Port-Based สามารถดำเนินการโดยมีขั้นตอนคร่าว ๆ ดังนี้

1. กำหนด VTP Domain เป็นอันดับแรก
2. กำหนดชื่อของเครือข่ายเสมือนจริง รวมทั้งเลขหมายของเครือข่าย
3. กำหนดหมายเลขพอร์ตให้กับเครือข่ายเสมือนจริงในแต่ละวงที่ถูกสร้างขึ้น

ข้อดีของเครือข่ายเสมือนชนิด Port-Based คือ สามารถย้ายกลุ่มจากเครือข่ายเสมือนจริงหนึ่งไปอีกเครือข่ายเสมือนจริงหนึ่งได้โดยง่าย ตัวอย่างเช่น สมมุติว่า อุปกรณ์กระจายสัญญาณตัวหนึ่งซึ่งมีจำนวนพอร์ตเท่ากับ 10 พอร์ต และทำการสร้างเครือข่ายเสมือนจริงไว้ 3 วง คือ VLAN 10, VLAN 11 และ VLAN 12 โดย VLAN 10 มีหมายเลขพอร์ตที่เป็นสมาชิก คือ 1, 2 และ 3 สำหรับ VLAN 11 มีพอร์ตสมาชิก คือ 4, 5 และ 6 สำหรับ VLAN 12 มีพอร์ตสมาชิกหมายเลข 7, 8 และ 9 ตามลำดับ ถ้าต้องการให้อุปกรณ์สื่อสารซึ่งเป็นสมาชิกของ VLAN 10 ไปเป็นสมาชิกของ VLAN 11 สามารถทำได้โดยปรับแต่งค่อนพิกัดเรซั่นผ่านซอฟต์แวร์บริหารจัดการบนอุปกรณ์กระจายสัญญาณ โดยเปลี่ยนความเป็นสมาชิกจาก VLAN 10 เป็น VLAN 11 เท่านั้น หรือสามารถปรับแต่งทางกายภาพโดยการย้ายอุปกรณ์สื่อสารจากพอร์ตที่เป็นสมาชิกบน VLAN 10 ไปยังพอร์ตที่เป็นสมาชิกของ VLAN 11 (เปลี่ยนจากพอร์ตหมายเลขระหว่าง 1 ถึง 3 บน VLAN 10 ไปยังพอร์ตหมายเลข 4 ถึง 6 บน VLAN 11)

2) เครือข่ายเสมือนจริงชนิด MAC Address-Based

เครือข่ายเสมือนจริง MAC Address-Based [25, 34] คือ การสร้างเครือข่ายเสมือนจริงโดยอาศัยที่อยู่ทางกายภาพ (MAC Address) ของอุปกรณ์เครือข่ายเป็นหลัก ซึ่งที่อยู่ทางกายภาพนี้ เป็นหมายเลขฐานสิบหกบิต 48 บิต ที่ไม่ซ้ำกัน โดยถูกกำหนดไว้กับเน็ตเวิร์คการดของอุปกรณ์ เครือข่ายทุก ๆ ตัว การแบ่งเครือข่ายเสมือนจริงด้วยวิธีการทางกายภาพนี้ง่ายต่อการปรับแต่งค่อนพิกัดเรซั่น (Configuration) มากร เนื่องจากไม่จำเป็นต้องกำหนดเลขหมายของพอร์ต และไม่ต้องสนใจว่า อุปกรณ์เครือข่ายจะติดตั้งอยู่บนพอร์ตหมายเลขใด และไม่ต้องกลัวว่าจะมีผู้ใดย้ายพอร์ตเพื่อเปลี่ยนวงของเครือข่ายเสมือน เนื่องจาก ไม่ว่าจะย้ายไปอยู่ที่ใด บนอุปกรณ์กระจายสัญญาณตัวใด ตราบใดที่ กำหนดที่อยู่ทางกายภาพให้กับเครือข่ายเสมือนจริงแล้ว จะเปลี่ยนแปลงสมาชิกของเครือข่ายเสมือน ได้ ก็ต่อเมื่อ มีการเปลี่ยนเน็ตเวิร์คการดเท่านั้น ดังรูปที่ 2.19



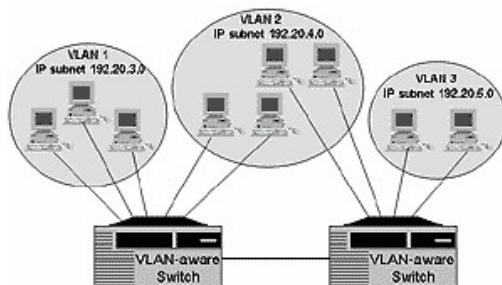
รูปที่ 2.19 ตัวอย่างเครือข่ายชนิด MAC Address-Based

ข้อจำกัดของ MAC Address-Based VLAN

พอร์ตที่ใช้งานร่วมกับ MAC-Based VLAN นั้นจะต้องไม่เป็น Static VLAN หมายความว่า จะต้องไม่มีการกำหนดหมายเลขพอร์ตที่ติดตัวให้กับ VLAN ได ๆ MAC Based VLAN ถูกออกแบบมาให้สามารถรองรับจำนวนผู้ใช้งาน 1 คน (Client) ต่อหนึ่งพอร์ตเท่านั้น แต่ปัจจุบันมีสวิตช์บางรุ่น สามารถรองรับจำนวนผู้ใช้งานมากกว่า 1 คน ต่อ 1 พอร์ตได้

3) Subnet-Based VLAN

Subnet-Based VLAN [25, 34] บางครั้งถูกเรียกว่า Layer-3 Based VLAN เป็น VLAN ที่ถูกสร้างขึ้นโดยอาศัยข้อมูลข่าวสารในระดับชั้นเน็ตเวิร์ค (Network Layer: Layer 3) โดยอุปกรณ์สวิตช์จะตรวจสอบข้อมูลไอพีที่ส่วนหัว (Header) ของแพ็กเก็ต ปกติ Subnet-based VLAN จะถูกติดตั้งบนสวิตช์ที่ทำงานในระดับชั้นเน็ตเวิร์คเท่านั้น ขณะที่ VLAN ชนิด MAC Address-Based จะทำงานบนระดับชั้นดาต้าลิงค์ (Data link layer: Layer 2)



รูปที่ 2.20 Subnet-Based VLAN [2]

จากรูป 2.20 แสดงการทำงานของอุปกรณ์สวิตช์ (Layer 3 Switching) เพื่อสร้าง VLAN จำนวน 2 กลุ่ม คือ VLAN 1 และ VLAN 2 จะสังเกตเห็นว่า VLAN ถูกแบ่งออกเป็นส่วน ๆ โดยใช้หมายเลขไอพี มาเป็นตัวกำหนด VLAN ที่ต่างกัน ข้อดีของการจัด VLAN ประเภทดังกล่าวนี้ ได้แก่ ความยืดหยุ่นของการประยุกต์ใช้งาน เนื่องจากสามารถปรับเปลี่ยน VLAN โดยการเปลี่ยนหมายเลขของไอพีเท่านั้น

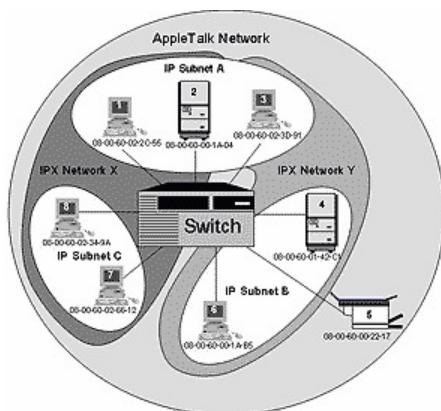
ผู้ใช้งานสามารถเชื่อมต่ออุปกรณ์สื่อสารบนพอร์ตที่มายเลขได้ก็ได้บนอุปกรณ์สวิตช์ โดยไม่มีความจำเป็นต้องแก้ไขคอนฟิกกูเรชันใหม่ วิธี Subnet-Based VLAN เหมาะสำหรับเครือข่ายที่ใช้โปรโทคอลทีซีพี/ไอพี (TCP/IP) เป็นหลัก ค่าใช้จ่ายในการดูแลรักษา VLAN ประเภทนี้ จะถูกกว่า MAC Address-Based มา

ข้อเสียของ Subnet-Based VLAN

ข้อเสียของ Subnet-Based VLAN คือ การสร้างกลุ่มของไอพีหลาย ๆ ชุดบนอุปกรณ์สวิตช์ตัวเดียวกัน ทำให้เกิดความสับสนได้ง่ายกว่าการสร้าง VLAN ประเภทอื่น ๆ รวมทั้งปัญหาของสวิตช์บางรุ่นที่อาจสนับสนุนหลายไอพีเดสบันพอร์ตเดียวกัน

4) Protocol-Based VLAN

รูปแบบของ VLAN ประเภทนี้ช่วยให้การสร้าง VLAN สามารถดำเนินการได้ง่ายกว่าการสร้าง VLAN แบบอื่น ๆ [25, 34] เนื่องจากการสร้าง VLAN ดังกล่าวจะอาศัยโปรโทคอลซึ่งอยู่ในระดับเน็ตเวิร์คเป็นตัวจำแนกการทำงานของแต่ละ VLAN ซึ่งโปรโทคอลเหล่านี้ได้แก่ IP และ IPX เป็นต้น Protocol-Based VLAN ถูกนำมาใช้ในสถานการณ์ที่ระบบเครือข่ายมีอุปกรณ์ที่มีความหลากหลาย โดยเฉพาะอย่างยิ่งมีการใช้งานโปรโทคอลที่แตกต่างกันทำงานอยู่ด้วยกัน หรือในสถานการณ์ที่ระบบเครือข่ายถูกแบ่งออกเป็นหลาย ๆ เซ็กเมนต์ (Segment) เป็นต้น จากรูป 2.21 แสดงรูปแบบการสร้าง Protocol-Based VLAN โดย VLAN 1 จะประกอบไปด้วยพอร์ตที่เป็นสมาชิกหมายเลข 1 ถึง 4 สำหรับประยุกต์ใช้กับโปรโทคอล IP และ VLAN 2 รองรับการทำงานโปรโทคอล IPX โดยมีสมาชิกพอร์ตหมายเลข 5 ถึง 8



รูปที่ 2.21 Protocol-Based VLAN [2]

ข้อดีของการใช้ Protocol-Based VLAN

ข้อดีของ Protocol-Based VLAN ได้แก่ความยืดหยุ่นในการใช้งาน เนื่องจากอุปกรณ์สื่อสารต่าง ๆ ไม่จำเป็นต้องติดตั้งอยู่ในสถานที่เดียวกัน สามารถติดตั้งกระจายอยู่บนเครือข่าย ณ

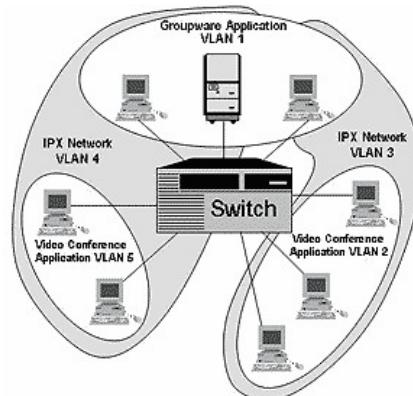
ตำแหน่งใด ๆ ก็ได้ แต่ละอุปกรณ์จะอยู่ภายใต้ VLAN เดียวกัน โดยอาศัยโพรโทคอลในการเชื่อมโยง อุปกรณ์เข้าด้วยกันแทน

ข้อเสียของการใช้ Protocol-Based VLAN

เครื่องคอมพิวเตอร์อาจติดตั้งและใช้งานโพรโทคอลหลายประเภทบนเครื่องเดียวกัน เช่น มีการใช้งาน IP กับ NetBIOS ร่วมกัน ซึ่งอาจจะส่งผลให้อุปกรณ์ที่ควบคุมดูแล VLAN ทำงานหนักมากกว่าปกติและอุปกรณ์ที่ควบคุม VLAN ต้องมีคุณภาพสูงและมีราคาแพงตามไปด้วย

5) Application-Based VLAN

Application-Based VLAN [25, 34] จะอาศัยประเภทของโปรแกรมประยุกต์ในการจัดกลุ่มความเป็นสมาชิกของแต่ละ VLAN ตัวอย่างเช่น โปรแกรมการประชุมทางไกล (Video conference) จะถูกจัดให้อยู่กลุ่มเดียวกัน (แสดงในรูปที่ 2.22) โดยอุปกรณ์คอมพิวเตอร์ที่สื่อสารกันจะอยู่บนเครือข่ายส่วนใหม่ก็ได้ ไม่จำเป็นต้องติดตั้งอยู่บนอุปกรณ์สวิตช์ตัวเดียวกัน จุดประสงค์ของการแยก VLAN โดยอาศัยโปรแกรมประยุกต์นี้ เพื่อเป็นการเอื้อประโยชน์ให้กับโปรแกรมประยุกต์แต่ละประเภท ให้สามารถใช้แบนด์วิชท์ได้อย่างเต็มประสิทธิภาพ อีกทั้งยังสามารถแยกประเภทของงานออกได้อย่างชัดเจน Application-Based VLAN จึงมีประโยชน์สำหรับหน่วยงานที่ต้องการใช้งานโปรแกรมประยุกต์ที่มีลักษณะจำเพาะเจาะจง ปัจจุบันอุปกรณ์ที่สนับสนุน VLAN ประเภท Application-Based VLAN ไม่เป็นที่นิยมใช้งาน เนื่องจากมีราคาแพง

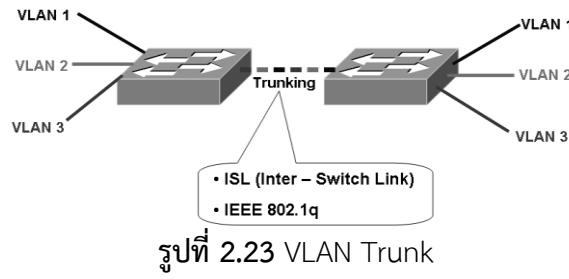


รูปที่ 2.22 Protocol-Based VLAN [2]

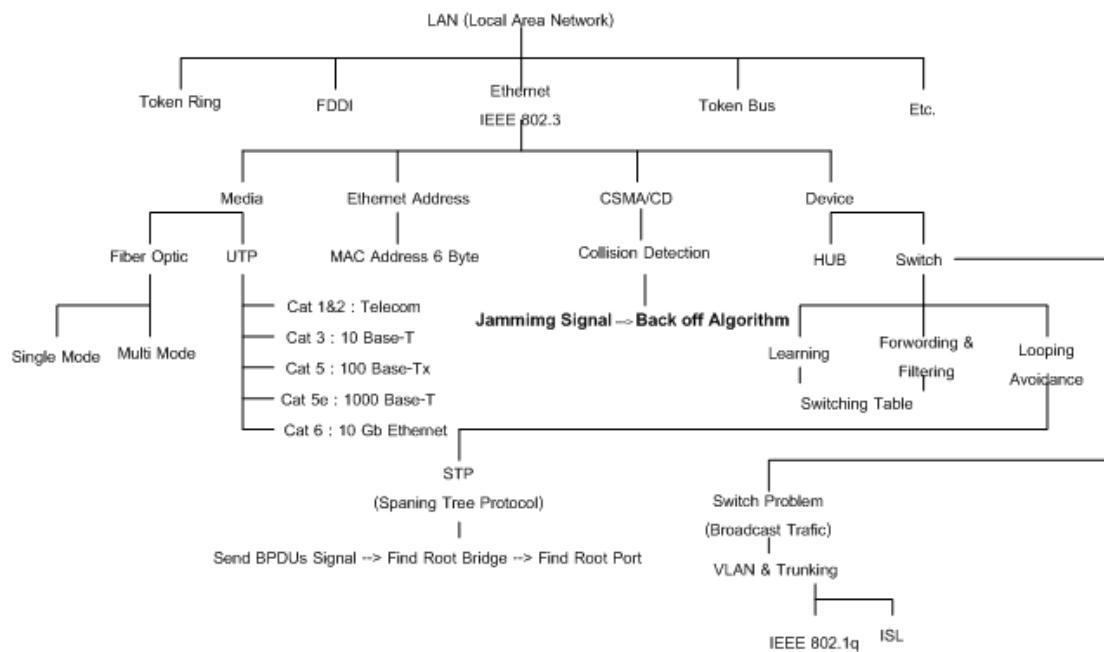
ข้อเสีย คือ ถ้าบางองค์กรใช้โปรแกรมประยุกต์ที่มีความหลากหลายและโปรแกรมประยุกต์เหล่านั้นมีปริมาณการใช้งานที่ต่างกันมาก จะส่งผลให้ขนาดของ VLAN มีขนาดใหญ่ไม่สมดุลกับ VLAN อื่น ๆ ทำให้เกิดปัญหาในเรื่องของบรอดแคสต์ได้

6) VLAN Trunk

VLAN Trunk มีหน้าที่เชื่อมต่อ Switching Hub ที่ติดตั้ง VLAN 2 Hub เข้าด้วยกัน และที่สำคัญได้แก่ การเชื่อมต่อ Switching Hub ที่ติดตั้ง VLAN กระจายไปตาม Hub ต่างๆ เหล่านี้ หลังจากที่เชื่อมต่อ Switches พร้อม VLAN เข้าด้วยกันแล้ว จะสามารถส่งผ่านสมาชิกของ VLAN หนึ่งจาก Switches Hub หนึ่ง ไปยัง Switches อีกหนึ่ง ภายใต้ VLAN ชุดเดียวกัน ด้วยสายสัญญาณ ที่เชื่อมต่อเป็น Trunk เพียงเส้นเดียว ไม่ว่า Switches Hub ตัวแรกจะมี VLAN กี่ชุดก็ตาม แต่เมื่อต้องการสื่อสารกับสมาชิก VLAN ซึ่งเดียวกัน แต่อยู่ต่าง Hub กัน ก็สามารถทำได้อย่างง่ายดาย ด้วย สายสัญญาณเพียงเส้นเดียว และนี่คือประโยชน์ของการใช้ VLAN Trunk ดังรูปที่ 2.23



รูปที่ 2.24 แสดงพัฒนาการของเทคโนโลยีเครือข่าย LAN ตั้งแต่อดีตจนถึงปัจจุบัน



รูปที่ 2.24 แสดงพัฒนาการของเทคโนโลยีเครือข่าย LAN

อุปกรณ์กระจายสัญญาณระดับผู้ใช้งาน (Access Layer)

ในส่วนของเลเยอร์ผู้ใช้งานนั้นไม่มีความซับซ้อนมากนัก โดยส่วนประกอบหลัก ๆ ที่จะต้องมีเมื่อต้องการต่อเข้ากับระบบเครือข่ายคือ

- เครื่องคอมพิวเตอร์ (PC) โดยต้องการมีการเซ็ตหมายเลขของไอพี ในกรณีที่ระบบไม่มีการใช้งาน DHCP คำสั่งเบื้องต้นที่ใช้สำหรับตรวจสอบหมายเลขไอพี คือคำสั่ง ipconfig ดังรูปที่ 2.25

```
C:\> Copyright 1985-2001 Microsoft Corp.
C:\Documents and Settings\Administrator>ipconfig /all
Windows IP Configuration

Host Name . . . . . : ken
Primary Dns Suffix . . . . . :
Node Type . . . . . : Unknown
IP Routing Enabled . . . . . : No
WINS Proxy Enabled . . . . . : No

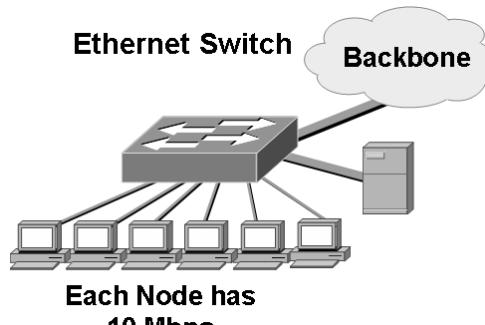
Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . . . . . :
Description . . . . . : Intel(R) PRO/1000 MT Mobile Connecti
on
Physical Address . . . . . : 00-0D-60-77-A6-1E
Dhcp Enabled . . . . . : No
IP Address . . . . . : 10.1.4.50
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.1.4.254
DNS Servers . . . . . : 202.28.32.1

C:\Documents and Settings\Administrator>
```

รูปที่ 2.25 แสดงการใช้คำสั่ง ipconfig

- เครื่องที่ต้องการเชื่อมต่อเข้าระบบเครือข่ายต้องมีกราดแลน (NIC Card)
- ทำการต่อเข้ากับอุปกรณ์เครือข่ายประเภท ยับ หรือสวิตช์ เพื่อเข้าสู่ระบบเครือข่ายหลัก ดังรูปที่ 2.26



รูปที่ 2.26 การเชื่อมต่อคอมพิวเตอร์เข้าสู่เครือข่าย

แบบฝึกหัดท้ายบท

- ประเภทของการเชื่อมต่อเครือข่าย (categories of topology) มีกี่ประเภท อะไรบ้าง
- ข้อดีและข้อเสียของการเชื่อมต่อเครือข่ายแบบ mesh คืออะไร
- การเชื่อมต่อแบบสตาร์ มีลักษณะเป็นอย่างไร และมีข้อดีข้อเสียอย่างไร
- การเชื่อมต่อแบบบัส ใช้ในงานในลักษณะใด และทำไม่จึงต้องใช้ลักษณะดังกล่าว

5. ในการแก้ปัญหาสำหรับการเชื่อมต่อแบบวงแหวน ในกรณีที่สายนำสัญญาณขัดข้อง สามารถทำได้อย่างไร จงอธิบาย
6. ประเภทของระบบเครือข่ายแบ่งได้กี่ประเภท อะไรบ้าง
7. ประเภทการเชื่อมต่อแบบ LAN มีลักษณะเป็นอย่างไร และสนับสนุนการสื่อสารข้อมูลในลักษณะใด
8. ประเภทการเชื่อมต่อแบบ MAN มีลักษณะเป็นอย่างไร และสนับสนุนการสื่อสารข้อมูลในลักษณะใด
9. ประเภทการเชื่อมต่อแบบ WAN มีลักษณะเป็นอย่างไร และสนับสนุนการสื่อสารข้อมูลในลักษณะใด
10. การเข้าถึงระบบเครือข่ายมี 3 ระดับประกอบไปด้วยอะไรบ้าง และแต่ละประเภทแตกต่างกันอย่างไร

บทที่ 3

อุปกรณ์เครือข่ายและการคอนฟิกกูเรชัน (Networking Devices and Configuration)



- Connecting Devices
- Command line interface
- System startup
- Managing system files
- Using show commands

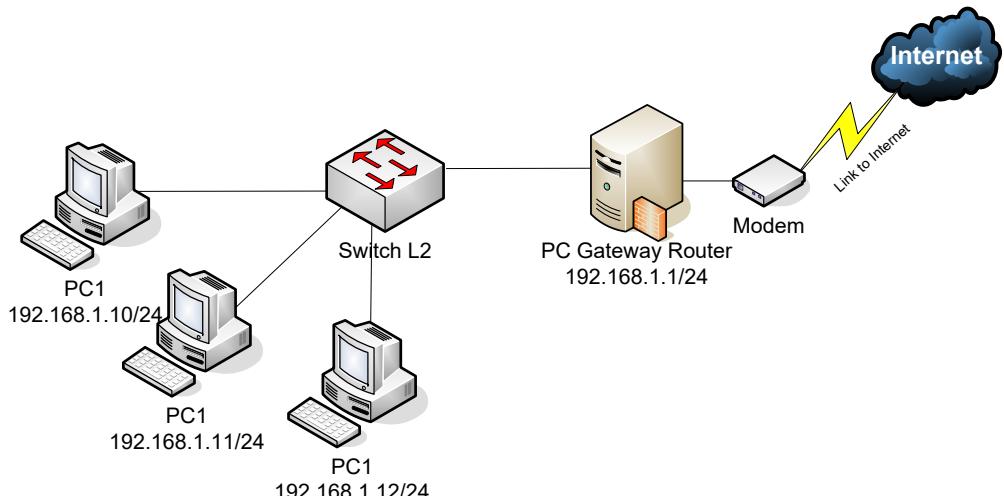
แนวคิด

การทำความเข้าใจถึงหน้าที่การทำงาน คุณลักษณะ ความสามารถต่างๆ ของอุปกรณ์เครือข่ายในแต่ละส่วนว่าเป็นอย่างไร (โดยเฉพาะเราเตอร์ และสวิชต์) จะทำให้ผู้ดูแลระบบสามารถกำหนดโครงสร้างและการทำงานของระบบเครือข่ายโดยรวมได้เป็นอย่างดี

วัตถุประสงค์

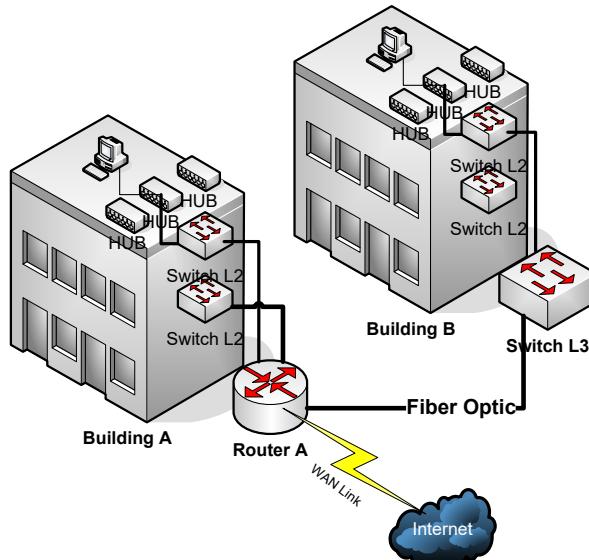
1. เพื่อให้ทราบถึงหน้าที่การทำงานของอุปกรณ์ที่ทำงานอยู่บนระบบเครือข่าย
2. เพื่อให้ทราบถึงประเภทของอุปกรณ์ ตำแหน่งที่ใช้สำหรับจัดวางอุปกรณ์บนระบบเครือข่าย ส่วนประกอบของอุปกรณ์แต่ละชนิด
3. เพื่อให้ทราบถึงโครงสร้างการทำงานของอาร์ดแวร์ในอุปกรณ์ระบบเครือข่าย
4. เพื่อให้ทราบถึงวิธีการติดตั้งและคอนฟิกอุปกรณ์ที่ทำหน้าที่สำคัญๆ บนระบบเครือข่าย

การแบ่งประเภทของอุปกรณ์ที่ทำงานอยู่บนระบบเครือข่ายสามารถแบ่งตามความสามารถในการส่งถ่ายปริมาณของข้อมูล ได้แก่ ชนิดที่สามารถส่งข้อมูลได้ในปริมาณที่มาก ๆ ควรจะทำการติดตั้งไว้สำหรับเป็นแกนหลักของระบบ (Core Layer) อุปกรณ์ที่ทำหน้าที่ในส่วนนี้ มักจะเป็นเราเตอร์ (Router) หรืออาจจะเป็นสวิตช์ที่สามารถทำงานได้ในเลเยอร์ที่สามได้ (Switch L3) หรือถ้าหน่วยงานที่มีงบประมาณที่จำกัดอาจจะมีเพียงสวิตช์เลเยอร์สองต่อเท้ากับคอมพิวเตอร์ที่ติดตั้งซอฟท์แวร์เพื่อทำหน้าที่ทางสื่อสารได้ เช่น ใช้ซอฟท์แวร์เราเตอร์ รูปที่ 3.1 แสดงลักษณะการเชื่อมต่อประเภทนี้



รูปที่ 3.1 แสดงการเชื่อมต่อโดยใช้ PC เป็นอุปกรณ์เลเยอร์สาม

ส่วนในองค์กรที่มีขนาดปานกลางถึงใหญ่ และมีงบประมาณในการจัดซื้ออุปกรณ์คันหนาเส้นทางได้ ก็ต้องมีการวางแผนในเบื้องต้นก่อนว่าจะออกแบบแกนหลักของเครือข่ายอย่างไร เช่น ติดตั้งอุปกรณ์เหล่านี้ตามพื้นที่ของอาคาร หรือติดตั้งตามหน้าที่การทำงานของผู้ใช้ เช่น แผนกการเงิน ก็ควรจะอยู่ใน VLAN (การสร้าง LAN ขึ้นมาโดยไม่ขึ้นกับทางกายภาพ) เดียวกัน ขึ้นอยู่กับความเหมาะสมสมกับลักษณะของปัญหาในแต่ละที่ แต่ละแห่ง จุดนี้ไม่สามารถที่จะกำหนดลงได้ ว่าการวางแผนอุปกรณ์แบบดังจะดีกว่ากัน แต่จากประสบการณ์ของผู้เขียน มีความเห็นส่วนตัวว่า การวางแผนตามลักษณะทางกายภาพของอาคารหรือตึก ดังรูปที่ 3.2 จะทำได้สะดวกมีแบบแผนและดูแลรักษาได้ง่าย ไม่สับสน เนื่องจากว่าอาคารต่าง ๆ มีลักษณะที่ไม่มีการเปลี่ยนแปลงเลยตลอดอายุการใช้งานของมัน และต้องวางผังหรือเขียน Network Diagram ออกมาราบกวนกำหนดได้อย่างชัดเจนว่า ตึกแต่ละตึกมีหมายเลขประจำตึกเป็นหมายเลขออะไร ชั้นที่เท่าไหร่ ซึ่งในตอนหลังจะสะดวกเป็นอย่างมากในการทำ VLAN



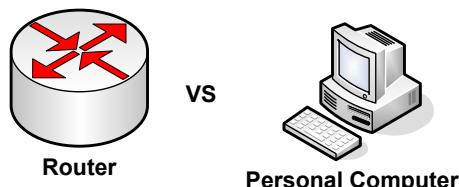
รูปที่ 3.2 การวางแผนอุปกรณ์หลักตามลักษณะทางกายภาพ

อุปกรณ์กระจายสัญญาณหลัก (Core Layer)

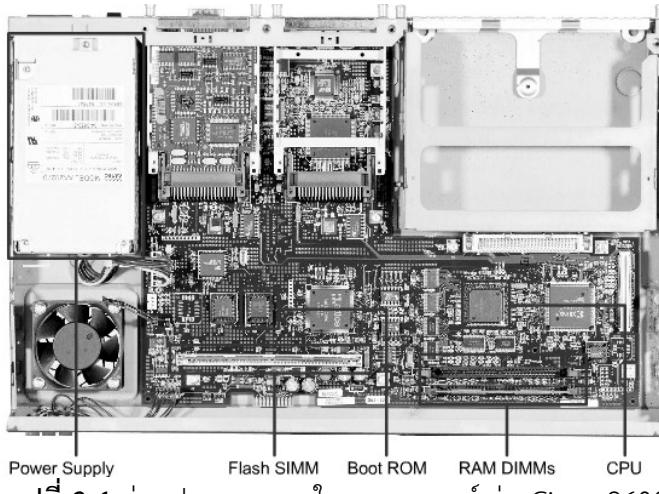
จากหลักการที่กล่าวมาแล้วว่า อุปกรณ์ที่ทำหน้าที่ใน Core Layer นั้นจะต้องโอนถ่ายข้อมูลได้ในปริมาณที่มากและรวดเร็ว พร้อมทั้งต้องฉลาดในการหาเส้นทางที่ดีที่สุดและสั้นที่สุดด้วย ซึ่งหนึ่งในพันอุปกรณ์ที่เรียกว่า "เราเตอร์" ในที่นี้จะกล่าวถึงเราเตอร์ของบริษัทซิสโก้ (Cisco) เป็นหลักเนื่องจากเป็นอุปกรณ์ที่ค่อนข้างจะได้รับความนิยมเป็นอย่างกว้างขวางและมีประสิทธิภาพในการทำงานที่มีเสถียรภาพเป็นอย่างมาก ซึ่งเมื่อเข้าใช้งานในอุปกรณ์ของบริษัทนี้แล้วก็ไม่เป็นภาระที่จะเรียนรู้และทำความเข้าใจกับอุปกรณ์ของบริษัทอื่น ๆ และที่สำคัญคือสามารถเชื่อมต่อที่ใช้งานบนอุปกรณ์เราเตอร์หรือสวิตช์ของบริษัทนี้นั้น เข้าใจได้่าย และมี Help ช่วยเหลือ ทำให้ผู้ดูแลระบบที่ไม่สามารถจำคำสั่งได้ทั้งหมด ก็ไม่จำเป็นต้องกังวลมากนัก ซึ่งการใช้งานคำสั่งก็จะกล่าวในลำดับต่อไป

เราเตอร์ (Router)

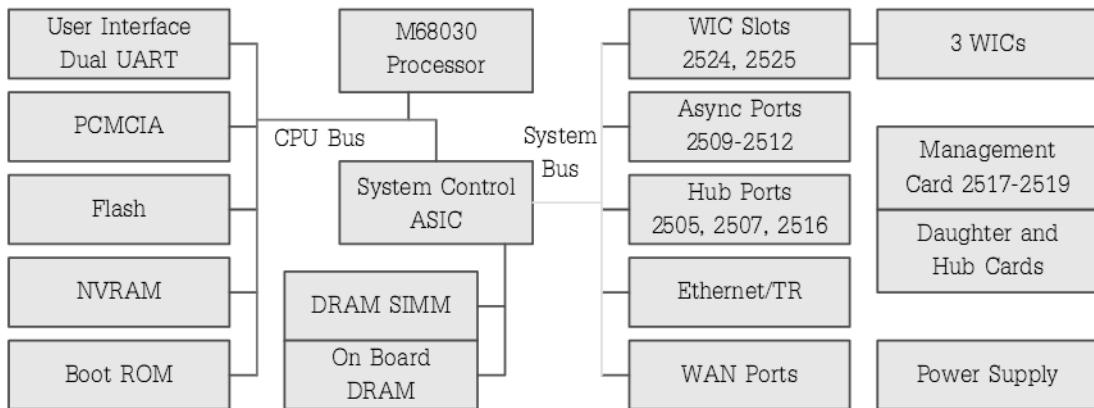
อุปกรณ์ที่ทำหน้าที่ใน Core Layer คือ Router หรือเราเตอร์ ซึ่งมีโครงสร้างของเครื่องคอมพิวเตอร์ที่ใช้งานกันอยู่ทั่วไปนั่นเอง ดังรูปที่ 3.3, 3.4 จะแตกต่างกันไปตรงที่เราเตอร์ถูกสร้างขึ้นมาทำงานเฉพาะทางเพื่อรับการโอนถ่ายข้อมูลให้เร็วที่สุดและค้นหาเส้นทางที่ดีที่สุดเป็นหลัก เราเตอร์ไม่สนใจเรื่องการคำนวณกราฟิก ไม่สนใจการประมวลผลเกมส์ต่าง ๆ ที่ PC ทำอยู่ในปัจจุบัน เราเตอร์สนใจแต่รับข้อมูลมาจากไหนและจะส่งข้อมูลไปทางไหนให้ดีที่สุดเท่านั้น



รูปที่ 3.3 โครงสร้างสถาปัตยกรรมของเราเตอร์คล้ายกับ PC



รูปที่ 3.4 ส่วนประกอบภายในของเราเตอร์ รุ่น Cisco 2600



รูปที่ 3.5 ผังโครงสร้างของเราเตอร์ Cisco

ส่วนประกอบหลัก ๆ ของเราเตอร์จะมีดังนี้ (ดังรูปที่ 3.5)

- ROM ทำหน้าที่เช็คค่าเริ่มต้นให้กับเราเตอร์ (POST) ดังรูปที่ 3.6 เมื่อ่อนกับการทำงานของ Bios ของ PC คือในรอมจะมีระบบปฏิบัติการขนาดเล็ก ๆ คอยจัดการเรื่องการตรวจสอบความพร้อมของฮาร์ดแวร์ เมื่อการทำงานของฮาร์ดแวร์ไม่มีข้อผิดพลาดแล้ว ก็จะส่งการทำงานต่อให้กับ IOS (เป็นระบบปฏิบัติการของบริษัท Cisco สร้างเพื่อทำงานกับเราเตอร์โดยเฉพาะ) ที่เก็บอยู่ใน Flash อีกทีหนึ่ง



- **Flash Memory** เป็นหน่วยความจำแบบกึ่งถาวร ที่กล่าวเข่นี้พราะว่าข้อมูลที่เก็บไว้จะสูญหายไปได้ถ้ามีการปล่อยกระแสไฟฟ้าเข้าไปยังตัวมัน (EPROM Erasable Programmable Read-Only Memory) มันจะทำหน้าที่เก็บระบบปฏิบัติการสำหรับควบคุมการทำงานของเราเตอร์ ดังรูปที่ 3.7



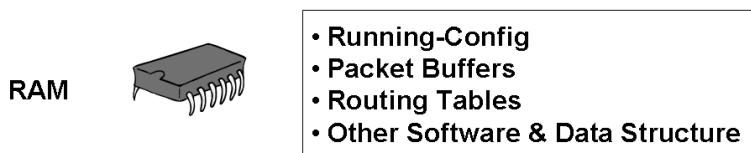
รูปที่ 3.7 Flash

- **NVRAM** (ดังรูปที่ 3.8) ทำหน้าที่เก็บคอนฟิกชันไฟล์ เมื่อ IOS เข้าควบคุมการทำงานของระบบเรียบร้อยแล้วมันจะโหลดคอนฟิกไฟล์ที่เก็บอยู่ใน NVRAM ไปทำงาน เมื่อเกิดการเปลี่ยนแปลงคอนฟิกเมื่อใด ๆ ในหน่วยความจำหลัก (Running Configuration) เราจะต้องใช้คำสั่ง Write Memory หรือ wr เพื่อบันทึกข้อมูลที่เปลี่ยนแปลงไปไว้ยัง NVRAM ด้วยมิเช่นนั้นถ้าเครื่องมีการเริ่มต้นทำงานใหม่ คอนฟิกที่เปลี่ยนแปลงจะหายไปด้วย



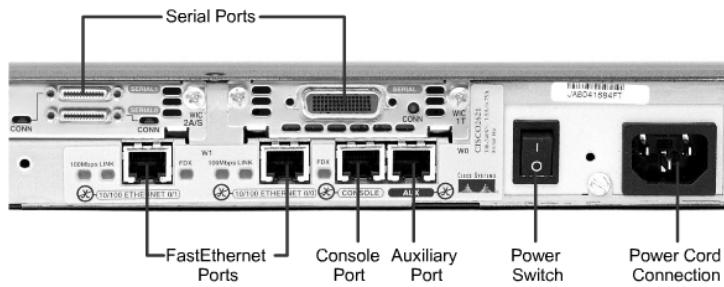
รูปที่ 3.8 NVRAM

- **RAM/DRAM** (ดังรูปที่ 3.9) ทำหน้าที่เป็นหน่วยความจำหลักของเราเตอร์ เก็บคำสั่งที่ทำงานในปัจจุบัน (Running Configuration) ซึ่งคำสั่งเหล่านี้ได้โหลดมาจาก NVRAM มาทำงาน ต่อมาเมื่อมีการเปลี่ยนแปลงคอนฟิกใหม่เพิ่มเติม มันจะเก็บข้อมูลที่เปลี่ยนแปลงไว้ในหน่วยความจำหลักนี้ ถ้าไม่มีการเขียนข้อมูลลง NVRAM ข้อมูลที่เปลี่ยนแปลงก็จะหายไปเมื่อมีการรีเซ็ตเราเตอร์

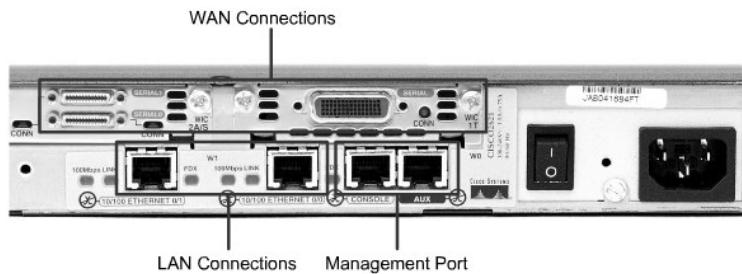


รูปที่ 3.9 หน่วยความจำหลักของเราเตอร์ RAM

- **Router Interface** คือจุดที่ใช้สำหรับเชื่อมต่ออุปกรณ์ภายนอกที่จะเข้ามาต่อกับตัวมัน ซึ่งอินเตอร์เฟสหลัก ๆ มี 2 ชนิดคือ อินเตอร์เฟสที่ใช้เชื่อมต่อกับ WAN Link และอินเตอร์เฟสที่ใช้เชื่อมต่อภายในเครือข่ายท้องถิ่น ดังตัวอย่างรูปที่ 3.10 และ 3.11 อินเตอร์เฟสเครือข่ายท้องถิ่นส่วนมากจะเป็นชนิด Fast Ethernet มีความเร็วที่ประมาณ 100 Mbps การอ้างถึงจะอ้างผ่าน slot และตามด้วยพอร์ต เช่น Fast Ethernet 0/0 (slot 0/port 0) ส่วนขา WAN จะเรียกว่า WIC ซึ่งเราเตอร์รุ่น 2600 จะมี 2 แบบคือ Serial และ BRI การอ้างถึงจะทำคล้าย ๆ Fast Ethernet เช่น Serial 0/0 เป็นต้น



รูปที่ 3.10 โครงสร้างภายนอกของเร้าเตอร์ Cisco รุ่น 2600

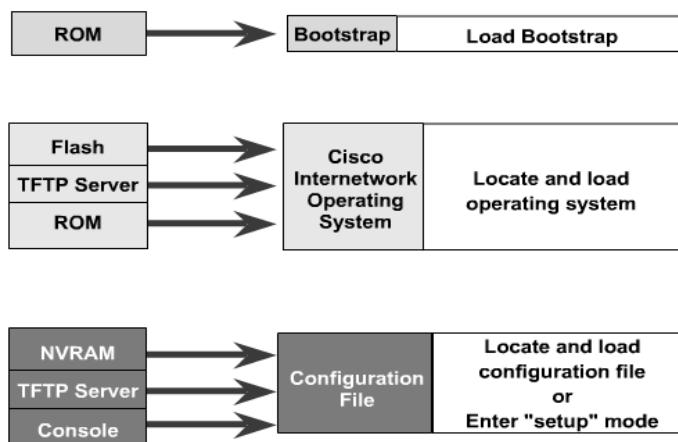


รูปที่ 3.11 ช่องทางสำหรับใช้เชื่อมต่อ กับ WAN และ LAN

- **IOS (Internetworking Operating System)** เป็นระบบปฏิบัติการที่ค่อยควบคุมการทำงานของเร้าเตอร์ พร้อมทั้งสนับสนุนการทำงานของ โอล็อตและการหาเส้นทางแบบต่าง ๆ ด้วย

ลำดับการทำงานของเร้าเตอร์

เริ่มต้นเมื่อทำการเปิด Power ของเร้าเตอร์ กระบวนการจะเริ่มทำงานดังรูปที่ 3.12



รูปที่ 3.12 ลำดับการทำงานของเร้าเตอร์

1. เราเตอร์จะเข้าไปอ่านโค้ดเต็ม ๆ ที่เก็บไว้ใน ROM เรียกชั้นตอนนี้ว่า Bootstrap จากนั้นจะทำการตรวจสอบอาร์ดแวร์ว่า ทำงานได้ถูกต้องหรือไม่ ถ้าไม่จะแสดงข้อผิดพลาดออกมา ก็จะทำข้อ 2 ต่อ
2. เมื่อทำ Bootstrap เสร็จแล้ว ลำดับต่อไปจะอ่านข้อมูลที่อยู่ใน Flash มาทำงานต่อ ซึ่งใน Flash นี้จะเก็บระบบปฏิบัติการตัวสมบูรณ์อยู่ ซึ่งเรียกว่า IOS ซึ่งชื่อของ IOS จะมีสัญลักษณ์บอกความหมายอยู่ เช่น XXXX-YYYY-WW Code XXXX จะบอกว่ามันเป็นรุ่นอะไร เช่น C2600 YYYY จะบอกถึงคุณลักษณะของมัน เช่น รองรับการทำงานแบบ IPX WW บอกถึงรูปแบบของไฟล์ เช่น เป็นอิมเมจที่มีการบีบอัดข้อมูลไว้ (.zip) ดังรูปที่ 3.13

The name has three parts separated by dashes, xxx-yyy-ww:

- xxxx = Platform
- yyyy = Features
- ww = Format - where the image runs and whether it has been zipped or compressed

Name Codes

Platform (Hardware) (Partial list)

c1005	1005
c1600	1600
c1700	1700, 1720, 1750
c2500	25xx, 3xxx, 5100, AO (11.2 and later only)
c2600	2600
c2800	Catalyst 2800
c2900	2910, 2950
c3620	3620

Features (Partial list)

b	Appletalk
boot	boot image
c	CommServer lite (CiscoPro)
drag	IOS based diagnostic images
g	ISDN subset (SNMP, IP, Bridging, ISDN, PPP, IPX, Atalk)
i	IP subset (SNMP, IP, Bridging, WAN, Remote Node, Terminal Services)
n	IPX
q	Async
t	Telco return (12.0)

Format (Where the image runs in the router)

f	flash
m	RAM
r	ROM
l	image will be relocated at run time

Compression Types

z	zip compressed (note lower case)
x	mzip compressed
w	"STAC" compressed

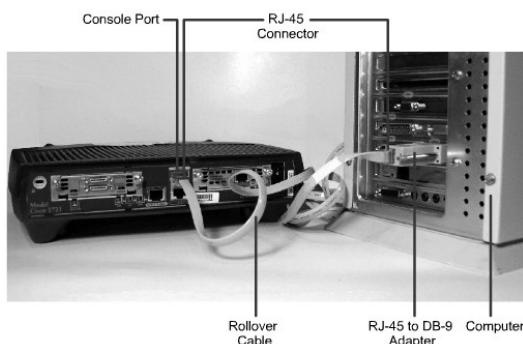
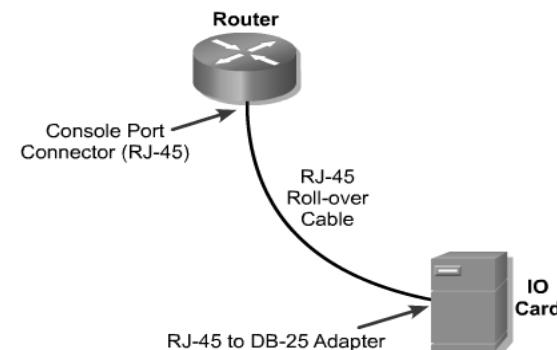
รูปที่ 3.13 แสดงความหมายรหัสของ IOS Image

ในขั้นตอนที่ 2 นี้ยังสามารถโหลด IOS ได้จากแหล่งที่เก็บอื่น ๆ ได้อีก เช่น โหลดจาก ROM ซึ่งจริง ๆ แล้วมันคือ IOS ตัวเล็ก ๆ ที่ไม่มีไฟล์เซอร์อะไรมากนัก หรือจะเลือกโหลด IOS ได้ผ่านทาง TFTP เซิร์ฟเวอร์ก็สามารถทำได้เช่นกัน

3. เมื่อ IOS เข้าควบคุมเราเตอร์โดยสมบูรณ์แล้ว มันจะโหลดคอนฟิกภูเรชันไฟล์จาก NVRAM เข้าไปทำงานในลำดับถัดไป เนื่องจากคอนฟิกภูเรชันไฟล์จะเก็บคำสั่งต่าง ๆ ที่สั่งให้เราเตอร์ ทำอะไรบ้าง เช่น ใช้โพรโทคอลหาเส้นทางแบบ RIP เป็นต้น ในส่วนนี้ก็สามารถทำการโหลด คอนฟิกไฟล์ได้จากแหล่งข้อมูลอื่น ๆ เช่นเดียวกัน เช่น โหลดมาจาก TFTP เซิร์ฟเวอร์ ในทาง กลับกัน เราสามารถเบ็คอับคอนฟิกภูเรชันไฟล์เป็นวิธียังที่ปลดภัยเพื่อไว้กรณี ผลลัพธ์คอน ฟิกไฟล์ทึ่ง ก็สามารถนำคอนฟิกที่เบ็คอับไว้มาราوترต่อได้ทันที
4. เมื่อ IOS โหลดคอนฟิกภูเรชันไฟล์มาทำงานเสร็จแล้ว ก็จะขึ้นข้อความพร้อมที่จะรับคำสั่งให้ ทำงานต่อไป ซึ่งก็เป็นการจบกระบวนการเริ่มต้นการทำงานของเราเตอร์

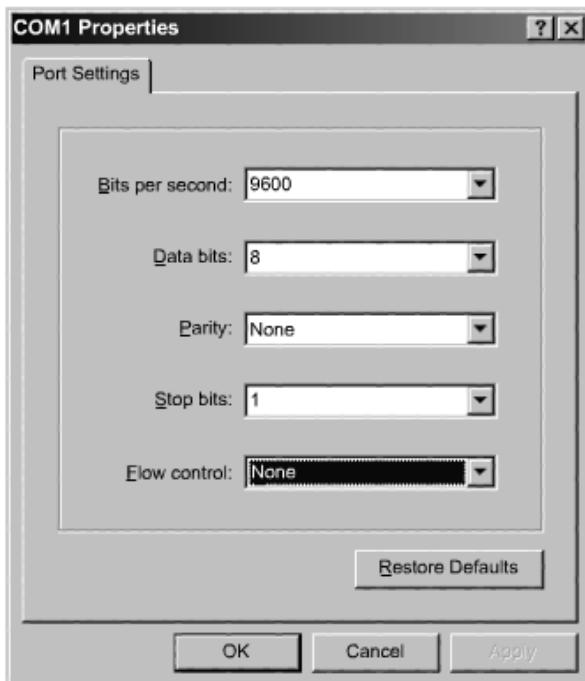
การเชื่อมต่อเพื่อคอนฟิกเราเตอร์ผ่าน孔โนโซล (Console Port)

จากรูปที่ 3.14 แสดงการเชื่อมต่อ PC เข้ากับเราเตอร์เพื่อคอนฟิก ปกติจะเชื่อมต่อโดยผ่าน Serial Port ของ PC แต่ปัจจุบันพอร์ต serial หายากส่วนมากจะเป็น USB ก็จำเป็นจะต้องหา อุปกรณ์ที่แปลงจาก USB เป็น serial จากนั้นก็จะต่อผ่านหัวต่อ RJ-45 to DB9 ด้วยสายชนิด Rollover เข้ากับ Console port ของเราเตอร์ด้วยหัวแบบ RJ-45 ดังรูป 3.14

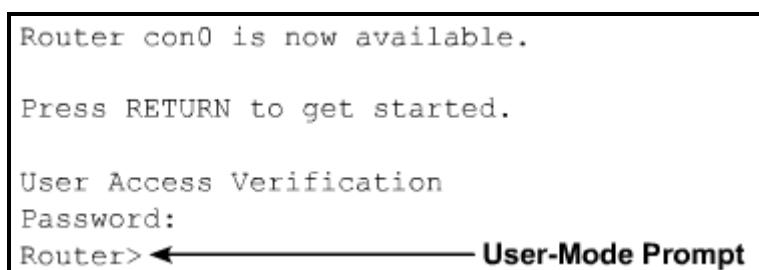


รูปที่ 3.14 แสดงการเชื่อมต่อ PC กับเราเตอร์เพื่อทำการคอนฟิกอุปกรณ์เราเตอร์

เมื่อต่อสายเรียบร้อยแล้ว ให้เปิดเครื่อง PC ส่วนการคอนฟิกจะต้องใช้โปรแกรมประเภท HyperTerminal หรือ SecureCRT และเซ็ตค่าต่าง ๆ ดังรูปที่ 3.15, 3.16



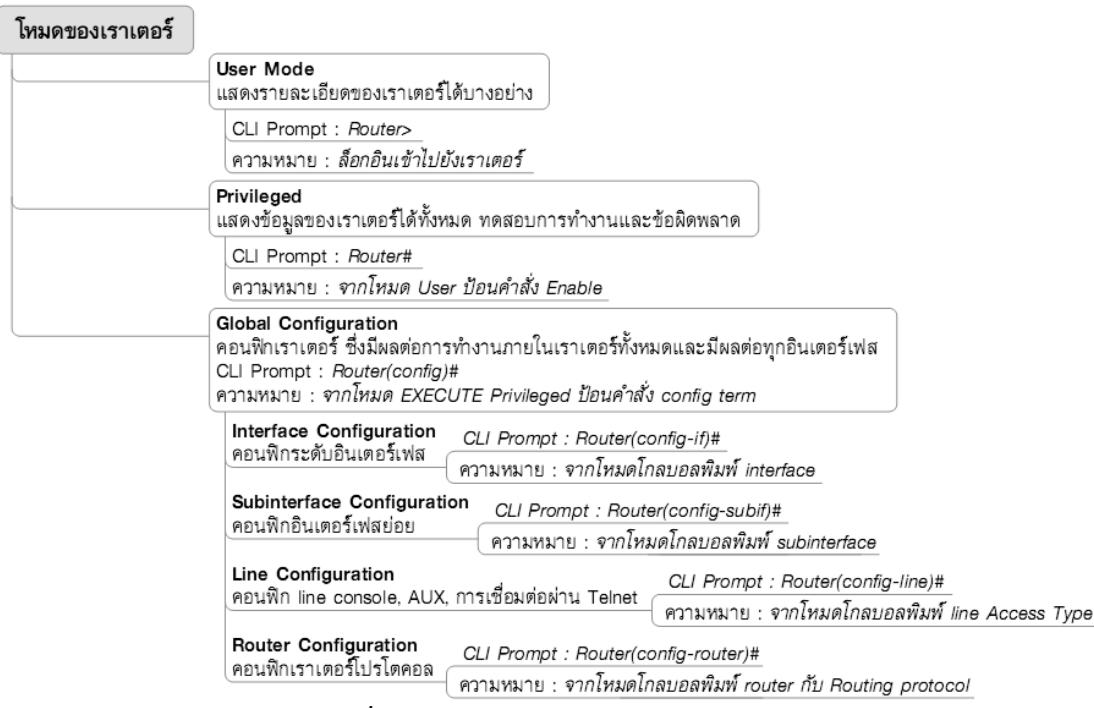
รูปที่ 3.15 การเซ็ตค่าต่าง ๆ เพื่อเชื่อมต่อกับเราเตอร์ผ่านทาง Console Port



รูปที่ 3.16 เราเตอร์พร้อมรับคำสั่ง

โหมดการทำงานของเราเตอร์ (Cisco Router Modes)

เราเตอร์ของ Cisco จะมีโหมดการทำงานหลายโหมด ดังรูป 3.17 เหตุผลที่เป็นเช่นนี้เนื่องจากเรื่องของความปลอดภัยเป็นหลัก บางโหมดจะยอมให้ผู้ใช้งานสามารถคอนฟิกได้ทุก ๆ อย่าง บางโหมดก็ยอมให้ทำงานได้บางอย่างเท่านั้น ในทวีชั่นนี้จะกล่าวถึงการทำงานของโหมดต่าง ๆ ที่มีอยู่ในเราเตอร์ เมื่อเราเตอร์พร้อมที่จะรับคำสั่งจะแสดง Prompt ให้ผู้ใช้ป้อนคำสั่ง ลักษณะคล้าย ๆ กับ Shell ของระบบปฏิบัติการยูนิกซ์หรือลีนุกซ์ แต่ Cisco เรียกว่า CLI (Command Line Interface) คือการประมวลผลคำสั่งจากผู้ใช้งานผ่านทาง "การคีย์ข้อมูลเป็นลักษณะบรรทัดต่อบรรทัด" คำสั่งที่ป้อนเข้าไปให้กับเราเตอร์จะมีผลทันที นั่นคือเราเตอร์จะประมวลผลคำสั่งทันทีเมื่อมีการกดคีย์ Enter



รูปที่ 3.17 โหมดการทำงานของเราเตอร์

User Mode

เป็นโหมดที่ระบบให้สิทธิ์บางอย่างเท่านั้น เช่น ดูข้อมูลของแต่ละอินเตอร์เฟส ปริมาณข้อมูลที่ส่งและรับ การใช้งานโหมดนี้จะทำงานได้จำกัดเท่าที่จำเป็นเท่านั้น เมื่อเข้าสู่การทำงานของโหมดนี้ สังเกตุง่าย ๆ ได้จาก Prompt จะมีลักษณะเป็น **Router>**

Privileged Mode

เป็นโหมดที่สามารถทำงานได้สูงมากขึ้นคือ สามารถแสดงข้อมูลของทุก ๆ อินเตอร์เฟส และรัน ning คุณพิก แสดงปริมาณการรับส่งข้อมูล ปริมาณการใช้งานของ CPU หน่วยความจำ โปรเซสที่ทำงานอยู่ และสามารถตรวจสอบข้อผิดพลาดของข้อมูลได้ว่าเกิดจากสาเหตุใด วิธีที่จะเข้ามาทำงานในโหมดนี้โดยการพิมพ์คำสั่ง enable จากโหมดของ User เมื่อเข้าสู่การทำงานของโหมดนี้สังเกตุง่าย ๆ ได้จาก Prompt จะมีลักษณะเป็น **Router#**

ตัวอย่าง **Router>enable ↲ Enter**

Router#

Global Configuration Mode

ในโหมด Privileged นี้ไม่สามารถทำการคุณพิกให้เราเตอร์สามารถทำงานได้ทั้งหมดทุกอย่าง จะทำได้เฉพาะบางอย่าง เช่น เพิ่ม达้าเบสของ VLAN เป็นต้น แต่สำหรับโหมดนี้จะสามารถคุณพิกเราเตอร์ได้เพิ่มขึ้น เช่น การสร้าง user การเข็มรหัสผ่าน เป็นต้น การเข้าสู่การทำงานของโหมดนี้ทำ

ได้ด้วยการพิมพ์ configuration terminal ที่โหมด Privileged เมื่อเข้าสู่การทำงานของโหมดนี้ สังเกตุง่าย ๆ ได้จาก Prompt จะมีลักษณะเป็น Router(config)#

ตัวอย่าง Router#configuration terminal ← Enter

Router(config)#

Interface Configuration Mode

เป็นโหมดการทำงานเสริมเพิ่มขึ้นจากโหมดโภบล สำหรับคอนฟิกเราเตอร์ในส่วนที่เกี่ยวข้อง กับอินเตอร์เฟส เช่น การกำหนดไอพีแอดเดรสให้แต่ละอินเตอร์เฟส การกำหนดความเร็วในการรับส่ง ข้อมูล การกำหนด ACL (ทำคล้าย ๆ ไฟล์วอล์ชของเราเตอร์) เป็นต้น การเข้าสู่โหมดนี้ทำได้ด้วยการ พิมพ์ Interface TYPE Slot/Port เช่น interface serial 0/1 ที่โหมด Configuration เมื่อเข้าสู่การ ทำงานของโหมดนี้สังเกตุง่าย ๆ ได้จาก Prompt จะมีลักษณะเป็น Router(config-if)#

ตัวอย่าง Router(config)#interface serial 0/0 ← Enter

Router(config-if)#

Subinterface Configuration Mode

ทำงานคล้ายกับโหมดของ Interface Configuration แต่ Cisco เราเตอร์จะยอมให้ในแต่ละ อินเตอร์เฟสสามารถทำอินเตอร์เฟสซ้อนเข้าไปได้อีก เช่น

Interface serial 0/0 ← Interface

Interface serial 0/0.100 ← subinterface

การเข้าสู่โหมดนี้ทำได้ด้วยการพิมพ์ Interface TYPE Slot/Subinterface จากโหมด Configuration เมื่อเข้าสู่การทำงานของโหมดนี้สังเกตุง่าย ๆ ได้จาก Prompt จะมีลักษณะเป็น Router(config-subif)#

ตัวอย่าง Router(config)# interface serial 0/0.100 ← Enter

Router(config-subif)#

Line Configuration Mode

เป็นโหมดที่ทำงานเกี่ยวกับการสร้างช่องทางสำหรับเข้ามาคอนฟิกตัวอุปกรณ์เราเตอร์ เช่น การคอนฟิกคอนโซล การคอนฟิกเกี่ยวกับ Telnet เป็นต้น การเข้าสู่โหมดนี้ทำได้ด้วยการพิมพ์ line VTY 0 4 (กรณี Telnet) จากโหมด Configuration เมื่อเข้าสู่การทำงานของโหมดนี้สังเกตุง่าย ๆ ได้ จาก Prompt จะมีลักษณะเป็น Router(config-line)#

ตัวอย่าง Router(config)# line VTY 0 4 ← Enter

Router(config-line)#

Router Configuration Mode

ทำหน้าที่หลักคือการคอนฟิกโพร็อโทคอลสำหรับทางเส้นทาง (Routing Protocol) เช่น rip igrp eigrp ospf bgp เป็นต้น การเข้าสู่โหมดนี้ทำได้ด้วยการพิมพ์ Router PROTOCOL_TYPE จากโหมด Configuration เมื่อเข้าสู่การทำงานของโหมดนี้สังเกตุง่าย ๆ ได้จาก Prompt จะมีลักษณะเป็น **Router(config-protocol)#**

ตัวอย่าง **Router(config)# router** **Enter**
Router(config-router)# rip **Enter**

TIPS : เมื่อต้องการออกจาก Configuration โหมดใดโหมดหนึ่งเพียง 1 ชั้นให้ใช้คำสั่ง exit แต่ถ้าต้องการออกจากโหมด Privileged ให้ใช้คำสั่ง end

TIPS: ถ้าจำคำสั่งของเราเตอร์ไม่ได้ให้ใช้คำสั่ง ? เช่น rou?

TIPS: คำสั่งของเราเตอร์สามารถเขียนแบบย่อ ๆ ก็ได้ ถ้าคำนั้นไม่ซ้ำกับคำอื่น ๆ เช่น config term (configuration terminal)

นอกจากโหมดการทำงานที่กล่าวมาทั้งหมดแล้วยังมีโหมดที่ถูกชื่อนว่าอีก 3 โหมดเพื่อใช้สำหรับทำงานบางอย่างที่พิเศษนอกเหนือไปจากนี้ เช่น การรีเซ็ตการตั้งค่า การล้างการตั้งค่า การเลือก IOS ตัวใหม่ เป็นต้น

Set up Mode

เป็นโหมดที่ใช้เมื่อเริ่มการทำงานของเราเตอร์ใหม่ครั้งแรกหรือพิมพ์คำสั่ง Setup ที่โหมด Configuration หรือเมื่อมีการเคลียร์ค่าของคอนฟิกบนเราเตอร์ทั้งหมดออก ด้วยคำสั่ง Write Erase แล้วเริ่มบูตระบบใหม่ ดังรูป 3.18 เป็นต้น

```
#setup

--System Configuration Dialog--
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.

Continue with configuration dialog? [yes/no].

First, would you like to see the current interface summary?
[yes/no]

Interface IP-Address OK? Method Status Protocol
TokenRing0 unassigned NO not set down down
Ethernet0 unassigned NO not set down down
Serial0 unassigned NO not set down down
Fddi0 unassigned NO not set down down
```

รูปที่ 3.18 Set up Mode

ROM Monitor

ใช้สำหรับแก้ไขค่าคอนฟิกของเราเตอร์ในกรณีที่เกิดการทำงานที่ไม่เป็นปกติ เช่น การเปลี่ยนแปลงการ Boot การเปลี่ยนรหัสผ่าน และการซีกค่าของระบบต่าง ๆ วิธีการเข้าสู่โหมดนี้โดยขณะที่เราเตอร์เริ่มต้น Boot ผ่านไปประมาณ 60 วินาที ให้กดปุ่ม CTL + Break เมื่อเข้าสู่โหมดนี้ Prompt จะมีลักษณะเป็น *rommon>* หรือ >

Subset IOS (ROM IOS หรือ RxBoot)

ปกติจะใช้ในกรณีที่ต้องการปรับปรุงระบบ Prompt จะเป็น *Router(boot)>*

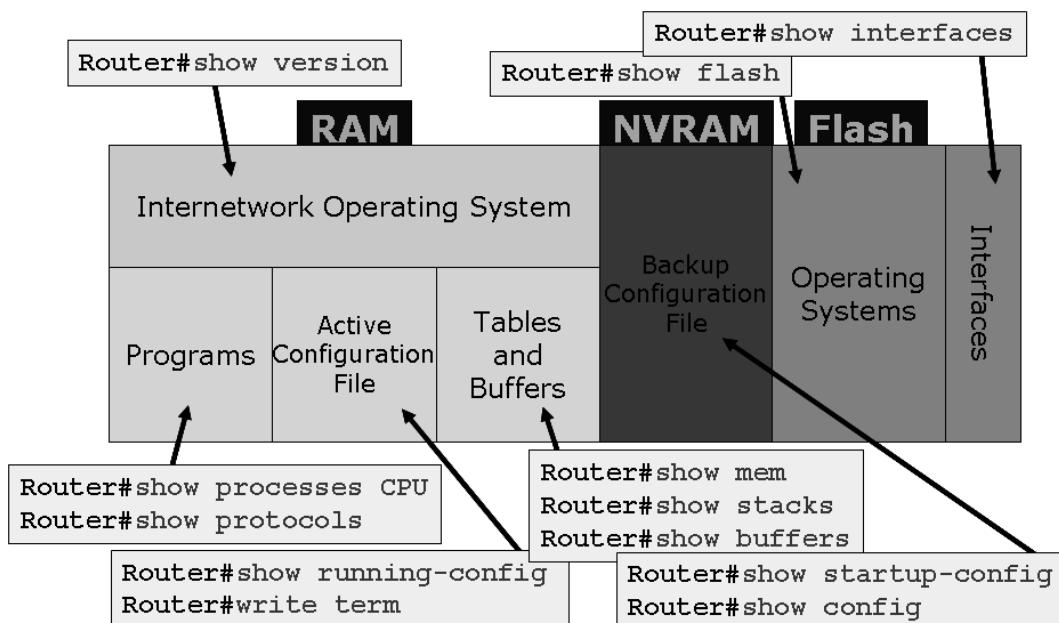
การแสดงข้อมูลต่าง ๆ ของเราเตอร์ด้วยคำสั่ง SHOW

เราเตอร์ของ Cisco จะใช้คำสั่ง show ดังตารางที่ 3.1 เมื่อต้องการแสดงค่าต่าง ๆ ของเราเตอร์ เช่น ถ้าต้องการแสดงเวอร์ชันของ IOS เราสามารถใช้คำสั่ง show version ที่โหมด Privileged จากรูปข้างล่างแสดงถึงคำสั่ง show ที่สัมพันธ์กับส่วนต่าง ๆ ของเราเตอร์ ตัวอย่างเช่นถ้าป้อนคำสั่ง show version เราเตอร์จะดึงข้อมูลมาจากในหน่วยความจำหลักที่เก็บ IOS อยู่มาแสดงเป็นต้น ดังรูปที่ 3.19

ตารางที่ 3.1 รูปแบบคำสั่ง show แบบต่าง ๆ

คำสั่ง	ส่วนที่ถูกแสดง	ความหมาย
Show version	ข้อมูลของ IOS ในหน่วยความจำหลัก	แสดงเวอร์ชันของ IOS, ปริมาณหน่วยความจำหลักจำนวนของหน่วยความจำ flash เป็นต้น
Show interfaces	ข้อมูลของอินเตอร์เฟส	แสดงข้อมูลรายละเอียดต่าง ๆ ของอินเตอร์เฟสทั้งหมด
Show flash	ข้อมูลของ flash	แสดงข้อมูลของ IOS ที่เก็บไว้ใน flash และประมาณหน่วยความจำ flash
Show process cpu	ข้อมูลเกี่ยวกับโปรแกรมที่กำลังทำงาน	แสดงเบอร์เซ็นต์การใช้งานของ CPU
Show mem	ข้อมูลเกี่ยวกับหน่วยความจำหลัก	แสดงเบอร์เซ็นต์การใช้งานของหน่วยความจำหลัก
Show protocols	ข้อมูลของ Routing โพรโทคอล	แสดงถึงโพรโทคอลที่ทำงานอยู่ปัจจุบันว่าเป็นแบบใดพร้อมทั้งรายละเอียดต่าง ๆ
Show running-config	ข้อมูลของคอนฟิกที่กำลังทำงานในปัจจุบัน	แสดงรายการทั้งหมดของคอนฟิกในรูปแบบไฟล์ที่กำลังทำงานอยู่ในปัจจุบันในหน่วยความจำหลัก
Show stack	ข้อมูลของ stack	แสดงข้อมูลทั้งหมดของ stack ที่กำลังใช้งานอยู่ว่าใช้งานเป็นอย่างไร
Show buffers	ข้อมูลของบัฟเฟอร์	แสดงการใช้งานบัฟเฟอร์ของเราเตอร์

Show startup-config	ข้อมูลของไฟล์คอนฟิกที่เก็บอยู่ใน NVRAM	แสดงข้อมูลของไฟล์คอนฟิกที่ถูกเก็บในลักษณะกึ่งถาวรโดยต้องมีการบันทึกจาก running-config มาเก็บไว้
---------------------	--	---



รูปที่ 3.19 แสดงการใช้คำสั่ง show

การ Boot เร้าเตอร์แบบต่าง ๆ ผ่านทางรีจิสเตอร์ (Register)

สามารถเปลี่ยนวิธีการ Boot ระบบของเราเร้าเตอร์ได้โดยการเซ็ตค่าผ่านรีจิสเตอร์ ซึ่งปกติแล้วจะไม่ได้ทำบ่อยนัก สถานการณ์ที่จะทำก็คือกรณีที่ IOS เกิดความเสียหาย หรือจำรหัสผ่านไม่ได้เป็นต้น ชนิดของการ Boot สามารถดูได้จากตารางที่ 3.2

ตารางที่ 3.2 ชนิดของการ Boot เร้าเตอร์โดยเซ็ตค่าผ่านทาง Register

Boot Field Value	Meaning
0x0---	Bypass network booting (TFTP)
0x2---	Attempt to boot the IOS from a network TFTP server first.
0x2100 or 0x0100	Boot to ROM monitor mode
0x2101 or 0x0101	Boot to subset IOS
0x2102 – 0x210F Or 0x0102 – 0x010F	Boot according to boot commands in saved configuration (default boot)
0x2142 – 0x214F Or 0x0142 – 0x014F	Boot default ROM software if network boot fails & Ignore NVRAM contents.

ขั้นตอนของการ Boot ผ่านรีจิสเตอร์

วิธีที่ 1 เมื่อสามารถเข้าระบบได้

1. เข้าสู่ Configuration Mode → **Router>enable** → **Router#configuration terminal**
2. เช็คค่าของ register เป็นค่าที่ต้องการให้ boot แบบใด เช่น ต้องการให้ boot ด้วย subset IOS → **Router(config)#config-register 0x101**
3. ออกจากโหมดคอนฟิกด้วย **exit**
4. ทดลอง show version ดูก่อนว่าเปลี่ยนแปลงหรือไม่ก่อนทำการ boot เราก่อตัวด้วยคำสั่ง **reload**

วิธีที่ 2 เมื่อยังไม่สามารถเข้าระบบได้

1. ปิดเราเตอร์แล้วเปิดเครื่องใหม่
2. รอเวลาให้เครื่องทำงานก่อนประมาณ 60 วินาทีแล้วจึงกดปุ่ม CTL กับ Break พร้อม ๆ กัน
3. ระบบจะเข้าสู่ ROM โดยแสดง Prompt เป็น **rommon>**
4. ใส่ค่าให้กับรีจิสเตอร์ตามที่ต้องการ สมบูรณ์ต้องการ boot เป็น 0x2142 → **rommon>confreg 0x2142**
5. สั่งคำสั่งให้เริ่มการทำงานใหม่ → **rommon>reset**

การ Backup และ Restore IOS และคอนฟิกกูเรชันไฟล์

IOS และ คอนฟิกกูเรชันไฟล์เป็นส่วนที่สำคัญเป็นอย่างยิ่งที่จะทำให้เราเตอร์สามารถทำงานได้อย่างราบรื่น แต่ก็มีปอยครั้งที่อุปกรณ์เกิดเสียขึ้นมาแบบไม่ทันตั้งตัว เช่น เกิดจากไฟฟ้ากระชาก ทำให้อุปกรณ์ต่าง ๆ ไม่สามารถทำงานต่อไปได้ ถ้าเกิดเหตุการณ์เช่นนี้ ก็จะต้องทำการเปลี่ยนอุปกรณ์ กันใหม่ ปัญหาคือ "แล้วคอนฟิกกูเรชันไฟล์ล่ะ?" จะเอามาจากไหน ถ้าตอบว่าก็คือเข้าไปใหม่ซิ วิธีนี้จะหมายความว่าระบบหรือองค์กรที่มีขนาดเล็ก ๆ แต่ถ้าองค์กรที่มีขนาดใหญ่ ๆ ล่ะ คำสั่งในคอนฟิกไฟล์ อย่างต่ำ ๆ ก็มีเป็นร้อย แล้วจะจำได้หรือ ? วิธีแก้ไขที่ดีที่สุดคือ "แบ็คอัพ" ไว้ที่เครื่องเดียวหนึ่งที่ปลอดภัย โดยการติดตั้งโปรแกรมประเภท TFTP หรือ FTP ไว้ เมื่อเกิดเหตุการณ์ฉุกเฉินก็ copy ข้อมูลจากเครื่องที่ทำการรองไว้เข้าไปในอุปกรณ์ตัวใหม่ได้ทันที ข้อมูลที่จะต้องเก็บมี 2 ส่วนคือ

1. Backup และ Restore IOS

อย่างที่ทราบแล้วว่า IOS จะทำหน้าที่เป็น OS ของระบบ แต่ก็จะมีคำถามว่าเมื่ออุปกรณ์ตัวใหม่มา มันจะมี OS มาให้อยู่แล้วทำไม่เจิงต้อง Backup เพื่อผลเนื่องจาก IOS ที่เราใช้งานจะมีการปรับปรุงอยู่เรื่อย ๆ บางครั้งอาจจะต้องมีการ Patch (คือการแก้ไขจุดบกพร่องของซอฟต์แวร์เมื่อพบ Bug) ดังนั้น IOS ของเราจะมีการอัพเดทอย่างเข้าไป ซึ่งถ้าเรา IOS ตัวใหม่มา

ทำงานอาจจะไม่สามารถทำงานได้เต็มประสิทธิภาพก็เป็นได้ และอีกรูปนึงคือ กรณีที่ไม่ได้นำอุปกรณ์ใหม่แกะกล่องมาใช้ แต่ห้าอุปกรณ์ตัวเก่าแต่มาใส่ IOS ใหม่ให้มันก็จะทำได้อย่างรวดเร็ว

วิธีการ Backup IOS

1. ทดลองแสดง IOS ว่าซื้ออะไร รุ่นอะไรมดูก่อนด้วยคำสั่ง **show flash** ที่ enable mode (privileged) ดังรูปที่ 3.20
2. จากนั้นก็ใช้คำสั่ง **copy flash tftp <enter>**
3. ใส่หมายเลขไอพีของเครื่องที่ TFTP เซิร์ฟเวอร์ แล้วกด enter ถ้าไม่ผิดพลาดอะไร โปรแกรมจะเริ่มทำการ copy IOS ไปยัง TFTP เซิร์ฟเวอร์ สถานะการณ์ก็จะเป็น สัญลักษณ์ !!!!!!!

```

Command
Router# show flash
4096 bytes of flash memory on embedded flash (in XX).

file      offset      length      name
0        0x40      1204637      xk09140z
[903848/2097152 bytes free]

Router# copy flash tftp
IP address of remote host [255.255.255.255]? 172.16.13.111
filename to write on tftp host? c4500-i
writing C4500-i !!!!!!!!!!!!!!!!
successful tftp write.
Router#

```

รูปที่ 3.20 แสดงคำสั่งสำหรับ Backup IOS

วิธีการ Restore IOS

ทำในลักษณะเดียวกันแต่มีทิศทางตรงกันข้าม ดังรูปที่ 3.21

```

Command
Router# copy tftp flash
IP address or name of remote host [255.255.255.255]? 172.16.13.111
Name of tftp filename to copy into flash []? c4500-aj-m
copy C4500-AJ-M from 172.16.13.111 into flash memory? [confirm] <Return>
xxxxxxxx bytes available for writing without erasure.
erase flash before writing? [confirm] <Return>
Clearing and initializing flash memory [please wait] #####...
Loading from 172.16.13.111: !!!!!!!!
!!!!!! (text omitted) [OK - 324572/524212 bytes]
Verifying checksum...
VVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVV
VVVVVVVV (text omitted)
Flash verification successful. Length = 1804637, checksum = 0xA5D3

```

รูปที่ 3.21 แสดงคำสั่งสำหรับ Restore IOS

2. Backup และ Restore คอนฟิกุเรชันไฟล์

running-config เป็นไฟล์ที่ถูกโหลดมาจาก NVRAM เข้ามายังหน่วยความจำหลักเพื่อทำงาน เมื่อมีการรีเซ็ตเครื่องจะทำให้ข้อมูลที่อยู่ในหน่วยความจำหลักหายไป ส่งผลให้ต้องโหลดไฟล์คอน

ฟิกจาก NVRAM มาทำงานใหม่อีกครั้ง ขณะที่เราป้อนคำสั่งหนึ่ง ๆ ลงไปที่ CLI จะส่งผลทันทีต่อเราเตอร์ เช่น เราสั่ง shutdown อินเตอร์เฟส จะทำให้อินเตอร์เฟสดังกล่าวนั้นหยุดทำงานทันที แต่ถ้าเรายังไม่มีการเขียนข้อมูลทั้งไปที่ NVRAM เมื่อมีการรีเซ็ตเราเตอร์ใหม่อีกครั้งจะทำให้อินเตอร์เฟสที่ลูก shutdown นั้นจะกลับมาทำงานอีกครั้งหนึ่ง ดังรูปที่ 3.22, 3.23

```
Command
tokyo#copy running-config tftp
Remote host []? 131.108.2.155
Name of configuration file to write[tokyo-config]?tokyo.2
Write file tokyo.2 to 131.108.2.155? [confirm] y
Writing tokyo.2 !!!!! [OK]
tokyo#
```



รูปที่ 3.22 การ Backup คอนฟิกยูเรชันไฟล์

```
Command
tokyo#copy tftp running-config
Host or network configuration file [host]?
IP address of remote host [255.255.255.255]? 131.108.2.155
Name of configuration file [Router-config]? tokyo.2
Configure using tokyo.2 from 131.108.2.155? [confirm] y
Booting tokyo.2 from 131.108.2.155:!! [OK-874/16000 bytes]
tokyo#
```

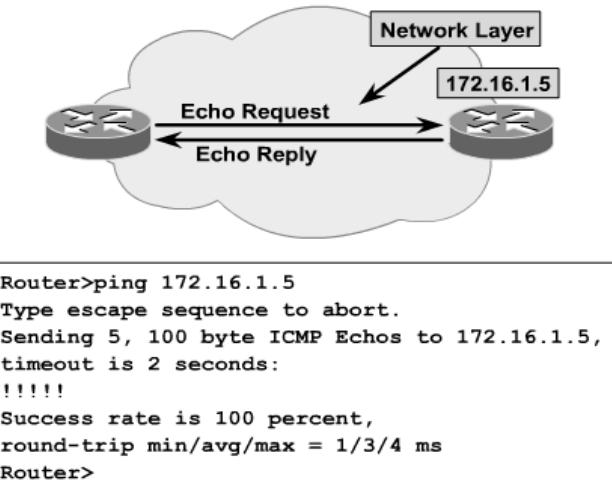


รูปที่ 3.23 การ Restore คอนฟิกยูเรชันไฟล์

การทดสอบการเชื่อมต่อระบบด้วยคำสั่ง ping

ในการตรวจสอบระบบเครือข่ายว่าสามารถทำงานได้จริงหรือไม่ ส่วนมากจะใช้โปรแกรมตัวหนึ่งคือ ping สำหรับตรวจสอบ โดยหลักการคือมันจะส่งแพ็กเกจจำนวนหนึ่ง (Echo Request) ไปยังปลายทาง ส่วนปลายทางเมื่อได้รับแพ็กเกจแล้วจะส่งข้อมูลกลับมา (Echo Reply) ต่อจากนั้นเรา

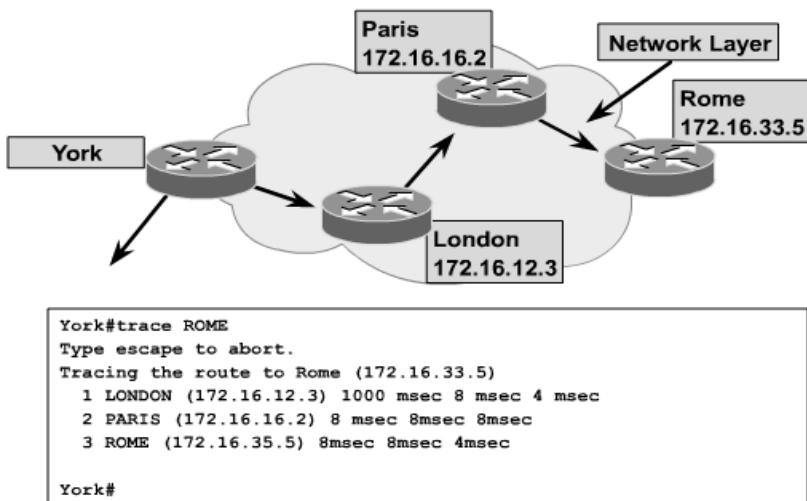
เตอร์ต้นทางจะประเมินค่าถ้าข้อมูลที่กลับนามีค่าตามที่กำหนดไว้ (TTL) ก็เป็นอันว่าสามารถสื่อสารกันได้ รูปแบบคำสั่งคือ *ping* หมายเลขอุปกรณ์ปลายทาง ดังรูปที่ 3.24



รูปที่ 3.24 แสดงการใช้คำสั่ง ping

การทดสอบการเชื่อมต่อระบบด้วยคำสั่ง trace

หลักการคล้ายกับ ping แต่ trace จะแสดงข้อมูลในรูปของเส้นทางจากต้นทางไปยังปลายทางว่าได้ผ่านเราเตอร์ตัวใดบ้างในเครือข่ายพร้อมกับระยะเวลาที่มั่นเดินทางไปถึงด้วย รูปแบบคำสั่งคือ *trace* หมายเลขอุปกรณ์ชื่อของเราเตอร์ ดังรูปที่ 3.25



รูปที่ 3.25 แสดงการใช้คำสั่ง trace

TIPS: ผู้ดูแลระบบสามารถเข้าไปคอนฟิกเราเตอร์ได้ผ่าน Remote คือการใช้คำสั่ง Telnet ชื่อโฮสต์

TIPS: ผู้ใช้งานสามารถลับ session ของ Telnet ที่เปิดไว้หลาย ๆ session ด้วยการกดคีย์ *ctrl + shift + 6 + x*

แบบฝึกหัดท้ายบท

1. การเข้าถึงระบบเครือข่ายมี 3 ระดับประกอบไปด้วยอะไรบ้าง และแต่ละประเภทแตกต่างกันอย่างไร
2. เราเตอร์มีหน้าที่อย่างไรบนระบบเครือข่าย พร้อมยกตัวอย่าง
3. จงอธิบายส่วนประกอบต่างๆ ของเราเตอร์ต่อไปนี้ว่าทำงานอย่างไร
 - ROM
 - Flash Memory
 - NVRAM
 - RAM/DRAM
 - Router Interface
 - IOS
4. จงอธิบายหลักการทำงานของเราเตอร์มาพอเข้าใจ
5. โหมดการทำงานบนเราเตอร์มีกี่โหมด อะไรบ้าง
6. จงอธิบายคำสั่งการทำงานของเราเตอร์ต่อไปนี้มาพอเข้าใจ
 - Show version
 - Show interface
 - Show flash
 - Show process
 - Show mem
 - Show protocol
 - Show running-config
 - Show startup-config
7. จงอธิบายกระบวนการ boot ผ่านรีจิสเตอร์ มาพอเข้าใจ
8. จงอธิบายวิธีการ backup และ restore ไฟล์ configuration ว่ามีขั้นตอนการทำงานอย่างไร

บทที่ 4

เริ่มต้นการคอนฟิกเราเตอร์ (Initial Router Configuration)

The screenshot shows a Windows command prompt window titled "Initial Router Configuration". The window displays the following configuration commands:

```
C:\WINNT\system32\telnet.exe
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
Received 201760 packets input, 11784807 bytes, 0 no buffer
0 input errors, 0 CRC, 0 frame, 0 runts, 0 giants
339107 packets output, 21366009 bytes, 0 overrun, 0 ignored, 0 abort
0 output errors, 0 collisions, 6 interface resets
0 output buffer failures, 0 output buffers swapped out
10 carrier transitions
      DCD=up DSR=up DTR=up RTS=up CTS=up
Router1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router1<config>#int s0
Router1<config-if>#ip addr 10.0.0.1 255.0.0.0
Router1<config-if>#no shut
Router1<config-if>#descr link to LA
Router1<config-if>#ena sec cisco
Router1<config>#int s0
Router1<config-if>#clock rate 56000
Router1<config-if>#int s1
```

- Hostname and interface descriptions
- System passwords
- Banners
- Interfaces
- Back-to-back router
- Cisco discovery protocol (CDP)

แนวคิด

ในส่วนนี้จะใช้สำหรับฝึกหัดการคอนฟิกระบบเครือข่ายผ่านตัวจำลอง เนื้อหาในแต่ละแล็บนั้นจะมีลักษณะที่ต่อเนื่องและเรียงลำดับตามความสำคัญของเนื้อหาในการติดตั้งและดูแลระบบเครือข่าย เมื่อจบในบทนี้แล้วผู้เรียนจะสามารถเข้าใจการติดตั้งและปรับแต่งอุปกรณ์เครือข่ายเบื้องต้นเพื่อนำไปสู่การปรับแต่งกับอุปกรณ์จริง

วัตถุประสงค์

- เพื่อให้สามารถปรับแต่งอุปกรณ์ที่ใช้งานบนระบบเครือข่ายได้อย่างเหมาะสม
- เพื่อสร้างความชำนาญในการใช้งานอุปกรณ์ที่สำคัญๆ บนระบบเครือข่าย

การตั้งค่าและปรับแต่งอุปกรณ์เครือข่าย (เช่น เร้าเตอร์ สวิตช์ เป็นต้น)

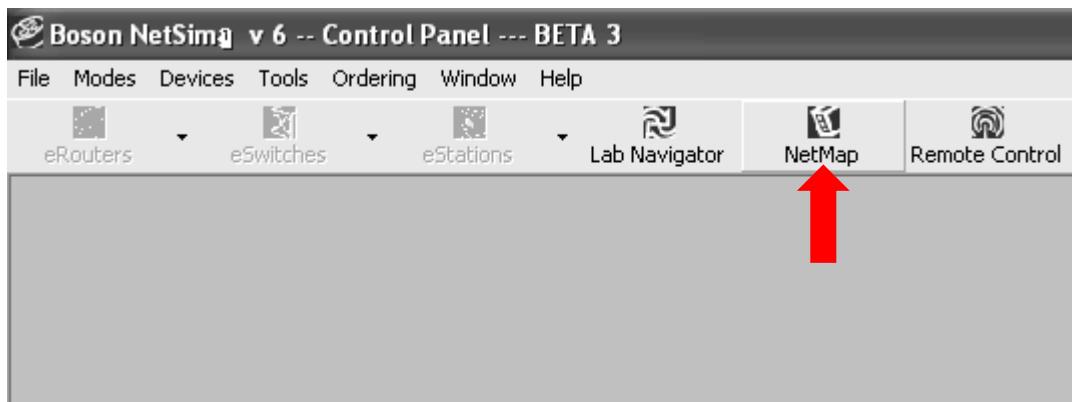
❑ Connecting และ Logging ไปยัง Cisco Router

จุดมุ่งหมาย : เริ่มต้นเรียนรู้คอนฟิกของเร้าเตอร์

เครื่องมือที่ใช้ทดลอง : ใช้เร้าเตอร์ 2 ตัว

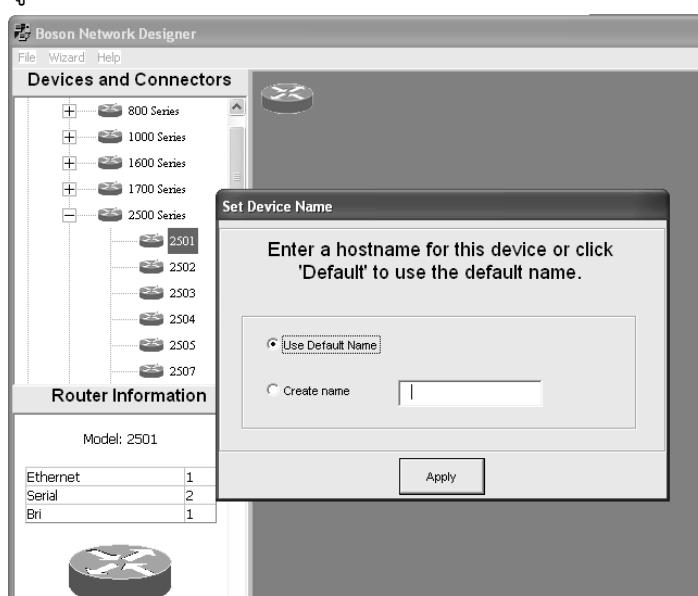
การสร้าง Network Map:

1. จากหน้าต่างหลัก (Simulator) คลิกเลือก NetMap เพื่อสร้างผังเน็ตเวิร์ค ดังรูปที่ 4.1



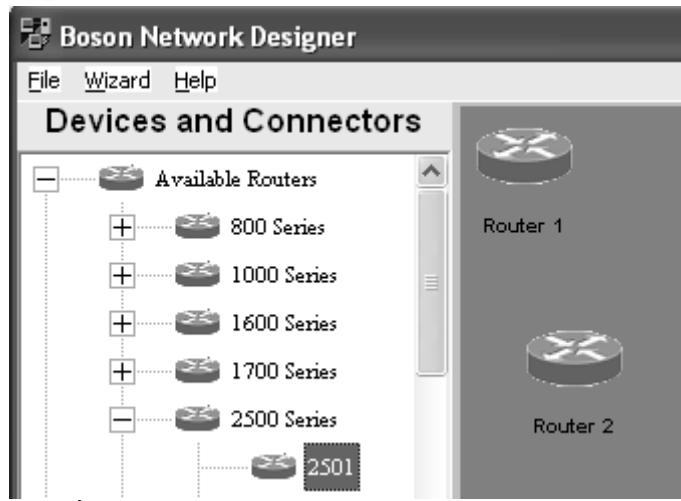
รูปที่ 4.1 หน้าต่างหลัก (Simulator) เลือก NetMap

2. ขั้นต่อไป จะปรากฏหน้าต่าง Boson Network Designer → คลิกเลือก Available Routers → คลิกเลือก 2500 series → เลือกรุ่น 2501 โดยการลากมาวางที่พื้นที่สีเทา ทางด้านขวาเมื่อ → ให้เลือก Use Default Name (เร้าเตอร์จะชื่อ Router1) → คลิก Apply ดังรูปที่ 4.2



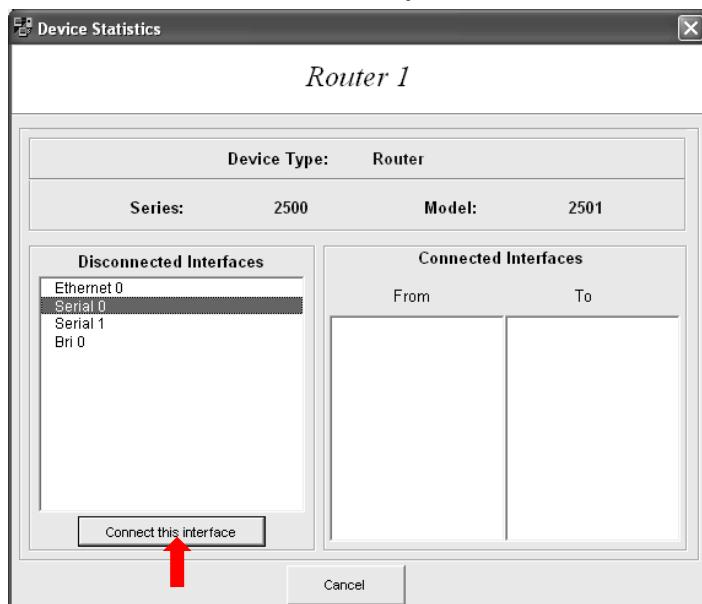
รูปที่ 4.2 สร้างเร้าเตอร์ชื่อ Router1

3. จะปรากฏอุปกรณ์เร้าเตอร์ (Router1) บนผังของเน็ตเวิร์ค จากนั้นให้สร้างเร้าเตอร์รุ่นเดิมอีก 1 ตัว (Router2) เนื่องจาก Boson NetSim จะยอมให้ค่อนพิกต้องมีอุปกรณ์ 2 ตัวขึ้นไป เชื่อมต่อกัน ดังรูปที่ 4.3



รูปที่ 4.3 สร้างเราเตอร์ 2 ตัวคือ Router1 และ Router2

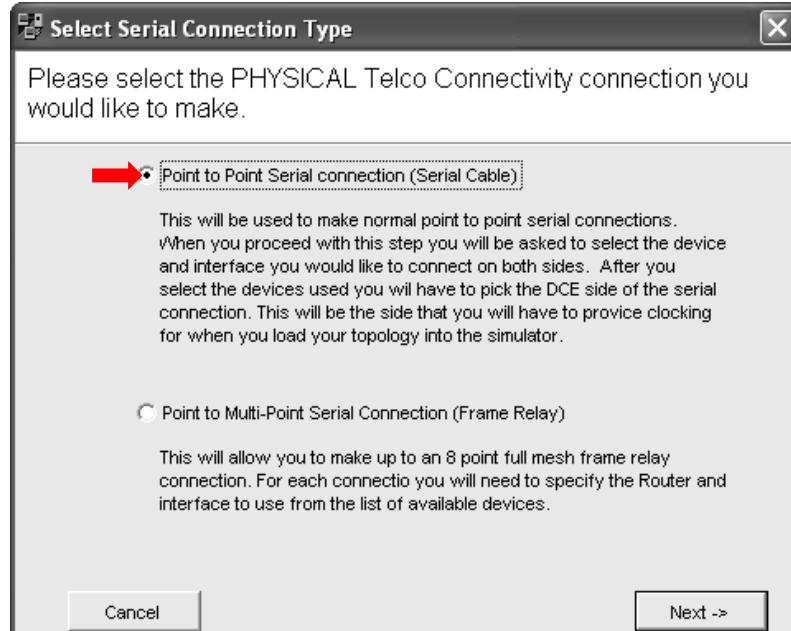
4. จากนั้นให้เชื่อมต่อเราเตอร์ 2 ตัวเข้าด้วยกันโดยการดับเบิลคลิกที่เราเตอร์ตัวใดตัวหนึ่งก็ได้ (หรือคลิกขวาที่ตัวเราเตอร์แล้วเลือก Add Connection to) ลำดับต่อไปให้เลือก อินเตอร์เฟสที่ต้องการใช้เชื่อมต่อ ให้ทำการเลือกอินเตอร์เฟสชื่อ serial 0 (สาย serial เป็นสายนำสัญญาณแบบจุดต่อจุด ส่วนใหญ่มีขนาดของแบนด์วิดธ์ไม่เกิน 2 เม็กกะบิตต่อวินาที) จากนั้นให้เลือก Connect this interface ดังรูปที่ 4.4



รูปที่ 4.4 เลือกอินเตอร์เฟสที่ต้องการเชื่อมต่อ

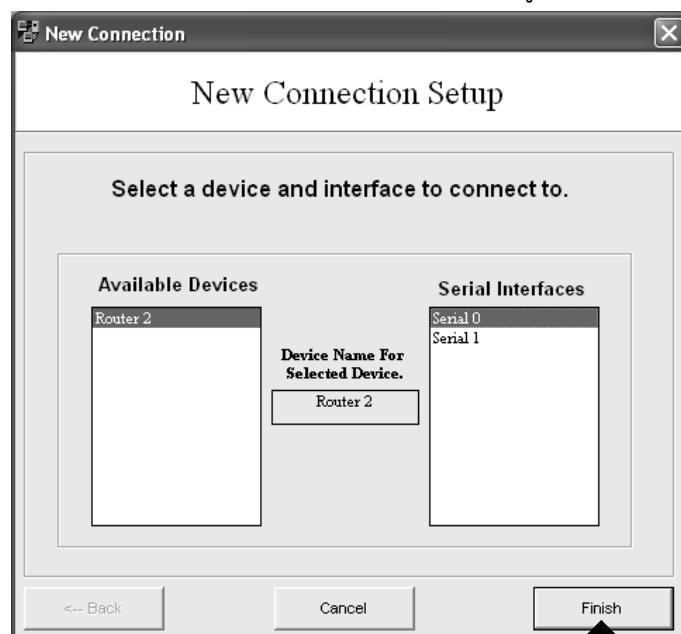
5. ลำดับต่อไปให้เลือกชนิดของการเชื่อมต่อซึ่งมีอยู่ 2 แบบคือ
 1. Point to Point Serial Connection (Serial Cable) คือสายที่มีการเชื่อมต่อกันแบบจุดต่อจุด หรือ 1 ต่อ 1

2. Point to Multi-Point Serial Connection (Frame Relay) เป็นการเชื่อมต่อแบบเพร์มรีเลย์ ซึ่งมีลักษณะ 1 ต่อ หลาย ๆ เครื่อง ให้เลือกการเชื่อมต่อเป็นแบบ Point to Point Connection แล้วคลิก Next-> ดังรูปที่ 4.5



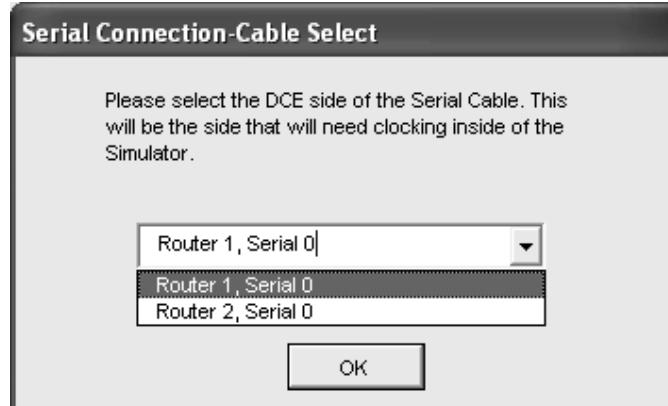
รูปที่ 4.5 ชนิดของการเชื่อมต่อ

6. จากนั้นจะปรากฏรายการของเราเตอร์ที่ต้องการจะเชื่อมต่อด้วยขึ้นมา ให้เลือกเชื่อมต่อกับอินเตอร์เฟสที่ต้องการในที่นี้ ในช่อง Available Devices เลือก Router 2 ในช่อง Serial Interfaces ให้เลือก Serial 0 เมื่อเลือกครบแล้วให้กดปุ่ม Finish ซึ่งเป็นการสิ้นสุดการเชื่อมต่ออินเตอร์เฟสของเราเตอร์ทั้งสองตัวเข้าด้วยกัน ดังรูปที่ 4.6



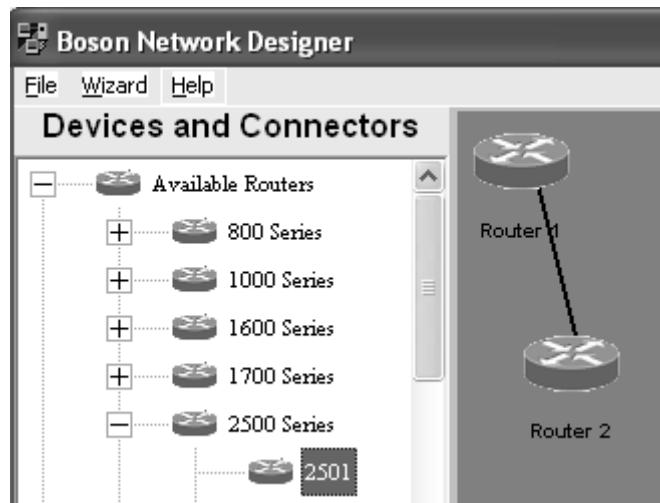
รูปที่ 4.6 เลือกเราเตอร์และอินเตอร์เฟส

7. ขั้นต่อไปจะปรากฏหน้าต่างเพื่อให้เลือกราเตอร์ที่เป็น DCE (เราเตอร์จะใช้อินเตอร์เฟสที่เป็น DCE (Data Circuit-Terminating Equipment) ในการสร้าง Clock เพื่อให้เราเตอร์ทั้ง 2 ฝ่ายทำงานได้พร้อมเพียงกัน) ในที่นี่ให้เลือกที่ Router 1 เป็น DCE และเราเตอร์อีกฝั่งจะเรียกว่า DTE (Data Terminating Equipment) ดังรูปที่ 4.7



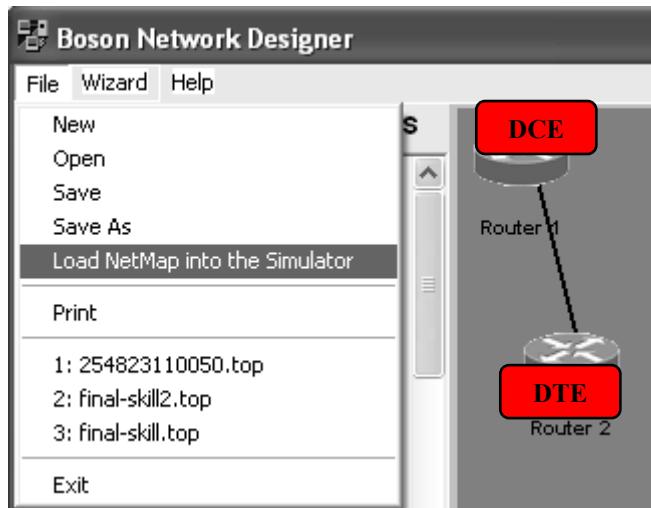
รูปที่ 4.7 เลือกราเตอร์ที่ให้เป็น DCE เพื่อเป็นตัวสร้าง Clock

8. ขั้นตอนต่อไป ที่หน้าต่างของ NetMap ให้เลือก File → Load NetMap into Simulator เพื่อโหลดผังเน็ตเวิร์คให้ไปทำงานที่หน้าต่างหลัก (Simulator) ซึ่งจะมีเมนูแสดงข้อมูลว่า ต้องการโหลดผังเน็ตเวิร์คเข้าไปยังหน้าต่างการทำงานหลักหรือไม่ (Simulator) ให้เลือก yes ดังรูปที่ 4.8



รูปที่ 4.8 แสดงการเชื่อมต่อที่ได้สร้างเสร็จแล้ว

9. เมื่อโหลดผังเน็ตเวิร์คไปยังหน้าต่างหลัก (Simulator) แล้ว เป็นอันว่าพร้อมที่จะเริ่มต้นคอนฟิกต่อไป ดังรูปที่ 4.9, 4.10



รูปที่ 4.9 แสดงวิธีการโหลดผังเน็ตเวิร์คเข้าไปยังหน้าต่างหลัก (Simulator)



รูปที่ 4.10 หน้าต่างหลัก (Simulator) พร้อมที่จะรับคำสั่งการทำงานต่อไป

หมายเหตุ : ควรจะมีการบันทึกผังเน็ตเวิร์คไว้ก่อนให้เลือก File → Save หรือ Save As (เมื่อไม่ต้องการซื้อที่โปรแกรมตั้งให้) ข้อมูลที่บันทึกจะมีนามสกุลจะเป็น .top

หมายเหตุ : ถ้าต้องการทราบว่าอินเตอร์เฟสของเราเตอร์ตัวไหนเป็น DCE ให้ใช้คำสั่ง

Router#show controllers ในโหมดการทำงาน privilege ดังรูปที่ 4.11

```
Router1#sh controllers
HD unit 0, idb = 0x1AE828, driver structure at 0x1B4BA0
buffer size 1524 HD unit 0
cpb = 0x7, eda = 0x58DC, cda = 0x58F0
RX ring with 16 entries at 0x4075800
00 bd_ptr=0x5800 pak=0x1B5E24 ds=0x4079108 status=80 pak_size=13
01 bd_ptr=0x5814 pak=0x1B85B8 ds=0x4080384 status=80 pak_size=13
02 bd_ptr=0x5828 pak=0x1B880C ds=0x4080A40 status=80 pak_size=69
```

รูปที่ 4.11 หน้าต่างหลักแสดงคำสั่ง show controllers

หน้าต่าง Simulator :

1. กดปุ่มคีย์บอร์ด **Enter** เพื่อเริ่มต้น(ถ้าต้องการดูແຜ່ນັງເຄືອຂ່າຍໃຫ້ເລືອກທີ່ເນຸ້ມ NetMap)
2. Router> [กำລັງຍູ້ໃນໂທມດ User]
3. Router>enable [ເຂົ້າສູ່ໂທມດ privileged ດ້ວຍคำສັ່ງ enable]

- | | |
|---------------------------|---|
| 4. Router# | [เข้าสู่โหมดของ privileged เรียบร้อยแล้ว] |
| 5. Router# <i>disable</i> | [กลับเข้าไปโหมด User] |
| 6. Router> <i>exit</i> | [ออกจากเราเตอร์ (ออกจาก console)] |
| 7. กดคีย์ <i>enter</i> | [เพื่อเข้าสู่โหมด User อีกครั้ง] |

หมายเหตุ : ลองทดสอบคำสั่ง Router>ena?

หมายเหตุ : การบันทึกข้อมูลจะมี 2 ส่วนคือที่หน้าต่าง Boson Network Designer (ใช้นามสกุล .top) และหน้าต่าง Boson Simulator (ใช้นามสกุล .nwc) ควรจะทำการตั้งชื่อให้ตรงกัน เช่น บันทึกผังเน็ตเวิร์กชื่อ lab1.top และควรบันทึกคอนฟิกไฟล์ใน simulator เป็น lab1.nwc ด้วยเช่นกัน เพราะจะเป็นการง่ายต่อการทำความเข้าใจและโปรแกรมจะทำงานได้อย่างถูกต้องด้วย

▣ เป็นต้นเกี่ยวกับวิธีการสื้อสารของผู้ใช้งานกับเราเตอร์

จุดมุ่งหมาย : เริ่มต้นเรียนรู้คำสั่งพื้นฐานที่เกี่ยวกับโหมดการทำงานของเราเตอร์และคำสั่งที่ใช้สำหรับช่วยเหลือ

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัว

การสร้าง Network Map: ให้สร้างผังเน็ตเวิร์กใหม่อีกหนึ่งที่ 4.8

หน้าต่าง Simulator:

- | | |
|-----------------|--|
| 1. <i>enter</i> | [กดคีย์บอร์ด enter เพื่อเข้าสู่โหมด User] |
| 2. Router>? | [แสดงคำสั่งทั้งหมดที่ใช้งานในโหมด User] ดังรูปที่ 4.12 |

Router>?	
show	Show running system information
enable	Turn on privileged commands
exit	Exit from the EXEC
help	Description of the interactive help system
disable	Turn off privileged commands
disconnect	Disconnect an existing network connection
logout	Exit from the EXEC
ping	Send echo messages
terminal	Set terminal line parameters
traceroute	Trace route to destination
lock	Lock the terminal
login	Log in as a particular user
mrinfo	Request neighbor and version information from a multicast router
mstat	Show statistics after multiple multicast traceroutes
mtrace	Trace reverse multicast path from destination to source
name-connection	Name an existing network connection
pad	Open a X.29 PAD connection
ppp	Start IETF Point-to-Point Protocol (PPP)
rlogin	Open an rlogin connection
slip	Start a Serial-line IP (SLIP)
systat	Display information about terminal lines
tunnel	Open a tunnel connection
udptn	Open an udptn connection

--MORE--

รูปที่ 4.12 ผลลัพธ์ของคำสั่ง ?

- | | |
|--------------------------|--|
| 3. Router> <i>enable</i> | [เข้าสู่โหมด privilege] |
| 4. Router#? | [แสดงคำสั่งทั้งหมดที่ใช้งานในโหมด privilege] |

5. Router#show ? [แสดงคำสั่งที่ใช้งานร่วมกับคำสั่ง show] ดังรูปที่ 4.13

```

Router#show ?
version          System hardware and software statu
cdp              CDP information
clock             Display the system clock
flash             display information about flash: :
history           Display the session command histo
hosts             IP domain-name, nameservers, and
interfaces        Interface status and configuration
protocols         Active network routing protocols
sessions          Information about Telnet connectio
terminal          Display terminal configuration pa
users             Display information about terminal
frame-relay       Frame-Relay information
isdn              ISDN information
ntp               Network time protocol
controllers       Interface controller status
running-config    Current operating configuration
startup-config   Contents of startup configuration
access-lists      List access lists
configuration     Contents of Non-Volatile memory
ip                IP information
arp               ARP table
isis              IS-IS routing information
clns             CLNS network information

--MORE--|

```

รูปที่ 4.13 ผลลัพธ์ของคำสั่ง show ?

6. Router#running-config [แสดงคอนฟิกุเรชันไฟล์ที่กำลังทำงานอยู่] ดังรูปที่ 4.14

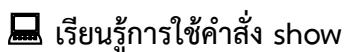
```

Router# XXXXXXXXXX
Building configuration...

!
Version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
!
ip subnet-zero
!
```

รูปที่ 4.14 ผลลัพธ์ของคำสั่ง show running-config

7. -MORE-- <spacebar> [more เป็นการแสดงหน้าต่อไป โดยกดคีย์ space bar]
8. Router#[exit/disable] [จากให้หมด privilege ให้ออกจากโหมดนี้ด้วยคำสั่ง exit
หรือ disable]



เรียนรู้การใช้คำสั่ง show

จุดมุ่งหมาย : เรียนรู้การใช้คำสั่ง show และคำสั่งที่ทำงานร่วมกับมัน

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัว

การสร้าง Network Map : ให้สร้างผังเน็ตเวิร์คเหมือน รูปที่ 4.8

หน้าต่าง Simulator :

1. Router> [เข้าสู่การทำงานของโหมด User]
2. Router>enable [เข้าสู่โหมด privilege]
3. Router#show running-config [แสดงคอนฟิกภูเรชันที่ทำงานอยู่ในหน่วยความจำหลัก ตอนพิกนี้จะหายไปหากมีการรีเซ็ตเราเตอร์]
4. Router#show flash [แสดงข้อมูลของระบบปฏิบัติการ (IOS) ที่เก็บอยู่บนหน่วยความจำที่เป็น flash หน่วยความจำแบบนี้ข้อมูลจะไม่สูญหายจากการรีเซ็ตระบบ] ดังรูปที่ 4.15

```
Router# [REDACTED]
System flash directory:
File Length Name/Status
1 5880916 [REDACTED]
[5880980 bytes used, 2507628 available, 8388608 total]
8192K bytes of processor board System flash (Read/Write)
```

รูปที่ 4.15 ผลลัพธ์ของคำสั่ง show flash

5. Router#show history [แสดงคำสั่งที่เคยใช้งานผ่านมาแล้ว CLI ของเราเตอร์จะเก็บ 10 คำสั่งที่เคยใช้งานแล้วไว้ในหน่วยความจำ]
6. Router#<ctl>P [แสดงคำสั่งที่เพิ่งใช้งานผ่านมา หรือจากคีย์บอร์ดลูกศรขึ้น]
7. Router#<ctl>N [แสดงคำสั่งต่อไป กรณีที่กลับไปใช้คำสั่งย้อนหลังแล้วต้องการกลับไปคำสั่งล่าสุด หรือกดคีย์บอร์ดลูกศรลง]
8. Router#show protocols [แสดงโพรโทคอลคันทรีเส้นทางที่กำลังทำงานอยู่]
9. Router#show version แสดงข้อมูลหลัก ๆ ของเราเตอร์ เช่น เราเตอร์แพลทฟอร์ม, รุ่นของ IOS, เวลาที่บูตระบบครั้งล่าสุด, ตำแหน่งที่อยู่ของไฟล์ ISO, หน่วยความจำ, จำนวนของอินเตอร์เฟสที่มีในระบบ, รีจิสเตอร์คอนฟิกภูเรชัน]
10. Router#show clock [แสดงสัญญาณนาฬิกาของเราเตอร์] ดังรูปที่ 4.16

```
Router#show clock
*00:38:35.755 UTC Mon Mar 1 1993
Router#
```

รูปที่ 4.16 ผลลัพธ์ของคำสั่ง show clock

11. Router#show hosts [แสดงรายชื่อของโಯส忒 และและแอดเดรสของมันทั้งหมด ในเบื้องต้นอาจจะไม่ปรากฏชื่อหรือข้อมูลใดๆ เนื่องจากยังไม่มีการคอนฟิก] ดังรูปที่ 4.17

```

Router#show hosts
Default domain is not set
Name/address lookup uses static mappings

Host Flags Age Type Address(es)
Router#

```

รูปที่ 4.17 ผลลัพธ์ของคำสั่ง show hosts

12. Router#*show users* [แสดงรายชื่อผู้ใช้งานทั้งหมดที่กำลังใช้งานอยู่ ในเบื้องต้นอาจจะไม่ปรากฏชื่อหรือข้อมูลใดๆ เนื่องจากยังไม่มีการคอนฟิก] ดังรูปที่ 4.18

```

Router#show users
Line User Host(s) Idle Location
* 0 con 0 idle 00:00:00

```

รูปที่ 4.18 ผลลัพธ์ของคำสั่ง show users

13. Router#*show interfaces* [แสดงรายละเอียดของอินเตอร์เฟสทั้งหมด]

14. Router#*show protocols* [แสดงชนิดของโปรโตคอลที่ใช้ทำงานและแสดงสถานะของอินเตอร์เฟสที่มีอยู่ในระบบทั้งหมด] ดังรูปที่ 4.19

```

Router#sh protocols
[s:
[redacted]
[redacted] protocol routing is enabled
Serial0 is administratively down, line protocol is down
[redacted]
[redacted] is administratively down, line protocol is down
Ethernet0 is administratively down, line protocol is down
Ethernet1 is administratively down, line protocol is down

```

รูปที่ 4.19 ผลลัพธ์ของคำสั่ง show protocols

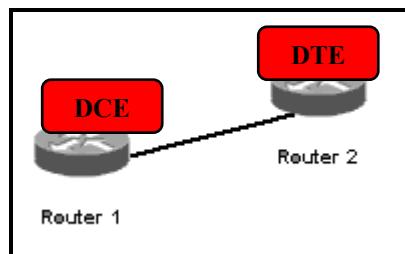
รู้จักกับ CDR (Cisco Discovery Protocol)

หน้าที่หลัก ๆ ของ CDP คือการส่งและรับข้อมูลของคอนฟิกภูเรียนพื้นฐานของเราเตอร์เพื่อบันทึกไว้ในการหาสาเหตุเมื่อเราเตอร์ทำงานผิดปกติ

จุดมุ่งหมาย : เข้าใจการทำงานของ CDR ว่ามีประโยชน์อย่างไร

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัวคือ R1 และ R2

การสร้าง Network Map : ให้สร้างผังเน็ตเวิร์คเหมือน ดังรูปที่ 4.20



รูปที่ 4.20 ผังเน็ตเวิร์คสำหรับ LAB 4

หน้าต่าง Simulator :

บนเราเตอร์ 1

1. Router>enable

Router#conf t

Router(config)# [เข้าสู่โหมดคอนฟิกกูเรชันของเราเตอร์ 1]

2. Router(config)#hostname R1
- R1(config)# [เปลี่ยนชื่อเราเตอร์จาก Router เป็น R1]
3. R1(config)#interface serial 0 [เข้าสู่อินเตอร์เฟส serial 0 ของเราเตอร์ R1]
- R1(config-if)#no shutdown [สั่งให้อินเตอร์เฟส serial 0 ของเราเตอร์ R1 ทำงาน]
- R1(config-if)#clock rate 5600 [เซ็ต Clock rate เพื่อให้ R1 สร้างสัญญาณนาฬิกาสำหรับใช้สื่อสารกันระหว่างแต่ละอินเตอร์ของเราเตอร์]

หมายเหตุ: ปกติทุก ๆ อินเตอร์เฟสจะถูก Disabled ไว้ (ไม่ทำงาน)

บันเราเตอร์ 2

1. Router>enable
- Router#conf t
- Router(config)# [เข้าสู่โหมดคอนฟิกกูเรชันของเราเตอร์ 2]
- Router(config)#hostname R2
- R2(config)# [เปลี่ยนชื่อเราเตอร์จาก Router เป็น R2]
2. R2(config)#interface serial 0 [เข้าสู่อินเตอร์เฟส serial 0 ของเราเตอร์ R2]
- R2(config-if)#no shutdown [สั่งให้อินเตอร์เฟส serial 0 ของเราเตอร์ R2 ทำงาน]

บันเราเตอร์ 1

1. R1(config)#interface ethernet 0 [เข้าสู่อินเตอร์เฟส ethernet 0 ของ R1]
- R1(config-if)#no shutdown [สั่งให้อินเตอร์เฟส ethernet 0 ของเราเตอร์ R1 ทำงาน]
- หมายเหตุ: CDR เป็นໂປຣໂທຄອລທີ່ໃຊ້ສໍາຮັບແລກເປົ້າຍັນຄອນພິກງູເຮັນທົ່ວໆ ຈະກ່ຽວຂ້ອງອຸປະກອນທີ່ເຊື່ອມຕ່ອງກັບມັນໂດຍຕຽນ ໂດຍທີ່ຕ້າ CDR ໄມໄດ້ເກີ່ວຂຶ້ອງກັບໂປຣໂທຄອລເຮົາຕິ່ງ ແລະ CDR ຈະຖຳການອູ່ທີ່ເລີຍອົບທີ່ 2
2. R1(config-if)#exit
- R1(config)#
- R1(config)#show cdp interface [แสดงข้อมูลของทุกอินเตอร์เฟสເມື່ອສັ່ງໃຫ້ CDP ทำงาน] ດັ່ງຮູບທີ່ 4.21

```
R1#sh cdp interface
Serial0 is up, line protocol is up
encapsulation HDLC
Sending CDP packets every 60 seconds
Holdtime is 180 seconds
```

ຮູບທີ່ 4.21 ຜົລັບພົບຂອງຄໍາສັ່ງ show cdp interface

3. R1#show cdp neighbors [แสดงรายชื่อของเพื่อนบ้านทั้งหมด ที่กำลังเชื่อมต่ออยู่] ดังรูปที่ 4.22

```
R1#sh cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
S -Switch, H - Host, i - IGMP, r - Repeater
Device ID      Local Intrfce     Holdtme   Capability Platform Port ID

```

รูปที่ 4.22 ผลลัพธ์ของคำสั่ง show cdp neighbors

หมายเหตุ: จากผลที่ได้จากคำสั่ง show cdp neighbors ที่เราเตอร์ R1 แสดงให้เห็นว่า R1 รับข้อมูลจาก R2 และส่งข้อมูลไปให้ R2 ด้วย โดยข้อมูลที่รับมีดังนี้

Device ID	คือ ชื่อของเราเตอร์ที่เชื่อมต่ออยู่ ในที่นี้คือเราเตอร์ R2
Local Interface	คืออินเตอร์เฟสที่เชื่อมต่อกัน ในที่นี้คือ Serial 0
Holdtme	คือระยะเวลาที่เราเตอร์ตั้งไว้สำหรับแก้ปัญหาการแลกเปลี่ยนข้อมูลระหว่างกัน ถ้าไม่มีการแลกเปลี่ยนข้อมูลเกินค่า Hold down time R1 จะลบข้อมูลของเพื่อนบ้านออกจากตารางเร้าติ้ง จากรูปมีค่าเท่ากับ 176 วินาที
Capability	แสดงว่ากำลังรับข้อมูลจาก R2
Platform	แสดงรุ่นของเราเตอร์เพื่อนบ้าน ในที่นี้ R2 คือรุ่น 2501
Port ID	เป็นพอร์ทที่เพื่อนบ้านใช้ในการเชื่อมต่อ ในที่นี้ R2 ใช้ serial 0 ติดต่อ

4. R1#show cdp neighbors detail [แสดงข้อมูลของเพื่อนบ้านโดยละเอียด] ดังรูปที่ 4.23

```
R1#sh cdp neighbors detail
-----
Device ID: R2
Entry address(es):
Platform: Boson 2501 , Capabilities: Router
Interface: Ser0, Port ID (outgoing port): 1
Holdtime: 174 sec

Version :
Boson Operating System Software
Software, Version 12.1(16), RELEASE SOFTWARE (fc2)
Copyright (c) 1986-2001 by Systems, Inc.
Compiled Fri 02-Mar-01 17:34 by dchih
```

รูปที่ 4.23 ผลลัพธ์ของคำสั่ง show cdp neighbors detail

5. R1#show cdp [แสดงระยะเวลาสำหรับทำการอัพเดทข้อมูลระหว่างกันรวมถึงระยะเวลาของ Hold Down Timer ที่ตั้งไว้โดยดีฟอลท์] ดังรูปที่ 4.24

```
R1#sh cdp
Global CDP information:
  Sending CDP packets every 60 seconds
  Sending a holdtime value of 180 seconds
  Sending CDPv2 advertisements is enabled
```

รูปที่ 4.24 ผลลัพธ์ของคำสั่ง show cdp

6. R1#conf t

R1(config)#cdp timer 45 [เปลี่ยนค่าระยะเวลาของการอัปเดทข้อมูลของ CDP] ดังรูปที่ 4.25

```
R1#sh cdp
Global CDP information:
    Sending a holdtime value of 180 seconds
    Sending CDPv2 advertisements is enabled
```

รูปที่ 4.25 ผลลัพธ์ของคำสั่ง cdp timer 45

7. R1#conf t

R1(config)#cdp holdtime 60 [เปลี่ยนระยะเวลาของ Hold Down Timer] ดังรูปที่ 4.26

```
R1#sh cdp
Global CDP information:
    Sending CDP packets every 45 seconds
```

รูปที่ 4.26 ผลลัพธ์ของคำสั่ง cdp holdtime 45

หมายเหตุ : เมื่อมีอุปกรณ์อื่นๆ เชื่อมต่ออยู่เลย (มีอุปกรณ์เราเตอร์เพียงตัวเดียว) ไม่ควรทำการ Enable CDP ไว้ เนื่องจากไม่เกิดประโยชน์อะไรและลดภาระของอุปกรณ์ลงด้วย

8. R1#conf t

R1(config)#no cdp run [Disable CDP บนเราเตอร์ R1]

9. R1#conf t

R1(config)#cdp run [Enable CDP บนเราเตอร์ R1]

หมายเหตุ : คุณสามารถ Disable CDP เนพะบางอินเตอร์เฟสก็ได้ ถ้าอินเตอร์เฟสนั้น ๆ มีช่องทางส่งสัญญาณแคบ

10. R1(config)#interface ethernet 0

R1(config-if)#no cdp enable [Disable CDP เนพะบางอินเตอร์เฟส ในที่นี่คือ อินเตอร์เฟส Ethernet 0 ของเราเตอร์ R1]

R1#show cdp interface [แสดงอินเตอร์เฟสที่ enable CDP ไว้ ส่วน ethernet 0 ไม่แสดงให้เห็นเพราะได้ Disable CDP ที่อินเตอร์เฟสนี้ไปแล้ว]

💻 เรียนรู้การการคونฟิกเราเตอร์พื้นฐาน

จุดมุ่งหมาย : เรียนรู้การคุณฟิกเราเตอร์ขั้นพื้นฐานเพิ่มเติม

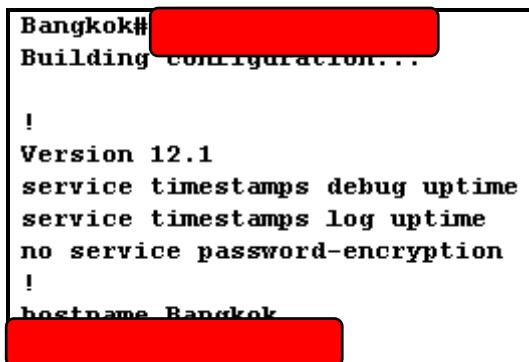
เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัว

การสร้าง Network Map : ให้สร้างผังเน็ตเวิร์คเหมือน รูปที่ 4.20

หน้าต่าง Simulator :

1. Router>enable
- Router#conf t
- Router(config)# [เข้าสู่โหมดคอนฟิกกูเรชันของเราเตอร์ 1]
2. Router(config)#hostname Bangkok [เปลี่ยนชื่อให้กับเราเตอร์ใหม่]
3. Bangkok(config)#enable password 123 [เซ็ตค่ารหัสผ่านที่จะเข้าสู่โหมด privilege เป็น 123 เนื่องจากโหมดนี้มีความสามารถเปลี่ยนแปลงค่าของคอนฟิกบางอย่างได้ จึงสมควรที่จะต้องมีการตรวจสอบสิทธิของผู้ที่จะเข้าใช้งาน]

หมายเหตุ : ให้ลอง Logout จากเราเตอร์และลอง Login ใหม่แล้วดูผลการทำงาน เนื่องจากการ enable รหัสผ่านแบบนี้จะไม่มีการเข้ารหัส จึงไม่ปลอดภัยและจำเป็นที่จะต้องใช้คำสั่งสำหรับเข้ารหัส เพื่อความปลอดภัยอีกครั้ง ดังรูปที่ 4.27



```
Bangkok#
Building configuration...
!
Version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Bangkok
```

รูปที่ 4.27 แสดงรหัสผ่านที่ไม่มีการเข้ารหัส

4. Bangkok(config)#enable secret 123 [เป็นการเข้ารหัสผ่านเพื่อเข้าโหมด privilege]
- หมายเหตุ : ถ้ามีการเซ็ตค่าของรหัสผ่านที่เป็นทั้งแบบไม่เข้ารหัสและแบบที่เข้ารหัสไว้ทั้งคู่ เราเตอร์จะให้ความสำคัญ secret มากกว่า เมื่อ login เข้ามาจะต้องป้อนรหัสของ secret
5. ทดลอง logout และ Login ใหม่อีกครั้ง และดูการเปลี่ยนแปลงที่เกิดขึ้น

█ การกำหนดแบบเนอร์ให้เราเตอร์ (MOTD Message of the Day)

MOTD เป็นคำสั่งที่ใช้สำหรับแสดงข้อความ เมื่อทำการ login เข้าสู่เราเตอร์ ซึ่งข้อความที่แนะนำให้เขียนควรจะเป็นข้อความที่เป็นคำเตือน เช่น ข้อความที่เกี่ยวกับการละเมิด การบุกรุก การโจมตี ยกตัวอย่าง "การบุกรุก ที่มีจุดประสงค์เพื่อสร้างความเสียหาย เป็นสิ่งที่ผิดกฎหมาย หากละเมิด จะถูกดำเนินการตามกฎหมาย" เป็นต้น อย่าแสดงข้อความที่แสดงถึง รุน ยี่ห้อ ของอุปกรณ์ เพราะจะเป็นสิ่งที่ผู้ไม่หวังดี นำไปโจมตี หรือบุกรุกได้

จุดมุ่งหมาย : เรียนรู้การสร้างแบบเนอร์ให้เราเตอร์

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัว

การสร้าง Network Map : ให้สร้างผังเน็ตเวิร์คเหมือน รูปที่ 4.20

หน้าต่าง Simulator :

1. Router>

Router>enable

Router#config t

Router(config)# [เข้าสู่โหมดคอนฟิกชันของเราเตอร์ 1]

2. Router(config)#banner motd z [เป็นการสร้างแบบเนอร์ให้กับเราเตอร์ โดยที่ตัวอักษร z เป็นสัญลักษณ์ที่ใช้สำหรับบอกให้ทราบว่าเป็นตัวปิดหัวและท้ายของข้อความ (delimiting) คือให้เราเตอร์ทราบว่าตัวอักษรที่ต่อจากคำสั่ง motd คือตัวที่แสดงจุดเริ่มต้นของข้อความและมันจะต้องเป็นตัวปิดท้ายของข้อความด้วย] ดังรูปที่ 4.28

หมายเหตุ : ตัวอักษรที่เป็นตัวปิดหัวและท้ายของข้อความจะเป็นตัวอักษรกีด้ แต่ต้องเหมือนกัน ทั้ง 2 ตัว

```
Router(config)# Enter the text followed by the 'z' to finish
```

```
Router(config)#
```

รูปที่ 4.28 แสดงการสร้างแบบเนอร์ของเราเตอร์

3. You do not have permission to be here. This router for eats hacker for lunch! z [ใส่ข้อความเตือน และวิจารณาด้วย z สำหรับใช้เป็นตัวปิด]

4. ทดลอง logout และ login ใหม่อีกครั้ง สังเกตข้อความที่แสดงบนเราเตอร์ ดังรูปที่ 4.29

```
You do not have permission to be here. this router for eats hacker lunch!
```

```
Router>
```

```
Router>
```

รูปที่ 4.29 เราเตอร์แสดงแบบเนอร์เมื่อทำการ login

█ เรียนรู้คำสั่ง Copy

คำสั่ง Copy มีไว้สำหรับใช้ประโยชน์ในการสำเนาข้อมูลของไฟล์ไปเก็บไว้อีกที่หนึ่ง เช่น การสำเนาข้อมูลจากรันนิ่งคอนฟิกชันที่อยู่ในหน่วยความจำหลักไปเก็บที่ NVRAM หรือสำเนาข้อมูลจาก NVRAM ไปยัง TFTP เซิร์ฟเวอร์ หรือสำเนาข้อมูลจาก TFTP เซิร์ฟเวอร์มาเก็บไว้ในรันนิ่งคอนฟิก ก็ได้

จุดมุ่งหมาย : เรียนรู้การใช้คำสั่ง Copy

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัว

การสร้าง Network Map : ให้สร้างผังเน็ตเวิร์คเหมือนรูปที่ 4.20

หน้าต่าง Simulator :

1. Router>

Router>enable

Router#

2. Router#*show running-config* [แสดงรันนิ่งคอนฟิกของเราเตอร์ที่กำลังทำงานอยู่ในหน่วยความจำหลัก รันนิ่งคอนฟิกจะไม่ได้ถูกบันทึกไว้ให้อัตโนมัติ ผู้ดูแลระบบจะต้องบันทึกเอง ถ้าเราเตอร์เริ่มทำงานใหม่รันนิ่งคอนฟิกนั้นจะสูญหายไป แต่นั่นจะโหลดคอนฟิกจาก NVRAM มาทำงาน เมื่อเราคอนฟิกที่ CLI ด้วยคำสั่งได ๆ ก็ตามถ้ายังไม่มีการบันทึกลง NVRAM คำสั่งนั้น ๆ จะจะสูญหายไปเนื่องจาก การรีเซ็ตระบบใหม่ ถ้าต้องการบันทึกคอนฟิกจะต้องใช้คำสั่ง copy]
3. Router#*show startup-config* [แสดงคอนฟิกไฟล์ที่อยู่ใน NVRAM]
4. Router#*copy running-config startup-config* [บันทึกค่าของรินนิ่งคอนฟิกที่กำลังทำงานอยู่ไปยัง NVRAM เพื่อป้องกันการเสียหายเนื่องจากการรีบูตเราเตอร์]
5. Router#*show startup-config* [ทดลองแสดง startup-config ใน NVRAM]
6. Router#*erase startup-config* [ลบคอนฟิกเรชันไฟล์ของเราเตอร์ทั้งหมด ออกซึ่งจะใช้มือต้องการเริ่มต้นการเริ่มต้นการคอนฟิกเราเตอร์ใหม่ทั้งหมด]
7. Router#*reload* [เมื่อลบคอนฟิกกูเรชันไฟล์ทั้งหมดออกแล้วให้สั่ง reload เพื่อเริ่มต้นการทำงานของเราเตอร์อีกครั้ง]
8. Router#*config terminal*

Router(config)#*hostname Bangkok*

Router(config)#*exit*

Router#*reload* [เปลี่ยนชื่อของเราเตอร์เป็น Bangkok และทดลอง reload เราเตอร์จะถามว่าให้บันทึกคอนฟิกกูเรชันไฟล์หรือไม่ให้ตอบว่า yes]

Bangkok> [ชื่อของเราเตอร์จะไม่หายไปเนื่องจากมีการบันทึกไว้ที่ NVRAM เรียบร้อยแล้ว]

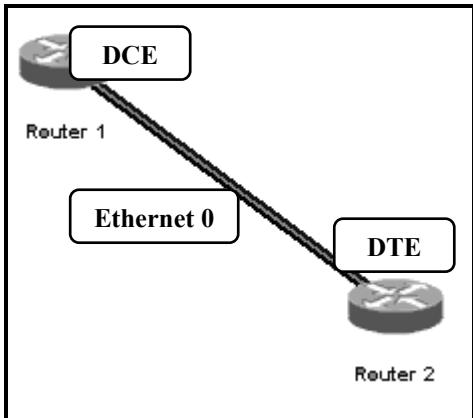
การคอนฟิกอินเตอร์เฟส

เราเตอร์มีอินเตอร์เฟสที่ใช้มต่อ กับ อุปกรณ์อื่น ๆ ได้หลายชนิด เช่น token ring, Ethernet, Serial, ISDN เป็นต้น ปอยครั้งที่เราจำเป็นที่จะต้องตรวจสอบสถานะของแต่ละอินเตอร์เฟส ซึ่งจะมีอยู่หลายคำสั่งที่ใช้แสดงข้อมูลของอินเตอร์เฟส ในส่วนนี้เราจะเรียนรู้ว่าจะสั่งให้แต่อินเตอร์เฟสทำงานได้อย่างไร และจะแสดงรายละเอียดของแต่ละอินเตอร์เฟสได้อย่างไร

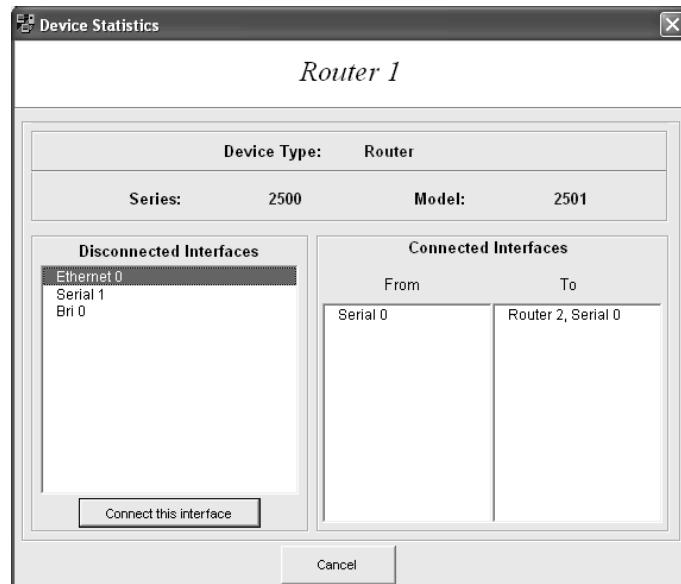
จุดมุ่งหมาย : เรียนรู้การใช้คำให้แต่อินเตอร์เฟสทำงานและหยุดทำงาน

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 2 ตัวคือ Router1 และ Router2

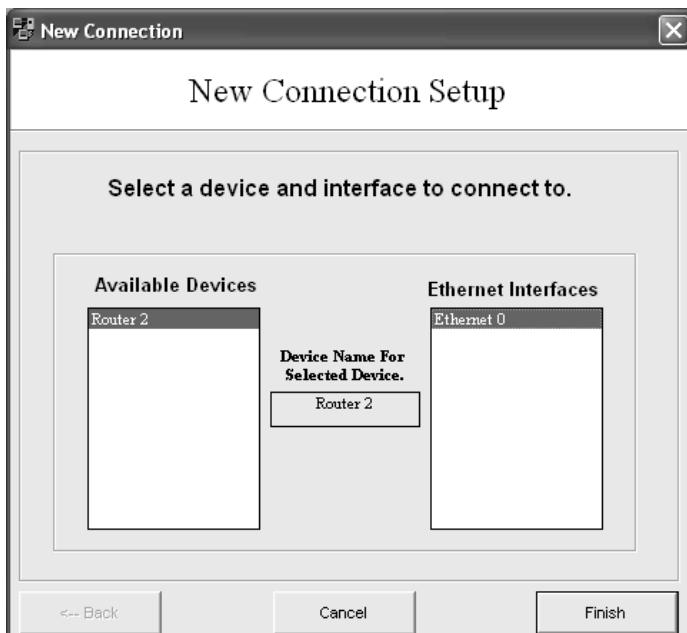
การสร้าง Network Map : ให้สร้างผังเน็ตเวิร์ค ดังรูปที่ 4.30



รูปที่ 4.30 ผังเน็ตเวิร์ค LAB 8



รูปที่ 4.31 การเชื่อมต่อ Ethernet to Ethernet ที่เราเตอร์ 1



รูปที่ 4.32 การเชื่อมต่อ Ethernet to Ethernet 2

หน้าต่าง Simulator :

1. Router>

Router>enable

Router#config t

Router(config)#hostname Router1 [เปลี่ยนชื่อเราเตอร์เป็น Router1]

2. Router1(config)#interface ethernet 0 [เข้าสู่โหมดคอนฟิกอินเตอร์เฟส
ethernet 0]

Router1(config-if)#? [แสดงคำสั่งทั้งหมดในโหมดอินเตอร์เฟส] ดังรูปที่ 4.33

Router1(config-if)	
exit	Exit from interface configuration mode
shutdown	Shutdown the selected interface
end	Exit Configuration Mode
cdp	CDP interface subcommands
ip	Interface Internet Protocol config commands
description	Interface specific description
interface	Select an interface to configure

รูปที่ 4.33 แสดงคำสั่ง ? ในโหมดคอนฟิกอินเตอร์เฟส

3. Router1(config-if)#no shutdown [ใช้สกุลเราเตอร์จะใช้คำสั่ง no เพื่อยกเลิกคำสั่งที่อยู่ในรันนิ่งคอนฟิกกูเรชัน ในการนี้ ethernet 0 จะไม่ทำงานโดยดีฟอลท์แต่ถ้า ใช้คำสั่งข้างต้นอินเตอร์เฟสนี้จะทำงานทันที] ดังรูปที่ 4.34

Router1(config-if)#no shutdown

Router1(config-if)#[red box]

รูปที่ 4.34 แสดงสถานะของอีเทอร์เน็ต 0 หลังจากใช้คำสั่ง no shutdown

4. Router1(config-if)#description Ethernet Interface on Router 1 [ใส่คำอธิบายไปยังอินเตอร์เฟสอีเทอร์เน็ต 0 เพื่อใช้อธิบายว่าอินเตอร์เฟสชนานี้ทำหน้าที่อะไร]

5. Router1(config-if)#end

Router1#show interface [แสดงรายละเอียดของแต่ละอินเตอร์เฟส ให้สังเกตว่าอินเตอร์เฟสอีเทอร์เน็ต 0 ว่ามีการเปลี่ยนแปลงอย่างไรบ้าง]

6. ให้เชื่อมต่อเข้าไปยังเราเตอร์ตัวที่สอง

Router>

Router>enable

Router#config term

Router(config)#hostname Router2 [กำหนดชื่อเราเตอร์เป็น Router2]

Router2(config)#interface ethernet 0 [เข้าสู่โหมดคอนฟิกอินเตอร์เฟสอีเทอร์เน็ต 0]

Router2(config-if)#[red box]

7. Router2(config-if)#no shutdown [สั่งให้อินเตอร์เฟสอีเทอร์เน็ต 0 ทำงาน]
8. Router2(config-if)#end
- Router2#show cdp neighbors [แสดงรายละเอียดเพื่อนบ้านที่เชื่อมต่ออยู่ด้วย]

ดังรูปที่ 4.35

Router2#sh cdp neighbors							
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge	S -Switch, H - Host, i - IGMP, r - Repeater	Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID

รูปที่ 4.35 แสดงรายละเอียดของเพื่อนบ้านโดยใช้ sh cdp neighbors

ตารางที่ 4.1 สถานะของอินเตอร์เฟส

Interface eth 0 is	Line protocol is	ความหมาย
Administratively Down	Down	สั่งด้วยคำสั่ง shutdown โดยผู้ดูแลระบบ
Up	Down	สายสัญญาณเชื่อมต่ออยู่แต่ ข้อมูลที่เรียกว่า keep alive ไม่มี
Down	Down	สายสัญญาณมีปัญหาหรือ DCE ไม่ได้จ่ายสัญญาณนาฬิกา หรือเราเตอร์ตัวอื่น ๆ ไม่ทำงาน
Up	Up	ทำงานได้เป็นปกติ (สิ่งที่เราต้องการ)

แบบฝึกหัดท้ายบท

1. เมื่อเข้าสู่การทำงานของโหมด user ถ้าต้องการจะแสดงคำสั่งทั้งหมดที่ใช้งานในโหมดนี้ คุณจะใช้คำสั่งอะไร? _____

2. เมื่อคุณต้องการเข้าสู่โหมด privilege จะต้องใช้คำสั่งอะไร? _____

3. เมื่อเข้าสู่การทำงานของโหมด privilege ถ้าต้องการจะแสดงคำสั่งทั้งหมดที่ใช้งานในโหมดนี้ คุณจะใช้คำสั่งอะไร? _____

4. ทำอย่างไรถ้าคุณต้องการแสดงคำสั่งที่ใช้งานร่วมกับคำสั่ง show (ใช้คำสั่งอะไร)? _____

5. ต้องการแสดงไฟล์คอนฟิกภูเรชันที่ทำงานในปัจจุบันจะต้องใช้คำสั่งอะไร? _____

6. เมื่อต้องการออกไปสู่โหมด User ควรจะใช้คำสั่งอะไร? _____

1. เมื่อคุณเข้าสู่เราเตอร์แล้วจะใช้คำสั่งอะไรเพื่อจะเข้าสู่ prompt ที่มีรูปแบบเป็น Router# ? _____

2. ต้องการแสดงรันนิ่งคอนฟิกจะใช้คำสั่งอะไร? _____

3. ต้องการแสดงข้อมูลของหน่วยความจำแบบแฟลสจะใช้คำสั่งอะไร? _____

- จงแสดงชื่อของ IOS? _____

- จงแสดงขนาดของ IOS? _____
 - จงแสดงปริมาณหน่วยความจำแฟร์สที่ยังไม่ได้ใช้งาน? _____
4. ต้องการแสดงไฟร์โทคอลคันหนาเส้นทางที่กำลังทำงานอยู่ใช้คำสั่งอะไร? _____
- ไฟร์โทคอลอะไรที่กำลังทำงาน? _____
 - มีจำนวนอินเตอร์เฟสเท่าไหร่ที่กำลังทำงาน (up) และมีอินเตอร์เฟสที่ผู้ดูแลระบบไม่สั่งให้ทำงาน (Administratively down)? _____
5. ใช้คำสั่งอะไรที่จะแสดงคำสั่งที่ได้ใช้งานผ่านมาแล้ว? _____
6. กดปุ่มคีย์บอร์ดอะไรที่แสดงคำสั่งก่อนหน้า? _____ และ _____
7. คำสั่งอะไรที่แสดง เรายาเตอร์แพลทฟอร์ม, รุ่นของ IOS, เวลาที่บูตระบบครั้งล่าสุด, ตำแหน่งที่อยู่ของไฟล์ ISO, หน่วยความจำ, จำนวนของอินเตอร์เฟสที่มี, รีจิสเตร์คอนฟิกกูเรชัน? _____
- IOS เก็บอยู่ที่ไหน ? _____
 - แพลทฟอร์มของเราเตอร์คืออะไร? _____
 - จำนวนความจุของ NVRAM? _____
 - ค่าของรีจิสเตร์คอนฟิกกูเรชัน? _____
 - จำนวนของอินเตอร์เฟสแบบอีเทอร์เน็ตและแบบซีเรียลมีจำนวนเท่าไหร? _____
8. คำสั่งอะไรที่แสดงวันและเวลาของเราเตอร์? _____
9. เราเตอร์ใช้เวลาไปทำอะไร? _____
10. คำสั่งอะไรใช้แสดงไอสต์ทั้งหมดบนเราเตอร์? _____
11. คำสั่งอะไรใช้แสดงผู้ใช้ทั้งหมดที่กำลังทำงานบนเราเตอร์? _____
12. คำสั่งอะไรแสดงค่าของโกลบอลและสถานะของอินเตอร์เฟส? _____
13. เมื่อคุณทำงานอยู่บนเราเตอร์แล้วต้องการแสดงคำสั่งทั้งหมดของเราเตอร์ที่มีอยู่ต้องใช้คำสั่งอะไร? _____
14. ถ้าต้องการเข้าสู่การทำงานในโหมด privilege จะต้องใช้คำสั่งอะไร? _____
15. เมื่อเข้าสู่โหมด privilege และต้องการดูคำสั่งทั้งหมดที่ใช้งานได้ จะต้องใช้คำสั่งอะไร? _____
16. จะคำสั่งอะไร เมื่อต้องการเข้าสู่โหมดคอนฟิก? _____
17. ถ้าต้องการเปลี่ยนชื่อของเราเตอร์เป็น Thailand จะต้องใช้คำสั่งอะไร? _____
18. ถ้าต้องการกำหนดรหัสผ่านเป็น abc ในโหมด privilege จะใช้คำสั่งอะไร? _____
19. เมื่อเปลี่ยนรหัสผ่านแล้ว ให้ทดสอบ logout และ login ใหม่อีกครั้ง
20. ถ้าต้องการกำหนดรหัสผ่านแบบเข้ารหัสเป็น 123 ในโหมด privilege จะใช้คำสั่งอะไร? _____

21. เมื่อเปลี่ยนรหัสผ่านแล้ว ให้ทดสอบ logout และ login ใหม่อีกรัง ให้ตรวจสอบว่าเราเตอร์ใช้รหัสผ่านตัวไหนเข้าสู่การทำงานของโหมด privilege

22. ให้ล็อกอินเข้าไปปั้งเราเตอร์และเข้าสู่โหมด privilege.

1. จงแสดงรันนิ่งคอนฟิก? _____
2. จงแสดงคอนฟิกภูเรชันไฟล์ที่อยู่ใน NVRAM? _____
3. ให้ก็อบปี้ running-config ไปยัง NVRAM? _____
4. จงแสดงคอนฟิกภูเรชันไฟล์ที่อยู่ใน NVRAM อีกรัง? _____
5. ลบคอนฟิกภูเรชันไฟล์ที่อยู่ใน NVRAM? _____
6. reload เราเตอร์และไม่ต้องบันทึกข้อมูล? _____
7. แสดงคอนฟิกภูเรชันไฟล์ที่อยู่ใน NVRAM อีกรัง? _____
8. เปลี่ยนชื่อของเราเตอร์เป็น bangkok? _____

23. reload เราเตอร์และให้บันทึกผลการทดลอง? _____

24. จงแสดงคำสั่งที่เมื่อต้องการเข้าคอนฟิกอินเตอร์เฟสอีเทอร์เน็ต 0 ? _____

1. ปกติทุก ๆ อินเตอร์เฟสจะมีสถานการณ์ทำงานเป็นอย่างไร? _____
2. คำสั่งอะไรที่ใช้สั่งให้อินเตอร์เฟสทำงาน? _____
3. คำสั่งอะไรที่สั่งให้อินเตอร์เฟสหยุด
ทำงาน? _____
4. แสดงคำสั่งที่แสดงถึงเพื่อนบ้านที่เชื่อมต่อ
อยู่? _____

บทที่ 5

การคอนฟิกอุปกรณ์สวิชต์ (Switch Configuration)

Switch Configuration

```
C:\WINNT\system32\telnet.exe
User Access Verification

Password:
Sales>ena
Password:
Sales#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Sales<config>#ena sec cisco2
Sales<config>#hostname Sales2
Sales2<config>#int vlan 1
Sales2<config-if>#ip addr 192.168.1.200 255.255.255.0
Sales2<config-if>#line vty 0 4
Sales2<config-line>#passw cisco
Sales2<config-line>#login
Sales2<config-line>#cdp run
Sales2<config>#int fa0/1
Sales2<config-if>#no shut
Sales2<config-if>#speed auto
Sales2<config-if>#int fa0/2
Sales2<config-if>#no shut
Sales2<config-if>#speed auto
Sales2<config>#int fa0/3
```

- Switch basics
- Virtual LANs (VLANs)
- Trunking
- Spanning tree
- Advanced switching

แนวคิด

ในส่วนนี้จะใช้สำหรับฝึกหัดการคอนฟิกอุปกรณ์สวิชต์บนระบบเครือข่ายผ่านตัวจำลองเนื้อหาในแต่ละแล็บนั้นจะมีลักษณะที่ต่อเนื่องและเรียงลำดับตามความสำคัญของเนื้อหาในการติดตั้งและดูแลระบบเครือข่าย

วัตถุประสงค์

- เพื่อให้สามารถปรับแต่งอุปกรณ์สวิชต์ที่ใช้งานบนระบบเครือข่ายได้อย่างเหมาะสมเพื่อสร้างความชำนาญในการใช้งานอุปกรณ์ที่สำคัญๆ บนระบบเครือข่าย

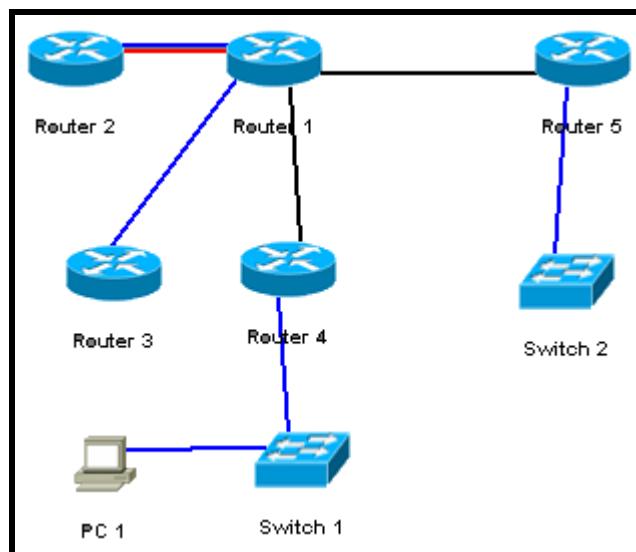
▣ ความรู้เบื้องต้นเกี่ยวกับอุปกรณ์สวิตช์

สวิตช์ทำงานที่เลเยอร์ 2 ของ OSI โมเดล (Data Link Layer) โดยหน้าที่หลักคือการรวมเครื่องผู้ใช้งานเข้ามาในเน็ตเวิร์ค และเชื่อมต่ออุปกรณ์อื่นเข้ามาสู่ระบบ เช่น เซิร์ฟเวอร์ สัมภาระ และสวิตช์ส่วนประกอบของสวิตช์ก็คล้าย ๆ กับ PC มันก็จะประกอบไปด้วย CPU, RAM และระบบปฏิบัติการ (IOS) การดูแลและจัดการกับสวิตช์ก็ทำได้เหมือนกับเราเตอร์คือ คอนฟิกผ่านพอร์ตคอนโซล ผ่านการเทลเน็ต และสามารถปรับเปลี่ยน IOS ได้ เช่น กับ สวิตช์จะใช้คำสั่ง show interface ต้องการหาข้อมูลเกี่ยวกับ IOS ให้ใช้คำสั่ง show version หรือเมื่อต้องการแสดงคอนฟิกเรียนที่กำลังทำงานอยู่จะใช้คำสั่ง show running-config และมีบางคำสั่งที่ไม่เหมือนกัน เช่น ต้องการทราบข้อมูลของ MAC Address จะใช้คำสั่ง show mac-address-table เป็นต้น

จุดมุ่งหมาย : เรียนรู้วิธีการคอนฟิกสวิตช์เบื้องต้น

เครื่องมือที่ใช้ทดลอง : ใช้สวิตช์ 1 ตัวคือ switch 1

การสร้าง Network Map : ใช้ผังเน็ตเวิร์คตามรูปที่ 5.1



รูปที่ 5.1 ผังเครือข่าย

หน้าต่าง Simulator :

1. บน Switch 1 เมื่อ Enter จะปรากฏ > แสดงว่าพร้อมรับคำสั่ง
>
2. แสดงเวอร์ชันของ IOS บนสวิตช์ ดังรูปที่ 5.2
>show version

```
>sh version
Boson Operating Switch Simulator (BOSS) 1900/2820 Enterprise Edition
Version V5.0
Copyright (c) Boson Software, Inc. 1998-2003
  uptime is 0 days, 0 hours, 55 minutes
Switch 1912 (BOSS) processor with 2048K/1024K bytes of simulated memory
Emulator revision is 5
Upgrade Status: No upgrade currently in progress.
Config File Status: No configuration upload/download is in progress
14 Fixed Ethernet/IEEE 802.3 interface(s)
Base Ethernet Address: 00-0C-69-01-96-31
```

รูปที่ 5.2 show version

- IOS เวอร์ชันอะไร?
 - หมายเลข Model ของสวิตซ์คือ ?
 - หมายเลข Base Internet Address คือ ?
3. แสดงอินเตอร์เฟสของสวิตซ์
- >show interface
- มีจำนวนของอินเตอร์เฟสที่มีความเร็ว 10 Mbps ?
 - มีจำนวนของอินเตอร์เฟสที่มีความเร็ว 100 Mbps ?
4. แสดงข้อมูลของตาราง MAC Address ดังรูปที่ 5.3

>show mac-address-table

Mac Address Table			
Vlan	Mac Address	Type	Ports
---	---	---	---
1	000C.4425.5252	DYNAMIC	Ethernet0/2

Total Mac Addresses for this criterion: 1

รูปที่ 5.3 show mac-address-table

- มีจำนวนของ MAC Address ในตาราง ?
5. แสดงคอนฟิกกูเรชันที่กำลังทำงานอยู่

>show running-config

💻 คำสั่งเบื้องต้นบนอุปกรณ์สวิตซ์

จุดมุ่งหมาย : เรียนรู้วิธีการคอนฟิกสวิตซ์เบื้องต้นกับสวิตซ์รุ่น 1912

เครื่องมือที่ใช้ทดลอง : ใช้สวิตซ์ 1 ตัวคือ switch 1

การสร้าง Network Map : ใช้ผังเน็ตเวิร์คตามรูปที่ 5.1

หน้าต่าง Simulator :

1. บน Switch 1 ให้กด Enter เพื่อเข้าสู่โหมด User

>

2. แสดงคำสั่งทั้งหมดที่สามารถใช้ได้ในโหมดผู้ใช้ (user) โดยใช้คำสั่ง ?

```
>?
```
 3. โหมด privilege จะมีความสามารถควบคุมการทำงานของสวิตช์ได้ทั้งหมด

```
>enable [เข้าสู่โหมด privilege]
```
 4. แสดงคำสั่งที่สามารถใช้งานได้ทั้งหมดในโหมด privilege

```
#?
```
 5. เมื่อต้องการคอนฟิกสวิตช์จะใช้คำสั่ง config terminal เมื่อونกับเราเทอร์มิ널

```
#config term
```
 6. กำหนดชื่อของ สวิตช์ ให้เป็น Boson

```
(config)#hostname Boson
```
 7. ในสวิตช์ตระกูล 1900 สามารถกำหนดรหัสผ่านที่เข้าสู่โหมด privilege ได้หลายระดับ

```
Boson(config)#enable password level 15 boson [กำหนดรหัสผ่านเป็น boson ระดับที่ 15]
```
 8. เมื่อกำหนดรหัสผ่านแล้วต้องการทดสอบ ให้ออกไปสู่โหมด User และล็อกอินเข้ามาใหม่อีกครั้ง

```
Boson(config)#exit
```
 9. รหัสผ่านที่กำหนดในขั้นตอนที่ 8 จะเป็นแบบ Plain text ซึ่งจำไม่มีการเข้ารหัสทำให้ไม่ปลอดภัย ตัวสวิตช์จึงเตรียมคำสั่งเพื่อให้ผู้ใช้งานสามารถเข้ารหัสผ่านได้

```
Boson#config terminal
```
- Boson(config)#enable secret level 15 cisco [กำหนดรหัสผ่านเป็น cisco และเข้ารหัสข้อมูลด้วย ถ้ามีทั้งรหัสผ่านแบบ plain text และ Secret สวิตช์จะมองว่า Secret สำคัญมากกว่า] ดังรูปที่ 5.4

```
!
enable secret 5 89E$S3634D$sd0923SD4837
enable password level 15 "boson"
!
```

รูปที่ 5.4 แสดงรหัสผ่านที่เป็นแบบ plain text และแบบ secret

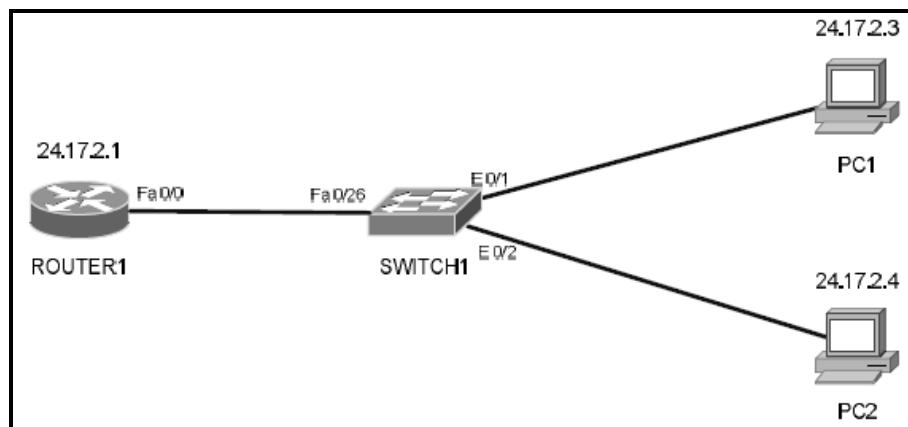
ออกไปสู่ User โหนดแล้วทดสอบใส่รหัสผ่านที่ตั้งไว้ สังเกตว่ารหัสผ่านชนิดไหนที่สามารถเข้าสู่โหนด privilege ได้

เบื้องต้นกับ VLAN

จุดมุ่งหมาย : เรียนรู้การคอนฟิก VLAN

เครื่องมือที่ใช้ทดลอง : ใช้เราเตอร์ 1 ตัว (Router1 2516) สวิตช์ 1 ตัว (1900 series) และ PC 2 เครื่อง

การสร้าง Network Map : ดังรูปที่ 5.5



รูปที่ 5.5 ผังเน็ตเวิร์ค

Simulator :

1. บนเราเตอร์ 1 ให้เปลี่ยนชื่อเป็น Router1 และกำหนดไอพีแอดเดรสเป็น 24.17.2.1 255.255.255.0 บนอินเตอร์เฟสอีเทอร์เน็ต 0

Router>enable

Router#config terminal

Router(config)#hostname Router1

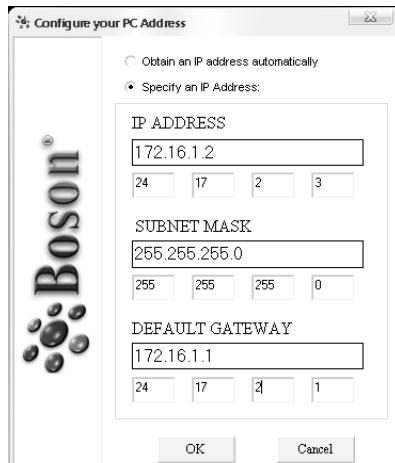
Router1(config)#

Router1(config)#interface ethernet 0

Router1(config-if)#ip address 24.17.2.1 255.255.255.0

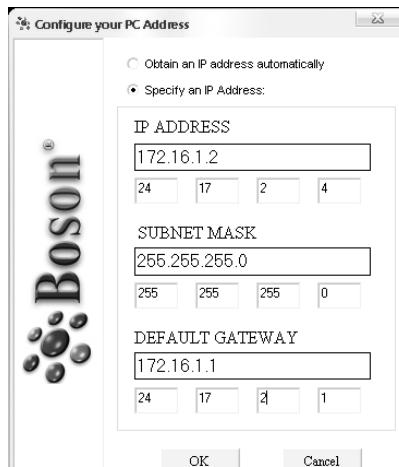
Router1(config-if)#no shutdown

2. บน PC1 ให้กำหนดไอพีแอดเดรสเป็น 24.17.2.3 255.255.255.0 gateway เป็น 24.17.2.1 โดยใช้คำสั่ง winipcfg ดังรูปที่ 5.6



รูปที่ 5.6 ระบบอุปกรณ์แอดเดรสให้กับ PC1 ด้วยคำสั่ง winipcfg

- บน PC2 ให้กำหนดอุปกรณ์แอดเดรสเป็น 24.17.2.4 255.255.255.0 gateway เป็น 24.17.2.1 ดังรูปที่ 5.7



รูปที่ 5.7 ระบบอุปกรณ์แอดเดรสให้กับ PC2 ด้วยคำสั่ง winipcfg

- ถึงขั้นตอนนี้คุณควรจะสามารถ ping จาก PC2 ไปยัง Router1 และ PC1 ได้แล้ว ดังรูปที่ 5.8, 5.9

```
C:>ping 24.17.2.1
Pinging 24.17.2.1 with 32 bytes of data:

Reply from 24.17.2.1: bytes=32 time=60ms TTL=241
```

รูปที่ 5.8 ping จาก PC2 ไปยัง Router1

```
C:>ping 24.17.2.3
Pinging 24.17.2.3 with 32 bytes of data:

Reply from 24.17.2.3: bytes=32 time=60ms TTL=241
```

รูปที่ 5.9 ping จาก PC2 ไปยัง PC1

5. ขั้นตอนนี้จะเป็นการค่อนพิก VLAN ปกติทุก ๆ อินเตอร์เฟสจะอยู่ที่ VLAN 1 อัตโนมัติ เราจะเริ่มแยก VLAN ออกจาก VLAN 1 ซึ่งจะเป็น VLAN ที่เราสร้างขึ้นมาเอง บนสวิตช์ 1 ให้สร้าง VLAN หมายเลข 22 มีชื่อเป็น pcs

>en

#config terminal

(config)#vlan 22 name pcs

6. เมื่อสร้าง vlan แล้ว ต่อไปจะต้องกำหนดพอร์ตที่จะใช้งานกับ vlan นี้ ในที่นี่กำหนดให้พอร์ต e0/1 เป็นสมาชิกของ vlan นี้

(config)#interface e0/1

(config-if)#vlan-membership static 22 [ระบุให้พอร์ตหรืออินเตอร์เฟส e0/1 เป็นสมาชิกของ vlan 22 แบบถาวร]

7. ถึงจุดนี้เราได้กำหนด vlan ให้กับ PC1 เป็นสมาชิกของ vlan 22 แล้วให้ทดสอบ ping จาก PC2 ไปยัง Router1 และ PC1 เมื่อ้อนในขั้นตอนที่ 4 อีกครั้ง บน PC2

C:>ping 24.17.2.1

C:>ping 24.17.2.3

ผลที่ได้เราไม่สามารถจะ ping PC1 ได้เนื่องจาก เราได้สร้าง vlan 22 และกำหนดให้ PC1 เป็นสมาชิกของ vlan นี้แล้วนั่นหมายความว่า PC1 ไม่ได้อยู่ที่ vlan 1 ซึ่งเป็นดีฟอลท์ vlan อีกแล้ว แต่เมื่อเรา ping ไปที่ Router1 จะสามารถ ping ได้เนื่องจากทั้ง PC2 และพอร์ตที่เชื่อมต่อไปยัง Router1 ยังคงเป็นสมาชิกของ vlan 1 เมื่อ้อนกัน

8. กลับไปที่สวิตช์แล้วทำการค่อนพิกให้พอร์ต e0/2 (ซึ่ง PC2 เชื่อมต่ออยู่) ให้เป็นสมาชิกของ vlan 22

(config-if)#exit

(config)#interface e0/2

(config-if)#vlan-membership static 22

9. กลับไปที่ PC2 อีกครั้งแล้วลองทดสอบ ping เมื่อ้อนเดิมอีกครั้ง บน PC2

C:>ping 24.17.2.1

C:>ping 24.17.2.3

ผลปรากฏว่าคราวนี้ไม่สามารถ ping Router1 ได้แล้ว เพราะอยู่คนละ vlan แต่สามารถ ping PC1 ได้เนื่องจากเป็นสมาชิกของ vlan หมายเลข 22 เมื่อ้อนกัน

10. กลับไปที่สวิตช์อีกครั้ง ลองแสดง vlan ที่ได้ค่อนพิกไปแล้วด้วยคำสั่ง show vlan และ show vlan-membership

```
(config-if)#end
#show vlan
#show vlan-membership ดังรูปที่ 5.10
```

#show vlan-membership			#show vlan-membership		
Port	VLAN	Membership Type	Port	VLAN	Membership Type
3	1	Static			
4	1	Static			
5	1	Static			
6	1	Static			

รูปที่ 5.10 แสดงคำสั่ง show vlan-membership

11. ขั้นตอนสุดท้ายให้ทำการคอนฟิกให้พอร์ต FA0/26 ซึ่งเป็นพอร์ตที่เชื่อมต่อไปยัง Router1 เป็นสมาชิกของ vlan 22

บนสวิตช์

```
#config t
```

```
(config)#interface FastEthernet 0/26
```

```
(config-if)#vlan-membership static 22
```

12. ทดสอบ ping ไปยัง PC1, 2 ให้ครบ

■ Virtual Trunking Protocol (VTP)

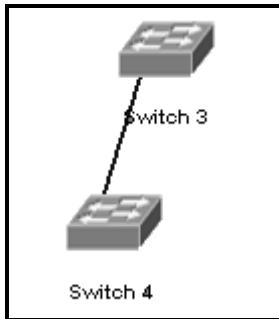
การสร้าง VTP เปรียบเสมือนการสร้างท่อขนส่งข้อมูลแบบจำลอง ๆ ขึ้นมา ซึ่งท่อแต่ละท่อจะมีข้อมูลที่สอดคล้องหรือเป็นกลุ่มเดียวกันหรือมาจากการแหล่งที่ใกล้เคียงกัน ถ้าจะยกตัวอย่างง่าย ๆ ของ VTP ก็คือ เราเห็นสายโทรศัพท์ที่เดินมาตามเสาไฟฟ้า (เปรียบเสมือน vlan) และเราเก็บทำการมัดรวมเอาสายโทรศัพท์เหล่านั้นเข้าไว้ด้วยกันเป็นมัดใหญ่ ๆ มัดหนึ่ง (VTP) ซึ่งก็เทียบเคียงได้เหมือนกับ Virtual Trunking เมื่อนั้น

จุดมุ่งหมาย : เรียนรู้การคอนฟิก VTP

เครื่องมือที่ใช้ทดลอง : ใช้สวิตช์ 2 ตัว (Catalyst 2950)

- คอนฟิก vlan บน Cisco Catalyst 2950
- ระบุพอร์ตให้กับ vlan
- คอนฟิก VTP ในโหมดของ server และ client
- คอนฟิก VTP ระหว่างสวิตช์
- ทดสอบการคอนฟิก VTP

การสร้าง Network Map : การเชื่อมตอกันระหว่าง Switch3 และ Switch4 ผ่านพอร์ต fastEthernet 0/12 (Trunk Port) ดังรูปที่ 5.11 และตารางที่ 5.1



รูปที่ 5.11 ผังเน็ตเวิร์คสำหรับ LAB VTP

ตารางที่ 5.1 รายละเอียดของอุปกรณ์ switch

อุปกรณ์	Switch 3	Switch 4
ชื่ออุปกรณ์	Switch3	Switch4
ไอพีแอดเดรส (VLAN 1)	10.1.1.1	10.1.1.2
Subnet mask	255.255.255.0	255.255.255.0

Simulator :

- เริ่มต้นด้วยการกำหนดหมายเลขไอพีแอดเดรสให้กับสวิตซ์ เปลี่ยนชื่อและสั่งให้ทำงานบนสวิตซ์หมายเลข 3


```

Switch>enable
Switch#config t
Switch(config)#hostname Switch3
Switch3(config)#interface vlan 1
Switch3(config-if)#ip address 10.1.1.1 255.255.255.0
Switch3(config-if)#no shutdown
Switch3(config-if)#end
Switch3#
      
```
- ทดสอบการเชื่อมต่อของสวิตซ์โดยทดสอบ ping จาก สวิตซ์ 4 มา�ัง สวิตซ์ 3 (ไม่สามารถ ping ได้)


```

      
```
- ขั้นต่อไป ให้สร้าง vlan 8 และ vlan 14 โดยกำหนดให้พอร์ต 0/2 ถึง 0/5 เป็นสมาชิกของ vlan 8 และ พอร์ต 0/6 ถึง 0/10 เป็นสมาชิกของ vlan 14


```

Switch3#vlan database [ฐานข้อมูลสำหรับเก็บข้อมูลของ vlan]
Switch3(vlan)#vlan 8 [สร้าง vlan หมายเลข 8]
Switch3(vlan)#vlan 14
Switch3(vlan)#exit
Switch3#config t
      
```

Switch3(config)#*interface range fast0/2 – 5* [กำหนดช่วงของพอร์ตที่ต้องการ]

Switch3(config-if)#*switchport access vlan 8* [กำหนดช่วงของพอร์ตที่ต้องการเข้า เป็นสมาชิกของ vlan]

Switch3(config)#*exit*

Switch3(config)#*interface range fast0/6 – 10*

Switch3(config-if)#*switchport access vlan 14*

Switch3(config-if)#*exit*

Switch3(config)#

- ใช้คำสั่ง show vlan เพื่อตรวจสอบความถูกต้องของคอนฟิกเรซั่น

Switch3(config)#*exit*

Switch3#*show vlan* ดังรูปที่ 5.12

Switch3#sh vlan			
VLAN	Name	Status	Ports
1	default	active	Ea0/1 Ea0/11 Ea0/12

รูปที่ 5.12 แสดงคำสั่ง show vlan

- โดยดีฟอลท์แล้ว Catalyst สวิตช์จะถูกตั้งเป็น mode Server เราจะคอนฟิกให้สวิตช์ 3 เป็น server และ สวิตช์ 4 เป็น mode client เปลี่ยน VTP Domain เป็น Boson และกำหนดรหัสผ่านเป็น rules

Switch3#*vlan database*

Switch3(vlan)#*vtp server* [กำหนด mode การทำงานของสวิตช์เป็น server]

Switch3(vlan)#*vtp domain Boson* [กำหนดชื่อของโดเมน]

Switch3(vlan)#*vtp password rules* [กำหนดรหัสผ่าน]

- เข้าไปยัง switch 4 คอนฟิก VTP

Switch4#*vlan database*

Switch4(vlan)#*vtp client* [กำหนด mode การทำงานของสวิตช์เป็น client]

Switch4(vlan)#*vtp domain Boson* [กำหนดชื่อของโดเมน]

Switch4(vlan)#*vtp password rules* [กำหนดรหัสผ่าน]

- กำหนดการทำงานของสวิตช์นั้นจะมีทั้ง mode 3 mode คือ server จะทำหน้าที่หลัก ๆ คือจะส่งข้อมูลของ vlan ไปให้ยังสมาชิกที่เชื่อมต่อกับมัน (client) และเป็นการสร้าง vlan จากจุด

ศูนย์กลางเพียงที่เดียวเท่านั้น ส่วน client จะไม่มีความสามารถพิเศษอะไรเพียงแต่รับข้อมูลที่ serve ส่งมาให้แล้วเก็บค่อนพิกเหล่านั้นไว้แล้วปรับปรุงข้อมูล vlan ของตัวเอง ส่วน transparent จะไม่สนใจข้อมูลที่ server ส่งให้มันจะสร้าง vlan ขึ้นมาเป็นของตัวมันเอง ขั้นตอนนี้เราจะทำการสร้างท่อเพื่อเชื่อมเอา switch3 และ switch4 เข้าหากันและส่งข้อมูลของ vlan แต่ละฝ่ายให้สามารถถูกกันได้ โดยต้องเรียกวิธีการ encapsulation ในที่นี้ให้เลือกเป็นมาตรฐานของ 802.1Q ใช้พอร์ต fast0/12 เป็น Trunk

```
Switch3(config)#interface fast0/12
Switch3(config-if)#switchport mode trunk
Switch3(config-if)#end
Switch4(config)#interface fast0/12
Switch4(config-if)#switchport mode trunk
Switch4(config-if)#end
8. เมื่อถึงขั้นตอนนี้แล้วการเชื่อมต่อทั้งหมดจะสามารถทำงานได้แล้วให้ทดสอบการเชื่อมต่อ
ด้วยการใช้คำสั่ง
show vlan
show vtp status
```

แบบฝึกหัดท้ายบท

- เมื่อต้องการดูคำสั่งที่ใช้งานในโหมดของ User จะต้องใช้คำสั่งอะไร? _____
- ถ้าต้องการเข้าสู่โหมด privilege จะต้องใช้คำสั่งอะไร? _____
- เมื่อต้องการดูคำสั่งที่ใช้งานในโหมดของ privilege จะต้องใช้คำสั่งอะไร? _____
- ถ้าต้องการเข้าสู่โหมดค่อนพิกจะต้องใช้คำสั่งอะไร? _____
- ต้องการเข็ตชื่อของสวิตซ์เป็นBoson จะต้องใช้คำสั่งอะไร? _____
- ต้องการกำหนดรหัสผ่านแบบ plain text จะต้องใช้คำสั่งอะไร? _____
- ต้องการทดสอบเมื่อเข้าทรัพย์ส่วนแล้วจะต้องอย่างไร? _____
- ต้องการกำหนดรหัสผ่านแบบ secret จะต้องใช้คำสั่งอะไร? _____
- ให้ logout ออกจากสวิตซ์แล้ว Login เข้าใหม่แล้วเข้าสู่โหมด privilege ต้องทำอย่างไร? _____