ชื่อ-สกุล น.ส. ลิขพร เลิศสิงหาคร กระดาษแผ่นที่ 1 รหัสนักศึกษา

for Staples

Chapter 1 Network Overview

- Network Diagram = โครงสร้าง Network ที่เราดูแล (มีการเชื่อมต่ออย่างไร)?
  - → physical → บอกว่าส่วนไหนเชื่อมต่อกันยังไง (ตรงไหนบ้าง) ex. ในตึก
  - → Logical → บอก ip
- Network protocol → TCP/UDP, FTP etc.
- Component of Network → HW
  - ↳ NW device มี 3 ประเภท
    - ① End device → หลังจากส่วนนี้จะเป็นคน (ตัว represent)
    - ② intermediary device : อยู่ระหว่างอุปกรณ์ ex. Network access device, hub, switch, ③ router
      - hub, repeater @ L_1 ⇒ ส่งพร้อมกัน → เกิด collision ∴ ใช้ CSMA/CD
      - switch, bridge @ L_2 ⇒ Learning / Flooding / Filtering / Forwarding / Aging
      - Routers @ L_3 ⇒ Routing
    - ③ Network Media = สื่อกลาง ex. fiber optic, wireless LAN —— straight ---- cross
- Type of Networks
  - → SW ① Switch เลือกทางออก ② router เลือกไปทางที่ดีที่สุด
  - → Size ① small home nw ⇒ มีใช้งานด้านนอกอย่างเดียว
    - ② small office / Home office → config จากด้านนอก เข้ามาใช้ด้านใน
    - ③ Medium to Large NW → มีการเชื่อมต่อมากกว่า 100 - 1000 เครื่อง
    - ④ World wide NW เช่น internet
  - ↳ infrastructure ทั่วไป ① LAN = มี 1 กลุ่ม admin ดูแล policy / NW ex. สำนักงาน — สถาบัน
    - ② WAN = มีหลาย admin

for Staples

- Reliable Network
  - → fault Tolerance → ทนต่อความผิดพลาด
  - → Scalability → สามารถปรับเปลี่ยนขนาดได้โดยไม่ส่งผลต่อ user
  - → Security → จำกัด / limit การเข้าถึง
  - ↳ Quality of Service (QoS) → แต่ละ service ให้ quality ได้เท่ากัน
- Type of Connection in a LAN
  - ↳ สายคู่ตีเกลียว (UTP, cat5) → BW = 100 Mbps → ยาว 100 m (can ต่อ hub, repeater, switch)
  - ↳ 2 ประเภท → ตรง
    - ↳ cross → ต่ออุปกรณ์เหมือนกัน except sw --- hub, pc -- router
  - ⇒ WAN Connection ⇒ เชื่อมระหว่าง router
    - ↳ 2 ฝั่ง → DCE (female) ⇒ มีการ command clock rate 56000
      - ↳ DTE (male)

Chapter 2 Basic Router Configuration

- Port Address → | 0 - 1023 ⇒ wellknown | (destination port) → | 1024 - 49151 → register port | → | 49152 - 65535 → random | (source port)
- Logical Address → IP (IPv4) @ L_3 มี 5 class A, B, C, D, E
  - A, B, C → host; D ↳ multicast ส่งไปให้ทุกคนที่อยู่ LAN เดียวกัน
  - ↳ แต่ละ nw / node มี logical name unique

A :	NW (0)	H	H	H	0 - 127	
B :	NW (10)	NW	H	H	128 - 191	
C :	NW (110)	NW	NW	H	192 - 223	
D :	1110				224 - 239	multicast
E :	1111				240 - 255	experimental

- private address → ip ที่ใช้งานภายในได้

		Prefix
A :	10.0.0.0 - 10.255.255.255	10.0.0.0 / 8
B :	172.16.0.0 - 172.16.255.255	172.16.0.0 / 12
C :	192.168.0.0 - 192.168.255.255	192.168.0.0 / 16

for Staples

- Physical Address : MAC Address → Ethernet = 48 bit ใช้เลข 2 = 0x ฐาน 16
  - ↳ ต้องลงทะเบียน IEEE กำหนด 3 bytes (24 bit) code (OUI)
    - ↳ 2 กฎ → ทุก MAC ที่กำหนดให้ NIC หรือ Ethernet device ต้อง → ใช้ OUI 3 ตัวหน้า
      - ↳ ทุก MAC ที่ same OUI ต้องกำหนด unique ที่ 3 byte สุดท้าย

⇒ Message Delivery
- ↳ Unicast = ส่งหาเครื่องปลายทางโดยตรง (NW เดียวกัน)
- ↳ Broadcast = ส่งหาทุกเครื่องใน NW เดียวกัน
- → Multicast ส่งหาหลายเครื่อง & รับเฉพาะเครื่องที่เปิด service → เริ่มด้วย 01-00-5E-xx-xx-xx

- Accessing a Cisco IOS Device
  ① Console port ② Telnet ③ Secure Shell (SSH) ④ Aux Port
  ↳ Terminal Emulation Program :: Putty, Hyper terminal etc
- Navigating the IOS → 2 mode ① User ">" ② Privileged (enable "#")
  ↳ Global Configuration Mode "(config)#"
  ↳ Other "(config-mode)#"
- The Command Structure
  ① Context Sensitive Help : "?"
  ② Command Syntax Check : enter แล้ว show ว่าผิดตรงไหน
  ③ Hot keys and Shortcut
  ④ IOS Examination Command → show...
- Getting Basic
  ① ตั้งชื่อ hostname → router (config)# hostname
  ② จำกัดการเข้าถึง → Banner msg. Router (config)# banner motd # text #
  ↳ Securing Device Access :: Enable password / secret, console pass, VTY pass, Encrypting pass display
  ③ กำหนด interface addressing ④ ตรวจสอบ config + การ์ดทำงาน ⑤ save config

③ Addressing Devices
1). เลือก interface to config
- Physical interface / Loopback interface
  Router (config)# interface type port
  " type slot/port
  " type slot/Port / subslot
- Switch virtual interface (SVIS)
  Switch (config)# interface vlan number

2). Set ip addr.
router (config-if)# ip address ip-address subnet-mask
" no shutdown

④ verify Connectivity → คำสั่งเพิ่มเติม section, include, exclude, begin
Router# show running-config → ดูว่า config อะไรไปบ้าง
show startup-config → set อะไรไปบ้าง
show ip route → ดู routing table
show interface
show ip interface } show info ของ interface ต่างๆ
show ip interface brief ↳ show interface แบบสรุป, show?, status?
traceroute
ping

PC > ping
tracert
route print
nslookup

Chapter 3 Static Routing & Dynamic Routing Protocol
- Function of Router → Characteristic :: ① Topology ② Speed ③ Cost ④ Security ⑤ Availability ⑥ Scalability ⑦ Reliability
  ↳ Packet Forwarding Methods ① Process Switching = ทุก packet ที่วิ่งเข้า router ต้อง process ใน CPU ว่าออก interface ไหน?
  ② Fast Switching : มีตัวช่วยทำให้ forward ได้เร็วขึ้น ③ Cisco Express Forwarding (CEF) : forward packet ได้เร็วสุด
- Connect Devices
  ↳ Default gateway → เป็นได้ ① first usable host (.1) ② last usable host (.254)
  ↳ ทางออกเชื่อมไป NW อื่น
  ↳ Enable IP on a Host ① Statically Assigned IP address
  ② Dynamically " ⇒ can กำหนด ip ได้ when NW มี DHCP (Dynamic Host Configuration Protocol)
- Switching Packet between NW
  ขั้นตอน : หา dest. ip (L#3) → ดูเทียบใน routing table → หาทางออกไป ดูเทียบนหา MAC address → ได้ dest. MAC (L2)
- Path Determination
  packet เข้า interface → match routing → dest ip match subnet → match interface? Y → check ARP cache
  ↓ N
  remote NW? Y → encap. frame → next hop
  ↓ N
  มี default route? Y → encap frame → next hop
  N → ทิ้ง packet & ส่ง ICMP กลับหา s. ip
  ↳ Load balancing = กรณี มี > 1 เส้นทาง
  ⇒ ใช้ทุกเส้นทางไปพร้อมกัน (ความน่าเชื่อถือของเส้นทางว่าไว้วางใจ) "trustworthiness"
  → Best Path : lowest metric (cost)
  ↳ Dynamic routing protocol ใช้
  ① Routing Information Protocol (RIP) : จำนวน hop
  ② Open Shortest Path First (OSPF) : BW มากสุด
  ③ Enhanced Interior Gateway Routing Protocol (EIGRP) : BW, delay, load, reliability
  ↳ Administrative Distance (AD)
  ↳ ระยะห่างทาง ผู้ดูแล ของ protocol ต่างๆ
  ↳ Connected = 0, Static = 1, Internal EIGRP = 90, OSPF = 110, RIP = 120
- Routing
  ① Static Routing → manual
  ข้อดี : security, ใช้ resource น้อยในการ process, ลด entry
  ข้อเสีย : ลำบากในการ scalability, ต้องแก้เปลี่ยนเส้นทาง
  ควรใช้เมื่อ : NW ขนาดเล็ก, จำนวน router น้อย, NW มีทางออกทางเดียว "stub NW"
  4 ชนิด ① standard ② Default : ใช้เมื่อ dest ip ไม่ match
  ③ summary ④ Floating : backup
  Router (config)# ip route nw-addr. subnet-mask {ip-addr. | exit-intf}

Config : Next-hop Option
ตัวอย่าง R (Config)# interface G 0/0
" -if# ip address 192.168.1.0 255.255.255.0 } Basic set ค่าทาง interface ให้กับ router
" no shut down
R1 (Config)# ip route 192.168.1.0 255.255.255.0 → 192.168.1.2 แบบที่ 1
↳ S 0/0/0 แบบที่ 2

Set default, static route
Router (config)# ip route 0.0.0.0 0.0.0.0 {ip-addr. | exit-intf}
② Dynamic Routing Protocol → auto
2.1 EGP (Exterior Gateway Routing Protocol) :: BGP
2.2 IGP (Interior ") :: RIP, OSPF, EIGRP, IS-IS → (Intermediate System to Intermediate System)
- Classful Addressing → update ตาม class
- Classless Inter-Domain Routing
  ↳ Summarization : เงื่อนไข → ① อยู่ติดกัน ② ไม่ทับซ้อนกัน
- VLSM
  ↳ Fixed Length Subnet Masking
  วิธีทำ ① prefix ใหม่ − prefix เก่า = จำนวน bit ที่เพิ่ม
  ② เพิ่ม ทุก bit เป็น 1 แล้ว แปลงค่าฐาน 10 ไป + ip เริ่มต้น +1 ก่อนไปเรื่อยๆ

ชื่อ-สกุล น.ส. ศิรพร เฉลิมโสภาร กระดาษแผ่นที่ 2 รหัสนักศึกษา [redacted]

for Staples

Chapter 4 Distance Vector Routing Protocol RIP ver 1

* info = information

▶ Dynamic Routing Protocol

- f^n = share info ระหว่าง router = auto update routing table เมื่อ topology เปลี่ยน (มีปัญหา) → หา best path
- purpose = หา remote NW (NW ที่อยู่ห่างไกล), ปรับปรุง routing info, เลือก best path ไปยัง dest NW, can หา new best path ถ้า path ปัจจุบันไม่ว่างยาวๆ
- Component = ① Algorithm = ใช้เพื่อสะดวก routing info และ best path
  ② Routing protocol messege = สำหรับหา neighbor และแลกเปลี่ยน routing info (best path)

	Dynamic routing	vs Static routing
ความยุ่งยากในการ Config	แปรผันกับขนาด NW (เหมือนเดิม)	ตามขนาด NW (เพิ่ม Command ทุก router)
Required ค.รู้ admin	Advanced (bec. config basic → NW ใหญ่ขึ้น ซับซ้อน ตามปัญหา)	No เพิ่มเติม (กำหนดเส้นทางเองโดย Command)
Topology change	ปรับ auto	admin config ใหม่ทั้งหมด
Scaling	เหมาะ simple & complex (router ที่ไม่ได้ directly ไม่ต้องทำอะไร)	เหมาะ simple topologies
Security	น้อยกว่า	มากกว่า
Resource usage	ใช้ CPU, mem (เก็บ routing info, link BW)	No เพิ่มเติม
Predictability	Route & current topology	Route → dest ตาม เดิมตลอด

▶ Classifying Routing Protocols

→ DRP

↳ (ภายใน AS) Interior Gateway Protocol (IGP)
- ↳ Distance Vector Protocol
  - ประกอบ vector (distance, direction)
  - incomplete view ของ NW topology
  - periodic update (รายงาน) ซ้ำๆ
  - → RIP (Routing Info Protocol) RIP v.2
  - → IGRP (Interior Gateway Routing Protocol) EIGRP (Enhanced Interior Gateway Routing P.)
- ↳ Link state P.
  - Complete NW topology is created (เก็บข้อมูล all)
  - update ไม่ใช่ periodic
  - ↳ OSPF (Open Shortest Path First)
  - ↳ IS-IS (Intermediate System to Intermediate System)

↳ (ภายนอก AS) Exterior Gateway Protocol (EGP)
- ↳ BGP (Border Gateway Protocol)

* AS = Autonomous System
↳ เป็นกลุ่มของ router ภายใต้การควบคุมของ single authority (1 กลุ่ม policy or ผู้ดูแล)

- ◉ 2 type
  ① Classful routing protocol → update ตาม class ไม่ส่ง subnet mask ใน routing update
  ② Classless " ——— " → ส่ง subnet mask ใน routing update
- ◉ Convergence = กำหนด เมื่อ routing table ของ router ทั้งหมด มีความคงที่ (ไม่เปลี่ยนแปลง)
  ↳ 2 type = Slower → RIP และ IGRP
  Faster → (มีการ update เมื่อเกิดการเปลี่ยนแปลงได้ >): EIGP & OSPF

for Staples

▶ Routing Protocol Metrics
- Metric : ค่าที่ใช้ในการวัด/หาในการไป dest. NW เลือก best path ไหน เช่น Hop count (RIP), Bandwidth (OSPF), Cost, Delay, Load, Reliability (IGRP)
- Load balancing : NW มีเส้นทาง > 1 ที่มี metric เท่ากัน → เฉลี่ยให้การใช้งานแต่ละเส้นทางเท่าๆกัน

▶ Administrative Distance of a Router (AD) ⇒ ใช้เลือก protocol ในการ routing
- วัตถุประสงค์ : ค่าตัวเลขที่เจาะจงสิทธิ์มิเติม ของ particular (ละเอียด) route ⇒

Route source	AD
Connected	0
Static	1
EIGRP sum route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
External EIGRP	170
Interal BGP	200

▶ Distance Vector Routing Protocol Ex. RIP, IGRP, EIGRP
- Distance Vector Technology ต้องรู้ 2 สิ่งมาใช้ตัด ① vector or direction, การเหตุเปลี่ยนควรถูกควบคุม ② Distance to final dest (cost)
- ลักษณะทั่วไป : periodic update (ตามเวลา update ที่แน่นอน), neighbor (⊗ใกล้เคียง), broadcast (255.255.255.255) update, เอา routing table ทั้งหมดไป update)
- ลักษณะทั่วไปของ Routing protocol : เกณฑ์ที่ใช้ check ว่า BW ดีไม่ดี ?
  ① Time to Convergence ⇒ เวลาที่เป็น steady state ของ routing table ที่เกิดการเปลี่ยนแปลงแล้ว ③ Resource usage
  ② Scalability ขนาดมากสุดเท่าไร ④ Implementation & maintanance

□ NW Discovery (ต้นแบบ) → ทำ basic config ก่อน)
มี 3 stage ① Cold State : Router Initial Start up
② Initial Exchange of Routing info. → ทำการแลกเปลี่ยน
③ Exchange of Routing info. → update (เพิ่ม hop count) routing info.
↳ บอก router อีกตัวว่าห่างเท่าไหร่

NW	intf	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	0
10.1.0.0	S0/0/0	1
10.4.0.0	FS0/0	1

update ③
10.2.0.0, 1
10.3.0.0, 1
10.1.0.0, 2
10.4.0.0, 2

□ Routing Table Maintanance
- Periodic Update : RIP update timer (default 30), Invalid timer (info ที่ lost) (default 180), Holddown timer (ถ้า down → hold ไว้ไม่ up หรือไม่) (default 180), Flush (ทิ้ง) timer (default 240)
- Bounded (ขอบเขต) Update : EIGRP → update แต่ที่เปลี่ยน
- Triggered Update → update เลยไม่รอ periodic time
- Random Jitter → ใช้ใน NW ที่เป็น multiple access router หลายตัวเชื่อมต่อกัน if เกิด update ชนกันพร้อมกัน ∴ จึงใช้ Random

□ ปัญหา Standard DV.
① Routing Loops เกิดเมื่อ interface ไหน down จะถูกดึงออกจาก table → ค่าที่ใน neighbor (จะกลับมา update (จน update → hop เพิ่ม → ∞)
↳ แก้ปัญหา: ① set max hop = 15 → if hop = 16 → unreachable (มัน down ไปแล้ว)
② holddown timer (ถ้า interface down → hold)
③ split Horizon Rule → ไม่ส่งข้อมูล update กลับไปทาง intf ที่ได้รับ update มา
④ Route Poisoning → ① เกิด down set unreachable ② ส่ง unreachable ที่เป็น poison ออกไป → ให้ intf นั้น hop = 16
⑤ ③ และ ④ → ถ้าเกิด unreachable จะ overrule split horizon โดยส่ง ip intf ที่ down (hop = 16) ⇌
⑥ IP & TTL (Time to Live) เกิดการวน update แต่จะหยุดเมื่อ TTL = 0

for Staples

	RIP_{v1}	RIP_{v2}	IGRP	EIGRP
speed convergence	slow	slow	slow	fast
Scalability - size NW	small	small	small	large
Use of VLSM	×	✓	×	✓
Resource usage	Low	Low	Low	Medium
implementation & Maintanance	Simple	Simp	Simp	Complex

▷ RIP version 1   AD : 120

- คุณสมบัติ : • classful, Distance vector (DV) routing protocol • Metric : hop count • hop count > 15 = unreachable • update broadcast ทุกๆ 30s

RIPv1 Message Format Transport Layer

encapsulated (บรรจุ) ใน UDP segment ด้วย source & dest port = 520

Data Link Frame Header	IP Packet Header	UDP Segment Header	RIP Message (512 Bytes ; Up to 25 routes)

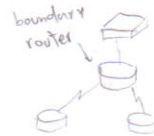
Request — reply — ขอ full routing table

0 … 7/8 … 15/16 … 23 24 … 31

header ← Command = 1 or 2	Version = 1	Must be zero
Address family identify (2 = IP)		Must be zero
IP Address (Network Access) → dest.		
Must be zero		
Must be zero		
1–15 ส่งจาก table, 16 → ส่งออก ← Metric (Hops)		
Multiple Route Entries, up to a maximum of 25		

- message มี 2 type
  - ① Request → ส่ง routing table
    - ↳ ไป interface ที่ config ไว้เมื่อเจอรับการ update
  - ② Response → ส่ง information ของ routing table
- IP Address → แบ่งตาม class A, B, C
- Basic RIPv1 Config
  - ① ทำ Basic config
  - ② สั่ง router rip → R1(config)# router rip
  - + ต่อสั่ง NW → R1(config-router)# network NW ip ที่ต่อตรงๆ กับ

- Verification (ตรวจสอบ) & trouble shooting (จัดการปัญหา) : show running-config or ip route or ip protocol, debug ip rip
  - passive interface command (ไม่ update interface ที่ไม่ต้องการ) R(config-router)# passive-interface intf-type (Fa/G/S) intf-number) (0/0, 0/0/0)
- Automatic Summerization : RIP Auto Summarize classful NW → เพื่อลด size routing table
  - ↳ ข้อดี : • ลด size routing update • single router แทนหลายๆ multiple route เร็วมากใน routing table
  - ↳ ข้อเสีย : • ไม่ support discontiguous NW (major NW เดียวกัน แต่ไม่อยู่ติดกัน → อาจจะเกิด load balancing ได้)



- boundary Routers : summarize RIP subnet from 1 major NW to another
- Processing RIP Update
  - ดูตัวที่ส่ง และตัวที่ update (ไป interface) ว่ามี classful เดียวกันไหม ? → yes : update subnet NW เช่น 172.16.1.0
  - no : update classful เช่น 172.16.0.0
- default route & RIPv1 ถ้าไม่เจอทางออกใน routing table (อาจจะเพราะคนละ protocol) → จะออก default route
  - R(config)# ip route 0.0.0.0 0.0.0.0 S0/0/1
  - default info. originate command → เพื่อ update ไปใน rip ว่าเชื่อม static ↔ dynamic
- Router ที่อยู่ระหว่าง 2 protocol → R(config-router)# default-information originate

Chapter 5 RIP version 2 & Access Control Lists

RIPv1	vs	RIPv2
classful (ไม่ส่ง subnet mask, ไม่ support CIDR)		classless (update subnet mask, support Variable Length Subnet Masking (VLSM), Support Route Summarization (Prefix Aggregation))
not support discontiguous subnet		update next hop address.
not support VLSM เพราะไม่ส่ง subnet mask (255.255.255.255)		ใช้ authentication routing (รองรับ discontiguous บางกรณี)
routing update ⇒ broadcast		Routing update ⇒ multicast

เหมือนกัน
- ใช้ Timer ป้องกัน routing loop
- ใช้ split horizon or split horizon with posion reverse
- ใช้ triggered update
- max hop count = 15

เป็น virtual interface สามารถเพิ่ม routing ให้ update ออกไปได้

- ข้อจำกัดของ RIPv1
  - loopback intf → ping ไป ip virtual intf → reply ได้
  - Null intf → ใช้ดักคนที่จะเข้ามา channel ที่เรา ไม่ต้องการ → ส่งออก null intf
    - ↳ packet discard เลย → timeout
  - Static route & null interface
    - ↳ null intf จะเป็นที่เป็นทางออกของ static route
    - R(config)# ip route summary-static-route subnet-mask Null0
    - ex. R2(config)# ip route 192.168.0.0 255.255.0.0 Null0

- Route redistribution (กระจาย) → อยากให้ rip รู้จัก static ตัวที่เพิ่มใหม่ (โดย rip ส่ง static ไปด้วย) R(config-router)# redistribute static
- Verify & Test Connectivity : show ip interface brief, ping (ผล : ! = ได้, U : ไม่ได้, . = timeout), trace route
- RIPv1 : classful, ไม่ส่ง subnet mask, summarize NW @ majorNW boundaries, if NW เป็น discontiguous & RIPv1 config convergence จะมาไม่ถึง
- ตรวจ routing table debug ip rip (content of routing update), ถ้าเป็น RIPv1 จะไม่ให้ subnet mask รวมใน NW addr.

▷ RIPv2

- Config • Enabling & verify (ตรวจสอบ) RIPv2 → show ip protocols
  - Config RIP → RIP v1 → can รองรับได้ทั้ง V1 & V2 แต่ส่งได้แค่ V1
    - ↳ RIP V2 → can รับ & ส่ง ได้แค่ V2
  - Auto-Summary & RIPV2 → auto sum route @ major nw boundaries
    - ↳ sum route ด้วย subnet mask ที่น้อยกว่า classful subnet mask
  - disabling Auto-Summary : no auto-summary เพราะ เมื่อใช้ NW topology ต่อแบบ discontiguous

RIPv2

0 … 7 8 … 15 16 … 23 24 … 31

Command = 1 or 2	Version = 2	Must be zero
Address Family identifier (2 = IP)		Route Tag
IP Address (Network Address)		
Subnet Mask		
Next Hop		
Metric (Hops)		
Multiple Route Entries, up to a maximum of 25		

Route Entry (IP Address – Metric)

- VLSM & CIDR → verify info. ที่ sent by RIPv2 debug ip rip
  - → VLSM → แพร่ NW addr. & subnet mask
  - → CIDR → ใช้ supernetting (= bunch ของ contiguous classful NW ที่เป็น addr. เหมือน single NW)
    - → verify show ip route, debug ip rip

ชื่อ-สกุล น.ส. สิจจพร เลิศสนิทการ กระดาษแผ่นที่ 3 รหัสนักศึกษา

for Staples

- Access Control list = ควบคุมการเข้าออก → ตรวจสอบ → check → source → dest ผ่านไหม?
  → ท่าอย่างนี้ (เช่น FTP) ผ่านไหม?
  ⇒ คุม Conversation
- Packet Filtering ดู ① dest, source @ L2 ② protocol ที่ใช้ ③ ไปทางไหน } → ทำการให้ผ่าน or block ไว้?
- Operation → ทำงานเป็น sequence statement
  → last statement เป็น implicit deny → block → discard
  inbound ACL ⇒ (X) outbound ACL ⇒
- Standard IPv4 ACLs vs Extended IPv4 ACLs
  - Standard:
    - Check source addr.
    - ปกติ permits or denies ทั้งชุด protocol
    - access-list 10 permit 192.168.30.0 0.0.0.255
    - number ACL : 1-99 & 1300-1999
  - Extended:
    - check source & destination addr.
    - ปกติ permits or denies specific (เฉพาะ) protocol
    - access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
    - Number ACL 100-199 & 2000-2699
- Wild card → invert ของ subnet mask
  → 0 = match/fix, 1 = ignore / อะไรก็ได้
  → วิธีหา จาก set ของ ip (match range) ① หาความสัมพันธ์ของ bit ที่ซ้ำกัน ให้ wild card mask ตำแหน่งนั้น = 0 ② bit ที่เหลือให้ 1
    if กรณีคิดไม่ได้ ลองแบบ Pattern or / and ส่วน คิด สุดท้าย wildcard จะเหมือนกัน
  → หา wild card ของ subnet = 255.255.255.255 - subnet mask
  → keyword → 0.0.0.0 = match all ใช้ host
    R1(Config) # access-list 1 permit 192.168.10.10 0.0.0.0
    " " " host 192.168.10.10
    → 255.255.255.255 = ignore all ใช้ any
    R1(Config) # access-list 1 permit 0.0.0.0 255.255.255.255
    " " any
- Guideline for (3Ps) → One ACL/protocol : ctrl traffic บน intf, ACL ต้อง define แต่ละ protocol enable บน interface
  ACL creation → One ACL / direction : ctrl traffic in 1 direction (Flow) at time on an intf, แยก ACL ctrl in & out bound traffic
  → One ACL / interface : ACL ctrl traffic for an intf, Ex. Golo
  - where → Extend ACL : @ close source → standard ACL @ close destination

for Staples

- Config ACLs (เลือกมา 1 แบบ; คำสั่งใน เหมือนกัน แต่ ข้อความของ มีเพิ่มคำสั่ง (permit, deny, remark) ได้เลย)
  → standard
  → number: Router(config) # access-list access-list-number (num ที่ต้องการสร้าง) deny | permit | remark : Comment source [source - wild card] [log]
    ถ้า remove all : no access-list
    ถ้าแก้ ① no access-list num# → สั่งใหม่ ② no # บรรทัด → # บรรทัด คำสั่ง
  → in intf → Router(config-if) # ip access-group (access-list num# permit) {access-list-number | access-list-name} {in | out}
    ถ้า remove all : no ip access-group
    ถ้าแก้ → บรรทัด # คำสั่งใหม่ → บรรทัดที่แทรก
  → ชื่อ: Router (Config) # ip access-list [standard | extended] name
- Verify : show ip interface, show access-list
- Security VTY port → อนุญาตให้คนนอกเข้ามา permit หมายถึง ที่เราต้องการให้เข้า → Router(Config-line) # access-class access-list-number {in [vrf-also] | out}

→ Extended : Filter : source / dest. addr., protocol, port number
  (standard 0-99, 1300-1999 Extended 100-199 2000-2699)
  access-list access-list-number {deny | permit | remark} protocol source (ต้นทางของ) [source - wildcard] [operator operand] [port port-number or name] (wild card source ของ port ไม่) destination [destination-wildcard] [operator operand] [port port-number or name] [established] (port dest)
  ip tcp, udp ใส่ก็ได้ / ไม่ใส่ก็ได้
  ที่เหลือ same standard มีได้ทั้งแบบ name & number
  - debug-output : debug ip packet ACL-number

Chapter 6 OSPF & DHCP

- Link-state Routing Protocol = เป็น protocol ที่ต้องสร้าง complete map (มี info. all) ของ NW topology ขึ้นก่อน → หา shortest path First (SPF) (ใช้ dijkstra)
  - เหมาะกับ : ① Large NW, ② fast convergence ③ admin ควรมีความรู้
  - ขั้นตอน update ① learn info. ของ link ② say hello neighbor ③ เอา info. มาสร้าง Link-state Packet (LSP) ④ router flood LSP to all neighbor → ได้รับแล้วเก็บลง db ⑤ router เอา all LSP เก็บใน db (รูปแบบtree) + Adding OSPF → routing table
  - ข้อดี : ① สร้าง topology map สามารถหา shortest path ② fast convergence ข้อมูลถ้าเปลี่ยนแปลง ③ LSP sent only when change topology (ได้เฉพาะ ข้อมูลที่เปลี่ยนแปลง) ④ hierarchical design (NW ใหญ่จะทำ) → ลด resource แบ่ง = แยกข้อมูลแต่ใน area (คำนวณ shortest path ←)
  - ข้อเสีย : ① ใช้ mem ในการเก็บ all link-state เยอะ ② ใช้ CPU ในการคำนวณ ③ ตอนส่ง LSP ครั้งแรกให้ BW มาก

for Staples

- OSPF AD = 110
  → 3 Table : ① Neighbor show ip ospf neighbor ② Topology (ทั้ง map) show ip ospf database ③ Routing (มี shortest path)
  - Message → Encapsulating : MAC Dest. = Multicast : 01-00-5E-00-00-05 or 01-00-5E-00-00-06
    Protocol field = 89
    → type OSPF Packet : 01 Hello → ทุก 10s (default : multiaccess & point to point NW), ทุก 30s (default : non-broadcast multiaccess [NBMA] NW), Cisco default 4 times (40s)
    : 02 Db Description (DBD) → synchronization db info.
    : 03 Link-state Request (LSR) → request link-state
    : 04 " Update (LSU) → Send update link-state
    : 05 " Acknowledgment (LSAck) → ตอบกลับได้เลย

- Operation : สร้างการเชื่อมต่อ ① Down state (เริ่มต้น) → ② init state (เริ่มส่ง hello) → ③ Two-way state (ตอบกลับ hello) → Exstate state
  → Exchange state → Loading state → Full state (ระหว่าง router update ข้อมูลเรียบร้อย)
- Config Single-Area OSPFv2 router ospf process-id → 1-65,535 , เป็น locally significant

  R(config-router)# router-id 1.1.1.1 → ถ้าไม่ set จะใช้ ② loopback , ③ active interface ip สูงสุด แต่ควรมี ① ไม่ก็ ②

  router ospf process-id
  network network-address wildcard-mask area area-id

- OSPF cost → ใช้ BW คำนวณ [default reference BW = 10^8]

  cost = $\frac{10^8 \text{ bps}}{\text{intf BW bps}}$
  - 10Gb Ethernet = 100×10^8 → cost = 1
  - Gb " = 10×10^8 → = 1
  - Fast " = 10^8 → = 1
  - Serial " = 1.544×10^6 → = 64

  auto-cost reference-bandwidth bandwidth-mbps
  auto-cost reference-bandwidth 10000
  1000
  100

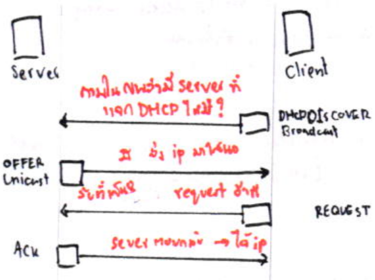
  → ต้องการเปลี่ยนค่า cost
  → เปลี่ยนค่า ref BW
  → เปลี่ยนค่า BW : R(config-if)# bandwidth 64 (EIGRP & OSPF จะใช้ได้)
  → เปลี่ยนค่า cost : ——— ip ospf cost 15625

- Verify OSPF show ip ospf neighbor , show ip protocol , show ip ospf interface brief , show ip ospf
- more config • Redistributing on OSPF Default Route
  R(config)# ip route 0.0.0.0 0.0.0.0 Loopback N
  ——— "#router ospf process-id
  R(config-router)# default-information originate
  ↳ เมื่อให้มันเอาค่า default route update ไปใน ospf

▶ DHCP (Dynamic Host Configuration Protocol) → ช่วย config ให้ host ให้ auto (เช่น ip, subnet mask, default gateway, dns)
- method ① Manual Allocation : admin assign เอง
  ② Automatic Allocation : DHCPv4 auto assign addr. จาก pool & & ไม่มี lease (เช่า) time
  ③ Dynamic Allocation : ใครเปิดขึ้นมาก็จะได้ ip ไป มี lease time → ถ้าหมด lease time ต้อง re ip ใหม่

- Operation

  Lease Origination



- Config

  R1(Config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9 ip ไม่ถูกแจก (ไม่เอา/ไม่ใช้)
  ——— " ip dhcp excluded-address 192.168.10.254
  ——— " ip dhcp pool LAN-POOL-1 สร้าง pool
  R1(dhcp-config)# network 192.168.10.0 255.255.255.0 NW ip ที่เราจะแจก
  " default-router 192.168.10.1 * ต้องมี
  " dns-server 192.168.11.5
  " domain-name example.com } มีไม่มีก็ได้
  " end
  R1#

- Verify
  show running-config | section dhcp
  show ip dhcp binding
  show ip dhcp server statistics
  • On the PC - issue the ipconfig /all command

• To disable dhcp - no service dhcp ยกเลิก
  ถ้า ip ไม่ขึ้น → cmd → ip config /release → ip-config /renew
  ↳ ติดลิขสิทธิ์ (@ หน้า set ip)

- Config DHCP Client (แจก ip ให้ client : -if)# ip address dhcp
  -if)# no shutdown

- debug → same Extended

ชื่อ-สกุล น.ส. สิจจพร เลิศสิลปศาสตร์ กระดาษแผ่นที่ 4 รหัสนักศึกษา 59011382

for Staples

Chapter 7 Basic Switch Address Resolution Protocol

▶ LAN Design → Borderless SW NW design : มีกฎทั่วไป :: - Hierarchical, Modularity, Resiliency, Flexibility

□ 2 ลักษณะ : ① 3-Tier LAN Design ประกอบด้วย :: 1) Core 2) Distribution 3) Access ② 2-Tier :: 1). Collapsed Core / Distribution 2) Access

ทุก Layer เป็น SW {
1. Core ⇒ อุปกรณ์ที่รองรับ SW สูงๆ → ทำให้มี speed ↑ ในการเชื่อมต่อ NW
2. Distribution ⇒ เชื่อมระหว่าง 1 & 2., Security Policy / Access Control
3. Access ⇒ ติดต่อ end device, Port Security, VLAN [Fa / Gig Ethernet], Power over Ethernet รองรับ 1 LAN or ไม่ได้ 1 LAN

} Layer 3 support [Gig / 10 Gig Ethernet] / Redundant component ⇒ มีหลายเส้นทาง } Link aggregation ↳ ทำให้ SW device เพิ่ม ↑ / Quality of Service (QoS)

□ เพิ่มความสามารถในการใช้ได้ LAN BW & ประสิทธิภาพให้ max

• function & ความ Server ① Enterprise s. (รวมทั้งองค์กร) ⇒ ควรติดตั้งที่ MDF (Main Distribution Facility : Core) power line ⟹ ทางเดินไปข้างนอก ที่ใกล้กับมัน
② Workshop s. (เฉพาะกลุ่ม) ⇒ ติดตั้งที่ IDF (Intermediate D. F. : Distribution) ⇒ เชื่อมระหว่าง cross กับ access ที่ใกล้กับมัน

VCC (Vertical cross-connection) : optical fibre : ⇒ MDF ⟺ IDF
HCC (Horizontal — " —) : UTP ⇒ Distribution → Access

• Collision detection issue (ปัญหาการตรวจจับการชน)
• Segmentation issue (ปัญหาการแบ่งส่วน) ⇒ ทำให้ปัญหาการใช้งาน NW มีประสิทธิภาพต่ำ
• Broadcast domain issue ⇒ ใน NW เกิดตลอด เช่น broadcast หา MAC Address ∴ broadcast NW ควรมีขนาดเล็ก

□ Segmentation เป็น process split single collision domain → smaller collision domain ลดจำนวน collision บน LAN segment :: L#2 device ให้เป็น bridges, SW

□ Broadcast domain รับหมดทุก port but routers (L#3) เป็นตัว filter / segment broadcast ไม่ให้กระจายออกจากตัวมัน

▶ SW Environment

□ SW Operation ① Learning : รับ frame เข้า SW จะเรียนรู้(จำ) Source Mac Addr. ว่าต่อกับ port ไหน (NW ไหน) + reset Aging
② Aging : ตามจุด MAC Addr. → กำหนด → ทิ้ง

กรณีส่งออก {
③ Flooding : ส่ง frame ออกทุก port ของ SW เมื่อ frame เป็น 1) brodcast 2) multicast 3) unknown unicast → กรณี dest ไม่อยู่ใน Table
④ Forwarding : ส่งไป dest. (กรณีพบใน table)
⑤ Flitering : if ได้รับ frame ไป dest ตาม port ที่เป็นทางไป dest (source และ dest บน interf. เดียวกัน) จะทำการ filter ทิ้ง
}

□ SW Methods : ① Store & Forward SW ⇒ check CRC. ว่า error ไหม → มี : ทิ้ง → ไม่มี : ส่ง, auto buffer
② Cut-through SW ⇒ check เฉพาะบางส่วน (dest, source อย่างน้อย 12 byte แรก) [10MS], No FCS & auto buffer
↳ 2 mode :: ① fast-forward ~ 12 byte ② fragment-free ~ 64 byte ; < 64 : 64 → หยุด → ไม่ส่ง

□ SW Domains :: ① Collision Domains ⇒ domain ที่เกิดการชนตอนที่ส่งข้อมูลมาพร้อมกัน "ดูที่ SW เป็นตัวแบ่ง"
② Broadcast ⟶ ⇒ domain ที่ส่ง broadcast → อยู่ domain เดียวกัน จะได้รับ "ดูที่ router เป็นตัวแบ่ง"

▶ Basic SW Concept & Configuration

manage intert: S(config)# interface Vlan num
same router { S(config-if)# ipaddress ip subnet / S(config-if)# no shutdown
default Gateway: S(config)# ip default-gateway ip

□ Basic SW Config • SW Boot Sequence • same router
• Preparing of basic SW Management : SW ไม่มี loopback ∴ กำหนดเป็น SVI (SW Virtual Interface) → VLAN
• Config SW Port → Duplex Communication
① Full ② Half (SW ที่ต่อกันต้องเลือกเหมือนกัน)
เข้าที่ intf → S(config-if)# duplex full → S(config-if)# speed 100 (เปลี่ยน speed)
→ Auto-MDIX : ปกติ SW จะเชื่อมกันด้วย cross-over but ต้องการเปลี่ยนเป็นสายตรง
เข้าที่ intf → S(config-if)# duplex auto → S(config-if)# speed auto → S(config-if)# mdix auto

→ Verify Port Config
show int fo/o, / startup-config, / running-config, / flash, / version, / history, / ip fo/o, / mac-address table

□ SW Security : Security Remote Access → SSH (Secure Shell) TCP port 22, Telnet : TCP port 23

Config : S_c(config)# ip domain-name (change host name) ชื่อ → # crypto key generate rsa → # username admin pass ccna → line vty 0 15 →
-line)# transport input ssh → -line)# login local [Verify SSH : show ip ssh, show ssh]

□ SW Port Security ⇒ กำหนด policy ว่าให้ MAC Addr. ไหนเข้า/ออก ได้-ไม่ได้
S(config-if)# Switchport mode access → # switchport port-security → เลือกมา 1 ประเภท
Secure MAC Addr. → ① Static :: S(config-if)# switchport ——— mac-address MAC-ADD
② dynamic :: S(config-if)# ——" ——" ——" sticky → learn frame ที่วิ่งมาก่อน → record ไว้

กรณีเพิ่มจำนวน MAC : # switchport port-security maximum MAX

Violation mode : ① protect :: security violation protect mode
② restrict :: ——— restrict mode ⇒ ส่งไปเก็บว่ามีอะไรเกิดขึ้นบ้าง
③ shutdown :: ——— shutdown mode ⇒ default
[verify : show port-security int fao/o, show port-security address]

Security Violation Mode

Forwards Traffic	Send Syslog Message	Display Error msg	Increase Violation Counter	Shutdown Port
No	No	No	No	No
No	Yes	No	Yes	No
No	No	No	Yes	Yes

□ Addr. Resolution Protocol (ARP) : ARP Cache เก็บ MAC Addr. ไว้ map หาทางไป dest. (if ไปข้างนอกต้องใช้ MAC gateway)

IPv4 : Classless [ต่อ P.1-2] :- Variable Length Subnet Masking (VLSM) : แบ่งทุก NW ไม่เท่ากัน :⇒ เลือกของอื่นใหญ่ → เล็ก
- Fixed ~ : แบ่งทุก NW เท่ากัน

Chapter 8 LAN Redundancy & Spanning Tree Protocol (STP)

▶ Issue with layer 1 Redundancy :: ① MAC Addr. Instability ⇒ MAC Addr. table จะไม่เสถียร เพราะเปลี่ยนแปลงบ่อย ② Broadcast storms : วนส่งไปทั่วไม่หยุด
③ Multiple frame transmission ⇒ start : unknown unicast → ทำให้ dest. ได้รับหลาย frame แต่ source ส่ง 1 frame

▶ STP ⇒ จุดหมาย : หา block port → block มัน → ไม่ให้เกิด traffic ขึ้น เพื่อแก้ปัญหา
■ ขั้นตอน ① หา Root Bridge = เอา priority min 1. 1 RB / 1 NW 2. 1 RP / 1 RB 3. 1 DP / segment
ที่ BPDU (Bridge protocol data unit) ② หา path cost all ③ หา Root port → path cost min → อันนี้จะเป็น Designated Port
③ ส่วน segment มี path cost เท่ากัน → ดู BID น้อยสุด เป็น designated port → อันนี้เป็น block port

สิ่งที่ใช้เปรียบเทียบ ① BPDU flag (ที่ priority น้อยสุด) ② BID น้อยสุด ③ Path Cost < ④ Sender's BID < ⑤ Sender's Port <

มี RB & Source [802.1D] ↲ □ Config : วิธีที่ 1 : S1(config)# spanning-tree VLAN 1. root primary
S2(config)# ——— 1. root secondary
วิธีที่ 2: S3(config)# spanning-tree VLAN 1 priority 24576 (เพิ่มให้ทีละ 2)
[Verify : show spanning tree]

for Staples

- แบบ Extended System ID : B. Priority (2byte) ⇒ B. Priority (per VLAN) (4bit) + Extended Sys ID (VLAN) (12 bit) + MAC Addr. (6 byte) ∴ BID = 8 byte
  - PVST+ (พัฒนามาจาก IEEE 802.1D STP) ⇒ เพื่อทำ load balancing โดยแยก root/vlan [verify : show spanning-tree active]
  - Rapid PVST+ ⇒ หา Alternate port (ไม่ให้ block ไปเลย สามารถรับส่งข้อมูลได้บางส่วน)
    - ↳ ทำการ set Edge Port @ port ต่อ host, router : -if)# spanning-tree portFast
    - ↳ link type : port ที่เชื่อมระหว่าง SW ต้องเป็น point-to-Point : -if)# spanning-tree bpduguard enable → port นี้ไม่ต้องการให้ BPDU ไปด้วย
  - Config : S1(config)# spanning-tree mode rapid-pvst → # int ที่ p-to-p → # spanning-tree link-type point-to-point

โค้ด clear all : clear spanning-tree detected-protocol

Protocol	Standard	Resource Needed	Convergence	Tree Calculation
STP	802.1D	Low	Slow	All VLANs
PVST+	Cisco	High	Slow	Per VLAN
RSTP	802.1W	Medium	Fast	All VLANs
Rapid PVST+	Cisco	Very high	Fast	Per VLAN
MSTP	802.1S Cisco	Medium/High	Fast	Per Instance

Chapter 9 VLANs & Inter VLAN

- VLAN = ทำ partition (แบ่งแยก : อยู่คนละ NW or broadcast domain กัน) Layer 2 เช่น SW และทำการต่อได้แค่ใน VLAN เดียวกัน
  - ข้อดี : - Security ดีขึ้น, - ลด cost, - ประสิทธิภาพดีขึ้น ↑, - broadcast domain เล็ก ↓ ประสิทธิภาพด้าน IT, จัดการประสิทธิภาพ [verify : show vlan brief]
  - ใน Multi-SW Environment
    - VLAN trunk : set ที่ interface ที่เชื่อมต่อระหว่าง SW กับ VLAN ⇒ can carry ข้อมูลได้ >1 VLAN
    - ③ ↳ Config : เข้า inf → -if)# switchport mode trunk [verify : show int f0/0 switchport]
    - Tagging Ethernet Frames (IEEE 802.1Q) : Ethernet Frame →

Dest MAC	Source MAC	TAG	Type/Length	Data	FCS

⇒ Tag ที่ให้ส่งถูก VLAN เมื่อส่งผ่าน Trunk

Native (base) VLAN ไม่มีการ tag & untag → ส่งผ่าน intf ได้เลย (เร็วขึ้น)

① สร้าง : I S(config)# vlan vlan_num → vlan)# name ชื่อ
II S# vlan database → (vlan)# vlan num name ชื่อ

- Assignment : VLAN number ⇒ 1-1005 เก็บ config @ vlan.dat (ใน flash)
  ⇒ 1006-4096 เก็บ config @ running-config (ใน NVRAM)
- assign port ให้ vlan : เข้า intf → -if)# switchport mode access → # switchport access vlan num (ถ้าแก้ : เติม no หน้า + ไม่ใส่ num) (ถ้าลบ : no vlan num)
  - verify : show vlan name ชื่อ, show vlan summary, show int vlan num
- Inter-VLAN Routing → router set เป็น Trunk โดยใช้เทคนิค "sub interface"
  - Config : ① set basic routing (set ip address, no shutdown) [verify : show vlan, show ip route, show running-config]
    ② R(config)# interface g0/0.10 → VLAN → -subif)# encapsulation dot1q 10 → # ip address ip subnet → mask

Chapter 10 VTP (VLAN Trunking Protocol) → ช่วย manage VLAN & NAT (NW Addr. Translation)

- VTP [msg : ISL or IEEE 802.1Q] → การ manage SW VTP จะเป็นการ manage ใน domain
  - Operation : - การ update VTP จะใช้เลข revision number 32 bits (0 - 4294927295) เชื่อตัวที่สูง >
  - ↳ 3 mode : ① Server → can สร้าง, remove, rename VLAN ทุกตัวใน domain ยกเว้น ③
    ② Client → รับข้อมูล VTP มา process, ส่ง VTP msg ออกทุก Trunk
    ③ Transparent → can สร้าง, remove rename แต่ในตัวมัน, เป็นแค่ทางผ่าน

Feature	Server	Client	Transparent
Source VTP Mesg	Yes	Yes	No
Listen to ⟶	Yes	Yes	No
Create VLANs	Yes	No	Yes*
Remember VLANs	Yes	No	Yes*

* Locally Significant only

- Config : 2 แบบ สิ่งสำคัญ : 1) SW cisco 2) มี trunk เชื่อมระหว่าง SW 3) ทั้ง domain 4) มี 3 mode
  ① in global configuration S(config)# vtp version 2 ⇒ # vtp domain ชื่อ → # vtp password pass ⇒ # vtp mode server | mode อื่น ⇒ # vtp server | client | Transparent
  ② in VLAN configuration S(vlan)# vtp v2-mode ⇒ [verify : show vtp status / counters]
- Pruning → manage ให้ traffic ที่ใช้งานใน interface นั้นๆ โดย config ที่ interface ต่างๆ เพื่อ remove ข้อมูลที่ไม่ได้ใช้
  S(vlan)# vtp pruning → เข้า interface → S(config-if)# switchport trunk pruning vlan remove vlan-num

- NAT → แปลง private ip ↔ publish / real ip
  - terminology : 4 type : ① Inside local Addr. (private ip) ② Outside local Addr. ③ Inside global Addr. ④ Outside global Addr. ↳ เหมือนกัน

Private Internet addresses are defined in RFC 1918;

Class	RFC 1918 Internal Address Range	CIDR Prefix
A	10.0.0.0 - 10.255.255.255	10.0.0.0/8
B	172.16.0.0 - 172.31.255.255	172.16.0.0/12
C	192.168.0.0 - 192.168.255.255	192.168.0.0/16

- type : ① Static : กำหนดค่าเอง [map : 1 to 1] → ① R(config)# ip nat inside source static local-ip global-ip
  ② Dynamic : มี pool ของ Global / Real ip [map : ทั้ง many ↔ 1] : real ไหนว่างก็ไปนั้น
  ③ PAT (Port Address Translation) → port กำหนดร่วมกับ NW addr. (map : many ↔ 1)
  (1.1) # ip nat pool ชื่อ start-ip end-ip {netmask netmask | prefix-length}
  (1.2) set ACL (1.3) ip nat inside source list ACL-num pool ชื่อ overload (PAT เพิ่มจาก dynamic)
- Config : 3 ส่วน ① NAT ② INSIDE : R(config-if)# ip nat inside ③ OUTSIDE : R(config-if)# ip nat outside

Config อีกวิธีของ PAT (Single Addr.) (1.1) สร้าง ACL (1.2) ip nat inside source list ACL-num interface f0/0 overload

- verify : show ip nat translations

ชื่อ-สกุล น.ส. สิรอพร เลิศสิภากร กระดาษแผ่นที่ ______ รหัสนักศึกษา [illegible]

for Staples

EIGRP IPv6 & Routing

EIGRP (Enhanced IGRP) : Cisco proprietary (เปิดใช้งานเฉพาะ cisco) แต่ปัจจุบันเป็น standard (1992) → large, multiprotocol NW

↳ **Feature**

- Diffusing Update Algorithm (DUAL) : การันตี loop-free & back up path ทั้งหมดของ routing domain (หา best path)
  → ทำให้ routing ทำงาน very fast convergent (มี convergent time < ของ OSPF
  รวมทั้งมี backpath ทำให้ทำงานเร็วขึ้น → if link down – เลือก path ใน back up มาเลย
- Establishing neighbor Adjacencies : track the status of these neighbors, เชื่อมต่อ กับ directly connected EIGRP routers
- Partial and Bounded updates (EIGRP can update) : update เฉพาะส่วนที่มีการเปลี่ยนแปลง / มีผลกระทบ (minimizing BW) ∴ update < RIP (update ตลอดเวลา)
- Equal (OSPF, RIPv2) and Unequal Cost Load Balancing : ช่วยให้ admin ดูแลระบบเพื่อกระจายการรับข้อมูลในเครือข่ายได้ดี
  : ค่า cost ไม่เท่าแต่ทำ load balance ได้
- Reliable Transport Protocol : use by DUAL

□ ใช้ protocol-dependent modules (PDMs) เพื่อรองรับ protocol ที่แตกต่างกัน เช่น IPv4, IPv6 legacy protocol IPX and Appletalk

↳ รับผิดชอบ :
- maintain EIGRP neighbor and topology table (Neighbor Table → สร้าง topology table → ใส่ routing table เป็น routing) (→ เก็บ shortest path กับ backup sht path)
- คำนวณ matric ที่ใช้ DUAL
- implement filtering and access list
- เชื่อมต่อ DUAL และ routing table
- ทำ redistribution with other routing protocol

□ RTP is EIGRP Transport layer protocol สำหรับ delivery & reception ของ EIGRP packets

↳ เป็น msg ที่อยู่ใน application layer ใช้ maintain ข้อมูล, msg ต่างๆที่ส่งของ EIGRP

□ ไม่ใช่ทั้งหมดของ RTP Packet จะส่งได้น่าเชื่อถือ (msg ≈ OSPF)
- Reliable packet require explicit (ชัดเจน) ack จาก dest.
  - Update, Query, Reply
- Unreliable packet don't require ack จาก dest.
  - Hello, ACK

□ รองรับ authentication (no encrypt routing update) และเป็นที่ยอมรับ (authen ≈ RIPv2, OSPF)

- **Packet Type**
  1. Hello → ที่ใช้ adjacencies ระหว่าง router 2 ตัวที่เป็น neighbor กัน, ไม่ต้องรอตอบ ACK, ส่ง unreliably
  2. Update → update info. ของ dest., update info ของ routing ไปให้ neighbor router
  3. ACK → พอได้รับการ update ก็ตอบ ack
  4. Query → request info. routing จาก neighbor router — multicast } เมื่อต้องการ info. ของ routing เพิ่มก็ query ให้กับ router ที่อยู่
  5. Reply → หลังจากได้รับ query ก็ reply — unicast } ข้างๆ แล้ว reply ตอบกลับว่ามีหรือไม่มี

  reliable (2–5)

- **EIGRP Msg**
  - Multicast dest – MAC : 01-00-5E-00-00-0A — IPv4 : ~~FF02::A~~ 224.0.0.10 – IPv6 : FF02::A
  - protocol : 88 — TLV (Type / Length / Value) Types : 0x0001

□ **Implement EIGRP for IPv4**

- Autonomous System (AS) : collection of NW ภายใต้การดูแลของ single authority (อ้างอิงถึง RFC 1930)
  ↳ AS number → ใช้ exchange routes between AS
  → managed by IANA & assigned by RIRs to ISPs, Internet Backbone providers, และ: ติดต่อสถาบันอื่นๆ
  → 16 bit : 0 - 65535 ⇒ since 2007, 32 bit : over 4 billion
  Δ verify : show ip neighbor, show ip protocol, show ip route
- Configure : R(config)# router eigrp AS-# (≈ router-id @ OSPF)
  (show ip protocol) R(config-router)# eigrp router-id → ถ้าไม่กำหนดจะใช้ loopback intf. → IPv4 adds. ที่สูงสุดที่ active
  R(config-router)# network NW-number [wildcard-mask] (ไม่ใส่ก็ได้)
  (มีหรือไม่มีก็ได้) R(config-router)# passive-interface type number [default] ไม่ให้ update ไปที่ intf. ไหน (เช่น LAN, G, F) (ไม่ใช้ serial)

EIGRP Metrics cost น้อย – ดี, delay น้อย – BW มาก

↳ **Default Composite Formula** : metric = [K1 × BW + K3 × delay] × 256 → $[(\frac{10,000,000}{bw}) + (\frac{sum\ of\ delay}{10})] \times 256$

Complete Composite Formula : metric = [K1 × BW + (K2 × BW)/(256 − load) + K3 × delay] × [K5 / (reliability + K4)]

- BW = lowest } default
- Delay = cumulative } default
- Reliability : worst
- Load : worst load on a link

Default value K1 (BW) : 1, K2 (load) : 0, K3 (delay) : 1, K4 (reliability) : 0, K5 : 0

Delay (DIY) : Giga = 10 μsec, Fast : 100 μsec, Serial : 20000 μsec

- R(config-router)# metric weights tos k1 k2 k3 k4 k5
- set bw : เข้า intf. → R(config-if)# bandwidth kilobits-bw-value

for Staples

for Staples

▫ EIGRP Operation

- Initial : ① R1 ส่ง hello ไป neighbor router ② → all eigrp enable intf. ②. R2 recieves → add to neighbor table → send update → send hello
  ③ R1 update neighbor table
- Discovery : ① R1 update topology table ② R1 → ACK ③. R1 → update (split horizone 4). R2 add information topology table
  ⑤ R2 → ACK

use DUAL → update best routes to each dest. → routing table

▫ DUAL and Topology Table (ใช้ Finite state Machine ในการเลือกเส้นทาง) → show ip eigrp topology [all-link], show ip route

- Successor (S) = route ไปที่ dest ดีที่สุด = neighbor router ที่จะส่งต่อไปที่ dest ที่มีค่า min น้อยสุด
- Feasible Successor (FS) = มีได้ไม่ต้อง Feasible condition] = Backup path
- Reported Distance (RD) = Distance ที่ neighbor ตัวอื่น report distance มาให้เรา = advertised distance จากตัวมัน → dest. มี cost เท่าไร → ลดการ loop
- Feasible Distance (FD) = ——— ที่ไปทาง S = ค่า distance ที่คำนวณไปทาง dest. NW เป็นค่า cost lowest → dest.
- Passive state : statble state & available for use · Active State = recomputed by DUAL

		Interior Gateway Protocol				Exterior Gateway Protocol
		Distance Vector		Link State		Path Vector
IPv4	Classful	RIP	IGRP			EGP
	Classless	RIPv2	EIGRP	OSPFv2	IS-IS	BGPv4
IPv6		RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6	BGPv4 for IPv6

IPv6 Network Address

IPv6	IPv4
128 bit	32 bit
base 16	base 10

▫ IPv4 Issue

- Need for IPv6 → เพราะว่า ip มีจำนวนจำกัด (private ip, NAT), ใช้ใน IoT ได้
- การใช้งานร่วมกัน (coexistence)
  - Migration IPv4 → IPv6 Techniques : ① DUAL Stack : วิ่งทั้งคู่ if ฝั่ง คนใหม่ ไม่กระทบ user
    ② Tunneling (อยากใช้ v6 แต่ core ไม่ support), เปลี่ยนเฉพาะเรา แล้ววิ่งผ่าน V4 เหมือนเดิม
    ③ Translation (คล้าย NAT) = IPv6 ⟷ IPv4

▫ IPv6 Addressing : 128 bit แบ่ง 8 ส่วน [1 ส่วนมี 2byte = 16 bit] → represent base 16 ทุกๆ 4 bit

- ลดแบบวิธีการกำหนดค่า IPv6
  - มี ① Omit Leading 0s · ไม่เขียน partition "0" ที่อยู่ตัวหน้าไม่ต้องใส่ไป 000x (0-F) , 00xx , 0xxx
  - ② Omit All 0 Segment : แทน segment ที่มี "0" ติดกันยาวๆ ด้วย "::" ใช้ได้ที่เดียว only
- Type of IPv6 Addressing
  - IPv6 → ① Unicast - single source
    - Global Unicast → 3 bit แรก "001" "01" 2000::/3" (static config)
    - Link-local → สร้าง Intf. ที่อยู่ติดกัน (ใน local เท่านั้น) FE80::/10
    - Unique Local
  - ② Multicast ③ Anycast = ส่งได้หลาย device
- △ IPv6 Prefix Length : 0-128 , most LANs is /64 เพราะ LAN เล็กสุด 64 bit

▫ IPv6 Routing

- Config Static Route

  R(config)# ipv6 route ipv6-prefix / prefix-length {ipv6-addr (next hop addr.) | exit-intf (exit-interface)} → ต้องมีได้

* ใช้ก่อน config ใช้งาน routing ipv6 ipv6 unicast-routing เพื่อเปิดการใช้งาน

▫ Verify : show ipv6 route static, show ip route ipv6, show running-config | section ipv6 route

▫ Default Static IPv6 Route

R(config)# ipv6 route ::/0 {ipv6-addr. | exit-intf}

▫ Verify : show ipv6 route static

- Config EIGRP for IPv6

  R(config)# ipv6 unicast-routing

  ——— ipv6 router eigrp AS-#

  R(config-rtr)# eigrp router ip 1.0.0.0 → รูปแบบ IPv4

  ———— no shutdown

  △ NW command : เข้า intf. → ipv6 eigrp AS-#

  แต่ passive-intface ต้องเข้าไปใน global config เหมือนเดิม

  △ verify : show ipv6 eigrp neighbor, show ipv6 protocols, show ipv6 route

- RIP v1 → hop count (>15 unreachable), broadcast every 30 secs
- V2 → hop count (max = 15), multicast 224.0.0.9
- OSPF → multicast
  - MAC dest 01-00-5E-00-00-05
    01-00-5E-00-00-06
  - IPv4 dest 224.00.5, 224.0.0.6
  - IPv6 dest FF02::5
  - protocol 89

10 secs multiaccess & point-to-point
30 secs non-broadcast [NBMA]