ชื่อ-สกุล นาย ธีรม คุณากรกุล กระดาษแผ่นที่ 1 รหัสนักศึกษา 88888888

for Staples

Chapter 1 Network Overview

- Network diagram = โครงสร้าง ทพ ที่เราดูแล มี ก.เชื่อมต่ออย่างไร
  - 2 type : ① Physical → บอกว่า port/interface ไหนเชื่อมต่อกันอย่างไร ที่ตน.ไหน ② Logical → บอก ip
- Network protocol ⇒ TCP/UDP, FTP, ARP, SMTP (mail), POP3, IMAP (email), ICMP (internet control message protocol) → ping เป็น 1 ใน command
  - (File transfer protocol) ⇒ รับส่ง file ระหว่าง client-server (Address Resolution Protocol) ⇒ map ระหว่าง IP addr. ว่ามี MAC addr. เป็นอะไร
  - NW Addr. : ① IP addr. (Logical addr.) ที่ L3 ② MAC addr. (Physical Addr.) ที่ L2 protocol & ทน media ③ Port number (Service Addr.) ที่ L4
- Components of network → HW → NW device มี 3 type
  - ① end devices = บริเวณที่มีคนใช้ (NW representation)
  - ▣ hub ▣ switch ⊗ router ② intermediary devices = อุปกรณ์ตัวกลาง เช่น NW access devices, Internetworking devices, Security devices
    - hub, repeater ที่ L1 ⇒ ส่งพร้อมกันเกิด collision ∴ ใช้ CSMA/CD (Carrier Sense Multiple Access with Collision Detection) เมื่อเกิด collision ก็หยุดส่ง
    - switch, bridges ที่ L2 ⇒ Learning / Flooding / Filtering / Forwarding / Aging
    - Router ที่ L3 ⇒ Routing
  - LAN ——— straight, - - - - cross; WAN ⚡ ③ Network media = สื่อกลาง เช่น copper, fiber optic, wireless
  - → SW → ① Switch เลือก port ออก ② Router เลือกปลายทางที่ดีที่สุด
- Types of Networks → Size ① small home nw ⇒ ไม่ใช้งานที่แนบอย่างเดียว ② small office/Home office ⇒ config จากที่แนบภายในได้
  - ③ Medium to Large nw ⇒ มี ก.ติดต่อภายใน 100-1000 เครื่อง ④ World wide NW เช่น Internet
  - → Infrastructure ทั่วไป ① Local Area NW (LAN) ⇒ มีกลุ่ม admin 1 กลุ่มดูแล policy/nw Ex. คณะ ↔ ห้องคอม
    - ② Wide Area NW (WAN) ⇒ มีหลายกลุ่ม admin
  - อื่นๆ : Metropolitan Area NW (MAN), Wireless LAN (WLAN), Storage Area NW (SAN), Personal Area NW (PAN)
- Reliable Network ① fault tolerance → ทนต่อ ค.ผิดพลาด ② Scalability → ปรับเปลี่ยนขนาดได้ไม่ส่งผลต่อ user เดิม
  - ③ Security → จำกัดการเข้าถึง ④ Quality of Service (QoS) → แต่ละ Service มี quality ไม่เท่ากัน
- Layer with TCP/IP & OSI Model

PDU : Protocol data unit

PDU	OSI Model	TCP Model
Data	⑦ Application ⑥ Presentation ⑤ Session	Application
Segment	④ Transport	Transport
Packet	③ NW	Internet
Frame Bits	② Data Link ① physical	NW Access

- Type of Connection in a Lan
  - ข้อจำกัด (UTP cat 5) : ① BW = 100 Mbps ② ยาว 100 m (can ต่อ hub, repeater, switch)
  - 2 type : ① ตรง ② cross → ต่ออุปกรณ์ same
  - ▸ WAN Connection ⇒ เชื่อมระหว่าง router
    - ↳ 2 ฝั่ง → DCE (female) ⇒ ต้องใส่ command clock rate 56000
    - ↳ DTE (male)
  - ▸ สาย console (Rollover Cable) ⇒ router - - → PC
    - RJ-45-to-DB-9 Adapter labeled TERMINAL
    - ⇒ manage command ต่างๆ

for Staples

Chapter 2 Basic Router Configuration

- Port Address ที่ L4 : กำหนดโดย (Internet Assigned Number Authority : IANA)
  - 0-1023 : requesting entities "well known ports" destination port
  - 1024-49151 : registered port = publish ให้ทุกคนใช้
  - 49152-65535 : dynamic or private port "Randomly generate" source port
  - Ex: 20 FTP (data), 21 FTP (control), 25 SMTP (Simple mail transfer), 53 DNS (domain name server) [TCP/UDP], 80 HTTP, 81 HOSTS2 Name Server
- Logical Address : IP Address (IPv4) ที่ L3
  - 5 class : A, B, C, D, E → reserved แตกต่างกันที่ จน. max ของ workstation required
    - ใช้เป็น host (A, B, C); multicast Addr ⇒ ส่งไปให้ทุกเครื่องในวง LAN เดียวกัน
  - แต่ละ nw/node or com ⇒ มี logical name (domain name) & ip unique

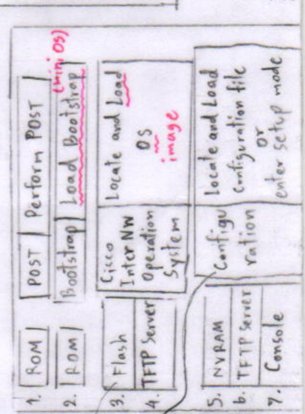
192.168.1 /24	→ prefix length
255.255.255.0	→ Subnet Mask
192.168.1.255	→ broadcast IP addr.
255.255.255.255	→ brodcast NW ↳ ทุกเครื่องใน NW นี้ได้รับ

Class	Structure	Range	
class A :	0 NW · Host · Host · Host (24 bits)	0-127	
class B :	10 NW · NW · H · H (16 bits)	128-191	
class C :	110 NW · NW · NW · H (8 bits)	192-223	
class D :	1110	224-239	multicast
class E :	1111	240-255	experimential

- ▸ private addressing → ip reuse ใช้ซ้ำได้
  - → ถ้าใช้ can ไปข้างนอก intenet ได้มาก

	RFC 1918 Internal Addr. Range	CIDR Prefix
Class A	10.0.0.0 - 10.255.255.255	10.0.0.0/8
class B	172.16.0.0 - 172.16.255.255	172.16.0.0/12
class C	192.168.0.0 - 192.168.255.255	192.168.0.0/16

- Physical Address : MAC Addr.
  - Ethernet : 48 bit ฐาน 2 = 12 ตัว ฐาน 16 → เขียนแบบ 0x ฐาน 16
  - ก่อตั้งโดย IEEE → กำหนด 3 byte (24 bit) code "Organizationally Unique Identifier (OUI)"
    - → 2 กฎ → ① ทุก MAC ที่กำหนดให้ NIC หรือ Ethernet devices อื่นๆ → ใช้ OUI ที่ 3 byte แรก
    - ② ทุก MAC ที่ SAME OUI ต้องกำหนด unique ที่ 3 byte สุดท้าย
- ▸ Message Delivery → ■ Unicast = ส่งจากเครื่องปลายทางโดยตรง ใน nw เดียวกัน
  - ■ Broadcast = ส่งหาทุกเครื่อง เช่น DHCP, ARP ใน nw เดียวกัน
    - → broadcast ip/nw = 255.255.255.255 = FF-FF-FF-FF-FF-FF
  - ■ Multicast = ส่งหาบางเครื่อง & รับเฉพาะเครื่องที่เปิด Service ไว้
    - → ขึ้นต้นด้วย 01-00-5E-xx-xx-xx



for Staples

- Cisco IOS (Internetworking OS)
  - function ① Addressing ② Interface ③ Routing ④ Managing Resource ⑤ Security ⑥ QoS
  - Router & Switch Boot Sequence
    - ROM ① POST (Power on Self Test) → check hw ว่าพร้อมใช้งานไหม
    - ② RUN boot loader SW
    - ROM ③ Boot loader does low-level CPU initialization (mini IOS)
    - Flash/ ④ " ——— " initialize the flash file system
    - TFTP Server ⑤ " ——— " locates & load a default IOS มา run บน ram

เวลาทำ Lab ควร clear configuration check: show startup-config

ปกติ OS จะเก็บที่ Flash ∴ load ขึ้นมาไว้ RAM จะใช้ได้เลย

router จะหยิบมานา startup-config ปกติอยู่ที่ NVRAM

- Accessing a CISCO IOS device
  ① Console Port ② Telnet ③ Secure Shell (SSH) ④ Aux Port
  ↳ Terminal Emulation Program : Putty, Tera Term, SecureCRT, HyperTerminal, OS X Terminal
- Navigating the IOS → 2 mode : ① user ">" ② privileged (enabled) "#"
  - ↳ 2.1 Global Configuration mode "(config)#"
  - ↳ 2.2 Other " ——— " "(config-mode)#"
- The Command Structure
  ① Context Sensitive Help = "?"
  ② Command Syntax Check = enter แล้ว show ว่าใช่, ผิด
  ③ Hot Keys and Shortcuts ④ IOS Examination Commands → show...
- Getting Basic ① ตั้งชื่อ hostname ② จำกัด ก. เข้าถึง ③ กำหนด interface addressing ④ ตรวจสอบ config + ก. เชื่อมต่อ ⑤ Save config
  router(config)# hostname name ↳ Banner msg. Router(config)# banner motd # text #

router# copy running-config startup-config

③ Addressing Devices

1) เลือก interface to config
   - Physical interface / Loopback interface
     Router(config)# interface type port
     " ——— " type slot/port
     " ——— " type slot/subslot/port
   - Switch virtual interface (SVIs)
     Switch(config)# interface vlan number
2) set ip address
   router(config-if)# ip address ip_addr. subnet_mask
   " ——— " no shutdown

④ Verify Connectivity → กำหนดเพิ่มเติม section, include, exclude, begin (ตามต้องการ, ที่ไม่ใช่, ตั้งแต่ตรงนั้นลงมา)

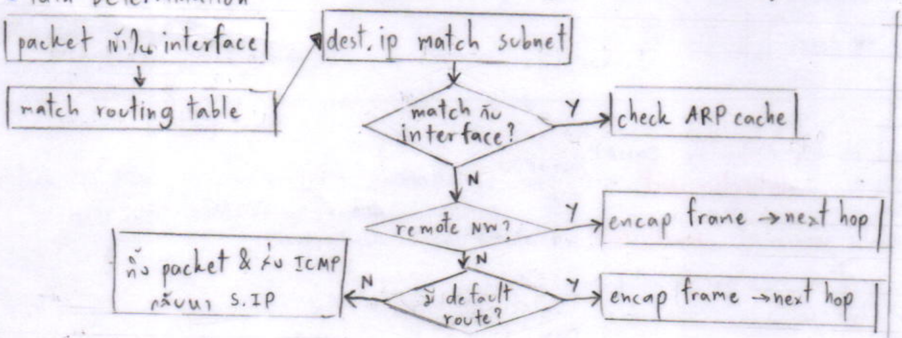
Router# show running-config → ดูว่า config อะไรไปบ้าง
" — " show startup-config → " set " — "
" — " show ip route → ดู routing table
" — " show interface
" — " show ip interface } show info ของ interface ต่างๆ
" — " show ip interface brief → show interface แบบสรุป, set?, status?
" — " trace route
" — " ping
PC > ping ; PC > route print
" — " tradeere ; " — " 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 → process ที่ CPU ↓ ออก interface ไหน?
    ② Fast switching = มีตัวช่วยทำให้ forward ได้เร็ว ↑ 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 addr.
    ② Dynamically " ——— " → กำหนด ip ได้เมื่อ nw มี DHCP (Dynamic Host Configuration Protocol)

รู้ได้มา ip ใน nw
ต้องใช้ ① subnet nw = nw ip & subnet mask
Ex. 192.168.1.0 = 192.168.1.1 & 255.255.255.0
② broadcast nw = เปลี่ยนส่วนที่เป็น host ใน subnet nw เป็น 1 หมด
Ex. = 192.168.1.255
ip อยู่ระหว่างนั้น

- Switching Packet between NW
  ขั้นตอน : นำ dest. ip (L#3) → ดูเทียบใน routing table → นำทางออกไปดูเทียบบน MAC addr. → ได้ dest. MAC (L#2)
- Path Determination



- ▸ Best Path : lowest metric (cost)
  ⇒ Dynamic routing protocol ใช้
  ① Routing Information Protocol (RIP) = จน. hop
  ② Open Shortest Path First (OSPF) = BW มากสุด
  ③ Enhanced Interior Gateway Routing Protocol (EIGRP) = BW, delay, load, reliability
- ▸ Load Balancing = กรณีมี > 1 เส้นทาง
  ⇒ ใช้ทุกเส้นทางได้เท่ากัน
- ▸ Administrative Distance (AD) : "trustworthiness"
  = ระยะน่าเชื่อถือของผู้ดูแลถึง protocol ต่างๆ
  → Connected = 0, Static = 1, Internal EIGRP = 90, OSPF = 110, RIP = 120

- The Routing Table

D 10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Serial0/0/0
① ② ③ ④ ⑤ ⑥ ⑦

① → Identifies how the network was learned by the router → C = directly connect, B = BGP, D = EIGRP, S = Static
② → " the destination network
③ → " administrative distance (trustworthiness) of the route sorce.
④ → " metric (cost) to reach the remote nw
⑤ → " next-hop IP addr. to reach the remote nw
⑥ → " amount of elapsed time since the nw was discovered
⑦ → " outgoing interface on the router to reach the dest. nw

- Classful Addressing → update ตาม class
- Classful Inter-Domain Routing
  - ▸ summarization : เงื่อนไข : ① อยู่ติดกัน ② ไปทางเดียวกัน
  - ! ถ้า set ใหม่ ต้อง no ip เก่าก่อน
  - วิธีทำ ① แปลง ip → ฐาน 2
    ② Group bit ที่เหมือนกันมากสุด
- VLSM
  - ▸ Fixed Length Subnet Masking
    วิธีทำ ① prefix ใหม่ − prefix เก่า = จน. bit ที่เปลี่ยน
    ② เปลี่ยนทุก bit เป็น 1 แล้วนำค่าฐาน 10 ไป + ip เริ่ม สลับกับ +1 ทำงานไปเรื่อยๆ

- Routing

① Static Routing → Manual
ข้อดี : security, ใช้ resource น้อยใน ก. process, ลด routing entry
ข้อเสีย : ลำบากใน ก. scalability, can เปลี่ยนเส้นทาง
ควรใช้เมื่อ : nw ขนาดเล็ก, จน. router น้อย, nw มีทางออกทางเดียว "stub nw"
4 type : 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 (① | ②)

Config : Next-hop Option
ระบบที่ R(config)# interface G0/0
" — "-if)# ip address 172.168.1.0 255.255.255.0
" ——— " no shutdown
} Basic set ทุก interface ได้ทั้ง router

ต่อไปนี้ →

R1(config)# ip route 172.168.1.0 255.255.255.0 → 172.168.1.2 แบบที่ 1 / ↳ S0/0/0 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)

ชื่อ-สกุล นาย ธีม คณาภรณ์ กระดาษแผ่นที่ 2 รหัสนักศึกษา

for Staples

Chapter 4 Distance Vector Routing Protocol RIP ver 1

▷ Dynamic Routing Protocol

- ▫ fn : share info ระหว่าง router • auto update routing table when topology เปลี่ยน (เมื่อมีปัญหา) • หา best path
- ▫ purpose : หา remote nw (nw ที่อยู่ห่างไกล) • ปรับปรุง routing info • เลือก best path ไป nw • can หา new best path ถ้า path ปจบ. ไม่ว่างยาว
- ▫ component : ① Algorithm : ใช้เพื่อจัดการ routing info & best path
  ② Routing protocol msg : ค้นพบหา neighbor & แลกเปลี่ยน routing info. (best path)

▫	Dynamic Routing vs.	Static Routing
ก. ยุ่งยากในก. Config	✗ ขนาด nw (same เดิม)	∝ ขนาด nw (เพิ่ม command ทุก router)
Required ก.รู้ admin	Advance (bec. config basic → nw. ใหญ่ + ซับซ้อนอาจมีปัญหา	No เพิ่มเติม (กำหนดเส้นทางเองโดย command)
Topology change	ปรับ auto	admin config ใหม่ all
Scaling	เหมาะ simple & complex (router ที่ไม่ใช่ directly ไม่ต้องทำไร)	เหมาะ simple topologies
Security	น้อยกว่า	มากกว่า
Resource usage	ใช้ CPU, mem (เก็บ routing info), link bandwidth	No เพิ่มเติม
Predictability	Route ∝ current topology	Route → Dest. ทางเดิมตลอด

▷ Classifying Routing Protocol

DRP
- (ภายใน AS) Interior Gateway Protocol (IGP)
  - Distance Vector P.
    - • ประกาศ vector [distance, direction]
    - • incomplete view ของ nw topology
    - • periodic update (ตามเวลา) (ซ้ำๆ)
    - RIP (Routing Info P.) → RIP v.2
    - IGRP (Interior Gateway Routing P.) → EIGRP (Enhanced)
  - Link-State P.
    - • complete nw topology (เก็บข้อมูล all)
    - • update ไม่ใช่ periodic
    - (Open Shortest Path Fist) OSPF
    - IS-IS (Intermediate System to Intermediate System)
- Exterior Gateway Protocol (EGP) (ภายนอก AS)
  - BGP (Border Gateway Protocol)
  - • Autonomous System (AS) เป็นกลุ่มของ router ภายใต้ ก.ควบคุม ของ Single authority (1 กลุ่ม policy or ผู้ดูแล)

▫ 2 type

for Staples

① Classful routing P. → update ตาม class ไม่ส่ง subnet mask ใน routing update
② Classless " → ส่ง subnet mask ใน routing update

- ▫ Convergence : ถูกกำหนด when routing table ของ all router มีสถานะคงที่ (ไม่เปลี่ยนแปลง)
  - ↳ 2 type : Slower → RIP & IGRP, Faster (มีก. update when เกิดก.เปลี่ยนแปลง ดีกว่า) → EIGP & OSPF

▷ Routing Protocol Metrics

- ▫ Metric : ค่าตัวที่ใช้ในก.คิด/หา ในก.ไป dest. NW จะเลือก best path ไหน เช่น Hop count, BW, Cost, Delay, Load, Reliability
- ▫ Load balancing : nw มีเส้นทาง > 1 ที่มี metric เท่ากัน → เฉลี่ยให้ ก.ใช้งานแต่ละเส้นทางเท่าๆกัน

▷ Administrative Distance of a Router (AD) → ใช้เลือก protocol ใน ก.routing

• วัตถุประสงค์ : ค่าตัวเลขที่เจาะจงลักษณะพิเศษของ particular (จำเพาะ) route

Route Source	Connected	Static	Internal EIGRP	OSPF	RIP	EIGRP Summary route	Extreme BGP	IGRP	IS-IS	External EIGRP	Internal BGP
AD	0	1	90	110	120	5	20	100	115	170	200

▷ Distance Vector Routing Protocol Ex. RIP, IGRP, EIGRP

- ▫ Distance Vector Technology ต้องรู้ 2 สิ่งเมื่อใช้คือ ① Vector or direction, ก.แลกเปลี่ยนตามจุดกำหนด ② Distance to final dest. (cost)
- ▫ ลักษณะทั่วไป : periodic update, neighbor (⊗ใกล้เคียง), broadcast (255.255.255.255) update, เอา routing table all ไป update
- ▫ " ของ Routing Protocol : เกณฑ์ที่ใช้บอกว่า routing protocol ดี/ไม่ดี ① Time to convergence → เวลาที่เป็น steady state ของ routing table ที่เกิดก.ปป.แล้ว ② Scalability ขยายขนาดได้เท่าไร ③ Resource usage ④ Implementation & maintenance
- ▫ NW Discovery (ค้นพบ) (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 อื่นตัวว่าห่างเท่าไร ?

Network	Interface	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	Fa0/0	1

เวลาส่งข้อมูล update ส่งแค่ 2 ช่องนี้

- ▫ Routing Table Maintenance

for Staples

  - • Periodic update : RIP update timer (default 30s), Invalid timer (info เริ่ม lost) (default 180), Holddown Timer (ถ้า down → hold ไว้ไม่ว่า up ตอนไหน) (default 180), Flush (ทิ้ง) timer (default 240)
  - • Bounded Update : EIGRP → update แค่ที่เปลี่ยน
  - • Triggered Update → update เลยไม่รอ periodic time
  - • Random Jitter → ใช้ใน nw ที่เป็น multiple access router หลายตัวเชื่อมต่อกัน → if เกิด update พร้อมกัน ∴ จึงใช้ Random

□ ปัญหา standard DV. ① Routing Loops เกิด when intf ไหน down จะถูกลบออกจาก table → ค่าเก่าใน neighbor จะกลับมา update (วน update → hop เพิ่มขึ้น → ∞)
↳ แก้ปัญหา : ① set max hop = 15 → if hop = 16 → unreachable (มัน down ไปแล้ว)
② holddown timer (ถ้า intf 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
⑤ ③ with ④ → ถ้าเกิด unreachable จะ over rule split horizon โดยส่ง ip intf ที่ down (hop = 16) ให้ intf นั้น hop = 16
⑥ IP&TTL (Time to Live) เกิด ก.วน update แต่จะหยุดเมื่อ TTL = 0

	RIPv1	RIPv2	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 & maintenance	simple	simple	simple	complex

▷ RIP version 1 AD = 120

□ คุณสมบัติ : • classful, DV • metric = hop count • hop count > 15 unreachable • update broadcast ทุก 30 s
→ encapsulated ใน UDP segment ด้วย source & dest port = 520

Transport Layer RIPv1 Message Format ของ full routing table

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

Bit 0 … 7 | 8 … 15 | 16 … 23 | 24 … 31

	Bits 0–7	Bits 8–15	Bits 16–31
header ←	Command = 1 or 2 (Request / Reply)	Version = 1	Must be zero
Route Entry	Address family identifier (2 = IP)		Must be zero
Route Entry	IP address (Network Address) → dest.		
Route Entry	Must be zero		
Route Entry	Must be zero		
Route Entry	Metric (Hops) → 1-15 ที่มาจาก table +1 → ส่งออก		
	Multiple Route Entries, up to maximum of 25		

□ msg มี 2 type
① Request → ส่ง routing table
→ ไป intf ที่ config ไว้เพื่อรอรับ ก.update
② Response → ส่ง info ของ routing table

□ ip addr. แบ่งตาม class A, B, C

□ Basic RIPv1 config
① ทำ basic config
② สั่ง router rip R1(config)# router rip
+ ค่าสั่ง nw R1(config-router)# network {nw ip ที่ต่อกับ router}

□ Verification & troubleshooting : show running-config or ip route or ip protocols, debug ip rip
• passive intf command (ไม่ update intf ที่ไม่ต้องการ) R1(config-router)# passive-interface {intf-type}(Fa/G/S) {intf-number}(0/0, 0/0/0)

□ Automatic Summarization : RIP Auto Summarize classful nw → เพื่อลด size routing table
↳ ข้อดี : • ลด size routing update • single routes are used to represent multiple routes → faster lookup in the routing table
↳ ข้อเสีย : • ไม่ support discontiguous nw (major nw เดียวกันแต่ไม่อยู่ติดกัน) → อาจจะเกิด load balancing ได้
• boundary Routers : summarize RIP subnet from 1 major nw to another (Boundary router = router ที่ต่อไป nw อื่น)
• Processing RIP update : ดูตัวที่ส่ง & ตัวที่ update ไป (intf) ว่ามี classful เดียวกันไหม → y : update subnet nw เช่น 172.16.1.0
↳ n : update classful เช่น 172.16.0.0

□ default route & RIPv1 ถ้าไม่เจอ nw ออกใน routing table (อาจจะเพราะคนละ protocol) → จะออก default route
R(config)# ip route 0.0.0.0 0.0.0.0 s0/0/1
defaut 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 VLSM (Variable Length Subnet Masking), Support Route Summarizing (Prefix Aggregation))
not support discontiguous subnet	update next hop addr.
not support VLSM bec. ไม่ส่ง subnet mask	ใช้ authentication routing (รองรับ discontiguous แบบนี้)
routing update → broadcast (255.255.255.255)	Routing update → multicast

ใช้ timer ป้องกัน routing loop
ใช้ split horizon or split horizon with poison reverse
ใช้ triggered update
max hop count = 15

□ ข้อจำกัดของ RIPv1
เป็น virtual interface can เพิ่มใน routing ให้ update ออกไปได้
{ • loopback intf → ping ไป → ip virtual intf → reply ให้
• Null intf → ใช้ดักคนที่จะเข้ามา channel ที่เราไม่ต้องการ → วิ่งออก null intf → packet discard เอง (timeout)
• Static route & null intf → null intf จะรับหน้าที่เป็นทางออกของ static route
R(config)# ip route {summary-static-route (major nw)} {subnet-mask} Null0 → ของ static supernet route

• Route redistribution (จัดสรร) → อยากให้ rip รู้จัก static ต้องทำเพิ่มในส่วนของ rip ส่ง static ไปด้วย)
R(config-router)# redistribute static
• Verify & test Connectivity : show ip interface brief, ping (ผล : ! = ได้, U = ไม่ได้, . = timeout), traceroute
• RIPv1 : classful, ไม่ส่ง subnet mask, summarize nw @ major nw boundaries, if nw เป็น discontiguous & RIPv1 config convergence จะมาไม่ถึง
• ตรวจ routing table : debug ip rip (content of routing update), ถ้าเป็น RIPv1 จะไม่ใส่ subnet mask รวมใน nw addr.

ชื่อ-สกุล ธีม คุณากรกุล กระดาษแผ่นที่ 3 รหัสนักศึกษา

▷ RIPv2 → show ip protocols

- config
  - Enabling & verify RIPv2
  - Config RIP → RIPv1 → can รับได้ทั้ง v1 & v2 แต่ส่งได้แค่ v1
    - → RIPv2 → can รับ & ส่งได้แค่ v.2
  - Auto-Summary & RIPv2 → auto sum route @ major nw boundaries
    - → sum route ด้วย subnet mask ที่น้อยกว่า classful subnet mask
  - disabling Auto-Summary : no auto-summary bec. when ที่ nw topology ต้องเป็น 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)		

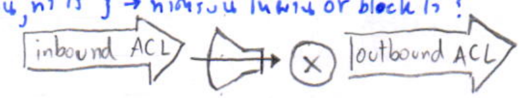
Route Entry: IP address, Subnet Mask, Next Hop, Metric (Hops)

Multiple Route Entries, up to a maximum of 25

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

▶ Access Control List = ควบคุม ก. เข้าออก → ตรวจสอบ → check → source → dest ผ่านไหม?
↳ ท่าอย่างนี้ (เช่น FTP) ผ่านไหม?
⇒ ดูจาก Conversation

- Packet filtering ดู ① dest, source @ L2 ② protocol อะไร ③ ไป nw ไหน, ทำไร } → ทำการนั้นให้ผ่าน or block ไว้?
- Operation → ทำงานเป็น sequence statement
  - → last statement เป็น implicit deny → block → discard



- Standard IPv4 ACLs vs. Extended IPv4 ACLs

Standard IPv4 ACLs	Extended IPv4 ACLs
- check source addr.	- check source & dest addr.
- ปกติ permits or denies ทั้งชุด protocol	- ปกติ permits or denies specefic (เฉพาะ) protocol
access-list 10 permit 192.168.30.0 0.0.0.255	access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
- number ACL : 1-99 & 1300-1999	- Number ACL 100-199 & 2000-2699

- wildcard → invert ของ subnet mask
  - → 0 = match/fix , 1 = ignore/อะไรก็ได้
  - → วิธีหาจาก set ของ ip (match range)
    - ① นำ ก. สัมพันธ์ของ bit ที่ซ้ำกันใส่ wildcard mask ตน.นั้น = 0
    - ② bit ที่เหลือใส่ 1
    - if กรณีคิดไม่ได้ลองแยก Pattern or/and ล้วนคิด สุดท้าย wildcard จะเหมือนกัน
  - → หา wildcard 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
    - R1(config)# access-list 1 permit host 192.168.10.10] same กัน
  - → 255.255.255.255 = ignore all ใช้ any
    - { R1(config)# access-list 1 permit 0.0.0.0 255.255.255.255
    - R1(config)# access-list 1 permit any] same กัน
- Guildline for (3Ps) → One ACL/protocol = ctrl traffic flow บน intf, ACL ต้อง define แต่ละ protocol enable on intf
  - ACL creation → One ACL/direction = ctrl traffic in 1 direction at time on an intf, แยก ACL ctrl in & out bound traffic
  - → One ACL/interface = ACL ctrl traffic on an intf, Ex. G0/0
  - where > Extended ACL : @ close source > standard ACL : @ close dest.
- Config ACLs > Standard ACL
  - → number (number ที่ต้องการ ก. สร้าง)

```
Router(config)# access-list access-list-number
                deny/permit/remark = comment
                source [source-wildcard] [log]
```

ถ้า remove all : no access-list
ถ้าแก้ ① no access-list num# → สั่งใหม่
② no บรรทัด# → บรรทัด# คำสั่ง

เลือกเอา
ตัวแบบ

- → in interface

```
Router(config-if)# ip access-group
    {access-list-number / access-list-name}
    {in/out}
```

ถ้า remove all : no ip access-group
ถ้าแก้ → บรรทัด# คำสั่งใหม่
↳ บรรทัดที่แทรก

- → name

```
Router(config)# ip access-list [standard | extended] name
Router(config-std-nacl)# [permit/deny/remark] {source
        (source-wildcard)} [log]
```

- Verify : show ip interface , show access-lists
- Securing VTY port → อนุญาตให้คนนอกเข้ามา permit เฉพาะที่เราต้อง ก. ให้เข้า

```
Router(config-line)# access-class access-list-number
                     {in [vrf-also] /out}
```


> Extended ACL : filter : source/dest. addr. , protocol, port number

```
access-list access-list-number {deny/permit/remark}
protocol source [source-wildcard] [operator operand]
[port port-number or name] destination [destination-wildcard]
[operator operand] [port port-number or name] [established]
```

ที่เหลือ same standard
มีได้ทั้งแบบ number & name
- debug-output
: debug ip packet ACL-num#

→ ใส่, ไม่ใส่ก็ได้
→ ip, tcp, udp
port dest.
คำสั่งใน [] หมายถึงใส่, ไม่ใส่ก็ได้

Chapter 6 OSPF & DHCP

▷ Link-State Routing Protocol = เป็น protocol ที่ต้องสร้าง complete map ของ nw topology ขึ้นก่อน → มา shortest path first (SPF) (โดย dijkstra) (↗ มี info all)

- ▫ เหมาะกับ : ① large nw, ② fast convergence, ③ admin ควรมีค.รู้ดี
- ▫ 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 neighbors → ได้รับแล้วเก็บลง db ⑤ router เอา all LSP เก็บใน db (รูปแบบ tree) + Adding OSPF → routing table
- ▫ ข้อดี : ① สร้าง topology map จึง can หา shortest path, ② fast convergence ข้อมูลทันที่ ปป. ③ LSP sent only when change topology (แลกเปลี่ยนเฉพาะข้อมูลที่ ปป. → คำนวณ shortest path) ④ hierarchical design (NW ใหญ่จะดี) → ลด resource bec. แลกข้อมูลแค่ใน area
- ▫ ข้อเสีย : ① ใช้ mem ในก.เก็บ all link-state เยอะ ② ใช้ CPU ในก. คำนวณ ③ ตอนส่ง LSP ตอนแรก ใช้ BW มากๆๆ

▷ 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 hello (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 : สร้าง ก.เชื่อมต่อ ① Down state (เริ่มต้น) → ② Init state (เริ่มส่ง hello) → ③ Two-Way state (ตอบกลับ hello) → Exstart state → Exchange State → Leading state → Full state (ระหว่าง router update ข้อมูลเรียบร้อย)
- ▫ Config Single-Area OSPF v2 : router ospf process-id → 1-65,535, เป็น locally significant
  R(config-router)# router-id 1.1.1.1 ① → ถ้าไม่ set can ใช้ loopback ②, active interface ip สูงสุด ③ but ควรมี 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]

$$cost = \frac{10^8 \text{ bps}}{\text{intf BW bps}}$$

10 Gb Ethernet	=	100×10^8	→ cost =	1
Gb	"	= 10×10^8	→ "	1
Fast	"	= 10^8	→ "	1
Serial	"	= 1.544×10^6	→ "	64

10 Gigabit Ethernet = 10000 (หน่วย 10^6)
Gigabit " = 1000 (หน่วย 10^6)
Fast " = 100 (หน่วย 10^6)

→ ต้องการเปลี่ยนค่า cost
→ เปลี่ยนที่ ref BW auto-cost reference-bandwidth bandwidth_mbps
→ เปลี่ยนที่ BW : R(config-if)# bandwidth 64 (EIGRP & OSPF can ใช้ได้)
→ " 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 an OSPF 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
```
→ เพื่อให้มันเอาค่า default route update ไปใน OSPF

▷ DHCP (Dynamic Host Configuration Protocol) → ช่วย config ใน host ให้ auto (เช่น ip, subnet mask, default gateway, dns)

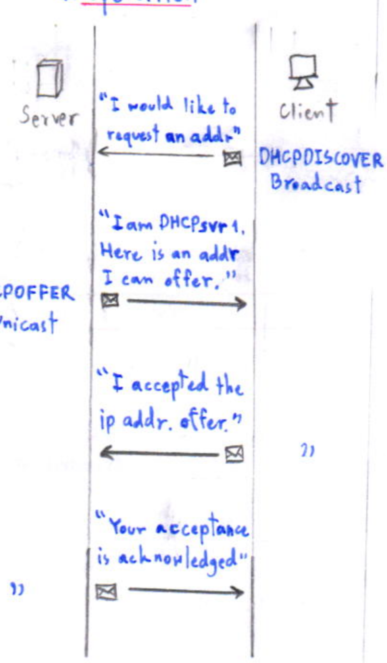
- ▫ method : ① Manual Allocation : admin assign เอง
  ② Automatic Allocation : DHCPv4 auto assign addr. จาก pool & ไม่มี lease (เช่า) time
  ③ Dynamic Allocation : ใครเปิดขึ้นมาก่อนจะได้ ip ไป มี lease time → ถ้าหมด lease time ต้อง re ip ใหม่
- ▫ config

```
R1(config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9
   "        ip dhcp excluded-address 192.168.10.254
   "        ip dhcp pool LAN-POOL-1   → ชื่อ
R1(dhcp-config)# network 192.168.10.0 255.255.255.0   (nw ip ที่จะแจก)
   "        default-router 192.168.10.1   ต้องมี
   "        dns-server 192.168.11.5       } มี/ไม่มี ก็ได้
   "        domain-name example.com       }
   "        end
```

to disable dhcp : no service dhcp
ถ้า ip ไม่ขึ้น → ipconfig /release → ipconfig /renew
→ ติ๊ก dhcp/static สลับไปมาใหม่ (หน้า set ipv4)

- ▫ Verify show running-config | section dhcp
  show ip dhcp binding
  show ip dhcp server statistics
  ใน PC ใช้ ipconfig /all
- ▫ Config DHCP Client (แจก ip ให้ client) : R1(config-if)# ip address dhcp
  " no shutdown (อย่าลืม)
- ▫ debug : same Extended

▫ operation



ชื่อ-สกุล ธาม คุณากรกุล กระดาษแผ่นที่ 4 รหัสนักศึกษา 59080698

for Staples

Chapter 7 Basic Switch Address Resolution Protocol

▷ LAN Design ⇒ Borderless sw nw design : มีอยู่ทั่วไป :: - Hierarchical, - Modularity, - Resiliency, - Flexibility

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

ทุก Layer เป็น SW {
[1] Core ⇒ อุปกรณ์ที่รองรับ BW สูง → ทำให้มี speed ↑ ใน ก.เชื่อมต่อ nw { Layer 3 Support, [Gig / 10 Gig Ethernet]
[2] Distribution ⇒ เชื่อมระหว่าง [1] & [2], Security Policy / Access Ctrl } Redundant components ⇒ มีแหล่งไฟสำรอง
[3] Access ⇒ ติดต่อ end device, Port Security, VLAN, [Fa (100 M) / Gig Ethernet], Power over (PoE) Ethernet รองรับ (↓ 1 LAN) or ไม่ก็ได้ (↳ 1 LAN, 1 power line)
} Link aggregation ↳ ทำให้ BW device เพิ่ม ↑, Quality of Service (QoS)

□ เพิ่ม ค.สามารถในการใช้ได้ LAN BW & ปสภ. ให้ max

• f^n & ก.วาง Server ① Enterprise S. (รวมทั้งองค์กร) ⇒ ติดตั้ง @ MDF (Main Distribution Facility : Core) ⇒ ทางผ่านไปข้างนอก ที่ใกล้กับมัน
② Workshop S. (เฉพาะกลุ่ม) ⇒ ติดตั้ง @ IDF (Intermediate D. F. : Distribution) ⇒ เชื่อมต่อระหว่าง cross กับ access ที่ใกล้กับมัน

• Collision detection issue (ปัญหา ก.ตรวจจับ ก.ชน)

• Segmentation issue (ปัญหา ก.แบ่งส่วน) ⇒ ทำให้ปัญหา ก.ใช้งาน nw มี ปสภ.ที่ดีขึ้น

VCC (Vertical cross-connect) : optical fibre :⇒ MDF ↔ IDF
HCC (Horizontal 〃) : UTP :⇒ Distribution ↔ Access

• Broadcast domain issue ⇒ ใน nw เกิดหลอดเช่น broadcast หา MAC Addr. ∴ broadcast NW ควรมีขนาดเล็ก

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

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

▷ SW Environment

□ SW Operation ① Learning :: รับ frame เข้า sw จะเรียน Source MAC Addr. ว่าต่อกับ port ไหน + reset Aging
② Aging :: อายุของ MAC Addr. → if หมด → ทิ้ง

กรณี ก.ส่งออก {
③ Flooding :: ส่ง frame ออกทุก port ของ sw when frame เป็น 1) broadcast, 2) multicast, 3) unknown unicast (↱ กรณี dest. ไม่อยู่ใน table)
④ Forwarding :: ส่งไป dest. (กรณีพบใน table)
⑤ Filtering :: if ได้รับ frame ไป dest. จาก port ที่เป็นทางไป dest. (source & dest. บน same interface) จะทำ ก. filter ทิ้ง
}

□ SW Methods :: ① Collision Domains ⇒ domain ที่เกิด ก.ชนตอนที่ส่งข้อมูลพร้อมกัน "ดู @ SW เป็นตัวแบ่ง"
② Broadcast ~ ⇒ domain ที่ส่ง broadcast → อยู่ domain เดียวกันจะได้รับ "ดู @ router เป็นตัวแบ่ง"

▷ Basic SW Concept & Configuration

□ Basic SW Config ◦ SW Boot Sequence = same router

manage intf : S(config)# interface vlan num
S(config-if)# ip address ip subnet
S(config-if)# no shutdown

default Gateway : S(config)# ip default-gateway ip

◦ Preparing of Basic SW Management : SW ไม่มี Loopback ∴ กำหนดเป็น SVI (SW Virtual interface) → VLAN

◦ Config SW Port → Duplex Communication: ① Full ② Half (SW ที่ต่อกันต้องเลือก same กัน)

for Staples

เข้าที่ intf → S(config-if)# duplex full → S(config-if)# speed 100 (ปรับ speed)

→ Auto MDIX : ปกติ SW จะเชื่อมกันด้วย cross-over แต่ต้องการเปลี่ยนเป็นสายตรง

เข้าที่ intf → S(config-if)# duplex auto → S(config-if)# speed auto → S(config-if)# mdix auto

→ Verify Port Config
show int f0/0, / startup-config, / running-config, / flash, / version, / history, / ip f0/0, / mac-address-table

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

Config : S_C(config)# (Change host name) ip domain 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/ออกได้ - ไม่ได้

S(config-if)# Switchport mode access → # switchport port-security → เลือกมา 1 ประเภท

Secure MAC Addr. : → ① Static :: S(config-if)# switchport port-security mac-address MAC-ADDR
② dynamic :: S(config-if)# switchport port-security mac-address sticky → learn frame ที่วิ่งมาก่อน → record ไว้

กรณีเพิ่ม จน. MAC : # switchport port-security maximum MAX

Violation mode : ① protect :: security violation protect mode
② restrict :: security violation restrict mode ⇒ ส่งไปเก็บว่ามีอะไรเกิดขึ้นบ้าง
③ shutdown :: security violation shutdown mode ⇒ default

[verify : show port-security int fa0/0, show port-security address]

Security Violation Modes

Violation Mode	Forward Traffic	Send Syslog Message	Displays Error Message	Increase Violation Counter	Shuts Down Port
Protect	No	No	No	No	No
Restrict	No	Yes	No	Yes	No
Shutdown	No	No	No	Yes	Yes

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

IPv4 : Classless [ต่อ P.1-2] : - Variable Length Subnet Masking (VLSM) : แบ่งทุก nw ไม่เท่ากัน ! ⇒ เลือกของ ชิ้นใหญ่ - เล็ก
- Fixed ~ : แบ่งทุก nw เท่ากัน

for Staples

Chapter 8 LAN Redundancy & Spanning Tree Protocol (STP)

▷ Issue with Layer 1 Redundancy :: ① MAC Addr. instability ⇒ MAC Addr. table จะไม่เสถียร bec. เปลี่ยนแปลงบ่อย
② Broadcast storms ⇒ วนส่งไปเรื่อยไม่หยุด ③ Multiple frame transmission ⇒ start: unknown unicast → ทำให้ dest ได้รับหลาย frame but source ส่ง 1 frame

▷ STP ⇒ จุดหมาย: หา block port → block มัน → ไม่ให้เกิด traffic ขึ้น เพื่อแก้ปัญหา

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

□ ขั้นตอน: ① หา Root Bridge = เลข priority min [Rule: [1] 1 RB/1 NW [2] 1 RP/1 RB [3] 1 DP/segment]
ดูที่ BPDU ② หา path cost all ③ หา Root Port → path cost min → อีกฝั่งจะเป็น Designated Port
(Bridge protocol data unit) ④ ถ้าใน segment มี path cost เท่ากัน → ดู BID min เป็น designated port → อีกฝั่งเป็น block port
↳ มี RB & source [802.1D]

□ Config: วิธีที่ I: S1(config)# spanning-tree VLAN 1 root primary
(if S1,S2 เป็น RB) S2(config)# spanning-tree VLAN 1 root secondary
วิธีที่ II: S3(config)# spanning-tree VLAN 1 priority 24576 ↑(เพิ่มให้มากๆ)
[verify: show spanning-tree]

□ แบบ Extended System ID: B.Priority (2 byte) ➡ B.Priority (per VLAN) (4 bit) + Extended Sys ID (VLAN) (12 bit) ➕ MAC Addr. (6 byte) ∴ BID = 8 byte
∘ PVST+ (มีก่อนมาตรฐาน IEEE 802.1D STP) → เพื่อทำ load balancing โดยแยก root/vlan
[verify: show spanning-tree active]
∘ Rapid PVST+ → มา alternate port (ไม่ให้ block ไปเลย can รับส่งข้อมูลได้บางส่วน)
↦ 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 นี้ไม่ต้อง ก.ให้ BPDU ไปด้วย
→ config: S1(config)# spanning-tree mode rapid-pvst → # int ทำ p-to-p → # spanning-tree link-type point-to-point
↳ ถ้า clear all: clear spanning-tree detected-protocol

Protocol	Standard	Resources 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	Med or High	Fast	Per Instance

Chapter 9 VLANs & Inter VLAN

▷ VLAN = ทำ partition (แบ่งแยก: อยู่คนละ nw or broadcast domain กัน) Layer 2 เช่น SW และทุก ก. ติดต่อได้แค่ใน VLAN เดียวกัน
□ ข้อดี: -security ดีขึ้น, -ลด cost, -ปสภ.ดี, -broadcast domain เล็ก↓, -ปสภ.ด้าน IT, จัดการ ปสภ. [verify: show vlan brief]
□ in a Multi-SW Environment
∘ VLAN Trunk: set ที่ intf ที่เชื่อมต่อระหว่าง SW ที่มี VLAN → can carry ข้อมูลได้ >1 VLAN
Native (base) VLAN ไม่ทำ ก.tag & untag → ส่งผ่าน intf ได้เลย (เร็วขึ้น)
③ ↳ config: เข้า intf → -if)# switchport mode trunk [verify: show int f0/0 switchport]
∘ Tagging Ethernet Frames (IEEE 802.1q): Ethernet Frame →

Dest MAC	Src MAC	Tag	Type/Length	Data	FCS

⇒ Tag ทำให้รู้ว่าอยู่ VLAN ไหน when ส่งผ่าน Trunk

□ Assignment: VLAN number ⇒ 1-1005 เก็บ config @ vlan.dat (ใน flash) ① สร้าง: I S(config)# vlan vlan_num → vlan)# name ชื่อ
⇒ 1006-4096 เก็บ config @ running-config (ใน NVRAM) II S# vlan database → (vlan)# vlan num name ชื่อ
② assign port ให้ vlan: เข้า intf → -if)# switchport mode access → # switchport access vlan num (if แก้: เติม no หน้า + ไม่ใส่ num)
-verify: show vlan name ชื่อ, show vlan summary, show int vlan num (if ยกเลิก: no vlan num)
▷ Inter-VLAN Routing → router set เป็น trunk โดยใช้เทคนิค "sub interface"
□ config: ① set basic routing (set ip address, no shutdown)
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] → ก.manage SW VTP จะเป็น ก.manage ใน domain
□ Operation: ก.update VTP จะใช้เลข revision number 32 bit (0-4294927295) เลือกค่าที่สูง>
↳ 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 แค่ในตัวมัน, เป็นแค่ทางผ่าน

Feature	Server	Client	Transparent
Source VTP Msg.	Yes	Yes	No
Listen to VTP Msg.	"	"	"
Create VLANs	"	No	Yes*
Remember VLANs	"	"	"*

*Locally Significant only

□ Config: 2 แบบ ต้องมีก่อน: 1) SW cisco 2) มี trunk เชื่อมระหว่าง SW 3) ตั้ง 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
② in VLAN configuration S(vlan)# vtp v2-mode [verify: show vtp status/counters] ↘ # vtp server|client|transparent
□ Pruning → manage ให้ traffic ที่ใช้งานใน intf น้อย↓ โดย config ที่ intf ต่างๆ เพื่อ remove ข้อมูลที่ไม่ได้ใช้
S(vlan)# vtp pruning → เข้า intf → S(config-if)# switchport trunk prunking vlan remove vlan num

▷ NAT
□ Terminology: 4 type: ① Inside local Addr. (private ip) ② Outside local Addr.
③ Inside global Addr. ④ Outside global Addr. ↦ same เดิม
□ type: ① static: กำหนดค่าเอง [map: 1↔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 ไหนว่าง ก็ใช้อันนั้น } (1.1) # ip nat pool ชื่อ start-ip end-ip {netmask netmask | prefix-length}
③ PAT (Port Addr. Translation) → port ทำงานร่วมกับ NW addr. (map: many↔1) } (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 อีกวิธี ของ 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

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

ชื่อ-สกุล นาย ธีม คุณากรกุล กระดาษแผ่นที่ 5 รหัสนักศึกษา

Chapter 11 EIGRP IPv6 & Routing

Dynamic Routing Protocol
- Interior Gateway P.
  - Distance Vector Routing P.
  - Link State Routing P.
- Exterior Gateway P.
  - Path Vector

	Distance Vector Routing P.		Link State Routing P.		Path Vector
Classful	RIP	IGRP			EGP
Classless	RIPv2	EIGRP	OSPFv2	IS-IS	BGPv4
IPv6	RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6	BGPv4 for IPv6

▷ EIGRP (Enhanced IGRP)

▫ Characteristics (ลักษณะประจำตัว)

o Basic features △ Cisco-proprietary (ใช้งานเฉพาะ protocol ของ cisco) เปิดให้ใช้ใน 1992

⊕ สร้าง classless version of IGRP ⊕ ทางเลือกที่เหมาะสมสำหรับ nw หลาย protocol, ขนาดใหญ่ ที่ส่วนใหญ่ใช้บน cisco router เป็นหลัก

✓ DUAL (Diffusing Update Algorithm) = การันตี loop-free & backup path ทั้งหมดของ routing domain → หา best path

→ ทำให้ routing ทำงาน very fast convergent (มี convergent time < ของ OSPF) รวมทั้งมี backup path (ทำให้ทำงานเร็ว ↑) → if link down จะเลือก path ใน backup ขึ้นมาเลย

(ก.ทำงาน ≈ OSPF เพื่อน router)

✓ Establishing Neighbor = เพื่อหา ก.สัมพันธ์ กับ directly connected EIGRP routers

Adjacencies = Adjacencies are used to track the status of these neighbors

อาศัย ✓ Reliable Transport Protocol (ของ cisco) = RTP provides delivery of EIGRP packets to neighbors

= RTP and neighbor adjacencies are used by DUAL (ใน ก. maintain)

update ทั้ง routing ตลอดเวลา: ไม่ว่า down หรือไม่

EIGRP can update

✓ Partial and Bounded = update เฉพาะส่วนที่มี ก.ปป. รวมทั้ง update ไปให้ เฉพาะส่วนที่มี ก.ปป. ∴ update < RIP

✓ Equal and Unequal Cost (OSPF, RIPv2) = ช่วยให้ admin ดูแลระบบ เพื่อกระจาย ก.รับ ข้อมูลในเครือข่ายได้ดีขึ้น

Load Balancing ↳ ค่า cost ≠ but ทำเป็น load balance ได้

△ ใช้ protocol-dependent modules (PDMs) เพื่อรองรับ protocol ที่แตกต่างกันเช่น IPv4, IPv6, legacy protocol IPX and AppleTalk

△ PDMs รับผิดชอบ:

⊖ maintain EIGRP neighbor and topology table

⊖ คำนวณ metric ที่ใช้ DUAL ⊖ เชื่อมต่อ DUAL และ routing table

⊖ implement filtering and access lists ⊖ 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 ต่างๆ ที่ส่งของ EIGRP

△ ไม่ใช่ทั้งหมดของ RTP packet จะส่งได้น่าเชื่อถือ (msg ≈ OSPF)

⊖ Reliable packet require explicit (ชัดเจน) ack จาก dest. ⊖ Update, Query, Reply

⊖ Unreliable packet do not require ack จาก dest. ⊖ Hello, ACK

△ รองรับ authentication (no encrypt routing update) และ is recommend (เป็นที่ยอมรับ) (authen ≈ RIPv2, OSPF)

o Packet Type (protocol 88 ใน transport layer) routing update or queries EIGRP multicast IPv4: 224.0.0.10 (01-00-5E-00-00-0A), IPv6: FF02::A อื่นๆ IGRP multicast 224.0.0.9, RIPv1 broadcast 255.255.255.255

① Hello → ที่ใช้ adjacencies ระหว่าง router 2 ตัวที่เป็น neighbor กัน, ไม่ต้องตอบ response, ส่ง unreliably

② Update → update info. ของ dest, update info ของ routing ไปให้ neighbor router

③ Acknowledgement → พอได้รับ ก.update ก็ตอบ ACK

④ Query → request info. routing จาก neighbor router } เวลาต้องการ ก. info ของ routing เพิ่มก็ query ไปกับ router ที่อยู่ข้างๆ แล้ว reply ตอบกลับว่ามีหรือไม่มี

⑤ Reply → หลังจากได้รับ query ก็ reply

□ Implement EIGRP for IPv4

△ Autonomous System (AS) is a collection of nw ภายใต้ ก.ดูแลของ single authority (อ้างอิง RFC 1930)

↳ AS number → ใช้ exchange routes between AS

→ managed by IANA & assigned by RIRs to ISPs, Internet Backbone providers, and institution ที่ต่อสาธารณะอื่นๆ (สาธารณะ)

→ 16 bit : 0-65535 ➡ since 2007, 32 bit : over 4 billion

△ verify : show ip eigrp neighbors, show ip protocols, show ip route

△ Configure : R(config)# router eigrp AS-#

show ip protocols ← R(config-router)# eigrp router-id → ถ้าไม่กำหนดจะใช้ loopback intf. → IPv4 addr ที่สูงสุดที่ active

R(config-router)# network nw-number [wildcard-mask] → มี or ไม่มีก็ได้

(มี or ไม่มี) R(config-router)# passive-interface type number [default] ; ไม่ให้ update ไปที่ intf. ไหน (เช่น LAN, G, F) → ที่ไม่ใช่ serial

□ Operation

o Initial Route Discovery (ค้นพบ) ① R1 say hello กับ neighbor router ② R2 อาจจะส่ง hello or update กลับมา ③ R1 ตอบ ack & update info.
④ ใช้ DUAL คำนวณ best route and update routing table

k1 = 1 k3 = 1 K4 = 0, K5 = 0 K2 = 0

- Metrics : BW [lowest], Delay [รวม], Reliability [worst], Load [worst] ดูค่า value: show interface

Default Composite Formula: metric = [K1 * bw + K3 * delay] * 256 = $\left[\left(\frac{10,000,000}{bw}\right) + \left(\frac{\text{sum of delay}}{10}\right)\right] * 256$

Complete: $= \left[k1 * bw + \frac{(K2 * bw)}{(256 - load)} + k3 * delay\right] * \left[\frac{K5}{reliability + K4}\right]$

- R(config-router)# metric weights tos k1 k2 k3 k4 k5 - set bw: เข้า intf. → R(config-if)# bandwidth kilobits-bw-value
- DUAL and the Topology Table (ใช้ FSM (Finite State Machine) ใน ก.เลือกเส้นทาง) → show ip eigrp topology [all-link], show ip route
  + Successor(S) [router ที่ไป dest. สั้นสุด] = neighbor router ที่จะเลือกมาไป dest ที่มีค่า min น้อยสุด
  + Feasible Successor (FS) [ไม่ได้ใน ∝ Feasible condition] = Backup path (เส้นทางสำรอง) if (RD < FD) → FS (จาก neighbor มา)
  + Reported Distance (RD) [distance ที่ neighbor ตัวอื่น report distance มาให้เรา] = "advertised distance" จากตัวมัน → dest มี cost เท่าไร" มองจาก hop
  + Feasible Distance (FD) [distance ที่ไปทาง S] = ค่า distance ที่คำนวณไปหา dest. ทพ เริ่มค่า cost lowest → dest.

▷ IPv6

IPv6	vs	IPv4
128 bit		32 bit
base 16		base 10

□ IPv4 Issue
- ⚠ Need for IPv6 → แก้ปัญหา ip มีจน.จำกัด (private ip, NAT), ใช้ใน IOT ได้
- ⚠ ก.ใช้งานร่วมกัน (coexistence)
  - Migration IPv4 → IPv6 Techniques :: ① Dual stack = run ทั้งคู่ if พังอันใหม่ ไม่กระทบ user
    ② Tunneling (อยากใช้ v6 but core ไม่ support) = ให้แพ็คเกจของเรา แล้ววิ่งผ่าน v4 same เดิม
    ③ Translation (คล้าย NAT) = IPv6 ↔ IPv4

□ IPv6 Addressing : 128 bit แบ่ง 8 ส่วน [1 ส่วนมี 2 byte = 16 bit] → represent base 16 ทุกๆ 4 bit
- ออกแบบวิธี ก.กำหนดค่า IPv6

Rule 1 - Omit Leading 0S = ในแต่ละ partition "0" ที่อยู่ด้านหน้าให้ตัดทิ้งไป 000x, 00xx, 0xxx (x → 0-F)

Rule 2 - Omit All 0 Segment = แทน segment ที่มี "0" ติดกันหลายๆตัวด้วย "::" ใช้ได้ที่เดียว only

□ Type of IPv6 Address
- IPv6 Addr. Type ① Unicast : ⊖ Global Unicast (→ 3 bit แรก: "001" or "2000::/3") ⊖ Link-local (→ ระหว่าง intf. ที่อยู่ติดกัน (@ใน local เดียวกัน)) ⊖ Unique Local
  - Link-local ↳ ในช่วง FE80::/10 = ใช้งานซ้ำได้ ถ้าเป็นระ: link-local (ใช้ได้เฉพาะใน local)
  - static config ← (Global Unicast)
    ↳ ipv6 address ipv6-addr/prefix-length → no shutdown
  ② Multicast
  ③ Anycast = ส่งได้หลาย device
- ⚠ IPv6 Prefix Length = 0-128, most LANs is /64 bec. LAN เล็กสุด 64 bit

□ IPv6 Routing
- config static route

  R(config)# ipv6 route ipv6-prefix/prefix-length {ipv6-addr | exit-intf} (→ ต้องใส่)

* ให้ก่อน config ใช้งาน routing ipv6 ipv6 unicast-routing เพื่อเปิด ก.ใช้งาน
  - ⚠ 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 : ~~[illegible]~~ show ipv6 route static
- config EIGRP for IPv6

R(config)# ipv6 unicast-routing

R(config)# ipv6 router eigrp AS-#

R(config-rtr)# eigrp router ip 2.0.0.0 → รูปแบบ IPv4

R(config-rtr)# no shutdown
  - ⚠ network command : เข้า intf → ipv6 eigrp AS-#
    but. passive-interface ต้องเข้าไปใน global config same เดิม
  - ⚠ verify : show ipv6 eigrp neighbors, show ipv6 protocols, show ipv6 route

ชื่อ-สกุล นาย ธีระ คุณวงศ์ กระดาษแผ่นที่ 6 รหัสนักศึกษา 68070088

for Staples

Summary Config

Router

① basic configuration

เข้า intf (เช่น Giga, Fa) ⇒ R(config-if)# ip address ip-addr. subnet-mask ⇒ R(config-if)# no shutdown

เข้า intf (serial ฝั่ง DCE) ⇒ ~ ⇒ R(config-if)# clock rate 56000 ⇒ ~

verify : show running-config → ดูว่า config อะไรไปบ้าง

show startup-config → ~ set ~

show ip route → ดู routing table

show interface
show ip interface } → show info. ของ intf. ต่างๆ

show ip interface brief show intf. แบบรวบ เช่น ip ?, status ?

② protocol

-static routing

next hop addr. exit-interface

เข้า intf. ⇒ R(config-if)# ip route nw-ip subnet-mask {ip addr. | exit-intf}

⚠ Default route ~ # ip route 0.0.0.0 0.0.0.0 {ip addr. | exit-intf}

- Dynamic routing
  - Interior Gateway P.
    - Distance Vector Routing P.
      - RIP : R(config)# router rip ⇒ R(config-router)# network nw-ip

        verify : show running-config, show ip route, show ip protocols, debug ip rip

        passive intf : R(config-router)# passive-interface intf-type intf-number

        เชื่อม RIP ↔ static : R(config)# router rip ⇒ R(config-router)# {redistribute static | default-information originate}

        โดน router นี้ก่อน set default route @ intf. ที่เชื่อมรวมกับ 2 protocol ต่างๆ !
      - RIPv2 : R(config)# router rip ⇒ R(config-router)# version 2 ⇒ no auto-summary ⇒ network nw-ip

        verify : same RIP เพิ่ม show ip interface brief

for Staples

      - EIGRP : R(config)# router eigrp AS-# ⇒ R(config-router)# eigrp router-id ⇒ network nw-ip [wildcard-mask]

        passive intf : R(config-router)# passive-interface intf-type intf-number

        verify : show ip protocols, show ip eigrp neighbors, show ip route, show ip eigrp topology [all link]

        metrics : R(config-router)# metric weights tos k1 k2 k3 k4 k5

        ** set bw : เข้า intf. ⇒ R(config-if)# bandwidth kbits-bw-value
    - Link State Routing P.
      - OSPF

        : R(config)# router ospf process-id (1-65,535) ⇒ -router)# router-id 1.1.1.1 ⇒ network nw-ip wildcard-mask area area-id

        set bw : เข้า intf ⇒ R(config-if)# bandwidth 64

        set cost : ip ospf cost 15625

        passive intf : R(config-router)# passive-interface intf-type intf-number

        verify : show ip protocols, show ip ospf neighbor, show ip ospf int brief, show ip ospf

        ลบ : clear ip ospf process

        redistribute (OSPF ↔ default route) : R(config)# ip route 0.0.0.0 0.0.0.0 loopback N

        R(config)# router ospf

        ⇒ R(config-router)# default-information originate

        redistribute (OSPF ↔ อื่นๆ) : R(config)# R(config)# router ospf process-id ⇒ R(config-router)# redistribute ?

③ อื่นๆ

- ACL : if ระบุ Name : R(config)# ip access-list [standard | extended] name

  set ACL : R(config)# access-list ACL-num {permit | deny | remark} source [source-wildcard] [log]

  set @ intf : เข้า intf ⇒ R(config-if)# ip access-group {ACL-num | ACL-name} {in | out}

  ลบ : no access-list ACL-num

  verify : show ip interface s0/0/0, show access-lists
- Securing VTP with standard IPv4 ACL : R(config-line)# access-class ACL-num {in [vrf-also] | out}
- Extended IPv4 ACL :

for Staples

R(config)# access-list ACL-num {deny | permit | remark} protocol source [source-wildcard] [operator operand] [port port-num or name] destination [dest-wildcard] [operator operand] [port port-num or name] [established]

- DHCP : R(config)# ip dhcp excluded-address ip-addr-start ip-addr-end
  R(config)# " ip-addr
  R(config)# ip dhcp pool LAN-POOL-1
  R(dhcp-config)# network nw-ip subnet-mask
  R(dhcp-config)# default-router ip-address-gateway
  verify: show running-config | section dhcp show ip dhcp binding show ip dhcp server statistics

Switch

① basic configuration
- management intf: S(config)# interface vlan N ⇨ S(config-if)# ip address ip-addr subnet-mask ⇨ no shutdown
- default gateway : S(config)# ip default-gateway ip
- verify : show running-config show ip interface brief

② configure switch port
- duplex communication : เข้า intf ⇨ S(config-if)# duplex full ⇨ speed 100
- auto-MDIX : เข้า intf ⇨ S(config-if)# duplex auto ⇨ speed auto ⇨ mdix auto
- verify : show int [intf-id], show startup-config, show running-config, show flash, show version, show history, show ip [intf-id], show mac-address-table
- security Remote Access
  + SSH (TCP port 22) : S(config)# ip domain-name cisco.com ⇨ crypto key generate rea ⇨ username admin password ccna ⇨ line vty 0 15 ⇨ S(config-line)# transport input ssh ⇨ login local
    - verify : show ip ssh, show ssh
  + Telnet (TCP port 23)
- switch Port Security : เข้า intf ⇨ S(config-if)# switchport mode access ⇨ switchport port-security
  + static secure MAC addr. : ⇨ switch port-security mac-address MAC-ADDR
  + dynamic " : ⇨ " sticky
  + max MAC addr. : ⇨ " maximum MAX
  + violation mode : ⇨ " violation {protect | restrict | shutdown} mode
  + verify : show port-security int fa0/3, show port-security address

③ STP : เบอร์ 1 S(config)# spanning-tree VLAN 1 root {primary | secondary}
  เบอร์ 2 S(config)# ——— " ——— priority 24576 → ยิ่ง < ยิ่ง priority ดี
  + verify : show spanning-tree [active], show running-config
  □ Rapid PVST+ PVST+
  + Port Fast : เข้า intf ⇨ S(config-if)# spanning-tree portfast
  + BPDU Guard : เข้า intf ⇨ S(config-if)# " bpduguard mode
  + Config : S(config)# spanning-tree mode rapid-pvst ⇨ เข้า intf. ⇨ S(config-if)# spanning-tree link-type point-to-point
  + clear STP ⇨ S# clear spanning-tree detected-protocol

④ VLAN : verify: show vlan name ชื่อ, show vlan summary, show int vlan num, show int fa0/0, switchport show vlan [brief]
1.) set VTP mode : S(config)# vtp version 2 ⇨ vtp mode {server | client | transparent} ⇨ vtp domain name ⇨ vtp password pass
2.) set trunk : เข้า intf. ⇨ S(config-if)# switchport mode trunk 3.) สร้าง VLAN @ Server : S(config)# vlan num ⇨ name name
4.) assign intf : เข้า intf. ⇨ S(config-if)# switchport mode access ⇨ switchport access vlan num
5.) set inter-VLAN : R(config)# int fa0/0.10 ⇨ description vlan 10 ⇨ encapsulation dot1q 10 ⇨ ip address ip subnet

⑤ NAT : verify: show ip nat translation [verbose], show ip nat statistics
- static : R(config)# ip nat inside source static local-ip global-ip ⇨ เข้า intf. ⇨ S(config-if)# ip nat inside outside
- dynamic : R(config)# ip nat pool name start-ip end ip {netmask netmask | prefix-length
  ⇨ access-list ACL-num permit source [source-wildcard] ⇨ ip nat inside source list ACL-num pool name
  ⇨ เข้า intf. ⇨ S(config-if)#

□ PAT = same dynamic เพิ่มว่า "overload"