ชื่อ-สกุล นาย ธนิก จันทร์ ส่ กระดาษแผ่นที่ 1 รหัสนักศึกษา 59080615

for Staples

- Network diagrams : โครงสร้าง nw ทำให้ดูแล มี ก. เชื่อมต่ออยู่ ดูได้
  - 2 type :: ① Physical → ดูอะไร port / interface ไหน เชื่อมต่อกันอย่างไร ② ตาม ไหน (เห็นภาพ) ② Logical → บอก ip
- Network protocol => TCP/UDP, FTP, ARP, SMTP (mail), POP3 (Email), IMAP, ICMP (internet control message protocol) => ping เป็นต้น
  (File Transfer protocol) => รับ-ส่ง file ระหว่าง client กับ server (Addr. Resolution Protocol) => map ระหว่าง IP addr. กับ Mac addr. เป็นต้น ใน command
  - NW Addr. :: ① IP addr (Logical addr) @ L3 ② MAC Addr. (Physical Addr) @ L2 protocol ตาม media ③ Port Number (Service Addr) @ L4
- Component of Network → HW → NW device มี 3 type NW Representation
  ① end devices = ที่ห่างจากผู้ใช้ เช่น PC
  ② intermediary devices = อยู่ระหว่างอุปกรณ์ เช่น NW access devices, Internetworking devices, Security device
  hub, switch, router
  - hub, repeater @ L1 => ส่งพร้อมกันเกิด collision :: ใช้ CSMA/CD (Carrier Sense Multiple) when เกิด collision
  - Switch, bridges @ L2 => Learning / Flooding / Filtering / Forwarding / Aging
  - Router @ L3 => Routing
  ③ network media = สื่อกลาง เช่น copper, fibre Optic, Wireless LAN —— straight - - - cross WAN
  → SW → ① Switch ต่อออกนอก ② router ต่อออกทางที่สุด

Chapter 2 Cisco IOS

1. ROM	POST	Perform POST	
2. ROM	Bootstrap	Load Bootstrap	
3. Flash / 4. TFTP Server	Cisco Internetwork Operation System	Locate and load Operating system image	
5. NVRAM / 6. TFTP Server / 7. console	Configuration	Locate and load configuration file or enter setup mode	

- Types of Networks → size ① small home nw => ใช้ในบ้านต่อกันไม่กี่เครื่อง ② small office / Home office => config ของตัวเองได้
  ③ Medium to Large NW => มาก ต่อกับ ตั้งแต่ 100 - 1000 เครื่อง ④ World Wide NW เช่น internet
  → infrastructure ① Local Area NW (LAN) => มี 1 กลุ่ม admin ดูแล policy / nw ต่างๆ Ex ภายในตึก, ชั้น
  ② Wide Area (WAN) => มีหลายกลุ่ม admin
  อื่นๆ :: Metropolitan Area NW (MAN), Wireless LAN (WLAN), Storage Area NW (SAN), Personal Area NW (PAN)

for Staples

- Layer with TCP/IP & OSI Model

PDU : Protocol data unit		OSI Model	TCP Model
Data	7	Application	Application
	6	Presentation	
	5	Session	
Segment	4	Transport	Transport
Packet	3	NW	Internet
Frame	2	DataLink	NW Access
Bits	1	Physical	

- Type of Connection in a LAN
  ข้อจำกัด (UTP cat 5) : ① BW = 100 Mbps ② ยาว 100 m (can do hub, repeater, switch)
  2 type :: ① ตรง ② cross → ต่ออุปกรณ์ same
- WAN Connection => เชื่อมต่อระหว่าง router
  ↳ 2 ฝั่ง DCE (female) => พิมพ์ command clock rate 56000
  ↳ DTE (male)
- สาย console (Rollover cable) => router — PC
  => manage command ต่างๆ RJ-45-to-DB-9 Adapter labeled Terminal

Chapter 2 Basic Router Configuration
- Port Address @ L4 : กำหนดโดย (Internet Assigned Number Authority : IANA)
  0 - 1023 : requesting entities "well known ports" destination port
  1024 - 49,151 : registered port = publish ให้คนอื่นใช้
  49,152 - 65,535 : 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 : www HTTP, 81 : Hosts 2 Name Server

192.168.1/24 → prefix range
255.255.255.0 → subnet mask
192.168.1.255 → broadcast ip addr
255.255.255.255 → broadcast NW
↳ ทุกเครื่องใน NW ได้รับ

- Logical Address : IP address (IPv4) @ L3
  - Class A [NW] [Host] [Host] [Host] 0 - 127
  - class B [NW] [NW] [Host] [Host] 128 - 191
  - C [NW] [NW] [NW] [Host] 192 - 223
  - D 1110 224 - 239 multicast
  - E 1111 240 - 255 Experimental

- Private addressing → ip can reuse ใช้ซ้ำได้
  → ข้างนอก can ใช้งาน internet ได้ภายใน

	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

for Staples

- Physical Addresses : MAC address
  - Ethernet : 48 bit เขียน 2 : 12 ตัว ฐาน 16 → 0 x ... 16
  - ข้อกำหนด IEEE → กำหนด 3 byte (24 bit) code "Organizationally Unique Identifier (OUI)"
    → 2 ข้อ ① ทุก MAC ที่ ... NIC และ Ethernet device ใช้ 3 byte แรก ② ทุก MAC ที่ same OUI ต้องกำหนด unique ที่ 3 byte สุดท้าย

▷ Message Delivery
① Unicast : ส่งข้อมูลปลายทางใน NW เดียวกัน
② Broadcast : ส่งทุกเครื่อง DHCP, ARP ใน NW เดียวกัน
→ broadcast ip/nw = 255.255.255.255 = FF-FF-FF-FF-FF-FF
▷ Multicast : ส่งทางกลุ่ม & รับเฉพาะกลุ่มที่ขอ service ไว้
ระบุ 01-00-5E-XX-XX-XX

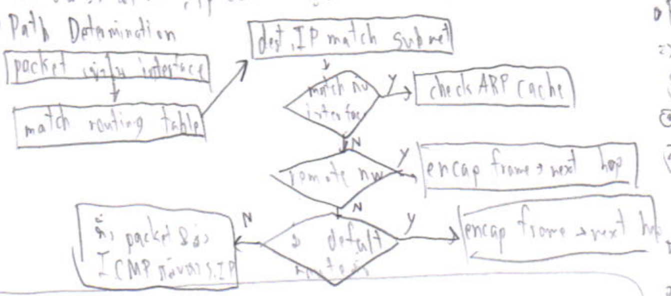
· Cisco IOS (Internetwork Operating System)
- function ① Addressing ② Interface ③ Routing ④ Managing Resource
- Router & Switch Boot sequence
Rom ① Post (Power On Self test) → check hw
② Run boot loader sw → IOS
ROM ③ Boot loader does low-level CPU initialization
Flash / ④ — initializes the flash file system
TFTP server ⑤ — locates & load a default IOS
RAM

· Accessing a Cisco IOS Device
① Console port ② Telnet ③ Secure Shell (SSH) ④ Aux Port
↳ Terminal Emulation Program :: PUTTY, Tera Term, Secure CRT, HyperTerminal, OS X Terminal
· Navigating the IOS → 2 mode :: ① user ">" ② privileged (enable) "#"
↳ 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 error
③ Hot keys and shortcuts
④ IOS Examination Commands → show
· Getting Basic ① ตั้งชื่อ host name ② จำกัดการเข้าถึง ③ กำหนด interface addressing ④ ตรวจสอบ config ⑤ save config
router(config)# hostname name ↳ Banner msg. Router(config)# banner motd # text #
↳ Securing Device Access :: Enable password / secret, console pass, VTY pass, Encrypting pass display
router # copy running-config startup-config

① Addressing Devices
1) interface / Loopback interface
Router(config)# interface type port
" — type slot/port
" — type slot/sublot/port
- 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 → section, include, exclude, begin
Router # show running-config → ดู config
" show startup-config
" show ip route → routing table
" show interface } show info ของ interface
" show ip interface
" show ip interface brief show interface up, set?, status?
" traceroute
" ping
PC > ping
" tracert
" route print
" nslookup

Chapter 3 Static Routing & Dynamic Routing Protocol
· Functions 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 ต้องผ่าน router → process ที่ CPU ออก interface
② Fast switching : forward ได้เร็ว ③ Cisco Express Forwarding (CEF) : Forward packet ได้เร็วที่สุด
· Connect Devices
▷ Default gateway → ① first usable host (.1) ② last usable host (.254)
▷ Enable IP on a Host :: ① Statically Assigned IP addr.
② Dynamically " — => ip DHCP (Dynamic Host Configuration Protocol)
· Switching Packet between NW
dest. ip (L#3) → routing table → MAC addr → dest. MAC (L#2)
· Path Determination



▷ Best Path : lowest metric (cost)
=> Dynamic routing protocol
① Routing Information Protocol (RIP) = hop
② Open Shortest Path First (OSPF) = BW
③ Enhanced Interior Gateway Routing Protocol (EIGRP) = BW, delay, load, reliability
▷ Load balancing
▷ Administrative Distance (AD) = protocol
= (connected = 0, static = 1, Internal EIGRP = 90, OSPF = 110, RIP = 120

The Routing Table

Config : Next-hop Option
R(config)# interface G0/0
" -if)# ip address 172.168.1.0 255.255.255.0 } Basic
" — no shutdown
R1(config)# ip route 172.168.1.0 255.255.255.0 ↳ 172.168.1.1
↳ S 0/0/0
set default static route
Router(config)# ip route 0.0.0.0 0.0.0.0 { ip-addr. | ext-intf }

scalability
"stub nw"
4 type : ① Standard ② Default ③ Summary ④ Floating : back up
Router(config)# ip route nw-addr. subnet-mask { 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)
· Routing
① Static Routing → Manual
security, resource, process, routing entry

· Classful Addressing → update class
· Classless Inter-Domain Routing
▷ Summarization ① ② ip
① ip → ② Group bit Same
· VLSM
▷ Fixed Length Subnet Masking ③ prefix - prefix
② bit

ชื่อ-สกุล ธนารัตน์ จันทร์... กระดาษแผ่นที่ 02 รหัสนักศึกษา 88888888

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 ที่อยู่ไกลๆ) แล้วรับ/ส่ง routing info แล้วเลือก best path ไป dest. nw และ can หา new best path ถ้า path เดิมใช้ไม่ได้
- component: ① Algorithm: ใช้ประมวลผล routing info. & best path
  ② Routing protocol msg: ส่งหา neighbor & แลกเปลี่ยน routing info (best path)

	Dynamic routing	vs	Static routing
ความยากใน config	∝ ขนาด nw (same เดิม)		∝ ขนาด nw (เพิ่ม command ทุก router)
Required ความรู้ admin	Advanced (bec. config basic ใน nw ใหญ่ + ต้องแก้ปัญหา)		No (เบื้องต้น) (กำหนดเส้นทางเองโดย command)
Topology change	ปรับ auto		admin config ใหม่ all
Scaling	เหมาะ simple & complex (route ที่ไม่ได้ directly ไม่ต้องทำให้)		เหมาะ simple topologies
Security	น้อยกว่า		มากกว่า
Resource usage	ใช้ CPU, mem (เก็บ routing info), link bandwidth		No ใช้ resource
Predictability	Route ∝ current topology		Route → dest หนเดียวตลอด

- Classifying Routing Protocol

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

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

- Convergence : ตกลงกันได้ when routing table ของ all router มีสถานะคงที่ (ไม่เปลี่ยนแปลง)
  แบ่ง 2 type : Slower : RIP & IGRP , Faster ถ้า update when เกิดการเปลี่ยนแปลงขึ้น >> : EIGP & OSPF
- Routing Protocol Metrics
  - Metric : ค่าที่ใช้ในการคิด/หา ในการไป dest. NW ว่าจะเลือก best path ได้ เช่น Hop count, BW, Cost, Delay, Load, Reliability (น่าเชื่อถือ)
  - load balancing: nw มีเส้นทาง ≥ 2 ที่มี metric เท่ากัน จะเลือกใช้ได้ ใช้ทางใดทางหนึ่ง จะสลับกันไปเท่าๆกัน
- Administrative Distance of a Router (AD) => ใช้เลือก protocol ในการ routing
  - วัตถุประสงค์ : ค่าที่บอกความน่าเชื่อถือของ particular (เฉพาะ) route

Route Source	Connected	Static	Internal EIGRP	OSPF	RIP	EIGRP Summary route	External 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 สิ่งที่นำมาใช้คือ ① Vector or direction, ทิศทางที่จะส่งตรงไป ② Distance to final dest. (cost)
  อธิบายการทำงาน: periodic (ตามรอบเวลา) update, neighbor (router ใกล้เคียง), broadcast (255.255.255.255) update, เอา routing table all ไป update
- " ของ Routing Protocol: เกณฑ์ที่ใช้ check ว่า DV ดี/ไม่ดี ① Time to convergence => เวลาที่เป็น steady state ของ routing table หลังเกิดการเปลี่ยนแปลง-ระงับ ② Scalability ขนาดของเครือข่ายที่ใช้ได้ ③ Resource usage ④ Implementation & maintenance
- NW Discovery (ค้นหา nw) (ทำ basic config ก่อน)
  มี 3 stage ① Cold stage : Router Initial Start up
  ② Initial Exchange of Routing info. → ทำตารางแลกเปลี่ยน
  ③ Exchange of Routing info → update (เพิ่ม hop counts) routing info
  → ของ router อื่นต่อไปเรื่อยๆ

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

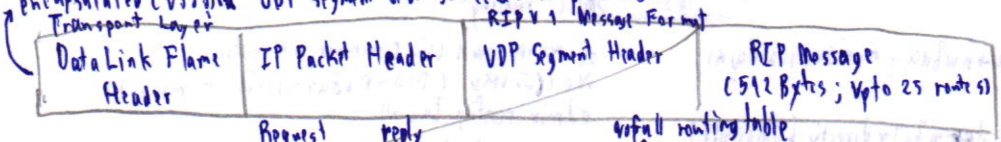
- Routing Table Maintenance
  - Periodic update: RIP update timer (default 30s), Invalid timer (info นั้น lost) (default 180), Holddown timer (ถ้า down → hold ไว้ไม่ให้ up ตอนไหน) (default 180), Flash เก็บ timer (default 240)
  - Bounded (จำกัดขอบเขต) Update : EIGRP → update เฉพาะที่เปลี่ยน
  - Triggered Update → update เลยไม่รอ periodic time
  - Random Jitter → ใช้ใน nw ที่มี multiple access router หลายตัว เชื่อมต่อกัน → if ส่ง update พร้อมกัน : จึงใช้ Random

for Staples

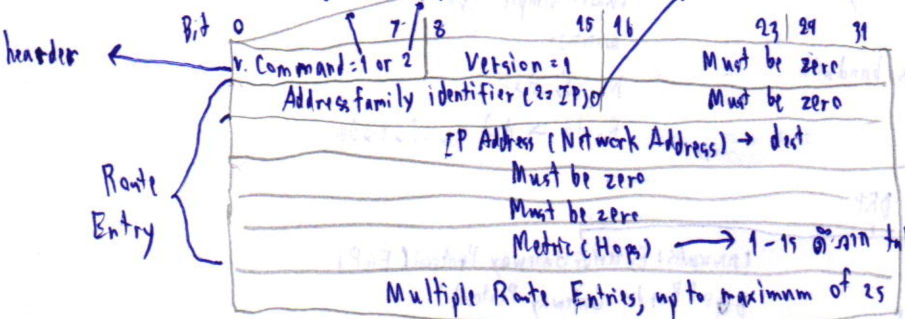
○ ปัญหา standard DV. ① Routing Loops เกิด when intf ใดๆ down จะถูกดึงออกมาจาก table → ค่าเก่าใน neighbor จะกลับมา update (วน update → hop ↓ ∞)
↳ แก้ปัญหา: ① set max hop = 15 → if hop = 16 → unreachable (มัน down ไปแล้ว)
② hold down timer (ถ้า intf down → hold)
③ Split Horizon Rule → ไม่ส่งข้อมูล update กลับไปทาง intf ที่ได้รับ update มา
④ Route Poisoning → ① เกิด down set unreachable ② ส่ง unreachable ที่เป็น poision ออกไปให้ intf นั้น hop = 16
⑤ ③ with ④ → ถ้าได้ unreachable จะ over rule split horizon โดยส่ง ip intf ที่ down (hop = 16)
⑥ IP TTL (Time to Live) เกิด ก. วน update but. จะหยุด when 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 (บรรจุใน) UDP segment ด้วย source & dest port = 520



○ msg มี 2 type
① Request → ส่ง routing table → ไป intf ที่ config ไว้เพื่อขอ upd.
② Responce → ส่ง info ของ routing table update
○ ip addr. แบ่งตาม class A, B, C
○ Basic RIP v1 Config
① ทำ basic config
① สั่ง router rip R1(config)# router rip
② ทำเลือก nw R1(config-router)# network nw ที่อยู่ติด



○ Verification (ตรวจสอบ) & troubleshooting (แก้ปัญหา) : show running-config or ip route or ip protocols, debug ip rip
. passive intf command (ไม่ update intf ที่ไม่ต้องการ) R(config-router)# passive-interface intf-type (Fa 1G 1S) interface-number (0/0, 0/0/0)
○ Automatic Summarization : RIP Auto summarizes classful nw → เพื่อลด size routing table
↳ ข้อดี : ลด size routing update . single router ระบุของมาจำนวน. มา multiple route เรียงตาม routing table
↳ ข้อเสีย : ไม่ support discontiguous nw (major nw เดียวกัน but. ไม่อยู่ติดกัน) → อาจจะเกิด load balancing ได้
. boundary Routers : summarize RIP subnet from 1 major nw to another
. Processing RIP update
ดูที่ส่ง & ดูว่า update ไป intf ว่าอยู่ classful เดียวกันไหม? → y : update subnet nw เช่น 172.16.1.0
↳ n : update classful เช่น 172.16.0.0
. default route & RIP v1 ถ้าไม่เจอเลยออกไปใน routing table (ลด bec. ลดของprotocol) → จะออก default route
R(config)# ip route 0.0.0.0 0.0.0.0 s0/0/1
default info originate command → ส่ง update ไปใน rip ว่าเป็น : static → dynamic
Router ที่อยู่ระหว่าง 2 protocol ← R(config-router)# default-information originate

Chapter 5 RIP version 2 & Access Control Lists

RIP v1	vs	RIP v2
classful (ไม่ส่ง subnet mask, ไม่ support CIDR)		classless (update subnet mask, support Variable Length Subnet Masking (VLSM), support Route Summarization (Prefix Aggregation))
not support discontigous subnet		update next hop addr.
not support VLSM bec. ไม่ส่ง subnet mask (255.255.255.255)		ใช้ authentication routing (ตรวจสอบ discontiguous ของ nw)
routing update => broad cast		Routing update => multicast

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

○ ข้อจำกัดของ RIP v1
สร้าง virtual interface
Can ให้ router routing ได้ update ออกไปได้
{ . loopback intf → ping ไป → ip virtual intf → reply ได้
. Null intf → ใช้ทำหน้าที่ = เข้าไป channel ที่ไม่ได้ออก. → ถ้าออก null intf → packet discount → timeout
. static route & null intf → null intf จะเป็นหน้าที่ในการปลายของ static route
R(config)# ip route summary-static-route subnet-mask Null 0
(major-nw) → static supernet route

. Route redistribution (จัดสรร) → อยากให้ rip รู้จัก static ต้องทำให้ในโซน rip นั้น static ได้ด้วย : R(config-router)# redistribute static
. Verify & test Connectivity : show ip interface brief , ping (ผล : ! = ได้ , . = ไม่ได้ , . = timeout), traceroute
. RIP v1 : classful, ไม่ส่ง subnet mask, summarize nw @ major nw boundaries, if nw เป็น discontiguous & RIP v1 config convergence จะไม่ได้

ชื่อ-สกุล นายจิณัฐ จันท่ส กระดาษแผ่นที่ 03 รหัสนักศึกษา 88880688

· ตรวจ routing table: debug ip rip (content of routing update, ถ้าเป็น RIP v_1 จะไม่ได้ subnet mask รวมใน nw address)

▷ RIPv2

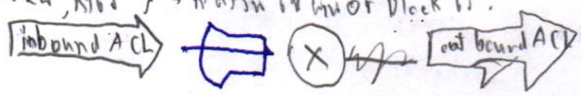
○ config
- · enabling & verify (ตรวจสอบ) RIP v_2 → show ip protocols
- · Config RIP → RIP v_1 → ควรรับได้ทั้ง v_1 & v_2 but ส่งได้แค่ v_1
  - → RIP v_2 → ควรรับ & ส่งได้แค่ v_2
- · Auto-Summary & RIP v_2 → auto sum route @ major nw boundaries
  - → sum route ด้วย subnet mask ที่น้อยกว่า classful subnet mask
- · disabling Auto-Summary : no auto-summary bcr. when วัน nw topology ที่เป็น discontiguous

○ VLSM & CIDR → verify info ที่ sent by RIP v_2 debug ip rip
- → VLSM → แชร์แพร่ nw addr. & subnet mask
- → CIDR → ใช้ supernetting (=บางส่วนของ 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?
- → ดูจาก conversation → ทำงานอะไร (เช่น FTP) ผ่านไหม?

○ Packet filtering ดู ① dest, source ② L2 ③ protocol ที่ใช้ ③ ไป nw ไหน, ที่ใด } → ตัดสินใจ ให้ผ่าน or block ไป?

○ 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 specific (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

○ wild card → invert ของ subnet mask
- → 0 : match / fix , 1 : ignore / จะไม่สนใจ
- → วิธีหา จาก set ของ ip (match range) ① หา ค. สัมพันธ์ของ bit ที่ซ้ำกัน ใส่ wild card mask ตำแหน่งนั้น : 0 ② bit ที่เหลือใส่ 1
  - if หาไม่ได้ ลอง แรก Pattern or / and ด้วย คือ สุดท้าย wildcard จะ same กัน
- → หา wild card ของ subnet = 255.255.255.255 - subnet mask
- → keyword → 0.0.0.0 → match all ใช้ host
  - → 255.255.255.255 = ignore all ใช้ any

```
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
R1(config)#   "    0.0.0.0  255.255.255.255
              "    any
```

○ Guideline for ACL creation (3Ps)
- → One ACL / protocol : ctrl traffic flow บน intf, ACL ต้อง define แต่ละ protocol enable on intf
- → 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 for an intf, Ex. G0/0

· where → Extend ACL : @ close source → standard ACL : @ close destination

○ Config ACLs
- → standard → number

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

ถ้า remove all : no access-list
ถ้าแก้ ① no access list num# → ลบทั้งหมด
② no ขบรทัด # → ขบรทัด # ที่ต้อง
access-list num# permit any

- → in intf

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

ถ้า remove all : no ip access-group
ถ้าแก้ ขบรทัด # ที่ไม่ได้
↳ ขบรทัดแรก

- → ชื่อ

```
Router(config)# ip access-list [standard | extended] name
```

หลังอันแรก 1 แบบ ที่ยัง ใช้ เหมือนกัน but ต่าง ตรง ขึ้นหัวส่ง (permit, deny, remark) ได้เลย

● Verify : show ip interface, show access-lists

○ Securing VTY port → อยากให้คนนอก เข้ามา permit เฉพาะคนที่ต้องการ ใช้ได้

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 [source-wildcard] [operator operand]
[port port-number or name] destination [destination wildcard]
[operator operand] [port port-number or name] [established]
```

ip, tcp, udp
ใส่ก็ได้ ไม่ใส่ก็ได้ (port src, port dest)

ที่เหลือ same standard
ใช้ได้ทั้ง แบบ number & name

- debug-output : debug ip packet ACL-number

Chapter 6 OSPF & DHCP

▷ **Link-state Routing Protocol** = เป็น protocol ที่ต้องรู้ complete map ของ nw topology (รู้ info all) ทั้งหมด → หา shortest path first (SPF) (ใช้ dijkstra)

แนะนำ : ① large nw, ② fast convergence ③ admin ควบคุม ต.รู้
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 → ได้รับและเก็บลง db ⑤ router เอา all LSP เก็บใน db (สู่ SPF tree) + Adding OSPF routing table

ข้อดี : ① สร้าง topology map can หา shortest part, ② fast convergence ข้อมูลส่งต่อเร็ว ③ LSP sent only when change topology → คำนวณ shortest path ④ hierarchical design (N ไม่ต้องทำ) → ลด resource bec. แยกข้อมูลแต่ละ area (ใช้ต่อเมื่อ ข้อมูลเปลี่ยน)
ข้อเสีย : ① ใช้ mem ใน ก.เก็บ all link-state เยอะ ② ใช้ CPU ใน ก.คำนวณ ③ ตอนส่ง LSP ต้อง ใช้ BW เยอะมากๆ

▷ **OSPF AD<110**

4 3 table : ① Neighbor show ip ospf neigbor ② Topology (เก็บ map) show ip ospf database ③ Routing (เก็บ shortest path)

message → Encapsulating : MAC Dest : Multicast : 01-00-5E-00-00-05 or 01-00-5E-00-00-06
Protocol fiel = 89

→ type OSPF Packet :
- 01 Hello → ทุก 10s (defult : multiaccess & point to point nw), ทุก 30s (defult : non-broadcast multiaccess (NBMA) nw), cisco default 4 times (40 s)
- 02 Db Description (DBD) → synchronization db info
- 03 Link-state Request (LSR) → request link-state
- 04 ~ Update (LSU) → send update link-state
- 05 ~ Acknowleadgement (LSAck) → ตอบกลับได้แล้ว

Operation : สถานะการเชื่อมต่อ ① Down state (เริ่มต้น) → ② Init state (เริ่มส่ง hello) → ③ Two-way state (ตอบกลับ hello) → ExStart state → Exchange state → loading state → Full state (ระหว่าง router update ข้อมูลเรียบร้อย)

config Single-Area OSPF ① router ospf process-id → 1-65535 เป็น locally significant
R(config-router)# router-id 1.1.1.1 → ถ้าไม่ set can ใช้ ② loopback, ③ active interface ip สูงสุด 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

→ ต้องการเปลี่ยนค่า cost
→ เปลี่ยน ref BW

```
auto-cost reference-bandwidth mbps
 - Fast Ethernet
   auto-cost reference-brandwidth 100
 - Gigabit Etheret
                  //            1000
 - 10 Gigabit Ethernet
                                10000
```

→ เปลี่ยนค่า BW R(config-if)# brandwidth 64 (EIGRP & OSPF can ใช้ได้)
→ 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 an OSPF Default Route
R(config)# ip route 0.0.0.0 0.0.0.0 loopback 1
R(config)# router ospf process-id
R(config-router)# defalt-information originate
```

▷ **DHCP (Dynamic Host Configuration Protocol)** → ทำหน้าที่ config ใน host ให้ auto เช่น ip, subnet mask, defalt gateway, dns
method
① manual Allocation : admin assign ให้
② Automatic Allocation : DHCP จะ auto assign addr. จาก pool ที่ไม่มี lease (เช่า) time
③ Dynamic Allocation : ใครต้องการก็จะได้ ip ไป มี lease time → ถ้าหมด lease time ต้องคืน ip ให้

Operation
- server ← DHCP Discover (ขอเลข DHCP?) ← client
- DHCPOffer → (มี)
- ← DHCP Request (request)
- DHCPack → (ตอบ)

config

```
R1 ~~~ # 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
dhcp-config)) # network 192.168.10.0 255.255.255.0
       # default-router 192.168.10.1
       # dns-server 192.168.11.5
       # domain-name example.com
```

Verify
show running-config
show ip dhcp binding
show ip dhcp server statistics

debug same Extend

Config DHCP client # ip address dhcp
no sh

ชื่อ-สกุล นายธนิก จิรพัร์ กระดาษแผ่นที่ 4 รหัสนักศึกษา 59810615

Chapter 7 Basic Switch Address Resolution Protocol

▸ LAN Design → Borderless sw nw design : มีกฎที่ใช้ :: - Hierarchical, - Modularity, - Resiliency, - Flexibility

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

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

▫ เพิ่ม ค. คล่องในการใช้ LAN BW & ปอท. ได้ max

. f" & ก. วาง Server ① Enterprise S (ของทั้งองค์กร) ➜ ต่อกับ @ MDF (Main Distribution Facility : Core) => ตอนผ่านไปข้างนอก ทำให้ใกล้กับมัน

② Workshop S. (เฉพาะกลุ่ม) ➜ ต่อกับ @ IDF (Intermediate D.F. : Distribution) => เชื่อมระหว่าง Core กับ access ทำให้ใกล้กับมัน

VCC (Vertical cross-connect) : Optical Fibre => MDF ↔ IDF
HCC (Horizontal ~) : VTP : @ Distribution ↔ Access

· Collision detection issue (ปัญหาการชนกันของ ก. ส่ง)

· Segmentation issue (ปัญ หาก. แบ่งส่วน) => ทำให้ ปัญหา ก. ใช้งาน nw มี ปอท. ที่ดีขึ้น

· Broadcast domain issue => ใน nw เกิด ตลอด เช่น broadcast หา MAC Addr. ∴ broadcast NW ควรมีขนาดเล็ก

▫ Segmentation เป็น process split single collision domain ลด จน. collision บน LAN segment :: L #2 device ใช้ เช่น bridges, SW

▫ Broadcast domain ส่งหมดทุก port but. router (L #3) เป็นตัว filter / segment broadcast ไม่ให้ กระจายออกจากตัวมัน

▸ SW Environment

▫ SW Operation ① Learning :: เมื่อ frame เข้า SW จะเรียน Source MAC Addr. ว่าต่อกับ port ไหน + reset Aging

② Aging :: ดูการจอง MAC Addr. → if หมด → ทิ้ง

กรณีก. 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 Store & Forward SW => check CRC ว่า error ไหม → มี : ทิ้ง → ไม่มี : ส่ง, auto buffer

② Cut-Through SW => check เฉพาะ บางส่วน (dest, source อย่างน้อย 12 byte แรก) [10 ms], No FCS & auto buffer

↳ 2 mode :: [1] fast-forward ~ 12 byte [2] Fragment-free ~ 64 byte :: < 64 : ขน → หยุด → ไม่ส่ง

▫ SW Domains :: ① Collision Domains => domain ที่เกิด ก. ชน ทุกที่ว่าเชื่อมถึงกันทางพร้อมกัน " @ ⓐ SW เป็นตัวแบ่ง"

 ② Broadcast ~ => domain ที่ส่ง broadcast → ทุก domain เชื่อมกันจะได้รับ " @ ⓐ router เป็นตัวแบ่ง"

▸ Basic SW Concept & Configuration

▫ Basic SW Config

▫ SW Boot Sequence : Same Router

Manage intf : S(config)# interface vlan num

same Router { S(config-if)# ip address ip subnet / S(config-if)# no shutdown }

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

→ Verify Port Config
show int fa0/0, /startup-config, /running-config, /flash, /version, /history, /ip f0/0, / mac-address-table

▫ Preparing of Basic SW Management : SW ไม่มี Loopback ∴ กำหนด เป็น SVI (SW Virtual interface) → VLAN

▫ Config SW Port → Duplex Communication : ① Full ② Half (SW ต้องปรับค่าต้องเลือก same กัน)

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

→ Auto-MDIX : ปกติ SW จะเชื่อมกันด้วย cross-over But ต้องปรับให้เป็น สายตรง)

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

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

Config : S:C(config)# 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 → เปิดใช้งาน 1 ประเภท

Secure MAC Addr. :: → ① static :: S(config-if)# Switchport port-security mac-address MAC-ADD

② dynamic :: S(config-if)# Switchport port-security mac-address sticky → learn from ตัวแรกก่อน → record ไว้

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

violation mode : ① protect :: Security violation protect mode

② restrict :: Security violation restrict mode => ส่งไปเก็บว่าเกิดอะไรขึ้นต่อที่ไหน ๆ

③ shutdown :: Security violation shutdown mode => default

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

Violation Mode	Forward traffic	Send Syslog Message	Display Error Message	Increase violation counter	Shut 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 ไม่อยู่ในวงเดียวกัน MAC gateway)

IPv4 : Classless (ต่อ P.1-2) : - Variable Length Subnet Masking (VLSM) : แบ่งทุก nw ไม่เท่ากัน :: => เริ่มจากวงที่ใหญ่ → 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 bec. ข้อมูลเปลี่ยนตลอด

② Broadcast storms => วนส่งไม่หยุด ③ Multiple frame transmission => start : unknown unicast → ทำให้ dest ได้รับ หลาย frame but. source ส่ง 1 frame

▸ STP => จุดหมายหา block port → block มัน → ไม่ให้เกิด traffic วน (เพื่อแก้ปัญหา)

ขั้นตอน : ① หา Root Bridge : priority min [Rule : [1] 1 RB / 1 nw [2] 1 RP / 1 RB [3] 1 DP / Segment]

ดูที่ BPDU (Bridge protocol data unit) ② หา path cost all ③ หา Root Port → path cost min → อันที่เหลือจะเป็น Designated Port

③ ส่วน segment มี path cost เท่ากัน → ดู BID min เป็น designated port → อันที่เหลือเป็น block port

โดยมี RB & Source [802.1D]

ลำดับการเลือก (ถ้าไม่เท่ากัน) ① BPDU flag (#priority น้อยสุด)
② BID น้อยสุด
③ Path Cost <
④ Sender's BID <
⑤ Sender's Port

▫ 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 ← (ต้องเป็นเลข 4096 ทวีคูณ)

[Verify : show spanning-tree]

▫ แบบ Extended System ID : B. Priority (2 byte) → B. Priority (4 bit) (ต่อ VLAN) + Extended Sys ID (VLAN) (12 bit) + MAC Addr. (6 bytes) ∴ BID = 8 byte

. PVST+ (ต่อยอดจาก IEEE 802.1D STP) → ช่วยทำ load balancing โดยแยก root / vlan
[verify: show spanning-tree active]

∘ Rapid PVST+ → มี Alternate port (ไม่ได้ block ไปเลย can รับส่งข้อมูลได้บางส่วน)
↳ ทำ n. set Edge Port @ port ต่อ host, router : -if)# spanning-tree port fast
↳ link type : port ที่เชื่อมระหว่าง SW ต้องเป็น point-to-point
-if)# spanning-tree bpduguard enable → port นี้ไม่ต้องการ BPDU ใดๆ ↱ ถ้า clear all : clear spanning-tree detected-protocol
→ config : S1(config)# spanning-tree mode rapid-pvst → # int ที่ p-to-p → # spanning-tree link-type point-to-point

---

Chapter 9 VLANs & Inter VLAN

▷ VLAN = ทำ partition (แบ่งแยก : อยู่คนละ nw or broadcast กัน) Layer 2 เช่น SW และทำ n. ต่อกันได้เฉพาะใน VLAN เดียวกัน

ข้อดี : - security ดีขึ้น, - ลด cost, - ประสิทธิภาพดีขึ้น ↑, - broadcast domain เล็กลง ↓, - ช่วยงาน IT, จัดการ ง่าย. [verify : show vlan brief]

▫ in a Multi-SW Environment

· VLAN Trunk ; set ที่ intf ที่เชื่อมต่อระหว่าง SW ทั้ง VLAN → can carry ข้อมูลได้ > 1 VLAN

Native (base) VLAN
ไม่ต้อง n. tag & untag → ระบุใน intf ได้เลย (เลือก)

③ ↳ config : เข้า intf → -if)# switchport mode trunk [verify : show int fa0/0 switchport]

. Tagging Ethernet Frame (IEEE 802.1q) : Ethernet Frame →

DestMAC	Src MAC	Tag	Type/Length	Data	FCS

⇒ Tag ที่ใช้ระบุ VLAN when วิ่งผ่าน Trunk

▫ Assignment : VLAN number ① 1-1005 เก็บ config ① vlan.dat (ใน flash) ③ สร้าง : I S(config)# vlan vlan_num → vlan)# name ชื่อ
② 1006-4096 เก็บ config ② running-config (ใน NVRAM)   II S# vlan database → (vlan)# vlan num name ชื่อ

②- assign port ให้ vlan : เข้า int f → -if)# switchport mode access → # switchport access vlan num (if ลบ : เติม no หน้าคำสั่งได้เลย)
(if ลบ : 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 "
[verify : show vlan, show ip route, show running-config]

▫ 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] → n. manage SW VTP จะเป็น n. manage ใน domain

▫ Operation : n. update VTP จะใช้เลข revision number 32 bits (0-4294967295) (เลขที่สูงกว่า)

L 3 mode ① Server → can สร้าง, remove, rename VLAN ทั้งหมดใน domain ระบุ ②
② Client → รับข้อมูล VTP มา process, ส่ง VTP msg ออกทาง trunk
③ Transparent → can สร้าง, remove, rename แค่ในของตัวเอง, เป็นแค่ทางผ่าน

* Locally Significant only

Feature	Server	Client	Transparent
Source VTP Messages	Yes	Yes	No
Listen to VTP Message	Yes	Yes	No
Create VLANs	Yes	No	Yes*
Remember VLANs	Yes	No	Yes*

▫ Config : 2 แบบ สิ่งสำคัญ : 1) SW cisco 2) มี trunk อย่างน้อยระหว่าง 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 ที่ไม่จำเป็นใน interface น้อยลง ↓ โดย config ที่ interface ต่างๆ เพื่อ remove ข้อมูลที่ไม่ใช้
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. ↳ same เลย

Private Internet address 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↔1) → ① R(config)# ip nat inside source static local-ip global-ip

② Dynamic : มี pool ของ Global / Real ip (map : many ↔ 1) ; real ip จะเปลี่ยนไปเรื่อยๆ } 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

▫ 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 fa0/0 overload (PAT เพิ่ม จาก dynamic)

▫ verify : show ip nat translations

Chapter 8

Protocol	Standard	Resources needed	Convergence	Tree Calculation
STP	802.1D	Low	Slow	All VLANs
PVST+	Cisco	High	Slow	Per VLANs
RSTP	802.1w	Medium	Fast	All VLANs
Rapid PVST+	Cisco	Very high	Fast	Per VLAN
MSTP	802.1s Cisco	Medium or high	Fast	Per Instance

ชื่อ-สกุล นายธนัท จันทร์ กระดาษแผ่นที่ 1 รหัสนักศึกษา 58010635

for Staples

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 (ลักษณะเฉพาะ)
    - Basic features : Cisco-proprietary (ใช้งานเฉพาะ) protocol ของ Cisco เปิดใช้ใน 1992
      - ① รองรับ classless version of IGRP ② ทางเลือกที่เหมาะสมสำหรับ NW หลาย protocol, ขนาดใหญ่ ที่ส่วนใหญ่บน Cisco router
- DUAL (Diffusing Update Algorithm) = การันตี loop-free & back up path ทั้งหมดของ routing domain → หา best path
  - → ทำให้ routing ทำงาน very fast convergent (มี convergent time < ของ OSPF) รวมทั้งมี backup path เก็บไว้ ทำงานเร็ว↑, → if link down จะเลือก path ใน back up มาแทน
  - (ก. ทำงาน ≈ OSPF ของ router)
- Establishing Neighbor Adjacencies = เชื่อมต่อสัมพันธ์กับ directly connected EIGRP routers.
  - = Adjacencies are used to track the status of these neighbors
- Reliable Transport Protocol
  - = RTP provides delivery of EIGRP packets to neighbors
  - = RTP and neighbor adjacencies are used by DUAL (ใช้ ก. maintain)
- EIGRP can update
- Partial and Bounded = update เฉพาะส่วนที่มี ก.ปป. รวมถึง update ไปให้เฉพาะส่วนที่มี ก.ป.ป. ∴ update < RIP (update ทั้ง routing table ทุก 30 วิ)
- Equal and Unequal Cost : ช่วยให้ admin ดูแลระบบเพื่อกระจาย ท. ส่งข้อมูลใน แต่ละ link ได้ดีขึ้น
  - Load Balancing → ค่า cost ≠ กัน, ก็ยังเป็น load balance ได้
  - ใช้ protocol-dependent modules (PDMs) เพื่อรองรับ protocol ที่แตกต่างกัน เช่น IPv4, IPv6, legacy protocol (IPX and AppleTalk)
  - PDMs รับผิดชอบ :
    - ⊖ maintain EIGRP neighbor and topology table (Neighbor Table → สร้าง Topology Table → ได้ routing table เป็น routing) (เก็บ topology ≈ OSPF → เก็บ shortest path กับ backup shortest path ด้วย DUAL)
    - ⊖ 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 ของ ใน application layer ใช้ maintain ข้อมูล, msg ต่างๆ ที่ส่งของ EIGRP
  - ไม่ใช้ทั้งหมดของ RTP packets จะส่งได้แน่นอนทั้ง (msg ≈ OSPF)
    - ⊖ Reliable packet require explicit ตอบกลับ 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 (เปิดรองรับ) (authen ≈ RIPv2, OSPF) (protocol ที่อยู่บน transport layer)
- Packet Type routing update & queries EIGRP multicast IPv4 : 224.0.0.10, IPv6 : FF02::A (01-00-5E-00-00-0A) ต่าง IGRP (multicast 224.0.0.9) RIPv1 broadcast 255.255.255.255
  - ① Hello → ทำให้ adjacencies ระหว่าง router 2 ตัว ที่เป็น neighbor กัน, ไม่ต้องรอง response, ส่ง unreliably
  - ② Update → update info. ของ dest, update info ของ routing ไปให้ neighbor router
  - ③ Acknowledgement → ใช้ยืนยัน ก. update ต้องการ Ack
  - ④ Query → request info. routing จาก neighbor router } ใช้ตอนต้องการ info ของ routing เมื่อ query ไปทุก router ที่อยู่ รอบๆ แล้ว reply ตอบกลับมาแล้วไม่มี
  - ⑤ Reply → ส่งตอบการได้รับ Query & reply

for Staples

○ Implement EIGRP for IPv4

△ Autonomous System (AS) is a collection of nw ภายใต้การดูแลของ single authority เดียวกัน RFC (1930)

↳ AS number → ใช้ exchange routes between AS

→ managed by IANA & assigned by RIPs to ISPs, Internet Backbone providers, and institution ที่ต้องการใช้งาน

→ 16 bit : 0 - 65535 → since 2007, 32 bit ; over 4 billion △ verify : show ip eigrp neighbors, protocols, route

△ Operation

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

○ Metrics = BW [lowest] (K1 = 1), Delay [สะสม] (K3 = 1), Reliability [worst] (K4 = 0, K5 = 0), Load [worst] (K2 = 0) ดูค่า value = show interface

Default Composite Formula : metric = [K1 * bw + K3 * delay] * 256

$$= \left[\left(\frac{10{,}000{,}000}{bw}\right) + \left(\frac{\text{sum of delay}}{10}\right)\right] \times 256$$

Complete : $= \left[K1 * bw + \frac{(K2 * bw)}{(256 - load)} + K3 * delay\right] * \left[\frac{K5}{reliability + K4}\right]$

- R(config-router)# metric weight tos k1 k2 k3 k4 k5 - set bw : เข้า int ก่อน → R(config-if)#

○ DUAL and the Topology Table (ใช้ FSM (Finite state Machine) ในการเลือกเส้นทาง)

+ Successor (S) [router ที่ไป dest ดีที่สุด] = neighbor router ที่จะเป็นทางไป dest ที่มีค่า ต่ำที่สุด

+ Feasible Successor (FS) [สำรองไว้ ใช้ Feasible condition] if (RD < FD) → FS = Backup path (เส้นทางสำรอง)

+ Reported Distance (RD) [distance ที่ neighbor ทำการ report distance มาให้เรา] = " advertised distance → dest ที่ cost จาก neighbor

+ Fesible Distance (FD) [distance ที่ไปทาง S] = ค่า distance ทั้งหมดที่ไปหา dest ทุก ๆ ค่า cost รวมจาก hop lowest ต่ำสุด

IPv6	vs	IPv4
128 bit		32 bit
base 16		base 10