ชื่อ-สกุล ภารัช แสงตระกูล กระดาษแผ่นที่ 1 รหัสนักศึกษา [illegible] 0 หน้า

CH.1 Network Overview

→ Network diagrams : โครงสร้าง NW ที่เราดูแล มีการเชื่อมต่อยังไง

↳ 1. Physical → บอกถึง port / inteface เชื่อมต่อกันอย่างไร อยู่ตำแหน่งไหน

2. Logical → บอก IP

→ Network protocol ⇒ Tcp / Udp, FTP, ARP, SMTP, Pop3, IMAP, ICMP

– NW Addr. 1. IP addr (Logical addr) @L3 2. MAC Addr. (Physical Addr.) @L2 ↙ Protocol ทาง media

3. Port Number (service Addr.) @L4

→ Components of Network → HW → NW device มี 3 type

1) end devices :

2) intermediary devices : อยู่ระหว่างอุปกรณ์ เช่น NW access devices, Internetworking devices, Security devices

- hub, repeater @L1 ⇒ ส่งพร้อมกันเกิด collision ∴ ใช้ CSMA

- switch, bridges @L2 ⇒ Learning / Flooding / Filtering / Forwarding / Aging

- Routers @L3 ⇒ Routing

3) network media = สื่อกลาง เช่น copper, fibre optic, wireless

→ Types of Networks → SW → 1. Switch เลือกทางออก 2 router เลือกไปทางที่ดีที่สุด

→ Size 1. Small home NW ⇒ ที่ใช้งานตามบ้าน อย่างเดียว 2. Small Office / Home Office ⇒ config บอกใช้...

3. Medium to Large NW ⇒ มีการติดต่อมากขึ้น 100-1000 4 World Wide NW เช่น Internet

→ infrastruture ทั่วไป 1. Local Area NW → มีกลุ่ม admin ดูแล policy / กฎ การอยู่ Ex. สถาบัน ↔ ริมหกอย

2. Wide Area NW → มีหลายกลุ่ม admin (อื่นๆ MAN, WLAN, SAN, PAN)

→ Relable Network 1. fault Tolerance ⇒ หยุดความเสียหาย 2 Scalability ⇒ สามารถปรับขนาดได้ โดยไม่ส่งผลต่อ USER เดิม

3. Security ⇒ จำกัด or limit ก.เข้าถึง 4 Quality of Service (Qos) ⇒ แต่ละ Service ให้ Quality ไม่เท่ากัน

→ TYPE of Connection in LAN ข้อจำกัด (UTP cat 5) 1 BW = 100 Mbps 2 ยาว 100 M (ระยะทางต่อ hub, repeater, switch)

2 type ∴ ① ตรง ② cross → ต่ออุปกรณ์ same ชนกัน SW – hub, R -- router

WAN Connection ⇒ เชื่อมระหว่าง router DCE (female) ⇒ พิมพ์ command clockrate 56000 DTE (male)

สาย console (Rollover Cable) ⇒ router --- PC ⇒ Manage command ต่างๆ

CH2 Basic Router Configuration

→ Port Address @L4 : กำหนดโดย (IANA)

0 - 1023 : requesting entities "well known ports" des. port

1024 - 49,151 : registered port = publish ให้ทุกคนใช้

49,152 - 65,535 : dynamic or private port = "Randomly generate" src port

Ex 20 : FTP (data), 21 : FTP Control, 25 : SMTP, 53 : DNS [TCP/UDP], 80 : WWW HTTP, 81 : HOSTS 2 Name Server

Logical Addresses : IP address (IPV4) @L3

- 5 class :: A, B, C, D, E → (reserved) แบ่งตาม ที่อยู่ max ของ workstation required

A, B, C ใช้เป็น host; D → multicast Addr. ส่งไปหาทุกเครื่องในวง LAN เดียวกัน

- แต่ละ NW / node or com ⇒ มี logical name (domain name) & ip unique

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 Addresses ; MAC Address

- Ethernet : 48 bit ฐาน 2 = 12 ตัว ฐาน 16 → เขียนแบบ 0x ฐาน 16

- ต้องทำโดย IEEE → กำหนด 3 byte (24 bit) code "Organizationally Unique Identifier (OUI)"

- ... NIC หรือ Ethernet device อื่นๆ → ใช้ OUI ที่ 3 byte แรก

→ 2 กฎ → 1. ทุก MAC ที่กำหนดให้ ...

2 ทุก MAC ที่ Same OUI ต้องกำหนด Unique ที่ 3 byte สุดท้าย

Message Delivery

· Unicast → ส่งจากเครื่องต้นทางไปยังปลายทาง ใน NW เดียวกัน

· Broadcast → ส่งหาทุกเครื่อง เช่น DHCP, ARP ใน NW เดียวกัน

Multicast ส่งหาหลายเครื่อง & รับเฉพาะเครื่องที่เปิด service ไว้

Cisco IOS

- function 1. Addressing 2. Interface 3. Routing 4. Managing Resource 5. Secure 6. Qos

- Router & Switch Boot Sequence

1. POST (Power On Self Test) → check HW [ROM]

2. Run boot loader SW

3. Boot Loader does low-level cpu initialization [ROM]

4. " — " initializes the flash file system

5. " — " locates & load a default IOS แล้ว run บน RAM Flash / TFTP server

ชื่อ-สกุล ภาธร แจ้งวิจารระกูล กระดาษแผ่นที่ 1 รหัสนักศึกษา 99011059 หลัง

or Staples

· Accesing 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 sentive Help : "?"
2. Command Syntax Check = enter แล้ว show ว่าไม่มีผิดพลาด
3. Hot keys and Short cuts =
4. IOS Examination Commands → show ...

router # copy running-config startup-config

· Getting Basic 1. ต้องตั้ง hostname 2. จำกัดการเข้าถึง 3. กำหนด interface addressing 4. ตรวจสอบ config + การเชื่อมต่อ 5. save config
router (config) # hostname name ↳ Banner mssg. Router (config)#
↳ Securing Device Access

CH. 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 ที่เข้า Router → process ที่ CPU ไป ออก interface ไหน
2. Fast switching = มีการช่วย ทำให้ forward ได้เร็วๆ 3. Cisco Express Forwarding (CEF) = forward packet ได้เร็วสุด

· Connect Devices
- Default gateway → เป็น ได้ 1 first usable host (.1) 2 last usable host (.254)
→ การออก เชื่อม ไปNWอื่น

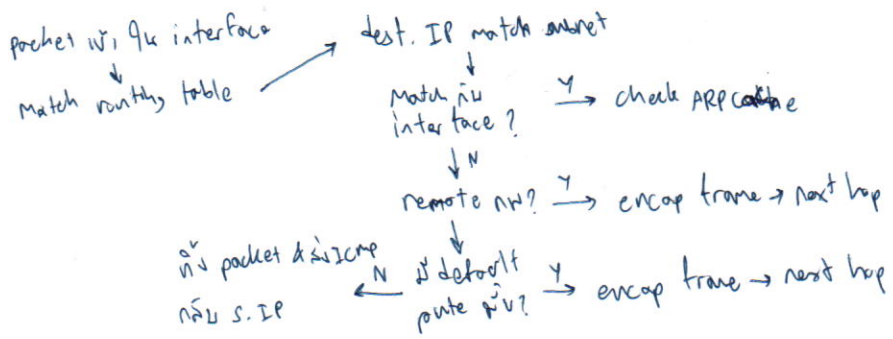
Enable IP on a Host 1 Statically Assigned IP addr
2 Dynamically "——" can กำหนด ip ได้ when NW มี DHCP (Dynamic Host Configuration Protocol)

or Staples

Switching Packet between NW

ขั้นตอน เช็ค dest. ip (L3) → ดูเทียบใน routing table → กำหนดออกไปที่ เทียบ กับ MAC addr. → ได้ dest MAC (L#2).

Path Determination



→ Best Path : lowest Metric (cost)
⇒ Dynamic routing protocol ดูจาก
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, relibility
> Load balancing = กรณี > 1 เส้นทาง
⇒ ใช้ หลายเส้นทางให้ เท่ากัน
ถ.ค่า cost เท่า แต่ ก. ไร้บ้างไป
> Administrative Distance (AD) : "trustworthiness"
= ระยะห่าง ของ ผู้ดูแล ที่ protocol ที่ใช้
→ connected = 0, static = 1, Internal EIGRP = 90
OSPF = 110 RIP = 120

Routing
1 static Routing → Manual
ข้อดี : security, ใช้ resource น้อย ในการ process ลง routing entry
ข้อเสีย : ดูแลยาก ในการ scalability can เปลี่ยนเส้นทาง
ตาม ได้ : NW ขนาดเล็ก, จน. router น้อย, NW มี ทางออก ทางเดียว "stub NW"
4 type : ① standard ② Default = ใช้ when dest. ip ไม่ match
3 Summary 4 Floating = back up
Router config # ip route nw-addr subnet-mask {ip-addr. | exit-intf}

Classful Addressing → update ตาม class
Classless Inter-Domain Routing
Summarization เมื่อไป 10ที่ ติดกัน 2 ไปตามนี้เก่า
ถ้า set ให้ ต้อง รอ ip ที่ ตามนี้ที่ 1 แปลง ip → ฐาน 2
2 Group bit ที่ same กันมากที่สุด

Config : Next hop Option
ระบุ R(config)# interface G0/0
"——"-if ip address 172.168.1.0 255.255.255.0 } Basic set
"——" no shutdown
R1 (config) # ip route 172.168.1.0 255.255.255.0 ↳ 172.168.1.2 แบบที่ 1
↳ S 0/0/0 แบบที่ 2

VLSM
Fixed Length Subnet Masking
วิธีทำ 1 prefix ใหม่ - prefix เก่า
= จน. bit ที่เปลี่ยน
2. เปลี่ยน ทุก bit เป็น 1 แล้ว
แล้วนำค่าที่ได้ ไป +ip เริ่มรวมกับ

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 +1 ห้องจะไปเร็วขึ้น to Intermediate System)

ชื่อ-สกุล ภาวัร แสงตระกูล กระดาษแผ่นที่ 2 รหัสนักศึกษา

for Staples

Chapter 4 Distance Vector Routing Protocol RIP VER 1

▶ Dynamic Routing Protocol

- f : 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 dest. nw.

- หา best path ใหม่ได้ถ้า path ปัจจุบัน ไม่ว่างยอด

component ① Algorithm : ใช้เพื่อประมวล routing info & best path

② Routing protocol msg : รับส่งกับ neighbor & แลกเปลี่ยน routing info (best path)

	Dynamic	VS Static
ความยุ่งยากในการ config	∅ ขนาด nw (เท่าเดิม)	∝ ขนาด nw (เพิ่ม command ทุก router
Required ความรู้ Admin	Advanced (เพราะ config basic → nw ใหญ่+ซับซ้อน	No เพิ่มเติม (กำหนดเส้นทางเองโดย command)
Topology change	ปรับ auto	admin config ใหม่ all
Scaling	เหมาะ : simple & complex (router ที่ไม่ใช่ directly ไม่ต้อง)	เหมาะ : simple topologies
Security	น้อยกว่า	มากกว่า
Resource usage	ใช้ CPU , mem (เก็บ routing info , link bandwidth	No เพิ่มเติม
Predictability	Route ∝ current topogy	Route → dest. ทางเดิมตลอด

▶ Classifying Routing Protocols

DRP

for Staples

(ภายใน AS) Interior Gateway Protocol (IGP) — (ภายนอก) Exterior Gateway Protocol (EGP)

Distance Vector P.

- ประกาศ vector [distance, direction]
- incomplete view ของ nw topology
- Periodic update (ส่งตามรอบ) ทั้งหมด

RIP (ROUTING Info Pro) — RIPv.2

IGRP (Interior Gateway Routing Pro.)

EIGRP (Enhanced Interior Gateway Routing Pro.)

Linkstate P.

- Complete nw topology (เก็บข้อมูล all)
- update ไม่ใช้ periodic

Open Shortest path First (OSPF)

Intermediate System to Intermediate System (IS-IS)

BGP (Border Gateway Protocol)

- Autonomous System (AS) เป็นกลุ่มของ router (กลุ่ม policy) ภายใต้ ก. ควบคุมของ single authority (หรือ มีคุณ)

- 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 ถูก กำหนด when routing table ของ all router ครบทั้งไม่เปลี่ยน
↳ 2 type : slower : RIP & IGRP , Faster (มีการเปลี่ยนแปลง when เกิดการเปลี่ยนแปลงเร็ว ดี >) : EIGP & OSPF

▷ Routing Protocol Metrics

- Metric : ค่าที่ใช้คำนวณ,คิด/หา ใน ก. ไป dest. nw จะเลือก best path ไหน เช่น Hop count , BW , Cost , Delay , Load , Reliability
- Load balancing : nw มีเส้นทาง >1 ที่มี metric เท่ากัน → เลือกใช้ ก. ใช้ งานแต่ละ เส้นทางเท่ากัน

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

ลักษณะสำคัญ : ค่าความน่าเชื่อถือของสิทธิ์ มีเกณฑ์ ของ particular (จะเลือก) 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 สิ่ง มาใช้คิด ① Vector or direction , ก. แลกเปลี่ยนการควบคุม ② Distance to final dest. (cost)

ลักษณะทั่วไป : periodic (ตามแต่รอบ) update , neighbor (ตัวใกล้เคียง) , broad cast (255.255.255.255) update , เอา routing table all ไป update

ลักษณะทั่วไปของ Routing Protocol : เหมาะกับทำใช้ check ว่า DV ดี / ไม่ดี ? ① Time to convergence ⇒ เวลาที่เป็น steady state ของ routing table ที่เกิดการเปลี่ยนแปลง ② scalability ขยายการรองรับ ③ Resource usage ④ Implementation & maintenance

▷ NW Discovery (ค้นพบ) (ทำ basic config ก่อน)

for Staples

มี 3 stage ① Cold state : Router Initial Startup

② Initial Exchange of Routing info → ทำการ แลกเปลี่ยน

③ Exchange of Routing info → Update (เพิ่ม hop count) routing info → ของ router อื่นที่ว่าห่างเท่าไร!

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 เริ่ม lost) (default 180) , Holddown timer (ถ้า down → hold ไว้ ไม่ว่า up หรือไม่) (default 180) , Flush (ทิ้ง) timer (default 240)

· Bounded (ขอบเขต) Update : EIGRP → Update แค่ที่เปลี่ยน
· Triggered Update → Update เลย ไม่รอ periodic time
· Random Jitter → จุดชุมนุม nw ที่เป็น multiple access router หลายตัวเชื่อมต่อกัน → if เกิด update ขนกันพร้อมกัน จึงใช้ Random

▷ ปัญหา standard DV. ① Routing Loops เกิด when intf ใดๆ down จะถูกลบออกไป จาก table → ถ้ายังไม่ neighbor จะกลับมา update (จน update → hop เพิ่มไปจนถึง ∞)
↳ แก้ปัญหา ① set max hop=15 → if hop=16 → unreachable (เน้นdown ไปแล้ว)
② holddown timer (ถ้า intf down → hold)
③ split Horizon Rule → ไม่ส่ง ข้อมูล update กลับไปทาง intf ที่ได้รับ update มา
④ Route Poisoning → ⓐ เกิดdown set unreachable ⓑ ส่ง unreachable ที่เป็น poison ออกไป
⑤ ③ with ④ → ถ้าเกิด unreachable จะ over rule splithorizon โดยส่ง ip intf ที่ down (hop=16)
⑥ IP & TTL (Time to Live) เกิดการวน update LP. จะหยุดเมื่อ TTL = 0

	RIPv1	RIPv2	IGRP	EIGRP
speed convergence	slow	slow	slow	fast
scalability - size nw	small	small	small	Large
use of VLSM	X	✓	X	✓
Resource usage	Low	Low	Low	Medium
implementation & maintenance	simple	simple	simple	complex

▷ RIP Version 1 AD=120
① คุณสมบัติ · classful, DV = Metric = hop count : hop count > 15 unreachable : update broadcast ทุก 30 s
↳ encapsulated (บรรจุ) ใน UDP segment ด้วย source & dest. port 520 Transport layer
RIPv1 Message Format
↳ Data Link Frame Header | IP Packet Header | UDP Segment Header | RIP Message (512 Byte; Up to 25 routes)

header ← Request / reply
V. Command = 1 or 2	Version = 1	Must be zero
Address family Identifier (2 = IP)		Must be zero
IP Address (Network Address) → dest.		
Must be zero		
Must be zero		
Metric (Hops) → 1-15 ส่งจาก table 1 → ส่งออก		

(v2 full routing table)

○ Msg มี 2 type
① Request → ส่ง routing table
→ ไปยัง intf ที่ config ไว้ เพื่อขอรับการ update
② Response → ส่ง info ของ routing table
□ ipaddr. แบ่งตาม class A, B, C
□ Basic RIPv1 Config ① ทำ basic config
② สั่ง router rip + กำหนด nw R1(config)# router rip
R1(config-router)# network
nw ip ที่ต้องการรู้จัก

- Verification (ตรวจสอบ) & trouble shooting (จัดการปัญหา) : 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/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 router แสดง บท ในการรวม multiple route เร็วขึ้นใน routing table
└ ข้อเสีย : ไม่ support discontiguous nw (major nw เดียวกัน แต่ไม่อยู่ติดกัน) → อาจจะ ทำ load balancing ได้
boundary Routers : summarize RIP subnet from 1 major nw to another
Processing RIP update
ดู ค่าที่ส่ง & ค่าที่ update ไป (intf) ว่าเป็น classful เดียวกันไหม ! → Y : update subnet nw เช่น 172.16.1.0
↳ N : update classful เช่น 172.16.0.0
• default route & RIPv1 ต้องไม่เอา หาออกใน routing table (อาจจะเพราะ คนละ protocol) → จะ ออก default route
R(config)# ip route 0.0.0.0 0.0.0.0 s0/0/1
default info. originate command → เพื่อ update ไปใน rip ว่าเชื่อม : static → dynamic
Router ที่อยู่ ระหว่าง 2 protocol ← R(config-router)# default-information originate

Chapter 5 RIP version 2 & Access Control Lists

RIPv1	vs	RIPv2
Classful (ไม่ส่ง subnet mask, ไม่ support CIDR)		Classless (update subnet mask, support Variable Length Subnet Masking (VLSM) summarization (Prefix Aggregation))
not support discontiguous subnet		update next hop addr.
not support VLSM เพราะ ไม่ส่ง subnet mask (255.255.255.255)		มี authentication routing (รองรับ discontiguous บางกรณี)
routing update ⇒ broadcast		

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

ข้อจำกัด ของ RIPv1
เป็น virtual interface สามารถ เพิ่ม routing ทำ update ออก ไปได้

loop back intf → ping ไป → ip virtual intf → reply ให้
Null intf → ใช้ ส่งคนที่จะ เข้ามา channel ที่เรา ไม่ ต้องการ → วิ่ง ออก null intf → packet discard เพราะ timeout
static route & null intf → null intf จะ รับ หน้าที่ เป็น ทางออก ของ static route
R(config)# ip route summary-static-route subnet-mask Null0
(major-nw) → ของ static supernet route

ชื่อ-สกุล ภาธิศ เนียวพระกุล กระดาษแผ่นที่ 3 รหัสนักศึกษา [illegible]

for Staples

· Route redistribution (จัดสรร) → อยากให้ rip รู้จัก static ก็ทำเพิ่ม (ให้ rip ส่ง static ไปด้วย): R(config-router)# redistribute static

· Verify & Test Connectivity : show ip interface brief, ping (ผล: ! = ได้, U = ไม่ได้, . = timeout), trace route

RIPv1 : Classful, ไม่ส่ง subnet mask, summarize nw ที่ 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.

▸ RIPv2

▫ Config · Enabling & verify (ตรวจสอบ) RIPv2 ↗ show ip protocols

· Config RIP → RIP v1 → สามารถรับได้ทั้ง v_1 & v_2 แต่ส่งได้แค่ v1

→ RIP v_2 → สามารถรับ & ส่งได้แค่ v.2

· Auto-Summary & RIP v_2 → auto sum route ที่ major nw boundaries

→ sum route ด้วย subnet mask ที่น้อยกว่า classful subnet mask

· disabling Auto-Summary : no auto-summary เพราะเมื่อขึ้น nw topology ที่เป็น discontiguous

VLSM & CIDR → verify info ที่ส่งโดย RIPv2 debug ip rip

→ VLSM → เผยแพร่ 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 = ควบคุมการเข้าออก → ตรวจสอบ → check → source → ผ่านไหม?

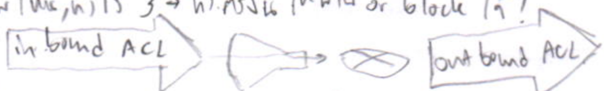
↳ ตัวอย่างนี้ (เช่น FTP) ผ่านไหม?

⇒ ดูว่า conversation

Packet Filtering ดู ① dest, source ที่ L2 ② protocol อะไร ③ไป nw ไหน, อะไร } → ทำครบนี้ให้ผ่าน or block ไว้?

Operation → ทำเป็น sequence statement

→ last statement เป็น implicit deny → block → discard



for Staples

· Standard IPv4 ACLs	vs	Extendend IPv4 ACLs
· - check source addr.		- check source & destination addr
- ปกติ permits or denies ทั้งชุด protocol		- ปกติ permits or denies specific เฉพาะ protocol
acces-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 & 2000-2649

▫ wild card → invert ของ subnet mask

→ 0 = match / fix, 1 = ignore / 0 = ไรก็ได้

→ วิธีหา set ip (match range) ① หาความเหมือนกันของ bit ที่ซ้ำกัน ใส่ wild card mask ตน.นั้น = 0

② bit ที่เหลือใส่ 1

if กรณีคิดไม่ได้ รองแยก Pattern or / and ส่วนต่างสุดท้าย wild card จะ = 1 เหมือนกัน

→ 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

Guideline 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 a intf, แยก ACL ctrl in & out bound traffic

→ One ACL / interface = ACL ctrl traffic for an intf, Ex G0/0

Where → Extend ACL : ที่ close source → standard ACL : ที่ close destination

config ACLs → standard → number

Router (config)# access-list access-list-number deny | permit | remark = comment source [source-wild card] [log]

ถ้า remove all : no access-list

ถ้าแก้ Ⅰ no access list num# → สั่งใหม่

Ⅱ no บรรทัด# → บรรทัด# ที่สั่ง

→ in intf

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

ถ้า remove all : no ip access-group

ถ้าแก้ → บรรทัด # คำสั่งใหม่ ↓ บรรทัดที่แก้

→ ชื่อ

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

· Verify : show ip interface, show access-lists

· Securing VTY port → อย่างจำกัดคนเข้า permit เฉพาะที่เราต้องการให้เข้า

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

→ Extended : filter : source / dest addr, protocol, port number

Chatper 6 OSPF & DHCP

▷ Link-state Routing Protocol = เป็น protocol ที่ต้องสร้าง complete map ของ nw topology ขึ้นก่อน → หา shortest path first (SPF) (↑ มี info all)

เหมาะกับ : ① large nw, ② fast convergence ③ admin ความรู้ความสามารถดี

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 เอาall LSP เก็บใน DB (รูปแบบ tree) + adding OSPF → routing table

ข้อดี : ① สร้าง topology map สามารถหา shortest path ② fast convergence ข้อมูลที่ไป ③ LSP sent only when change topology → ก็แค่หา shortest path ④ hierarchical design (NW ใหญ่ๆจะดี) → ลด resource เพราะแยกข้อมูลแค่ในarea (ไม่ต้องเก็บ=ข้อมูลทั้งหมด)

ข้อเสีย : ① ใช้ mem ในการเก็บall link-state เยอะ ② ใช้ cpu ในการคำนวณ ③ ตอนส่ง LSP พร้อมๆ ใช้ BW เยอะ.

▷ OSPF AD < 110

↳ 3 table : ① Neighbor show ip ospf neighbor ② Topology show ip ospf database ③ Routing (มี shortest path)

Message → Encapsulating : MAC Dest = Multi-cast : 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
02 Db Description (DBD) → synchronization db info
03 Link-state Request (LSR) → request link-state
04 " Update (LSU) → send update link-state
05 " Acknowledgment (LSAck) → ตอบกลับได้เลย

operation : สร้างการเชื่อมต่อ ① Down state (เริ่มต้น) → ② Init state (เริ่มส่ง hello) → ③ Two-way state ตอบกลับhello → Exstate start
→ Exchange state → Loading state → Full state (ระหว่าง router update ข้อมูลเรียบร้อย)

Config Single-Area OSPF v2 router ospf process id → 1-65535 เป็น locally significant
R(config-router) # router-id 1.1.1.1 → ถ้าไม่set สามารถใช้ loop back, active interface ip สูงสุด แต่การ มี① ไม่มี②
router ospf process-id
network network-address wild card-mask area area-id

OSPF cost → ใช้ BW กำหนด [default reference BW = 10^8

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

→ ต้องการเปลี่ยนค่า cost
→ เปลี่ยนที่ ref BW ถ้าไม่ต้องการให้ cost เท่ากัน ก็เปลี่ยนค่า cost กัน
auto-cost reference-bandwidth bandwidth-mbps
- Fast Ethernet F=G=10G=1
auto-cost reference-bandwidth 100 (หน่วย 10^8)
- Gigabit Ethernet F=10, G=10G=1
auto-cost reference-bandwidth 1000 (หน่วย 10^8)
- 10 Gigabit Ethernet F=100, G=10, 10G=1
auto-cost reference-bandwidth 10000 (หน่วย 10^8)

→ เปลี่ยนที่ BW R(config-if) # bandwidth 64 (EIGRP & OSPF สามารถใช้ได้
→ เปลี่ยนที่ 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 N
R(config)# router ospf process-id
R(config-router) # default-information originate

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

method ① Manual Allocation : admin assignเอง
② Automatic Allocation : DHCPv4 auto assign addr จาก pool & ไม่มี lease (เช่า) time
③ Dynamic Allocation : ใครเปิดขึ้นมาก่อจะได้ ip ไป มี leave time → ถ้าหมด leave time ต้อง re ip ใหม่

ชื่อ-สกุล สาธร แจ้วพระกุล กระดาษแผ่นที่ 4 รหัสนักศึกษา 88888888

for Staples

Chapter 7 Basic Switch Address Resolution Protocol

- ▷ LAN Design → Borderless sw nw design : มักทำให้ :: - Hierarchical, - Modularity, Resiliency, flexibility

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

[1] Core → อุปกรณ์ที่รองรับ BW รวม → ทำให้ speed ↑ จุดเชื่อมต่อภายนอก } Layer 3 Support [Gig/10 Gig Ethernet]

[2] Distribution ⇒ เชื่อมต่อระหว่าง [1] & [2], Security Policy / Access Ctrl

[3] Access ⇒ ติดต่อ end device, Port security, VLAN, [Fa/Gig Ethernet], Power over Ethernet รองรับ ↓ 1 LAN or ไม่ก็ได้ ใน 1 LAN, ipv6 line

ทุก layer เป็น SW

Redundant component ⇒ มีสำรองเผื่อพังไป

Link aggregation → ทำให้ BW device เพิ่ม ↑ Quality of service (QoS)

□ เพิ่มความสามารถในการใช้ได้ LAN BW & ประสิทธิภาพ ให้ Max

- f^n & การวาง Server ① Enterprise S. (รวมทั้งองค์กร) → ติดตั้ง @ MDF (Main Distribution Facility : Core ⇒ การเดินไปข้างนอก

② Workshop S. (แผนก = กลุ่ม) ⇒ ติดตั้ง @ IDF (Intermediate D.F. : Distribution) ⇒ เชื่อมต่อระหว่าง Cross กับ Access ที่ใกล้กันมั้ย

VCC (Vertical cross-connect) : optical fibre : → MDF ↔ IDF

HCC (Horizontal ") : UTP : ⇒ Distribution ↔ Access

- Collision detection issue (ปัญหา การตรวจจับการชน)
- Segmentation issue ⇒ (ปัญหา ก. แบ่งส่วน) → ทำให้ ปัญ. การ ก. ใช้งาน nw มีปส.ก. ที่ขึ้น

Broadcast domain issue → ใน nw เกิด traffic เช่น broadcast หา MAC Addr. ∴ broadcast nw ควรมีขนาดเล็ก

Segmentation เป็น process split single collision domain → smaller collision domain ลดจำนวน collision บน LAN segment

Broadcast domain รับทุกอย่างทุก port but - router (L#3) เป็นตัว filter / segment broadcast ไม่ให้กระจายออก L#2 device ใช้ เช่น bridge, SW

▷ SW Enviroment

□ SW Operation ① Learning :: รับ frame เข้า จะจำเรียน Source MAC Addr ว่าต่อกับ Port ไหน & reset Aging

② Aging :: อายุ ของ MAC Addr. → if หมด → ทิ้ง

หน้าที่การส่งของ ③ Flooding ส่ง frame ออก ทุก port ของ SW เมื่อ frame เป็น 1) broadcast, 2) multicast, 3) unknown unicast → กรณี dest. ไม่อยู่ใน table

④ Forwarding :: ส่งไป dest' (กรณี พบใน table)

⑤ Filtering :: if ได้รับ frame ไป dest จาก port ที่เป็นของไป dest. (source & dest. บน same interface) จะทำการ filter ทิ้ง

for Staples

SW Methods ① Store & forward SW ⇒ check CRC ว่า error ไหม → มี ทิ้ง → ไม่มี ส่ง, auto buffer

② Cut-Through SW ⇒ (check เฉพาะ ส่วน (dest source อย่างน้อย 12 byte แรก) NO FCS & auto buffer

↳ 2 mode :: [1] Fast-forward ~ 12 byte [2] Fragment-free ~ 64 byte :: < 64 = ชน → ทิ้ง → ไม่ส่ง

SW Domains ① Collision Domains ⇒ domain ที่เกิด ก. ชน ของ ที่ส่งข้อมูล มาพร้อมกัน "ก @ SW เป็นตัวแบ่ง"

② Broadcast ~ ⇒ domain ที่ส่ง broadcast → อยู่ domain เดียวกันจะได้รับ "ก @ router เป็นตัวแบ่ง"

▷ Basic SW Concept & Configuration

□ Basic SW Config ∘ SW Boot Sequence = same Router ∘ Preparing of Basic SW Management : จะไม่มี loopback ∴ กำหนด เป็น SVI (SW Virtual interface) → vlan ∘ Config SW Port → Duplex Communication : ① Full ② Half (เหมือนกันกับ ต้องเลือก same กัน) → แต่ง speed ได้ (เปลี่ยน speed) → Auto-MDIX : ปกติ SW จะเชื่อมต่อกัน สาย cross-over เข้าที่ intf → S(config-if)# duplex full → S(config-if)# speed 100 (เปลี่ยน speed) → Auto-MDIX : ปกติ SW จะเชื่อมต่อกัน สาย cross-over

but. ต้องการเปลี่ยนเป็น รายตรง 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(config)# ip domain-name ชื่อ → # crypto key generate rsa → # username admin pass cc กด → line vty 0 15 → -line)# transport input ssh → -line)# login local [verify SSH : show ip ssh, show ssh]

□ SW Port Security ⇒ กำหนด policy ว่า จะให้ MAC Address ไหนเข้า / ออก ได้ได้

S(config-if)# switchport mode access → # switchport port-security → เลือกหมด / ประกาศ

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 frame ที่เข้ามาก่อน → record ไว้

กรณี เพิ่ม ของ MAC # switchport port-security violation protect mode

for Staples

Violation mode : ① protect : security violation protect mode

② restrict : security violation restrict mode → ส่งไป เก็บว่ามีอะไร ที่ผิดบ้าง

③ shutdown : security violation shutdown mode → default

[verify : show port-security int fa0/1, show port-security address]

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

IPv4 : Classless [ต่อ P. 1-2] : Variable Length Subnet Masking (VSLM) : แบ่งตามที่ใช้

: เลือกของขนาดไม่ให้ไปเล็ก

Fixed ~ : แบ่งทุก NW เท่ากัน

Chapter 8 LAN Redundancy & Spanning Tree Protocol (STP)

▷ Issue with Layer 1 Redundancy ① MAC Addr. Instability ⇒ MAC Addr. table จะไม่ได้รับ เพราะเปลี่ยนแปลงบ่อย ② Broadcast storms ⇒ วนวิ่งไปทั่วไม่จบ ③ Multiple frame transmission ⇒ start: unknown unicast → ทำให้ dest ได้รับหลาย frame but source ส่ง 1 frame ▷ STP ⇒ ดูจุดหมาย: ทำ block port → block ไม่ให้เกิด traffic ขึ้น (เพื่อแก้ปัญหา)

ขั้นตอน: ① หา Root Bridge = ใช้ priority min [Role: [1] 1 RB/เครือข่าย [2] 1 RP/1 RB [3] 1 DP/segment] ② หา path cost all ③ หา Root port → path cost min → อีกที่เหลือจะเป็น Designated Port ③ ส่วน segment ที่ path cost เท่ากัน → ดู BID min เป็น designated port → อีกฝั่งจะเป็น block port

ดูที่ BPDU (Bridge protected data unit) → มี RB & source [802.1D]

สิ่งที่ใช้แต่ละเครื่อง ① BPDU flag (priority น้อยสุด) ② BID น้อยสุด ③ Path cost < ④ Sender's BID ⑤ Sender's Port <

config: วิธีที่ 1: S1(config)# spanning-tree VLAN 1 root primary
วิธีที่ 2: S3(config)# spanning-tree VLAN 1 priority 24576 เพื่อให้มากขึ้น (if-
(if S1, S2 [1] เป็น RB) S2 (config)# spanning-tree VLAN 1 root secondary (Verify: show spanning-tree)

□ แบบ Extended System ID: B. Priority → B. Priority (per VLAN) + Extended SysID (VLAN) + MAC Addr. : BID
(2 byte, 4 bit, 12 bit, 6 bytes, 8 bytes)

o PVST+ (พัฒนามาจาก IEEE 802.1D STP) → เพื่อทำ load balancing โดยแยก root / vlan [Verify: show spanning-tree active]

o Rapid PVST → มี Alternate port (ไม่ให้ block ไปเลย config เร็ว ขึ้น → ได้ทันที)
→ ทำ Edge Port @ port ต่อ {a host, router: -if)# spanning-tree port fast (Port นี้ไม่ต้องรับ BPDU ด้วย)
→ link type: port ที่เชื่อม s: ระหว่าง sw ที่เชื่อม point to point -if)# spanning tree bpduguard enable
→ config: S1(config)# spanning-tree mode rapid-pvst → # int ทำ p-to-p → # spanning-tree link-type point-to-point

Chapter 9 VLANs & Inter VLAN

▷ VLAN = ทำ partition (แบ่งแยก: ข้อมูล & broadcast domain กัน) Layer 2 เป็น switch ทำการต่อ ได้แยก VLAN เสร็จ
o ข้อดี: - security ดีขึ้น, - ลด cost, - ประสิทธิภาพดีขึ้น, - broadcast domain เล็กลง, - ปัญหา ด้าน IT จัดการ ง่าย

□ in a Multi-SW Environment
· VLAN Trunk: set ที่ intf ที่เชื่อมระหว่าง sw ที่มี VLAN → can carry ข้อมูลได้ > 1 VLAN
↳ config: เข้า intf → if)# switchport mode trunk [verify: show int f0/0 switchport] (TAG ทั้งหมด VLAN ผ่าน Trunk)

Tagging Ethernet Frames (IEEE 802.1Q): Ethernet Frame → Dest MAC | Src MAC | Tag | Type/Length | Data | FCS |

Assignment: VLAN number ⇒ 1-1005 เก็บ config @ vlan.dat (ใน flash)
⇒ 1006-4096 เก็บ config @ running-config (ใน NVRAM)

สร้าง: I S(config)# vlan vlan-num → vlan)# name ชื่อ
I S# vlan database → (vlan)# vlan num name ชื่อ
| assign port ให้ vlan: เข้า intf → if # switchport mode access → # switch access vlan

▷ Inter-VLAN Routing → router set เป็น Trunk โดยจะทำการ "sub interface"
Config ① set basic routing (set ip address, no shutdown)
② R(config)# interface g0/0 (ต่อ VLAN → sub if) # 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 จะใช้ revision number 32 bit
↳ 3 modes: ① server → can สร้าง, remove, rename VLAN บนทั้งใน domain เห็น ② Client → รับข้อมูล VTP มา process, ส่ง VTP msg ออก ทาง Trunk ③ Transparent → can สร้าง, remove, rename แต่ ใน ตัวมัน, เป็นแค่ทางผ่าน

□ Config: 2 แบบ สิ่งที่ต้องมี: 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 vlan configuration S(vlan)# vtp v2-mode [verify: show vtp status] → # vtp server/client/transparent

□ Pruning → manage ให้ traffic ที่ไม่จำเป็นไปยัง interface น้อยลง โดย config ที่ Interface ต่าง เพื่อremove ข้อมูลที่ไม่ได้ใช้
S(vlan)# vtp pruning → เข้า interface → s(config-if)# switchport trunk pruning vlan remove vlan-num

□ NAT ⇒ แปลง private ip ⇔ publish (real ip)
Terminology: 4 Types ① Inside local Addr. (private ip) ② Outside local Addr ③ Inside global Addr ④ Outside global Addr

□ type ① static: กำหนด ตายตัว [map: 1 to 1] → ① R(config)# ip nat inside source static local-ip global-ip
② Dynamic: มี pool ของ Global / Pest ip [map ทั้ง many ↔ 1]: real ไม่พอ ใช้ไม่ได้ } (1.1) # ip nat pool ชื่อ start-ip end-ip {net mask netmask / prefix-length}
(1.2) set ACL (1.3) ip nat inside source list ACL-num pool
③ PAT (Port Addr. Translation) → port ที่แปลงร่วมกัน ได้ addr. (Map: many ↔ 1)

□ Config: 3 ส่วน ① NAT ② INSIDE: R(config-if)# ip nat inside ③ Outside: R(config-if)# ip nat outside

ชื่อ-สกุล ภาธร แจ่มจิตรตระกูล กระดาษแผ่นที่ 5 รหัสนักศึกษา

for Staples

□ Operation

Initial Route Discovery (ค้นหา) 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

○ 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 : $metric = [k1*bw + k3*delay]*256 = [(\frac{10,000,000}{bw}) + (\frac{sum\ of\ delay}{10})]*256$ complete: $= [k1*bw + \frac{(k2*bw)}{(256-load)} + k3*delay] * [\frac{k5}{reliability + k4}]$

- 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) ในการเลือกเส้นทาง) → show ip eigrp topology [all-link], show ip route

+ Successor(s) [router ที่ไป dest. สั้นสุด] = neighbor router ที่จะเลือกทางไป dest ที่มีค่า path น้อยสุด

+ Feasible Successor (FS) [มีได้ถ้าผ่าน Feasible condition] = Back up path (เส้นทางสำรอง)

+ Reported Distance (RD) [distance ที่ neighbor ตัวอื่นๆ report distance มาให้เรา] = "advertised distance" จากเราไปมัน → dest มี cost เท่าไร" มองจาก hop

+ Feasible Distance (FD) [distance ที่ไปทั้งหมด] = ค่า distance ที่คำนวณไปหา dest หาเป็นค่า cost lowest → dest

▷ IPv6

□ IPv4 Issue

Need for IPv6 → แก้ปัญหา ip มีจำนวนจำกัด (private ip, NAT), ใช้กับ IoT ได้

IPv6	vs IPv4
128 bit	32 bit
base 16	base 10

ก. ใช้ร่วมกัน (coexistence)

- Migration IPv4 → IPv6 techniques ① Dual stack = run ทั้งคู่ if พร้อมเปลี่ยนใหม่ ไม่กระทบ user
② Tunneling (อยากใช้ v6 but core ไม่ support) = เปลี่ยน (ห่อ) แล้ววิ่งผ่าน v4 แบบเดิม
③ Translation (คล้าย NAT) = IPv6 ↔ IPv4

for Staples

□ IPv6 Addressing = 128 bit แบ่ง 8 ส่วน [1 ส่วน มี 2 byte = 16 bit] → represent base 16 ทุกๆ 4 bit

○ ออกแบบวิธีกำหนดค่า IPv6

Rule 1 - Omit Leading 0s = ใช้แต่ละ partition "0" ที่อยู่ข้างหน้าให้ตัดทิ้งไป 000x (→ 0-F), 00xx, 0xxx

Rule 2 - Omit All 0 segment = แต่ละ segment ที่เป็น "0" ติดกันยาวๆ ตัวย่อ "::" ใช้ได้ที่เดียว only

□ Type of IPv6 Address

IPv6 Addr. Type ① Unicast : ⊖ Global Unicast (→ 3 bit แรก "001" or 2000::/3) ⊖ Link-local (→ สร้าง intf. ที่อยู่ติดกัน (ใน Local เดียวกัน) ↳ ใน router FE80::/10 = ใช้งานเมื่อไม่ได้กำหนด link-local (ใช้ได้เฉพาะใน local)) ⊖ Unique Local

Static config ←

↳ ipv6 address ipv6-addr/prefix-length → no shutdown

② Multicast

③ Anycast = ส่งไปได้หลาย device

IPv6 Prefix Length = 0-128, most LANs is /64 bcz LAN เล็กสุด 64 bit

□ IPv6 Routing

Config static route

R(config)# ipv6 route

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

ไปก่อน 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-int}

Verify 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 2.00.0 → 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 neighbor, show ipv6 protocols

show ipv6 route

for Staples

Chapter 11 EIGRP IPv6 Routing ▷ EIGRP (Enhanced IGRP)

Characteristics (ลักษณะประจำตัว)

Basic features ▷ Cisco-proprietary (เจ้าของเดียว) protocol ของ Cisco เปิดให้ใช้ใน 1992

① สร้าง classless version of IGRP + หลายอย่างที่เหมาะสม ซึ่งนับว่ามี หลาย protocol, หลายที่ใหญ่ ก็สร้างขึ้นบน Cisco router เป็นหลัก

Dual (Diffusing Update Algorithm) = การไม่มี loop-free & backup path ทั้งหมด ของ routing domain → หา best path

→ ทำให้ routing ทำงาน very fast convergent (ยิ่ง convergent time < ของ ospf) รวมทั้งมี backup path (ทำให้ พร้อม)

(การทำงาน ≈ OSPF เชื่อม router) เร็ว → if link down จะเลือก path ใน backup ขึ้นมาเลย

✓ Establishing Neighbor = เชื่อม ค. สัมพันธ์ กับ directly connected EIGRP router

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

Reliable Transport Protocol = RTP provides delivery of EIGRP packet to neighbors

= RTP and neighbor adjacencies are used by Dual (ใน ค. maintain)

EIGRP can update = update เฉพาะ ส่วนที่มี ค.ปป. รวมทั้ง update ไป ให้ เฉพาะ ส่วนที่มี ค.ปป ∴ update < RTP

✓ Partial and Bounded

✓ Equal and UnEqual Cost = ทำให้ traffic ดู แพร่ระบาย เพื่อ กระจาย การรับข้อมูล ในเครือข่ายได้ดีขึ้น

Load Balancing แม้ว่า cost ต่าง ก็เป็น 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

⊖ 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 ซึ่งรับ delivery & reception ของ EIGRP packets

= เป็น msg ที่อยู่ ใน application layer ใช้ maintain ข้อมูล msg ต่างๆ ที่ส่ง ของ EIGRP

ไม่ใช่ ทั้งหมด ของ RIP packet จะส่งได้ น่าเชื่อถือ (msg ≈ OSPF)

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

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

รองรับ authentication (no encrypt routing update) และ เป็น ที่ ยอมรับ (auth ≈ RIPv2 OSPF)

Packet Type routing update or queries EIGRP multicast IPv4

① Hello → ทำให้ adjacencies ระหว่าง router 2 ตัวที่เป็น neighbor กัน, ไม่ต้องรอ Response, ไม่ reliably

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

③ Acknowledgement → พอได้รับ การ update ก็ตอบ Ack

④ Query → request info routing จาก neighbor routers } ขอข้อมูล info ของ routing เพิ่มเติมให้ กับ router ที่อยู่ ข้างๆ

⑤ Reply → หลังจากได้รับ query ก็ reply } reply ตอบกลับ ว่ามี หรือไม่มี

Implement EIGRP for IPv4

Autonomous System (AS) is a collection of nw ภายใต้ การดูแล ของ single authority

↳ AS number → ใช้ exchange routes between AS

→ Managed by IANA & assigned by RIRs to ISP, Internet Backbone

→ 16 bit : 0-65535 → since 2007, 32 bit : over 4 billion verify show ip route

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

R(config-router)# eigrp router-id