418351 Friday, September 19, 2008 ชื่อ กมล มูลศรีแก้ว รหัสนิสิต

บันทึกช่วยจำ

Comparison LS, DV.

การส่งข้อความ

LS จะส่งข้อความไปให้ทุก node.

DV ส่งเฉพาะ node ที่ติดต่อกันเท่านั้น

ความเร็ว

LS $O(n^2)$

DV ค่าไม่แน่นอน

ถ้า router ส่งผิดพลาด

LS ดีเพราะให้ทุก node ทราบค่า Link ที่ผิด

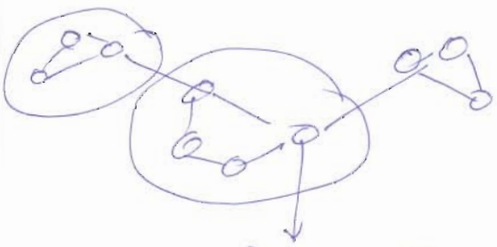
Hierarchical Routing

ใช้ AS ในการ Routing.

router ใน AS เดียวกันจะรันใน protocol เดียวกัน

เรียกว่า Intra-AS.

Gateway router. ใช้เชื่อม router ของ AS.



Intra คือ Inter AS.

เพราะเชื่อมทั้งภายในและภายนอก AS.

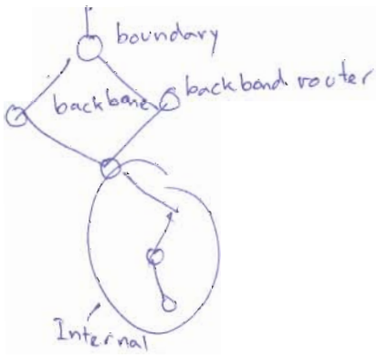
OSPF จะบอกข้อมูล 1 ครั้งต่อ 1 router.

มี security

multiple same cost path allowed.

เก็บได้หลายทาง.

hierarchical OSPF in large domain



BGP ใช้ได้ประมาณเท่าใช้เป็นมาตรฐาน

de facto standard

มี policy กับ reachability สิ้นที่สุด

เพื่อขาย เลือกตามปัจจัย

ไม่ forward ต่อ

Intra AS มีอีกชื่อคือ Interior Gateway Protocol (IGP)

มีหลายยี่ห้อ

RIP : Routing Information Protocol → DV, hop max = 15 hops

OSPF : Open shortest Path First. → LS

IGRP : Interior Gateway routing protocol.

คิดต่ออีกนิด ...

418351 Friday, September 19, 2008 ชื่อนายกฤษฎา [illegible] รหัสนิสิต 49210040

บันทึกช่วยจำ

*ICMP : Internet control mosgage PITC

ไปไกลออกเมื่อเกิด mes เป็นการ บ่งชัด อินเตอร์เน็ต

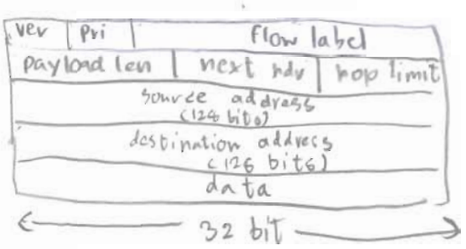
- ใช้ Host และ router ทำงานกับ ICMP ใช้ร่วมกับ Network Layer
- ส่งข้อมูลไปยังทาง router ระหว่างทางข้อมูลเกิดอะไรขึ้นบ้าง ไม่ได้ถึง ไม่รู้ว่า ตายตรงไหน
- data ของ Icmp มีข้อมูลอยู่ภายใน

Traceroute and Icmp

- ส่ง Uup ไปยังทาง ส่งไปที่ TTL = 1 , TTL = 2 เพิ่มไป 0 เรื่อยๆ
- ส่ง type 11 codes ตรง 3 pig เพิ่ม TTL เพิ่ม 3 เรื่อยๆ

ตรงนี้เกิด ① ไม่ถึงปลายทาง ② ไม่ถึง ③ บาง Host ไม่ใช้

IP V.6



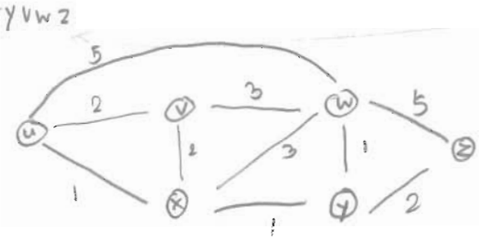
Tunneling

IP6 IP6 IP4 IP4 IP6 IP6

O O O O O

Dijkstra's algor

step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



ความดี : มากครับจะได้เรียนรู้

418351 Friday, September 19, 2008 ชื่อ นาย กิตติพงศ์ รักษา รหัสนิสิต 4921000973

บันทึกช่วยจำ

Robustness : ความอลวลของ Algorithm
LS : จะเก็บความยืดหยุ่นของสายสื่อสาร
DV : ข้อมูลของ Link ที่ยืดหยุ่นกับเส้นทางขึ้นอยู่กับ

เน็ตเร็กจะแบ่งออกเป็นส่วนย่อยๆ (autonomous system) สามารถใช้ Internet ภายในองค์กรของเราเองได้ แม้ไม่ได้ต่อ Internet

Gateway router : เชื่อมระหว่าง องค์กร
flooding : Broard cast ไปทั่ว Network

ข้อแตกต่างระหว่าง Transport vs. network
- network layer ส่งข้อมูลระหว่างเครื่องหนึ่งไปยังอีกเครื่องหนึ่งไปจนถึงปลายทาง
- transport layer มองแค่ process ของต้นทางไปที่ปลายทาง
  - congestion control : TCP ต้นทาง ควบคุม Data ไม่ให้ส่งครั้งเดียวไป
  - flow control : ควบคุม เมื่อไม่ให้ต้นรับปลายทางรับไม่ทัน
  - connection setup : ขอปลายทางก่อนว่าจะส่งหรือ และจะส่งที่ port ไหน

Dijkstra's algorithm

step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2, u	5, u	1, u	∞	∞
1.	ux	2, u	4, x		2, x	∞
2.	uxy	2, u	3, y			4, y
3.	uxyv		3, y			4, y
4.	uxyvw					4, y
5.	uxyvwz					

DST	PRE	COST
x	v	1
y	x	2
v	u	2
w	y	3
z	y	4

418351 Friday, September 19, 2008 ชื่อ ... รหัสนิสิต 4721012 3

บันทึกช่วยจำ

ICMP ...
Internet
- ... host(com) และ router ... ICMP ...
...
- ... ICMP ... ① ... ② code ...
type code
8 | 0 ...

Traceroute and ICMP
- ... UDP ... TTL=1, TTL=2
- ... type 11 code 0 ... TTL ...
... ① ... ② ... ③ ...

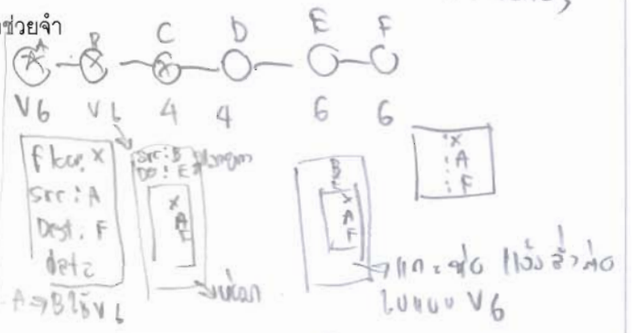
IPV6. ... fragment ...
... ...

ver | pri | flow Label
payload len | next hdr | hop lim
source add
destinati
data
(32 bit)

ver <4 bit บอก version>
priority, flow label ... priority ... packet ...
pay load len ... byte
next hdr ... Protocol ...

... IPV4
- checksum
- soure router ... list ...
- ... message type ... fragment

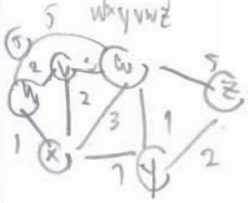
Transtion IPV4 to IPV6
- ... router ... delay ... route ...
- IPV4 ... IPV6 ... IPV4 ...
tunneling ...



① ... link state ... router ... Dynamic
② Decentralized ...
Static ⇒ routing table ... router ...
Dynamic ⇒ ... ได้ ...

Dijkstra's ... (ใช้ cost ...)
Ex ... u ... v, x, w
v ... x ... w
... node ...

step	N'	D(v),P(v)	D(w),P(w)	x	x	y	y	z	z
0	u	2y	5w	1w		∞		∞	
1	ux	2w	4x			2x		∞	
2	uxy	2u	3y					4y	
3	uxyv		3y					4y	
4	uxyvw							4y	
5	uxyvwz								



418351 Friday, September 19, 2008 ชื่อ จิรภาวุธ ขึ้นแล้ว รหัสนิสิต 4921013)

บันทึกช่วยจำ

ICMP Protocol ใช้ Msg ของทุก Network ใช้ระดับชั้น error preporting โดย host & Router เพื่อสื่อสาร ซึ่งอยู่เหนือชั้น ใน network leger

Tracerroute คือ ICMP ใช้ส่งข้อมูล จาก host ต้น ไป host ปลาย

- IP Router ไม่ can ทราบไป ถึง host ได้ ยกเว้น แต่ ส่ง msg type 3 ไปให้ host เพื่อแจ้ง error
- เมื่อ data ไปยัง router ชั้น n แล้ว TTL ของ ทุก ชิ้นใน แต่ละส่ง ICMP ไป ที่บอก type 11
- ใช้ หา ชื่อ และ ที่อยู่ ของ router ระหว่าง source & dest

IPV 6 32 bit ไม่พอ เพิ่ม header format ของใหม่ขึ้น 40 byte เพื่อลดขั้นตอน ใน IP Packet ให้เร็วขึ้น

ข้อแตกต่าง ไม่มี check sum เพื่อช่วยแก้ไขปัญหาในชั้น ขึ้นไป

IPv.4 กับ IPv 6 Tunneling

IPv 6 datagram ทั้งหมดใส่ไว้ใน field data ของ IPv.4 และ สมุติ address ของ IPv.6 ที่อยู่ห่างไกลกัน ก็ส่งไปใน network ก่อนๆ ไปเหมือนกับ IPv6 ที่ไม่มีอยู่ เหมือนกับว่าง node IPv 6 นี้ ที่ถูกต้องไม่ ทราง

ประเภทของ Runting Algorithm

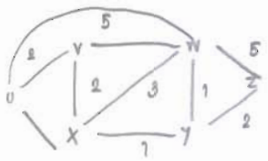
1. Global ทราบ cost ทั้งหมด ต้องทราบทุก node ทุก link
2. decentralized รู้แค่ว่า node ที่ติดอยู่กัน

ชนิด

1. static เปลี่ยนแปลงทาง เมื่อ node และ link เปลี่ยน admin เป็น ผู้ กำหนด
2. Dynamic เปลี่ยนแปลง เส้นทาง Auto

Dijkstra' Algorithm ค้นหาทาง network topology

และ link ทุกที่ทราบแล้ว เริ่ม ที่ node ทุก node ทำการ boradcast ให้ทุกๆ เครื่องที่อยู่ ของ เครื่องข่ายตอบกลับไปยัง node, router ทุกที่ทางใน network เรียก " Link state " เมื่อรู้ผลลัพธ์ทางแล้ว ก็ส่งไปยังเครื่องที่ Router ทำหน้าที่ สร้างเป็น routing table ที่ node ทุกที่ทาง ภายใน network



N'	D(v), P(v)	D(w), P(w)	x	y	z
u	2,u	5,u	1,u	∞	∞
ux	2,u	4,x		2,x	4,y
uxy	2,u	3,y			4,y
uxyv					4,y
uxyvw					
uxyvwz					

node ขั้นตอนที่ทำงานหาที่สั้น = n * (n+1) /2

worst cost $O(n^2)$

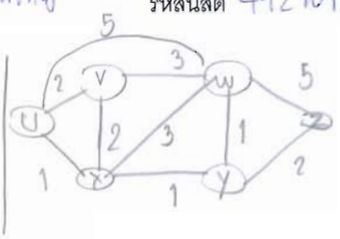
ข้อเสีย เมื่อ oscilation ที่ต้องการให้ทุกๆ ใช้ algorithm เดียวกัน execute ~~ทุกๆ node ในเครือข่าย~~ ทุกๆ ชั่วโมง

418351 Friday, September 19, 2008 ชื่อ น.ส.สุไรยวรรณ์ ชิมไทย รหัสนิสิต 4921015 6

บันทึกช่วยจำ

Dijkstra's algorithm

step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



CIDR (Classless InterDomainRouting)

- การจัดสรรแบ่งเครือข่าย IP ตามสถานที่ใช้เครื่องแม่ข่าย ตามตัวอย่างเดิมของ ISP ก็
- ข้อมูลเครือข่าย = Prefix และ เลขโฮสต์ = Suffix
เช่น 128.10.0.0/16

DHCP (Dynamic Host Configuration Protocol)
= Protocol ที่ใช้ในการกำหนด IP Add อัตโนมัติแก่ลูกข่าย ในระบบที่ติดต่อ TCP/IP
- DHCP sever = ส่งหน้าที่แจก IP ในเครือข่ายไปให้ลูกข่าย ตลอดเวลาที่เปิดเครื่อง

ขั้นตอนการต่อเครื่องลูกข่าย DHCP sever

① ลูกข่ายเครื่อง DHCP S_ ในเครือข่าย โดยส่ง DHCP discover เพื่อร้องขอ IP add_
② DHCP S_ จะค้นหา IP ที่ว่างอยู่ในฐานข้อมูล แล้วส่ง DHCP offer กลับให้ลูก
③ เมื่อลูกได้รับ IP ก็จะส่งสัญญาณตอบกลับ DHCP Request ในทันทีทันใด
④ DHCP S_ ส่งสัญญาณ DHCP Ack ให้ลูกเพื่อแจ้งว่าใช้งานได้

*** NAT (Network Address Translation)**

- ตารางแปลง IP บนสุดๆ ต่างที่ใช้ สามารถในเครือข่ายนี้ติดต่อกับเครือข่ายอื่นโดยใช้ IP เดียวกัน

ขั้นตอนการทำงาน → เมื่อ NAT เริ่มทำงานจะสร้างตารางการแปลงไว้ให้สำหรับเก็บข้อมูล IP add_ ของเครื่องในเครือข่าย ตรงในส่ง packet ผ่าน NAT device และกำหนดขั้นตอนการเข้าไปเก็บหมายเลข Port ที่ถูกใช้ไปโดย outside IP add_ เพื่อการส่ง Packet จากเครือข่าย ใน → นอก ดังนี้

① บันทึกข้อมูล Source IP add_ และ S-port number ไว้ในตารางที่เตรียมไว้
② แทนที่ IP ของ packet ด้วย IP จากของ NAT d_ ของ
③ assign เลข Port ใหม่ใน Packet และบันทึก ค่า Port นี้ไว้ใน ตร. และกำหนดค่าที่ลงใน S-p_ ของ packet นั้น
④ ขั้นนี้จะคำนวณหา IP, TCP checksum อีกครั้งเพื่อตรวจสอบว่าถูกต้อง

* NAT → ป้องกันไวรัสเข้าไม่ได้
* N → แก้ปัญหาการขาดแคลน Address

เก็บของ

418351 Friday, September 19, 2008 ชื่อ นายอภิเดช ทศกัณฑ์ รหัสนิสิต 4921016-4

บันทึกช่วยจำ

Transport services and protocols
- ให้ segments
- ใช้ TCP, UDP

Transport vs network layer

เลือกรับส่งกับโปรแกรม

network - Host

Transport - processes

Internet transport
in order (TCP)
congestion control ควบคุมความคับคั่ง
flow control ควบคุมการส่ง
in order (UDP)

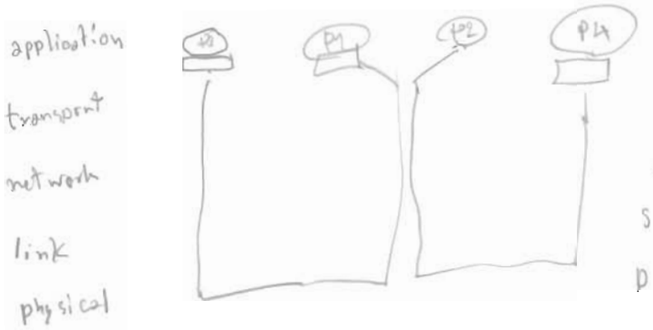
มัลติเพล็ก

multiplexing / demultiplexing

จะมีเลขหมาย (socket)

	N'	D(v),P(v)	D(w),P(w)	D(x),P(x)	D(y),P(y)	D(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	4,y
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					
5	uxyvwz					

UDP socket
source
ดาย dest ip & port number



host1 Host2

418351 Friday, September 19, 2008 ชื่อ นายศักดิ์ชัย วิสุทธิ์ยนต์ รหัสนิสิต 49210255

บันทึกช่วยจำ

Network Layer

ส่ง segment จากต้นทาง → ปลายทาง

① forwarding

② routing การหาเส้นทางของ Packet จะส่งถึงปลายทางได้อย่างไรบ้าง

transition From IPv4 To IPv6

- ไม่สามารถเปลี่ยน router ทุกตัวในเวลาเดียวกันได้ จะมี delay ระหว่าง router ต่อ router ดังนั้นไม่สามารถเชื่อมต่อกันได้

- IPv4 เปลี่ยนเป็น IPv6 ก่อนๆ เปลี่ยนไปใช้ Tunneling (อุโมงค์)

ทุก router ต่อเข้ากับ router table costs: เพื่อดูว่าไปใน link มี costs เท่าไหร่ หรือ (ค่าใช้จ่าย)

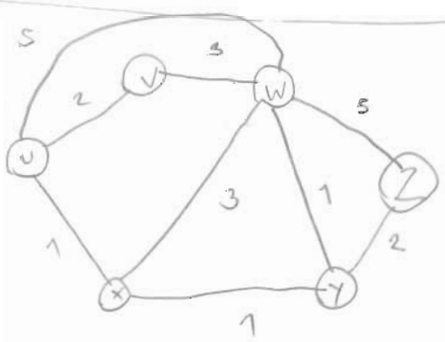
Routing Algorithm classification

① Global หรือ decentralized ขึ้นอยู่กับการเชื่อมต่อของ

- router ทุก router เชื่อมต่อกับ network

② Decentralized

- router ไม่จำเป็นต้องเชื่อมต่อกับทุกๆ ตัว แต่จะใช้ข้อมูลจากเพื่อนบ้านที่เชื่อมต่อกับตัวเองเท่านั้น



step	N'	D(v), P(v)	D(w), P(w)	D(x), P(x)	D(y), P(y)	D(z), P(z)
0	u	2, u	5, u	∞	∞	∞
1	uv	2, u	4, x			
2	uxy	2, u	3, y			
3	uxyv		3, y			
4	uxyvw					
5	uxyvwz					

* ความดี * รถน่าซื้อไม่

418351 Friday, September 19, 2008 ชื่อ นลดองกมล แก้วศรี รหัสนิสิต 49210271

บันทึกช่วยจำ

ICMP โปรโตคอลที่ใช้ Message ควบคุม Network ทำให้เกิดกรณีที่มี error reporting ถูกใช้โดย hosts & router เพื่อสื่อสารข้อมูลที่เกิดขึ้นใน network layer

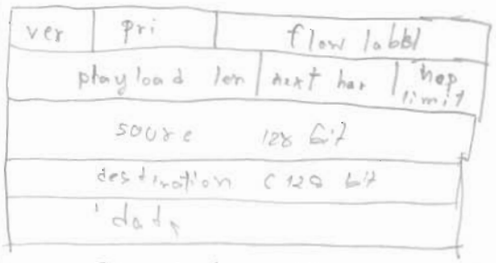
type code

8 0 echo request (ping)

Traceroute and ICMP ใช้งานได้จาก host ทำงานไปที่ host ปลายทาง

- IP router ไม่สามารถหาที่อยู่ host ได้ สร้าง และส่ง msg type 3 ไปให้ host เพื่อแจ้งว่าหาไม่เจอ
- เมื่อ Data ไปถึง router ที่ทำหน้าที่ TTL เป็นศูนย์ (หรือใกล้ศูนย์) ICMP ไปยังต้นทาง (type 11)
- ใช้งานร่วมกับฟังก์ชันของ router ในการค้นหาเส้นทางของ packet

IPv6 32-bit ไม่พอใช้งานต่อไปในอนาคต header format ของใหม่ ใช้งานได้ 40 byte เป็นค่าคงที่ ทำให้การประมวลผล IP packet ได้เร็วขึ้น



ข้อเสียคือ ไม่มี checksum เพื่อหาความผิดพลาดได้เร็ว เพิ่มความเร็วในการประมวลผลที่ router ไม่ต้องคำนวณ checksum ใหม่ทุกครั้งที่ผ่าน router ไม่มีการ fragment ที่ router ต้องใช้ ICMP ในการแจ้งขนาดที่ใหญ่ที่สุดที่ path รับได้

การย้าย IPv4 ไป IPv6 ใช้ Tunneling

IPv6 datagram ที่ผ่านไปใน field ข้อมูลของ IPv4 หน้าที่ของ address IPv6 ต้องทำงานร่วมกับ node ตามในอุโมงค์ IPv6 ต้องรู้ด้วยว่าต้องส่งต่อ IPv6 ต่อไปยัง node ถัดไปในอุโมงค์ IPv6 ต้องรู้ว่าจะต้องส่งต่อ IPv6 ออกจากอุโมงค์ IPv6 ได้อย่างไร

Global การจัดเส้นทางแบบ decentralized ผู้ดูแลระบบ

Static เปลี่ยนแปลงเส้นทางเมื่อ node เปลี่ยนไป ต้อง admin ทำงาน

Dynamic เปลี่ยนแปลงอัตโนมัติ

Dijkstra หาจาก Network topology และ link cost ทั้งหมด และเป็น node ต่อ node ทำงานแบบ broadcast ต้องมีการแจ้งข่าวสารเกี่ยวกับเส้นทางไปยัง node ทุกๆ node ในเครือข่าย

N'	D(v), p(v)	w	x	y	z
u	2,u	5,u	1,u	∞	∞
ux	2,u	4,x		2,x	∞
uxy	2,u	3,y			4,y
uxyv		3y			4y
uxyvw					4y
uxyvwz					

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

การคำนวณต้องการข้อมูลเกี่ยวกับค่าใช้จ่ายของการเชื่อมต่อทั้งหมดในเครือข่าย ทำงานแบบรวมศูนย์ ต้องรู้ node ทุก node ทำให้ต้องมีการ set ข้อมูลที่ cost ต่ำสุดเพื่อไปยังโหนดอื่นๆ ในแต่ละรอบ ทำให้ต้องมีการคำนวณรอบ ถ้ามี n โหนด ต้องมีการคำนวณ n-1 รอบ ในแต่ละรอบต้องส่งข้อมูลไปยัง node ที่เชื่อมต่อกันทั้งหมด ถ้ามี n node ต้องส่งข้อความ n-2 node ไปเรื่อยๆ จนกว่าจะครบทุก node ที่ต้องการ

ต้องคำนวณทั้งหมด $= n*(n+1)/2$

อยู่ใน worst case $= O(n^2)$

ข้อเสีย อาจจะเกิดการ Oscilation ต่อการกระทำหนึ่งๆ ในการใช้ algorithm ในหนึ่งๆ node ไม่ทำการ execute พร้อมกันทุกรอบ

Distance Vector Algorithm

$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

ตารางดี วิชาเพื่อนมากเรียน

19/09/08

418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต 49210297

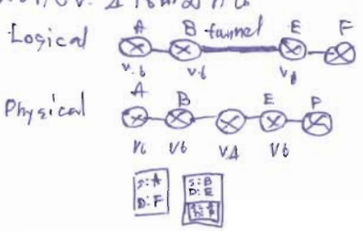
บันทึกช่วยจำ

- ICMP : Internet control message Protocol
  - ใช้บอก error ตามหลังบอกว่าเครื่องปลายทางเปิดหรือไม่ - ใช้ request / reply โดยการ ping (ส่งไปกลับ)
- Traceroute and ICMP
  - หาเส้นทางที่ใกล้ที่สุด ต้น→ปลาย - ทำส่งไปที่ละเพจ: TTL epired จะส่ง ICMP เป็น message error
- IPv.6
  - v.6 มี 128 bit v.4 มี 32 bit - ลด TTL ที่ header (มี length 40 byte) - ไม่รับ fragmentation

Version v.6	priority	
ความยาว	ของ Protocol TCP=6	hop time
source		
destination		
data		

ไม่มี checksum กับ option

v.6 กับ v.4 ใช้ร่วมกันได้



Static แก้ยาก ถ้ามีข้อมูลผิดต้องแก้

Dynamic ปรับปรุงได้เปลี่ยน, ใน cost อันใดอันหนึ่งเสีย

- Graph abstraction : costs บ่อยมี
  - c(w,z) = 5

- Routing Algor classification
  - Global มีศูนย์กลางบอกรายชื่อของ - Decentralized : รู้แค่ตัวใกล้ข้างใน
  - distance vector ระยะทาง

- A link-state Routing Algor
  Dijstra's algor.
  - ทราบราคา All ของ link state broadcast ให้ทุก node - ทราบ cost บ่อยครั้ง
  - C(x,y) ค่า cost ระหว่าง x,y
  - D(v) ค่าระยะรอบ ของ cost ทุก link → v
  - P(v) หน้าก่อนของ v
  - N' cost น้อยครั้ง

step	N'	D(v), P(v)	D(w), P(w)	ทำที่เชื่อมไปไม่ได้ใน ∞ ไปก่อน
	u			

- ทุก router ทำมาจากเปลี่ยนแปลงได้ตาม cost O(n log n)
- OSPF ส่งไปหมด

418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต 4[illegible]

บันทึกช่วยจำ

Distance Vector ... Link-state

จะใช้ข้อมูล router ใกล้เคียงเท่านั้น ไม่ต้องสนใจทั้งหมด แต่จะไม่รู้ว่าเส้นทางไหนสั้นที่สุด ถ้าได้เส้นทางไปแล้วก็จะใช้เส้นทางนั้นไปเรื่อยๆ

OSPF (open shortest Path first)

ใช้ Link-state Algor

จะอัพเดตให้ทั้งหมด 1 entry per neighbor router

Link-state Dijstra's Algorithm

ตารางเป็นแบบนี้

step	N'	D(v), p(v)	w	x	y	z
0	u	2,u	5,u	1,u	∞	∞
	ux	2,u	4,x		2,x	∞
	uxy	2,u	3,y			4,y
	uxyv		3,y			4,y
	uxyvw					4,y
	uxyvwz					

Internet - transport-layer Protocols

TCP (จะเป็นเมื่อข้อมูลที่เข้ามาถึงก่อน) เป็น in-order delivery

- congestion control ควบคุมปริมาณ
- flow control
- สร้างการเชื่อมต่อ

UDP : ไม่น่าเชื่อถือ unreliable, unordered delivery

- no frill's extension of "best-effort" IP

logic

network layer คือ เชื่อมต่อระหว่าง host

transport layer คือ การเชื่อมต่อระหว่าง process

ทำดี : สร้างผลงานที่ดีที่สุดเท่าที่จะทำได้

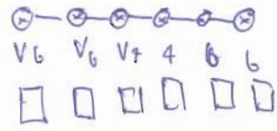
418351 Friday, September 19, 2008 ชื่อ นายปฏิญญา ยังรู้ รหัสนิสิต 49210420

บันทึกช่วยจำ

Transition From IPv4 To Pv6

ไม่สามารถเปลี่ยน router ทุกตัวในครั้งเดียวแต่จะมี delay เช่น ที่นี่ 1 router ดูแลกันได้ แต่จะเชื่อมกันอยู่ V4 เปลี่ยน V6 ล้อมรอบเปลี่ยน โดยใช้ Tunneling

Tunneling



ทุก Router ต้องมีตารางที่ Router table costs น้อยที่สุดไป 1 link cost เท่านั้น

1 ตาราง Link มี cost เดียว

Routing Algorithm classification

1 Global - ข้อมูลที่ใช้ทั้งหมดเป็นข้อมูลรวม

- router ทุก router เก็บตารางข้อมูลของทั้ง network link state ต้องรู้จนหมดว่า router ตัวอื่นเชื่อมอยู่ใช้แบบ dynamic

2 Decentralized

- router ไม่มีข้อมูลที่ถูกต้องเป็นของตัวเองเท่านั้น ต้องใช้ประสบการณ์ข้อมูลจากเพื่อนบ้านมาตัดสินใจเลือกเส้นทาง [distance vetor ระยะทาง (cost)]

static → routing table จะไม่เปลี่ยน เว้นแต่ router เชื่อมกันเสียหาย

dynamic → ต้องมีการเปลี่ยนแปลงตารางไปเรื่อยๆ ตลอดเวลา

Priority
Flow label } ใช้ร่วมกันบางเครื่องจะเลือกอย่างใดอย่างหนึ่งได้ packet ขนาดใด

Payload len ความยาวของ Data มีหน่วยเป็น byte

next hdr data ที่ใช้ Potocal อะไร เพราะมีหลายตัวเลือก

link state dynamic

NAT

ตัดการเปลี่ยนแปลงของ Ip Adress

UPP ส่งข้อมูลมั่วๆ

TCP จัดเรียงข้อมูล

ข้อมูลที่ส่งมาใน Network leicer จะเปลี่ยนไปที่ Transport layer

Ipv6 128 bit ไม่ขอยังไม่ใช้ Ver ใหม่

จดหน้าต่างไป

418351 Friday, September 19, 2008 ชื่อ ปรัชญา คนอมหาด รหัสนิสิต 49210438

บันทึกช่วยจำ

IP Add

มีขนาด 32 bit เช่น 11111111.
IP แบ่งออกเป็น 2 ส่วน — ส่วนหน้าเรียก network part
ส่วนหลังเรียก host part

class A /8 0.0.0.0 → 127. 225.255. 255
class B /16 128.0.0.0 → 191. 255. .
class C /24 192.0.0.0 → 223, "
class d /32 224.0.0.0 → 239, "

192.168.100.0/27 → อยู่ใน คลาส C/19 เพราะว่า ยืม 3 bit

Net work IP = 192.168.100.111 xxxxx
Subnet mask = 255-31 = 224 = 225.255.255.224
จำนวน subnet = 2^{n} bit ที่ยืม − 2 (จำนวนที่เป็นไปได้) = $2^3 - 2 = 6$ subnet
จำนวน host = $2^{5} - 2 = 30$ host
Broadcast IP = 192.168.100.255
Range host IP

subnet zero	192.168.100.0	- 192.168.100.31	
1	"	.32 - "	.63
2	"	.64 - "	.95
3	"	.96 - "	.127
4	"	.128 - "	.
5	"	.160 - "	.
6	"	.192 - "	.
Broadcast	"	.224 - "	.255

subnet mask = ตัวเลข bit ด้านซ้ายมือของหลัง bit ที่ยืมมาเป็นหมด

CIDR

- เป็นการอ้างอิงหมายเลข IP ตามมาตรฐาน โดยใช้เครื่องหมาย /(slash) ตามด้วยขนาดของมาสก์
- Tag เลขเครือข่าย = Prefix และเลขโฮส = suffix
เช่น 128.10.0.0/16

* DHCP

คือ โปรโตคอลที่ใช้ในการกำหนด IP Add อัตโนมัติ แก่เครื่องลูกข่ายบนระบบที่ติดตั้ง TCP/IP

- DHCP sever → มันจำที่ แจก IP ในเครือข่าย ไม่ให้ซ้ำ เป็นการลดความซ้ำซ้อน

ขั้นตอนการทำงานของเครื่องลูกข่าย DHCP S—

1) ลูกข่ายเครื่อง DHCP S— ในเครือข่าย โดยส่ง DHCP discover เพื่อร้องขอ IP add
2) DHCP S— จะค้นหา IP ที่ว่างอยู่ในฐานข้อมูล แล้วส่ง DHCP offer กลับให้ลูก
3) เมื่อลูกได้รับ IP ก็จะส่งสัญญาณตอบกลับ DHCP Request ให้แม่ทราบ
4) DHCP S— ส่งสัญญาณ DHCP Ack ให้ลูกเพื่อยืนยันว่าเริ่มใช้งานได้

NAT

- สามารถแปลง IP หลายๆ ตัวที่ใช้งานในเครือข่ายให้กลายเป็น IP เดียวกัน

ขั้นตอนการทำงาน

เมื่อ NAT เริ่มทำงาน มันจะสร้างตารางขึ้นมาสำหรับเก็บข้อมูล IP add ของเครื่องในเครือข่าย ที่ผ่าน NAT device และตารางนี้จะมีการสร้างตารางใส่เลขหมายเลข port ที่ถูกใช้ในโดย Outside IP add เมื่อมีการส่ง packet จากเครือข่ายใน → นอก NAT จะ

1) บันทึกข้อมูล source IP add และ S— port number ในตารางที่เกี่ยวข้อง
2) แทนที่ IP ของ packet ด้วย IP ของ NAT ด— ไอ
3) assign เลข port ในตารางให้ packet และบันทึก ค่า port นี้ไว้ในตาราง เพื่อตามหาค่าที่ส่งไป ว— p— n ของ packet นั้น
4) จากนั้นจะคำนวณหา IP, TCP checksum ขึ้นใหม่อีกครั้ง เพื่อตรวจสอบความถูกต้อง

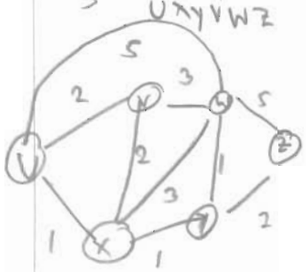
NAT → ยังคงไว้ซึ่งไม่ได้

" → แก้ปัญหาการขาดแคลน Add

418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต [illegible]

Dijkstra's algorithm : example บันทึกช่วยจำ

step	N'	c(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Bellman-Ford example

$d_u(z) = \min\{c(u,v) + d_v(z),\ c(u,x) + d_x(z),\ c(u,w) + d_w(z)\}$

$= \min\{2+5,\ 1+3,\ 5+3\} = 4$

Internet transport layer protocols.

Tcp ให้บริการที่เชื่อถือได้ ให้ in-order delivery - congestion control
- flow control
- connection setup

UDP ตรงข้ามกับ Tcp. เป็น unordered delivery
- no-frills extension of "best-effort" IP

Transport

ก็ยังมี Traffic ของ cost ที่ขึ้นอยู่กับ

แต่ต่างกันที่ LS ส่ง message เพื่อหาข้อมูลจากทั่วๆ DV

DV ที่แก้ปัญหาของ ที่ใช้งานได้ยากขึ้น

- count to infinity problem

Robustness : ความสามารถของ Algo

LS = มีการตรวจสอบการทำงานของตัวเอง

DV = ถ้าของ link ที่ต่อกับเราเสียไป

network ที่เราเชื่อมต่อไปยังๆ (autonomous system) ของ Internet จะรู้ได้

ว่า ที่เราเชื่อมต่อไปนั้นมีการทำงานยังไง เช่น) ของ ใช้

418351 Friday, September 19, 2008 ชื่อ น.ส.พัชรพรรณี เหลืองสอาดกุล รหัสนิสิต 4921051l

บันทึกช่วยจำ

DHCP (Dynamic Host Configuration Protocol)

คือ โปรโตคอลที่ใช้ในการกำหนด IP address อัตโนมัติแก่เครื่องลูกข่ายในระบบที่ใช้ TCP/IP

- DHCP server → มีหน้าที่แจก IP ในเครือข่ายโดยอัตโนมัติ ในกรณีลดความซ้ำซ้อน

NAT (Network Address Translation)

- สามารถแปลง IP หลายๆ ตัว ที่ใช้ในเครือข่ายใน ก็สื่อต่อกับเครือข่าย ที่ใช้ IP เดียวกัน

ขั้นตอนการทำงาน

เมื่อ NAT เริ่มทำงานขึ้น จะสร้างตารางขึ้นมาเพื่อเก็บข้อมูล IP address ของเครื่องในเครือข่ายที่ส่ง packet ผ่าน NAT device เช่น จากนั้นจะกำหนดเลข port ที่ถูกใช้ใน outside IP address เพื่อแทนการส่ง packet จากเครื่องใน ไปยัง NAT

① บันทึกข้อมูล source IP address และ S_port number ไว้ในตารางที่เตรียมไว้

② แทนที่ IP ของ packet ด้วย IP ของ NAT และ S_port

③ assign เลข port ใหม่ ให้ Packet ที่จะออกไป

ICMP : Internet control Message Protocol

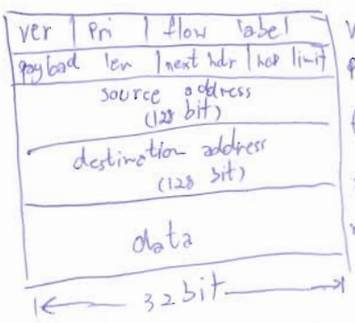
- ใช้ โดย host(com) และ router ทำงานร่วมกับ ICMP เพื่อสื่อสารใน network layer
- ส่งข้อมูลผิดพลาด กลับ router ตามเส้นทางเดียวกับ
  ข้อมูลกลับ ถูกต้อง → ไม่มีปัญหา
  ผิดพลาด → ส่งข้อความแจ้งกลับ
- data ของ ICMP ที่อยู่ภายใน → ชนิดของข้อมูล, code คือ ข้อมูลในส่วนของ

Traceroute and ICMP

- ส่ง UDP ไปยังปลายทาง โดยที่ TTL=1, ①, TTL=2 ②
  ถึงที่หมาย ก็จะส่งข้อความกลับมา
- ส่ง type และ code ของ ping เพิ่ม TTL ขึ้น 3 ไปเรื่อยๆ จนกว่าจะได้ผลลัพธ์ 1. ไม่ถึงปลายทาง 2 ถึงแล้ว 3 ปิด host ไม่ให้

IPv6

ออกแบบมาเพื่อรองรับปัญหาการใช้งานอินเทอร์เน็ต, ไม่รองรับการ fragment



Ver (4bit) ของ version
Pricrity } ใช้ร่วมกันในการจัด priority ของแต่ละ
flow label } ลำดับการส่ง packet นั้นๆ
payload len คือ ความยาวของ data
next hdr data ชั้นถัดไป ถ้าไม่มีจะเป็นค่าว่าง

Transition From IPv4 to IPv6

- โปรแกรมเปลี่ยน router ทุกตัวในวันเดียว, ทำให้เกิด delay ได้ คือ ทั้ง 2 router ต้องใช้ IPv4 เหมือนกันอยู่
- IPv4 เปลี่ยนเป็น V6 ต้องใช้ tunneling (อุโมงค์)

Routing Algorithm classification

1. Global หรือ decentralized
   - router ทุก router เข้าใจภาพรวมของทั้ง Network
2. Decentralized
   - router ไม่จำเป็นต้องเข้าใจภาพรวมของทั้งหมด ใช้วิธีการแลกเปลี่ยนข้อมูลกับเพื่อนบ้าน ค่อยๆ เรียนรู้ และเลือกเส้นทางที่ดีที่สุด

static → routing table ของแต่ละ router จะไม่เปลี่ยนแปลงตามเวลาที่ผ่านไป

Dynamic → ค่าในตารางเปลี่ยนแปลงตามเวลาที่ผ่านไป หรือตามเหตุการณ์

A4 link - state Routing

Dijkstra's Algorithm ทำงานทั้งหมดทุก router

418351 Friday, September 19, 2008 ชื่อ ภูชณตาธรณ์ สอนชิน รหัสนิสิต 4921056C

บันทึกช่วยจำ

Virtual circuits ส่วนมากจะใช้กับโทรศัพท์

Virtual circuits : signaling protocols

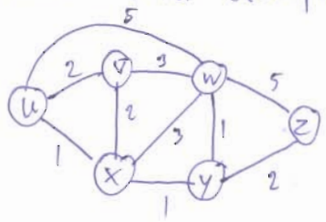
- ต้นทาง → ปลายทาง จะต้องบอกปลายทางก่อนว่าจะส่งไป แล้วปลายทางตอบรับมาก็เริ่มส่งได้
- ใช้ ATM (Protocol ชนิดหนึ่ง), frame-relay, X.25

Datagram networks

ไม่ต้องให้รู้ อยากส่งก็ส่งเลย ทางฝั่ง routers router ไม่รู้สถานะของเส้นทางปลายทาง

- ในแต่ละ packets forwarded ด้วย destination host address ว่าเป็น IP อะไร ก็ส่งไปตามนั้นเรื่อยๆ

Bellman – Ford example รูปแต่ละเส้นทาง



หาเส้นทางที่น้อยที่สุด

Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

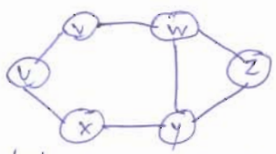
B-F equation says

$$d_u(z) = \min\{c(u,v) + d_v(z), c(u,x) + d_x(z), c(u,w) + d_w(z)\}$$

$$= \min\{2+5, 1+3, 5+3\} = 4$$

Dijkstra's algorithm :

เพื่อคำนวณ ดูว่ามี node ใดบ้างติดกับ node A บ้าง

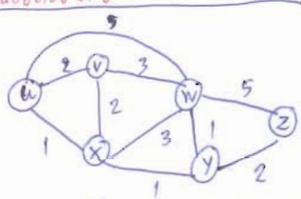


destination	link	
v	(u, v)	①
~~x~~	(u, x)	②
y	(u, x)	③
w	(w, x)	④
z	(u, x)	⑤

step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)
0	u	2, u	5, u	1, u	∞
1	ux	2, u	4, x		2, x
2	uxv	2, u	3, y		
3	uxyv		3, y		
4	uxyvw				
5	uxyvwz				

สิ้นสุดตัวที่น้อยดีของมันบ

D(z), p(z)
∞
∞
4, y
4, y
4, y



1, u หมายถึงไปถึง x ได้โดยขณะก่อนหน้าเป็น u ไปถึง x โดยขณะที่น้อยที่สุดคือ u

* หา cost ที่น้อยที่สุด cost เป็น ∞ เพราะไม่ได้เชื่อมอยู่โดยตรง

Distance Vector Algorith

$$d_x(y) = \min\{c(x,v) + d_v(y)\}$$

การเปลี่ยน IPv4 To IPv6 ไม่สามารถเปลี่ยนทุก router ได้พร้อมกัน

- ทุกเครื่องของ network ต้องเป็น version เดียวกัน

Tunneling การใช้ Tunnel (อุโมง) ทำให้ IPv4, IPv6 เชื่อมถึงกันได้

network ate อยู่ห่อรอบ

network call network ที่ช่วยทำ routing

ลาก่อนดี ขับรถดูกกฎจราจร

ตื่นเช้า

418351 Friday, September 19, 2008 ชื่อ น.ส.วิมล กฤษณเพ็ญ รหัสนิสิต 49210586

บันทึกช่วยจำ

Router Architecture Overview

ประกอบด้วย 2 router functions

1. routing algor / Potocol เช่น RIP, OSPF, BGP
2. forwarding datagram จาก link นอก.

ส่วนประกอบใน router มี 3 ส่วน

1. input 2. output 3. switch.

Input Por Functions

* queueing เป็นตัวหักเพื่อรอ process ที่ให้ข้อมูล delay ถ้าเต็มจะทิ้งไป.

• Input Port Queuing

- เกิดปัญหา (HOL) = Head-of-the-line เกิดที่หัวแถว
- เกิดการรอ (queuing delay)

* ถ้าการเคลื่อนไม่ได้เร็วเท่าการเข้าของ ให้รอในแถวก่อน ที่ไม่ packet ที่จะส่งต่อก็จะไม่ได้ เพราะติดที่หัวแถว เรียก "queuing delay" เช่น รถติด.

• Output Port จะเหมือนกับ input ports

- มี Q ขาดเร็ว Router ที่ความเร็ว
- มัก Q ขาoutput เต็มข้อมูลจากหายไป.

• Subnets

- ถ้า subnet part เหมือนกัน แสดงว่าอยู่ใน Network เดียวกัน.

• IP Addresses : How to get one?

ที่เครื่องจะได้ IP Add ความคิดนี้ มี 2 วิธี

1. ไปขอ set เอา (Manual)
2. แบบ Auto คือมี Protocol กำหนดไว้เป็นแบบปรับเอง Host Dinamic

NAT = router

NAT ที่การเปลี่ยน IP ให้ข้อมูลที่ส่งให้หมด สามารถพิมพ์ได้ถูก.

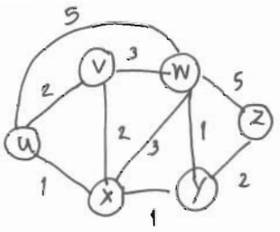
ลักษณะที่ NAT

- เมื่อได้ IP มาอันเดียว แต่อยากใช้กับทุก Add.
- เปลี่ยน IP ในแลนจากได้โดยไม่ต้องบอก

* NAT แต่ปัญหาเมื่อมี Add ไม่พอ.

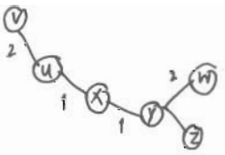
Dijkstra's algorithm :

step	N'	D(v), P(v)	D(w), P(w)	D(x), P(x)	D(y), P(y)	D(z), P(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



ต.ไส้.

DST	PRE	COST
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4



Traceroute and ICMP

- ส่ง UDP ไปปลายทาง
- คำนวณที่ส่งทางไส้
- ผลที่ตอบกลับคือ "ถึงไปไม่ถึง" (host unreachable)

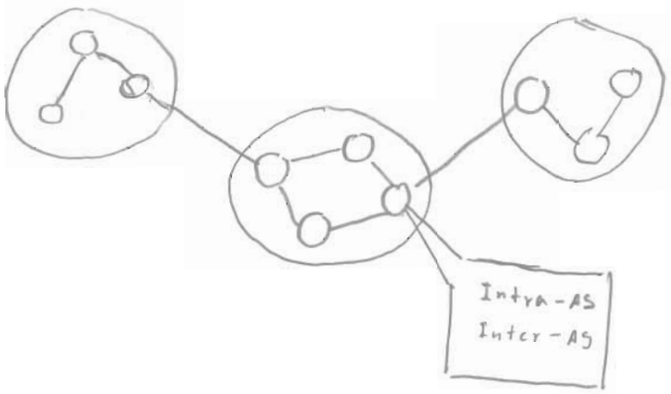
IPv.6 ใช้ v.4 มี 32 bit ไม่พอ.

418351 Friday, September 19, 2008 ชื่อ นายปรเมธา์ จิรมหรักษ์ รหัสนิสิต 49210594

บันทึกช่วยจำ

Interconnected ASes

* AS : Autonomous System



ภายในต้องใช้ Protocol เดียวกัน
Inter AS ภายในใช้ Protocal อย่างหนึ่งก็ได้

RIP (Routing Information Protocal)

- เป็นของ UNIX
- ใช้ได้ไม่เกิน 15 Hops
- ไม่มีระบบ Security

OSPF (Open Shortest Path First)

- มีระบบ Security

~~BGP (Border Gateway Protocol)~~

418351 Friday, September 19, 2008 ชื่อ วสิษฐ์ ศรีรัชญ์ รหัสนิสิต 4921064

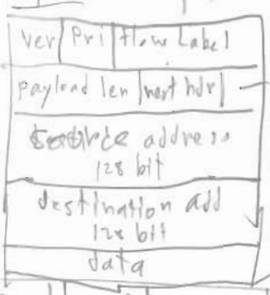
บันทึกช่วยจำ

ICMP: Internet control Message Protocol ใช้ทำการส่งข้อความที่เกี่ยวกับ message เป็น control
- ใช้โดย host (dom) หรือ ⑤ ทำงานร่วมกันของ ICMP ใช้ร่วมกันเพื่อหาข้อมูลใน Network internet
- ใช้ Data ในการส่ง ⑤ : แบ่งเป็น 2 ประเภท คือ ข้อมูลที่ต้องการ ← ใช้ในการตรวจสอบ, ใช้ในการแก้ไข
- data ของ ICMP มี Data อยู่ส่วนใน ขนาดของ data
code คือ แจ้งให้ทราบถึงความผิดพลาด

Traceroute and ICMP
- ใช้ UDP ในการส่ง ① ใช้ TTL = 1 ② TTL = 2 คือ ค่าที่ใช้แจ้งเมื่อข้อความไปยังตำแหน่ง
- ใช้ type code ของ ping เพิ่ม TTL ไปเรื่อยๆ (ความหมายคือทางใกล้)

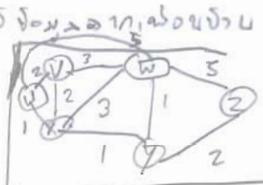
ผลที่ได้ คือ ใกล้ชิด, ช้า host ใช้ได้

ข้อดีของ IPv6 ที่เหนือกว่า คือ - ใช้เร็วกว่า
- Source router
- V6 เพิ่ม message type ใช้ใน fragment



Routing Alg.. classification
1) Global หรือ decentralized - router ทุก r. เป็นการรวบรวมข้อมูลของ network
2) Decentralized - ⑤ ใช้คำนวณเส้นทางที่ดีที่สุดโดยใช้ข้อมูลจากเพื่อนบ้าน
ที่มีลักษณะการคำนวณที่ทำซ้ำๆ

static routing table : ใช้เปลี่ยนเส้นทางโดย ⑤ ระดับต่ำ



DST	PRE	cost
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4

Ex u ต่อกับ v, x และ
v — w, x
⋮
ทุก node ใช้วิธีเดียวกันหมด

d(x,y) คือ cost ของ link ที่เชื่อม x, y
D(v) : ค่าใช้จ่ายของเส้นทางที่ดีที่สุด
P(v) : ตัวก่อนหน้าบนเส้นทางที่ดีที่สุดถึง v
N : เซตของ node ที่ cost น้อยที่สุด

Dijkstra's

Step	N'	D(v), P(v)	D(w), P(w)	D(x), P(x)	D(y), P(y)	D(z), P(z)
0	u	2, u	5, u	1, u	∞	∞
1	ux	2, u	4x		2, x	∞
2	uxy	2, u	3y			4, y
3	uxyv		3y			4, y
4	uxyvw					4, y
5	uxyvwz					

418351 Friday, September 19, 2008 ชื่อ นาย.อิสระ นามสกุล รหัสนิสิต 4921065?

บันทึกช่วยจำ

IP datagram format.

① Ver ตัวนี้ v.4 [4 bit ระบุเป็น version ที่ตรงกับตัวเราใช้ version เดียวกันเพราะถ้าคนละ version กัน v. ไม่รู้จักกันใช้ไม่ได้]
② headleten ของ datagram เป็น byte <ส่วนใหญ่ 20byte>
③ type of service บริการพิเศษ <เฉยๆ>
④ length <16 bit> ความยาว packet <65535 byte>
⑤ 16bit identifier IP ของ fragment
⑥ flgs ธงแสดงสถานะ
⑦ fragment offset ลำดับที่ของ data เดิม
⑧ time to live packet ของ router ตัวหนึ่งวน loop ทิ้งเลยกันเสีย
⑨ upper layer ชั้นที่สูงกว่า router <ชั้นของเรา> ส่งมาให้
⑩ header checksum ตรวจ header ของตัวเองในแต่ละครั้งที่ router รับมาทำงาน

①	②	③	④
⑤	⑥	⑦	⑧
⑨	⑩	⑪	
32 bit source			
32 bit destin			

ไม่ได้ตรวจสำหรับ message ที่เป็น control Internet.

ICMP : Internet control message protocol
- ใช้โดย host (com), และ router ทำงานกับ ICMP.
- ส่งข้อมูลไปยังตัวเราเมื่อ router รับมาไม่ได้

IPV6 32 bit ไม่พอใช้ของเดิม ที่เหมือน IPv. ใหม่ ตรงกันข้ามกับ v.4 { ไม่ checksum
v.6 ไม่มี masage types ไม่ทำ fragment

- source router => router ต้องรู้ว่าต่อไปต้องไปทางไหน เส้นทางถูกเก็บอยู่ใน list หรือ <ตัวตามไป list. ของ router ต่อๆ ใน list นั้น>

Tunneling
* ถ้า v.6 ใช้ของ v. 4 ต้องส่งไปที่ v. 6 ตัวต่อไปโดยตรงไม่ได้ ต้องส่งไป v.4 ก่อนที่จะถึง v. 6.

Tunneling ดูโดยรวม.
- ทั้งคู่ router ดูแลกันไม่ได้ ต่างเครือข่ายกัน.

A-Link-State Routing

Dijkstra's algorithem. => ก่อนการทำงานต้องรู้ router ใน network และ cost ต่อกันของทุก router.
ต้องมีการแพร่กระจายข้อมูลที่ดีที่สุด.

Ex บางเส้นทางต้องผ่าน V > x, w
V ~~~~~ x, w.
ทุก node ใช้งานเหมือนกันหมด

C(x,y) : cost ของ link ที่เชื่อม x,y
D(v) : ค่าระยะทางจากไปยัง
P(v) : ตัวก่อนหน้า (ตัวสุดท้ายที่ไปยัง v
N' : เส้นทางที่ cost ต่ำสุดแล้ว

Step	N'	D(v), P(v).
0	u	
1	ux.	
2		
3		

418351 Friday, September 19, 2008 ชื่อ นส วิภาวัลย์ เพชรศิริกฤษณ์ รหัสนิสิต 4930610693

บันทึกช่วยจำ

ICMP → ทำในชั้น network layer เป็น protocol ที่ใช้ message ควบคุม internet (แต่ IP ไม่ต่อในท่อไป)

Traceroute ด้วย ICMP

ฝั่งเส้นทางตรงข้ามทางกลับทางตั้ง ปลายทาง สามารถค้นหาเส้นทางได้ ที่ตรงนี้ UDP ที่ยังมองว่าเป็นอย่างไร

IPV6 ↔ IPV4 → ถ้า packet ในชั้น router จะทำการแบ่ง packet ของท่านให้ตรงกับที่ต้องการ

ในหมวดว่าตัวว่า เมื่อมี packet ไปแล้วถ้าขนาดใหญ่เกินไปจะต้องลบ 6 Packet To Big เพื่อให้ส่งใหม่

ทั้ง V6 และ V4 ใช้งานร่วมกันได้ แต่ V6 จะแปลงในนั้น V4 ด้านนอก เรียกว่า → tunneling

Intra-AS (ภายใน) → RIP, OSPF, IGRP

RIP ระยะทาง cost = 1 นอกจากนี้จะมี max hops เพื่อการกระโดดข้ามไป DES

* ถ้ามีการ hops เกิน 15 ครั้ง ต้องกันการ ∞ loop *

จะทำการ update ทุก 30 sec แต่ถ้ามีข้อมูลใหม่ก็ส่งไปได้เลย

OSPF

บอกข้อมูลทั้งหมดครั้ง DES นายครั้งจะส่งข้อมูลครั้ง ต้อง broadcast ไปเรื่อยๆ

Graph abstraction : costs (ต้นทุน)

แบ่งได้หลายแบบ Ex. Bandwidth (ความเร็ว)

Routing Algorithm จะมี 2 แบบ

1. link state algorithm
2. distance vector

① จะทำการ broadcast ทุก link เพื่อค้นหาเส้นทางที่สั้นที่สุด (ตาม topology)

ข้อดี เมื่อ link ไหนมีค่า cost ต่อ link อื่นๆ จะรู้ว่าต่อ

ข้อเสีย ทำการ broadcast จะทำให้การทำงานช้าลงเมื่อมี link มาก

ขั้นตอน Dijkstra's algorithm

C(x,y) = cost ที่เชื่อมระหว่าง x กับ y

D(v) = ระยะทาง (cost) ที่น้อยสุด to ปลายทาง

p(v) = ตัวก่อนไป Ex. A → B → C → D เป็นต้นของ C

N' = เส้นทางที่ลงมาแล้วว่า cost ที่น้อยสุด

Ex routing table

DST	PRE	COST
x	u	1
y	x	2
v	~~u~~	2
w	y	3
z	y	4

ดูจากการที่ต้องเข้าน้ำ ป่อก เข้าน

② จะส่งข้อมูลระหว่างกันที่ละเอียดระหว่าง link)

ข้อดี ทำงานเร็ว เมื่อ cost ลดลงจะ update ทันที

ข้อเสีย count to ∞ เกิดจาก cost เพิ่มขึ้นทำทางส่งไปเรื่อยๆ ละ 1 จนกว่าจะมีค่าที่ดีกว่า (เรียกว่าปลอบขวัญ) ทำให้การทำงานช้า

418351 Friday, September 19, 2008 ชื่อ นายเกียรณพ หอมเนียม รหัสนิสิต 4421068S

บันทึกช่วยจำ

Transport Layer

ความน่าเชื่อถือ ไม่น่าเชื่อถือ

1 reliable, in-order delivery (TCP)

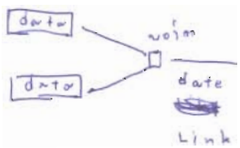
- congestion control = การควบคุมความหนาแน่นของข้อมูล
- flow control = ควบคุมการไหลของข้อมูล
- connection setup = ตั้งค่าการเชื่อมต่อ setup ที่ source

2 unreliable, unordered delivery (UDP) (ไม่น่าเชื่อถือ, ไม่เรียงลำดับ, ไม่ควบคุมการไหล)

- no-frills extension of "best-effort" IP

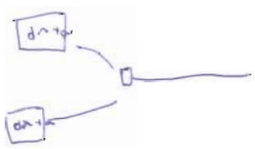
Multiplexing

- เป็นการรวม data หลายๆ อันเข้าด้วยกัน ในเครื่องเดียวกัน ที่ data Layer

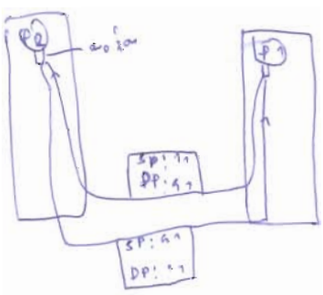


Demultiplexing

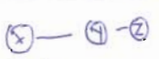
การแยกข้อมูลที่ถูกรวมกันมาแล้ว



TCP socket
- source IP address
- source port num
- dest IP add
- dest port num



Dijkstra's algorithm



Table

dest	Link
x	y

Distance

Data gram networks
- connectionless

ความรู้

1. ช่วงนี้งานเยอะ ไม่ค่อยมีเวลาเข้าเว็บ ตั้งแต่ทำงานมาก็ไม่ได้เข้าเว็บเลย ก็ไม่รู้ว่าจะเขียนอะไรดี ก็เลยเขียนเรื่องการเชื่อมต่อแบบต่างๆ ก็แล้วกัน

ขอบคุณที่มาจาก ตึก F ชั้นนี้ชั้นไม่ค่อยได้ขึ้นเลย ขอบคุณมากๆ ครับ

418351 Friday, September 19, 2008 ชื่อ นาย ภิษกฤ พิมพ์ภิมิ์ รหัสนิสิต 4932107727

บันทึกช่วยจำ

Transport layer ให้บริการ แก่ network layer อีกที หลายต่อหนึ่ง

เป็นการส่งแบบ ไม่น่าเชื่อถือ ไม่ reliable

ส่งแบบไม่ล่วงหน้า

ไม่มีการควบคุม ความเร็วการรับ IP

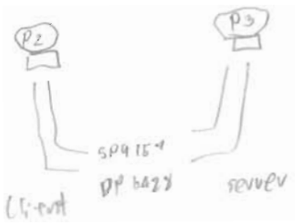
ไม่การันตี delay ใดๆ

ไม่การันตี bandwidth

source port ระบุหมายเลข return address

มีการใช้ multiplex เอง R ระหว่างหลาย data ผ่านเส้น ที่เหมาะสมกับการส่งในครั้งนั้น

setup connection ก่อนที่จะส่งข้อมูลจริง



ผ่าน Socket

เชื่อมต่อกับ Process

UDP ประกอบด้วย (dest IP address, dest port number)

Dijkstra's

step	N'	D(v), P(v)	D(w), P(w)	D(x), P(x)	D(y), P(y)	D(z), P(z)
0	u	2, u	5, u	1, u	∞	∞
1	ux	2, v	4, x		2, x	
2	uxy	2, u	3, y			4, y
3	uxyv		3, y			4, y
4	uxyvw					4, y
5	uxyvwz					

สุดเหนื่อย ไปหาอาหารทานเช้า กันเพื่อน

418351 Friday, September 19, 2008 ชื่อ นาย สรวิชา วรยศสินธ์ รหัสนิสิต 4942107824

บันทึกช่วยจำ

ICMP : Internet control Message Protocol

โปรโตคอลที่ใช้ร่วมกับ Message เป็นการ control Internet

ใช้ดูว่า can (host) และ router ทำงานอยู่ ICMP ใช้ร่วมกับเพื่อส่งสาร ใน N.L.

Send Data to R → have Feed Back

Data ของ ICMP มี type Data

Code Destination Know

Traceroute and RCMP.

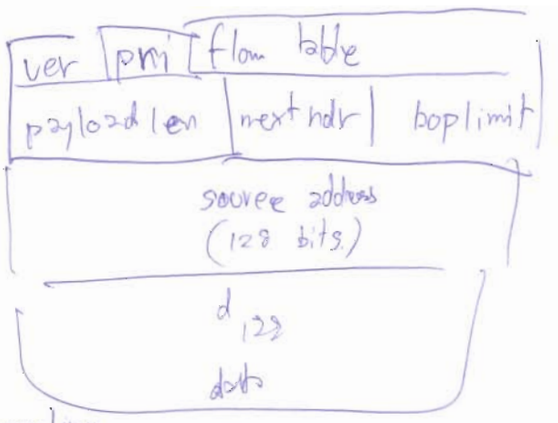
- ส่ง UPP ไป datagrams ตัวไปจับ TTL=1 TTL=2 0 หมด

send type 11 code ของ ping

ไปยังเครื่อง ไปที่เครื่อง on test ไปที่

Dijkstra's

IP v6 QoS.



Tunneling

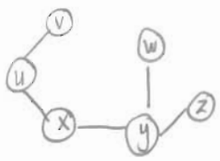
DV ระหว่าง N → N

ความคิดเห็น : ไม่มีข้อสอบ

418351 Friday, September 19, 2008 ชื่อ น.ส. ศุภิชา เชื้อสันทร์ รหัสนิสิต 49210800

บันทึกช่วยจำ

Dijkstra's algorithm:



destinational	link
v	(u, v)
x	(u, x)
y	(u, x)
w	(u, x)
z	(u, x)

ข้อเสีย เมื่อมี router เพิ่มจะต้องคำนวณ link ใหม่หมด

Bellman-Ford => หาเส้นทางที่สั้นสุด รู้แต่ระยะทาง แต่ไม่รู้ว่าแต่ละช่วงจะไปในทางต่อ (ทำจากรูปด้านล่าง)

Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

B-F equation ~~$d_v z = \min$~~

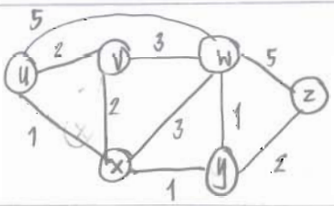
$d_u(z) = \min\{ c(u,v) + d_v(z), c(u,x) + d_x(z), c(u,w) + d_w(z) \}$

$= \min\{ 2+5, 1+3, 5+3 \} = 4$

Dijkstra's algorithm!

Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(x)	D(z), p(z)
0	u	2, u	5, u	1, u	∞	∞
1	ux	2, u	4, x		2, x	∞
2	uxy	2, u	3, y			4, y
3	uxyv		3, y			4, y
4	uxyvw					4, y
5	uxyvwz					

v → ① 2
x → ① 1
y → ② 2
w → ③ 3
z → ② 4



DST	PRE	COST
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4

⑥ การเปลี่ยนจาก IPv4 → IPv6

- ไม่สามารถเปลี่ยน router ทุก router พร้อมๆกันได้
- ทุกเครื่องของ network ต้องเป็น version เดียวกัน
- เชื่อมต่อ IPv6 ได้โดยใช้ Tunneling ข้อดี ทำให้ 2 router ซึ่งทำงานใน IPv6 สามารถรู้จักกันได้

network gate => อยู่ขอบ

network call => Network ที่ช่วยทำ routing

⑥ อะไรที่เปลี่ยนจาก IPv4

- ไม่ทำ checksum
- IPv4 การยอมให้ส่งที่เวลาใช้ในการ Process Data เปลี่ยนไป การทำงาน Option ไม่ทำงานบน head

"Packet Too Big" เป็นการเริ่มต้น

418351 Friday, September 19, 2008 ชื่อ น.ส. [illegible] รหัสนิสิต 49210818

บันทึกช่วยจำ

- ICMP - ใช้ในการแจ้งเตือน error reporting , - ping (type=8 code=0)
(Traceroute and ICMP) - แต่ละข้อความ host ต้นทาง → ปลายทาง
data ชิ้นแรก ส่งค่า TTL=1 ต่อๆไป เพิ่มขึ้นทีละ 1 ไปเรื่อยๆ จนกว่าจะได้ข้อความที่ต้องการ

- IPv6 Header
- เพิ่ม IP add_ จาก 32 bit → 128 bit

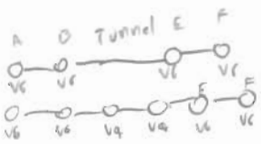
Ver = บอกเวอร์ชั่นของ IP 4 bit
priority = 8 bit ใช้แยกแพ็กเก็ต ตามคลาส เช่น: ต้องการให้ส่งเร็ว
payload len = 16 bit บอกขนาดของ payload
next hdr = ชี้ protocol อะไรถัดไป
hop limit = ค่าของ field ของแต่ละ packet ถ้าผ่าน router ใช้ -1 ถ้าเหลือ 0 ทิ้งเลย
source add_ , destination add_ , data =

* เปลี่ยนแปลงจาก IPv4
- ไม่ check sum ทำงานเร็วขึ้น
- options ที่ใช้ ไม่อยู่ใน IP header แต่จะอยู่ใน field next header เพื่อเพิ่มความยืดหยุ่น

* ICMP v6 : คล้าย v4
- Packet too Big * ปรับขนาดใหม่

* v4 TO v6
- Flag days * → ให้เปลี่ยนพร้อมๆกันทุก router ในวันเดียว

* Tunnelling



ใช้ v6 ใส่ใน field data (payload) ของ v4 data_
data_ จะวิ่งไปหา des_ add_ ก็ต่อ add_ ของ v6 ที่อยู่ปลายทาง
พอถึงจะส่ง data เข้า ถึงจะได้ data จะเห็น v6 ตรงนี้ แล้วส่งต่อ

* Graph abstraction
ส่ง pac_ จาก sending host → des_

* Routing algorithm classification
แบ่ง Global or decentralized
- อันแรกจะต้องรู้ว่า cost ต่ำสุด ทางเข้าของ node ทุก node
link ทุก link ก่อนถึงจุดหมาย เรียกว่า "link state"
node ที่รู้ cost ของ link ที่ต่อกับมัน โดยการแจ้งเพื่อนบ้าน
ต้องรู้จุดหมาย และแจกแจงกลับไปหาทุก node ที่อยู่ใกล้ ๆ ถึง des_
เรียกว่า " distance vector "

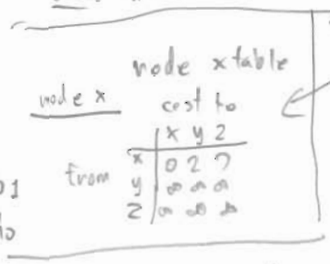
Or static or dynamic → เปลี่ยนแปลงตามอัตโนมัติ
admin จัดการเปลี่ยน

* Dijkstra's al_____

step	N'	D(v), p(v)			
0	u	2, v	6, u	1, u	∞
1	ux				
2					
3	uxy				
4					
5					

D(v): cost ของเส้นทางจาก source node → d_
ที่ cost ต่ำที่สุดของแต่ละ
p(v) : node ก่อนหน้าจาก so_ ไป (v)
N' : set ของ node
ความซับซ้อน .
$n*(n+1)/2 \rightarrow O(n^2)$
ข้อเสีย เกิด oscillation ได้

* Distance



* $d_x(y) = \min\{c(x,v) + d_v(y)\}$

$c(x,y)$: cost ของ link จาก node x ไป node y

418351 Friday, September 19, 2008 ชื่อ น.ส. สุวดี สิงห์ลา รหัสนิสิต 49210826

บันทึกช่วยจำ

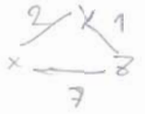
อะไรที่เปลี่ยนจาก IPV4
- ไม่ check sum
- source router → router ต้องอยู่ใน list แล้ว (ซึ่งตามไปรู้จาก router ที่อยู่ในลิสต์นั้น)
- V6. เพิ่ม massage types ไม่ทำ Fragment

transition จาก IPV4 To IpV6.
- ไม่สามารถเปลี่ยน router ทุกตัวในวันเดียว แต่จะมี delay ข้อดี ทำให้ router คุยกันได้ แม้จะเชื่อมกันอยู่
- IPV4 กับ IPV6 ต้องใช้เปลี่ยนไปใช้ Tunneling

Routing Algo — classifica —
1. Global หรือ decentralized มีศูนย์กลางเป็นจุดรวม
- router ทุก router เข้าถึงข้อมูลของ network
2. Decentra —
- router ไม่ทำเป็นต้องเห็นภาพทั้งหมดมีจุดใช้วิธีประมาณการส่งไปยังข้อมูลจากเพื่อนบ้าน ถ้ามีคนเสนอเส้นทางที่ดีกว่า ก็จะเลือกเส้นทางนั้น
- ~~ใช้ข้อมูล~~ [distance vector (cost)
static → routing table จะไม่เปลี่ยน เว้นแต่ router เป็นคนเปลี่ยนค่าเอง
A-link - state Routing
Dijkstra's alg → ต้องการทั้งหมดของทุก router ใน network ทุก router ต้องมีภาพนี้ เพื่อหาเส้นทางสั้นที่สุด

IP Addressing : introduction
interface จุดเชื่อมต่อระหว่าง host กับ link
router มีหลาย inter face host มี interface เดียว

Ex link state ใช้ dynamic ต้องทุกช่วงเวลา จะต้อง state borad case เมื่อใน router มีการเปลี่ยนแปลงในแม่



$D_x(z) = \min[c(x,y) + D_y(z), c(x,z) + D_z(z)]$
$= \min(2+1, 7+0)$
$= 3$ ✓

(X)

	x	y	z
x	0	2	7
y	—	—	—
z	—	—	—

	x	y	z
x	0	2	3
y	2	0	1
z	7	0	1

(Y)

	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

(Z)

	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

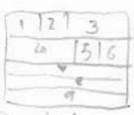
มาเรียนทันเวลา

418351 Friday, September 19, 2008 ชื่อ นายนนทวัฒน์ แซ่จึง รหัสนิสิต 4921085?

บันทึกช่วยจำ

ICMP
- ถูกใช้ในการแจ้ง error reporting
- ping program type = 8 code = 0

IPv.6 Header
- ขนาด IP add_ จาก 32 bit → 128 bit



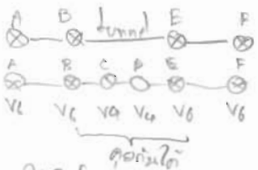
① Ver. : 4 bit หมายเลข version ของ IP
② Priority : 4 bit ให้ความสำคัญกับ Packet ตาม class ของการบริการ
③ Flow label : 20 bit ใช้ระบุ Flow ของ Packet
④ Payload len : 16 bit ขนาดของข้อมูล Payload
⑤ next hdr. : 8 bit ระบุว่าใช้ Protocol อะไรในการส่ง
⑥ hop limit : ค่าของ field ลดลง 1 ทุกๆ packet ที่ผ่าน router นึง = 0 packet จะถูกทิ้งไป
⑦ Source add_ :
⑧ Destination add_ :
⑨ data :

ที่แตกต่างจาก IPv.4
- ไม่ checksum เพื่อการทำงานที่เร็วขึ้น
- options ไม่อยู่ใน IP header แต่จะอยู่ใน field next header เพื่อระบุว่ามี option ที่ใช้

ICMPv.6 สร้างขึ้นมาใหม่
- "Packet Too Big" ขณะที่ข้อมูลไหลเข้าไป

IPv.4 to IPv.6
- "Flag days" ที่การทำงานเปลี่ยนทุก router ในเวลาที่เราเลือกไว้ ไม่มีการทำงานร่วมกันระหว่างเครื่องทุกๆ เครื่องใน Internet

- Tunneling

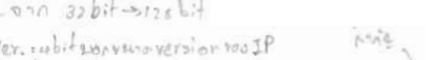


กล่อง V6 data_ ที่ผ่านมาในรูปที่ field data(payload) ของ V.6 data data_ จะระบุว่า dest add_ ของ node V.6 ที่อยู่ปลายอุโมงค์ ขณะที่ส่ง data ไป ถ้า router ที่อยู่ในอุโมงค์ถึงจะ V.4 data_ ไม่รู้จะทำอย่างไร ก็จะส่งกลับมาว่า data_ นี้ของ V.6 data อยู่ ถึงจะมาถึง V.6 ต้นทางจะรู้ว่า V.6 ถึงปลายทางแล้ว

Routing algorithm
- Graph abstraction
  ต้องส่ง packet จาก sending host → dest host

R_a_ classification
แบ่งเป็น Global or decentralized
- ต้องการรู้ว่าเส้นทางที่มี cost ต่ำสุด ต้องรู้ข้อมูลของ node ทุก node ทุก link ทุก link ก่อนถึงจะ
- เรียกอีกอย่างว่า "link state" algorithm
- node จะรู้แค่ cost ของ link ที่ต่อกับตัวมันเองกับ node ที่อยู่ใกล้เคียง ทำงานร่วมกับ node ที่อยู่ใกล้เคียงเพื่อหาเส้นทางที่ดีที่สุดไปยัง dest node เรียกว่า "distance vector"

ใช้ static or dynamic
เปลี่ยนตามน้ำหนักของ load traffic หรือ link ที่เปลี่ยนโดย admin การจัดการ



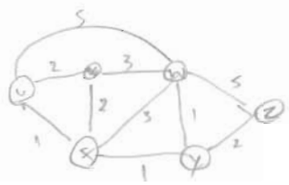
link-state R_A_
อัลกอริทึม (Dijkstra's algor_)
c(x,y) : cost ของ link จาก x ไป y ถ้า x,y ไม่ได้ต่อตรงกัน จะถือว่า c(x,y) = ∞
D(v) : cost ของเส้นทางจาก source node → dest v ที่มี cost น้อยที่สุดในขณะนั้น
P(v) : เป็น node ก่อนหน้าบนเส้นทางที่สั้นที่สุดจาก source ไป v
N' : เป็น set ของ node ที่ได้ทราบเส้นทางที่สั้นที่สุดจาก source
Dij _____

418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต 4921087 5

บันทึกช่วยจำ

Ex Dijkstra algorithm : example

step	N'	D(v), P(v)	D(w), P(w)	D(x), P(x)	D(y), P(y)	D(z), P(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					4,y

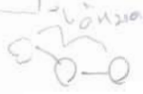


ตาราง u

DST	PRE	COST
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4

vt

network layer = ส่งข้อมูลจากเครื่องไปยังเครื่องอื่นโดยผ่านเน็ตเวิร์ก



transport layer : ส่งข้อมูลจากต้นทาง - ปลายทาง

- ถ้ามีการ Congestion ของข้อมูล - Flowcontrol - ควบคุมปริมาณการส่งข้อมูลเพื่อไม่ให้ข้อมูลล้น

- การสร้างการเชื่อมต่อ connection setup

- ถ้าไม่ต้องการความน่าเชื่อถือของข้อมูล ไม่ต้องการ unreliable, unordered delivery : UDP

- ถ้าไม่ต้องการ delay ไม่ได้

Multiplexing/demultiplexing (การรวมข้อมูล / การแยกข้อมูล)

TCP ใช้งานอย่าง - IP ปลายทาง
port ปลายทาง
IP ปลายทาง
PORT ปลายทาง

ตามนี้
1. ไม่มี Quiz
2. [illegible]

418351 Friday, September 19, 2008 ชื่อ น.ส. ธารุมา เรืองทอง รหัสนิสิต 49210909

บันทึกช่วยจำ

IP Addressing : Introduction

IP แบ่งเป็น 2 ส่วน
- ส่วนหน้า → network port
- ส่วนหลัง → host port

class A /8 1.0.0.0 - 127.255.255.255
B /16 128.0.. - 191.255..
C /24 192.0.. - 223.255..
D /32 224.0.. - 239.255.

Ex 192.168.100.0/27 อยู่ใน class C กลมล่าสุด 3 bit

network IP = 192.168.100.111xxxxx (5)
subnet mask = 255.255.255.224
จน. subnet $2^n - 2 = 2^3 - 2 = 6$ subnet
จน. host $2^5 - 2 = 30$ host
boadcast IP = 192.168.100.255

host IP

subnet 0	192.168.100.0	192.168.100.31
1	——— .32	——— .63
3	——— .64	——— .95
⋮		
broadcast 6	192.168.100.128	192.168.100.255

CIDR : จน. ใช้ตามที่ routing ของ network port ที่ต้องการ

Dijkstra's algorithm

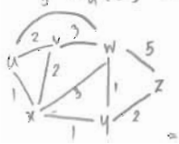
step	N'	D(v), p(v)	w	x	y	z
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			
4	uxyvw					
5	uxyvwz					

DST (ปลายทาง)	PRO (ผ่าน)	cost
x	u	1
y	x	2

Bellman - ford

clearly, $d_v(z) = 5, d_x(z) = 3, d_w(z) = 3$

$d_u(z) = \min\{ c(u,v) + d_v(z), c(u,x) + d_x(z), c(u,w) + d_w(z) \}$

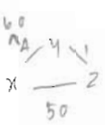


$= \min\{2+5, 1+3, 5+3\} = 4$ เอาค่าที่น้อยสุด

Distance Vector : link cost changes

ข้อดี "good news travels fast"

ถ้าจาก z → y เมื่อ cost จาก y → x ลดลง y แจ้งให้ z รู้



cost เพิ่ม ปัญหาใหญ่

~~cost~~ LS หรือ DV

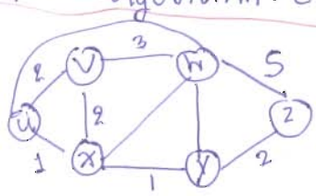
LS - ส่งข่าวมาก
- ลัดเร็ว
- ถ้า error จะเกิดเฉพาะ link

DV - ส่งข่าวน้อย
- error เกิดที่ทั่วทั้งหมด

418351 Friday, September 19, 2008 ชื่อ น.ส.อัธรา บุญมาเอิศ รหัสนิสิต 4921091Y

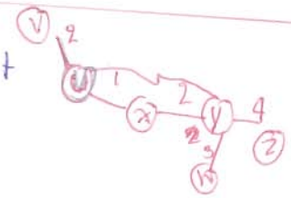
broadcast → ส่งเข้าที่ปลายหนึ่งเครื่อง เข้าถึงทุกเครื่อง บันทึกช่วยจำ

Dijkstra's algorithm: example

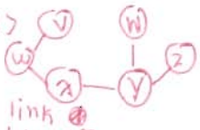


step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

DST	PRE	cost
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4



example 2)



destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

IP Fragmentation & Reassembly

network layer ของ MTU ของลิงค์ชั้นล่างของเฟรม IP Packet ก็อาจใหญ่เกินไป, แต่ละลิงค์ในเครือข่ายอาจใช้ router ไป router ที่ link ต่างชนิดกันมีค่า MTU ของ frame ได้ไม่เท่ากัน ทำให้ IP Packet ต้องถูกแบ่งออกเป็นหลายๆ ส่วนย่อย ๆ เรียกว่า fragment แต่ละส่วนจะมี IP header ของตัวเอง และส่งไปที่ปลายทางแต่ละส่วนถ้ายังไม่ถึงปลายทางก็ต้องส่งต่อไปอีก จนกว่าจะถึงปลายทาง เพื่อที่จะประกอบกลับ

DHCP → โปรโตคอลที่ใช้ในการกำหนด IP Add

DHCP sever → มีหน้าที่แจก IP ในเครือข่ายที่ให้ขอ

เป็นการขอตามชื่อเรียก

NAT → แปลง IP Add หนึ่ง ไปยัง IP Add หนึ่ง

ใช้ผ่านกับแปล IP vs. IP ตามรุ่น และ IP ตัวใดตัวหนึ่ง

can แก้ข้อจำกัดของการใช้งาน IP ตามรุ่นได้

เมื่อ DNAT เรียกทำงานจะสร้างตารางรายการขึ้นมาเก็บข้อมูล IP add ของเครื่องในเครือข่ายภายในที่ส่ง Packet ผ่าน NAT device และสร้างตารางรายการเก็บข้อมูลของ Port ที่ถูกใช้ไปโดย outside IP add เมื่อมีการส่ง Packet จากเครื่องข่ายภายในผ่าน NAT

Hub ส่งออกทุกลิงค์ยกเว้นลิงค์ที่เป็นอินพุต

ICMP : เป็นโปรโตคอลที่ใช้ในการรายงานสถานะของตัวแทน รวมไปถึงมีหลักการทำงานอื่นๆ

error reporting ของ error ที่คาดไม่ถึง

echo request ตามชื่อบอกว่าเรื่องอะไรอยู่ข้างล่าง

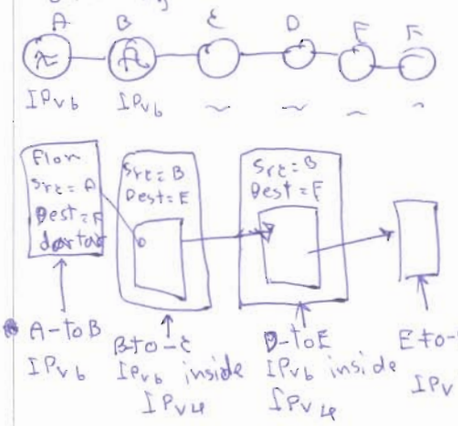
418351 Friday, September 19, 2008 ชื่อ น.ส.อรยา จุนพลพิทักษ์ รหัสนิสิต 4921098J

บันทึกช่วยจำ

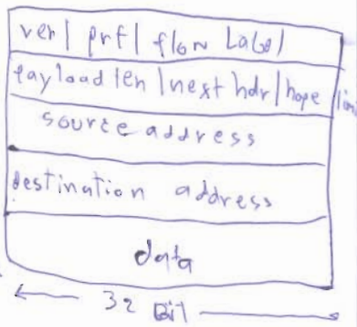
ICMP: Internet control Massage protocol = โปรโตคอลที่เกี่ยวกับ message เวลา control Internet
- ให้ host และ router ส่งข้อความแจ้งข้อผิดพลาดกลับมา ← ไม่ส่งกลับ ไม่ใช้ก็ได้

IPv6 ตัดการแบ่งข้อมูลออก, ไม่รับทำ fragment

Tunneling



ทุก router ต้องรู้จักกัน
→ route table
Costs: ค่าใช้จ่ายที่ต้องเสียไปในแต่ละ link
→ Link มี cost ต่อ 1



IP Fragmentation and Reassembly
MTU ขนาดสูงสุดของแพ็กเก็ตที่ link นั้นรับได้
ข้อมูล IP ขนาดใหญ่ต้องแบ่งเป็นหลายๆ byte ก่อนส่งออกไป
ถ้าแบ่งเป็น Datagram ละ MTU = 1500 byte
1) ข้อมูลมี Header 20
∴ 4000 byte = 4000 - 20 = 3980
∴ เลือกแบ่งได้ไม่เกิน 3980 byte

ต้นเค้า *
Router ทำงานหลัก 1) run routing algorithms
2) Forwarding datagrams

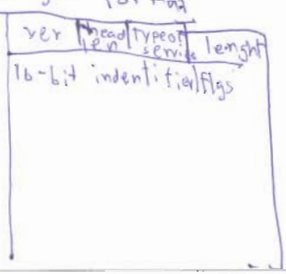
Input Port Data → ตรวจสอบ ข้อมูล → Lan card → queue ing → switching Fabric → output

การทำงานที่ Input Port ต้องมี queue
- เพื่อทำให้ไม่เกิด Head-of-Line blocking (การที่ข้อมูล 2 ข้อมูลออกทางเดียวกันแล้วชนกัน)
- ถ้าข้อมูลเข้ามามากๆ ถูกจัดเก็บไว้ใน queue ทำให้เกิด Queueing delay (เสียเวลา)

output Port Switch Fabric → queue → Lan card → ออกสู่ภายนอก

Transport layer
- TCP ข้อมูลถูกส่งถึงปลายทางแน่นอน แต่ความเร็วในการส่งข้อมูลช้า
- UDP ข้อมูลถูกส่งปลายทางแต่ข้อมูลอาจสูญหายได้ แต่ทำงานได้เร็ว เช่น VDO, ออกไม่ค่อยกระตุก

IP datagram format



418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต 4938075

บันทึกช่วยจำ

Dijsktra's Algorithm

```
1  Initialization:
2  N' = {u}
3  for all nodes v
4  if v adjacent to u
5    then D(v) = c(u,v)
6    else D(v) = ∞
7
8  Loop
9  find w not in N' such that D(w) is a minimum
10   add w to N'
11   update D(v) for all v adjacent to w and not in N'
12     D(v) = min( D(v), D(w) + c(w,v) )
13   until all nodes in N'
```

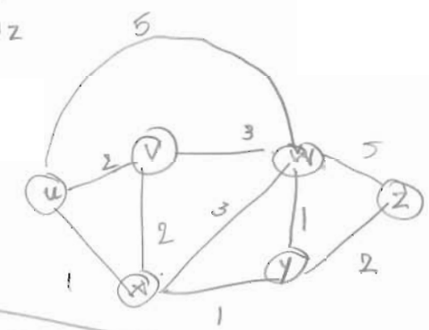
Bellman-Ford

Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

B-F equation says:

$$d_u(z) = \min\{ c(u,v) + d_v(z), \; c(u,x) + d_x(z), \; c(u,w) + d_w(z) \}$$
$$= \min\{ 2+5, 1+3, 5+3 \} = 4$$

Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2, u	5, u	1, u	∞	∞
1	ux	2, u	4, x		2, x	
2	uxy	2, u	3, y			4, y
3	uxyv		3, y			4, y
4	uxyvw					4, y
5	uxyvwz					



PST	PRE	COST
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4

Distance vector algorithm

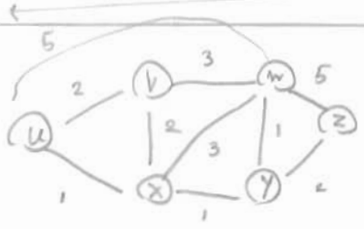
$D_x(y) \leftarrow \min_v\{ c(x,v) + D_v(y) \}$ for each node $y \in N$

418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต 49370158

บันทึกช่วยจำ

Dijkstra's algorithm. example

step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2,u	5u	1u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



v → ① 2
x → ① 1
y → ② 2, 3
w → ③, 3
z → ②, 4

Dst	PRI	cost
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4

ได้ Resulting table

Ex link ถูกใช้งานหนักมาก ทำให้เพิ่ม cost ใน link ทำให้เปลี่ยนเส้นทาง - ทำให้เป็น dynamic แต่ต้องไม่ link state broadcast ไปทั่ว

* เพราะทุกคน update พร้อมกัน
* แก้ปัญหาโดย อยู่ update ต่างกัน

Dijkstra's algorithm, discussion

- ซับซ้อนกับจำนวน node ที่ใน network i router อย่างเดียว
- link ออกจะไม่ทราบได้ ถ้า processor ที่เร็ว เช่น ตรวจ link ที่เปลี่ยน ๆ ค่า ข้อมูลต้องใช้ cost ที่เหมาะสมด้วย

Hierarchical Routing

- Network ใหญ่ขึ้นเรื่อย ๆ ต้องใช้ routing protocol เปลี่ยน

OSPF (Open shortest Path First - ส่ง update ไปทั่วทุกเส้น
- ทุก node ได้ข้อมูลเดียวกัน

OSPF "advanced" features (not in RIP)
- ใช้ security ตรวจสอบก่อนว่าเป็น router
- เส้น paths ไปยังเครือข่ายได้หลายเส้นทาง
- ช่วยให้ครับงานกระจายกัน แต่ละเส้นทาง packet เท่า ๆ กันได้ ถ้า router ใหม่รับงานมาก

ความหมาย hierarchical routing ได้ - integrated hierarchical

Internet inter-AS routing: BGP.

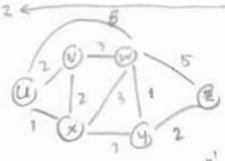
- เป็นที่ยอมรับกันว่าใช้งานได้ดี แต่ยังมาก
- BGP รู้จักเส้นทางเครือข่ายต่าง ๆ ที่เชื่อมต่อกันได้ Auto
- ระหว่าง AS ต่าง ๆ ใน Auto sys
- BGP ตรวจสอบว่าเส้นทางไปยังเครือข่ายต่าง ๆ ยังใช้งานได้อยู่หรือไม่

418351 Friday, September 19, 2008 ชื่อ ชัยศักดิ์ แสวงโชคมงคล รหัสนิสิต 4934701126

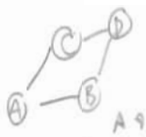
บันทึกช่วยจำ

Dijkstra (LS) Link stable

step	N'	D(v), P(v)	D(w), P(w)	D(x), P(x)	D(y), P(y)	D(z), P(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



Distance Vector เร็วกว่า ของเสื่อมบันทึกข้อมูลแล้วส่งต่อ กันเรื่อยๆ



A อยู่บอกแค่ C, B เท่านั้น

(AS) Autonomous system. — ใช้ routing protocol ได้อย่าง.

Network ขึ้นอยู่ กับแบ่งออกมาเป็นหลาย Network

ต่อ อยู่ใน network ต่อเข้ากับ intra-As.

ต้องมีอย่างน้อย: ต้อง run routing protocol อย่าง.

ต้นที่ 1 เรื่อง Intra-As ต้นที่ 2 เรื่อง Inter-As ใช้ติดต่อกับภายนอก

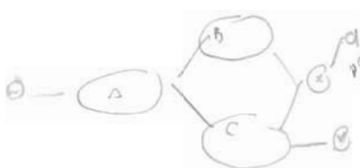
Intra-As routing

RIP Routing Information Protocol ใช้ระบบ Unix ใช้ได้แค่ 15 hops. เร็วกว่า Distant Vector,

OSPF Open shortest Path First มีหลายอันเร็ว เก็บได้มากกว่า เร็วกว่า (LS)

IGRP

Inter-AS BGP เร็วที่สุด multicast เร็วกว่าสุด ต้องการ short path.



dual-homed,

ถ้าไม่ forward packet ถ้า B ต่อ x ไม่ได้ ผ่าน x ไม่ได้ C, B.

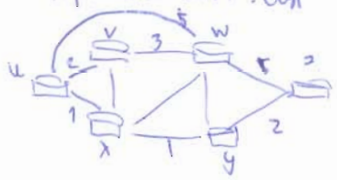
ตามนี้

อัดเสียงไว้ด้วย

418351 Friday, September 19, 2008 ชื่อ นาย นักบาส ใจหาญ รหัสนิสิต 4937018?

บันทึกช่วยจำ

Graph abstraction : cost link vs cost



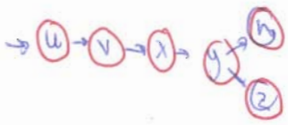
ใช้ link ที่มี cost น้อยที่สุด

คำนวณหาเส้นทาง Dijkstra's algorithm

Step	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

หาเส้นทาง

Dst	PRE	cost
x	v	1
y	x	2
v	u	2
w	y	3
z	y	4



หาตาราง forwarding table ที่ u:

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Routing Algorithm จัดเส้นทาง มี 2 แบบ

1. Global : ต้องรู้ข้อมูลของเครือข่ายทั้งหมด link state ใช้ข้อมูลของ router ทั้งหมดเพื่อคำนวณ cost ที่ดีที่สุด
2. Decentralized : router ไม่มีข้อมูลครบทั้งเครือข่าย ใช้ข้อมูลของเพื่อนบ้านและการคำนวณของเพื่อนบ้านมาคำนวณต่อ → distance vector
   ↳ ค่าใช้จ่ายน้อยที่สุดในการเดินทางจากต้นทางไปปลายทาง

Distance Vector Algorithm : $d_x(y) = \min_v \{ c(x,v) + d_v(y) \}$.

เส้นทางที่สั้นที่สุดใช้ Bellman–Ford

ex. Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

B-F equation say: $d_u(z) = \min \{ c(u,v) + d_v(z), c(u,x) + d_x(z), c(u,w) + d_w(z) \}$

$= \min \{ 2+5, 1+3, 5+3 \} = 4$

update x table

node	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

	x	y	z
x	0	∞	∞
y	2	0	1
z	∞	∞	∞

	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

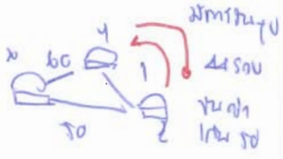
	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

ไม่มีการ update

	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

ข้อดีข้อเสีย : ดูแลรักษาง่าย

ข้อเสีย : เกิดปัญหา

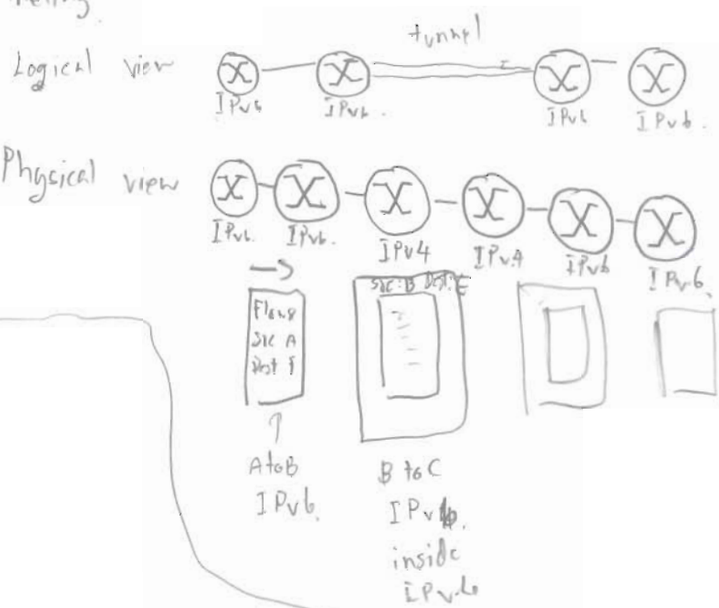


418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต 4937021.

บันทึกช่วยจำ

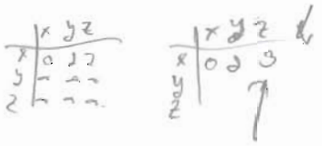
IPv6

Tunneling



Dijsktra's Algorithm

1. Initialization:
2 N' = {u}
3 for all nodes v
4 if v adjacent to u
5. then D(v) = c(u,v)
6 else D(v) = ∞
7
8 Loop
9 find w not in N' such that D(w) is a minimum
10 add w to N'
11 update D(v) for all v adjacent to w and not in N':
12 D(v) = min [D(v), D(w) + c(w,v)]
13 shortest path cost to w plus cost from w to v
15 until all nodes in N'



$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\}$
$= \min\{2+0, 7+1\} 2.$

step	N'	D(x),p(x)	D(w),p(w)	D(z),p(z)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	
1	ux	2,u	4,x		2,x	
2	uxy		3,y			
3	uxyv					
4	uxyvw					
5	uxyvwz					

418351 Friday, September 19, 2008 ชื่อ นาย วุฒิพงศ์ ลีพรหมรักษ์ รหัสนิสิต 49370307

บันทึกช่วยจำ

Traceroute → ทำ Ping 3 ครั้ง ครั้น TTL ทีละหัน

↓ หา

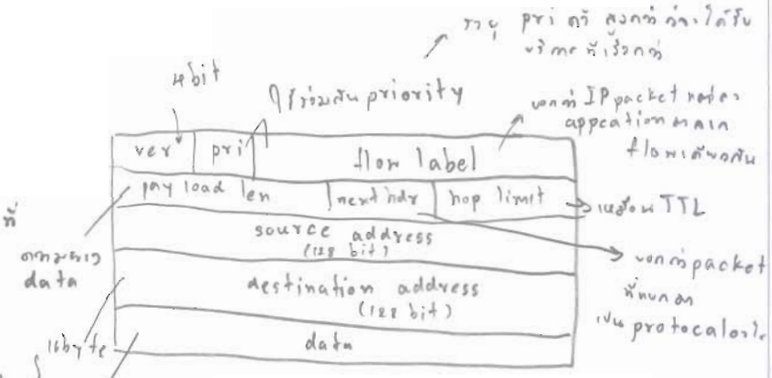
1 ไปถึงปลายทาง

2 ไปไม่ถึง

3 บน host ไม่ได้

IPv6

header มี 40 byte ทางคอมพิวเตอร์ไม่รับทำ fragment



ICMP ย่อมาจาก

ping คือ IP address เป็นไปได้

2 แบบ

1 host ทำ ping เพื่อดูว่าได้รับ massage ตอบกลับ

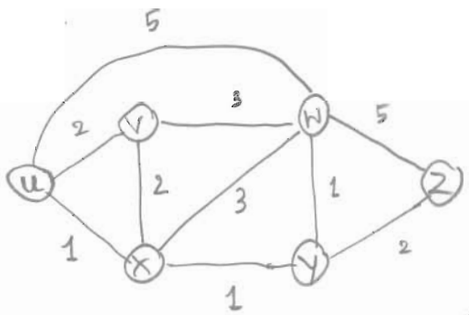
2 host ทำ pi

ได้ IP มากขึ้น 2 เท่า

Dijkstra's algorithm

step	N'	D(v), p(v)	D(r), p(r)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2, u	5, u	1, u	∞	∞
1	ux	2, u	4, x		2, x	∞
2	uxy	2, u	3, y			4, y
3	uxyv		3, y			4, y
4	uxyvw					4, y
5	uxyvwz					

DST	PRE	COST
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4



v → ① 2

x → ② 1

y → ② 2

w → ② 3

z → ② 4

Tunnel

ทำได้ 2 router ที่ห่างกัน IPv6 หรือ router ที่ 2 อาจเหมือนกันอยู่ ระหว่าง router ทั้งไม่รู้จัก IPv6

อาจจะใช้ ทางนี้ทาง กับ ปลายทาง ต่อสัมผัสกลาง

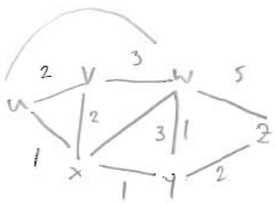
src อาจเป็น data packet ที่เป็น v6 ที่ปรับจูนเป็น data ที่จะเป็น IPv4 ตามปลายทาง ได้รับ ก็อาจกลายเป็นออกมา หลาย ๆ ต่อ โดยไม่มีปัญหา

418351 Friday, September 19, 2008 ชื่อ [illegible] รหัสนิสิต [illegible]

บันทึกช่วยจำ

Dijkstra's algorithm

step	N'	D(v) p(v)	D(w) p(w)	D(x) p(x)	D(y) p(y)	D(z) p(z)
0	u					
1	ux	2, u	5, u	1, u	∞	∞
2	uxy	2, u	4, x		2, x	∞
3	uxyv		3, y			4, y
4	uxyvw					4, y
5	uxyvwz					



PST	PRE	cost
x	u	1
y	x	2
v	u	2
w	y	3
z	y	4

multiplexing - [illegible] ของ SP [illegible] เพื่อส่ง [illegible]

Demultiplexing - [illegible] SP [illegible] packet

[illegible] node [illegible]

LS กับ DV

LS : [illegible] broadcast

DV : [illegible]

ข้อดี DV - [illegible]

ปัญหา routing loop

แก้ count-to infinity

LS : [illegible]

DV : [illegible]

hierarchical - [illegible] network [illegible] routing protocol [illegible]

Intra - AS Routing มี 3 แบบคือ

- rip
- OSPF
- IGRP

OSPF - [illegible] update [illegible] node [illegible] advertisement : [illegible] node [illegible] security [illegible] paths [illegible]

rip - distance vector algorithm

[illegible] 1 cost : hop = 15 [illegible] loop

Internet - BGP

[illegible] - [illegible] system [illegible]

- [illegible] system
- [illegible]