APNIC Training

Internet Fundamentals

20-21 November 2009 – Nadi, Fiji

Sixth PacNOG Meeting, Conference and Educational Workshop

In conjunction with PITA

Introduction

- Presenters
  - Nurul Islam Roman
    - Technical Training Officer
    - nurul@apnic.net

APNIC Asia Pacific Network Information Centre

Assumptions & Objectives

Assumptions

- Entry/Mid level engineers working in ISP/service provider network
- Are not familiar or up-to-date with technology detail
- Has not got advance experience to work with network equipment
- Are interested in Internetworking technologies

Objectives

- To provide an understanding of current Internet protocols
- To provide a working knowledge of the procedures managing Internet
- To keep up updated knowledge of future Internet technology

APNIC Asia Pacific Network Information Centre

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

Signal, Data and Information

- Data is transmitted over a physical network as a sequence of binary digits (bits - 0s and 1s).
- The "sending" process involves the source device generating a pattern of signals (voltages, light patterns, wavelengths).
- The pattern of signals generated represents the sequence of bits making up the data.
- These signals can be "read" by any device attached to the same physical network.
- "Reading" means identifying the signals to receive the same pattern of bits as generated by the sender.

What is Protocols

- All data is transmitted in the same way irrespective of what the data refers to, whether it is clear or encrypted.
- The data communication protocols define the structure or pattern for the data transferred – this gives it its meaning.
- The Protocols define
  - *functions* or *processes* that need to be carried out in order to implement the data exchange and the
  - *information* required by these processes in order for them to accomplish this

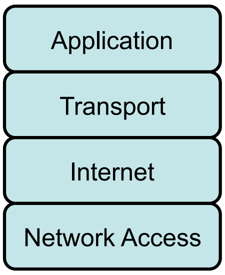
The OSI Model

Layer	Function
Application	Access to the network
Presentation	Manipulate data (Translate, encrypt)
Session	Manage sessions (connections)
Transport	Provide reliable delivery
Network	Internetwork - move packets from source to destination
Data Link	Configure data for direct delivery by physical layer
Physical	Physical delivery - electrical specs etc

Protocol Models

- In the late 1970s the ISO (International Standards Organisation) introduced a model defining the functions for data communications between two computers in a 7 layer model - The OSI (Open System Interconnection) Model
- Not a protocol but a framework intended to facilitate the design of protocols for inter-computer communication.
- Defines the processes required at each of the modularised layers
- OSI is “protocol independent”

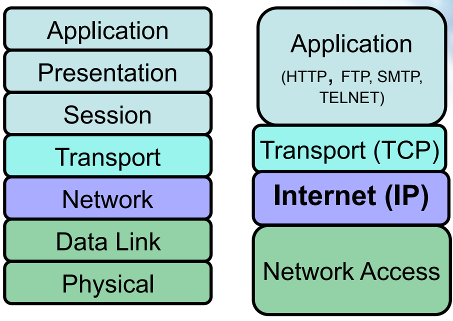
The Four Layers of TCP/IP

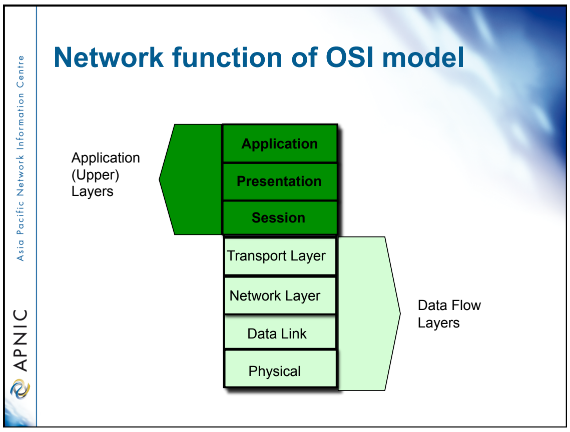


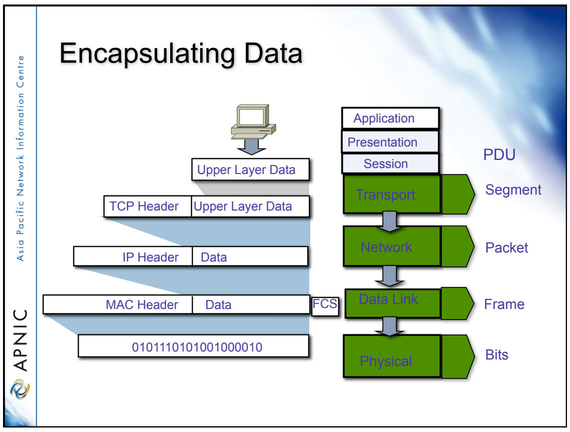
TCP/IP and the OSI Model

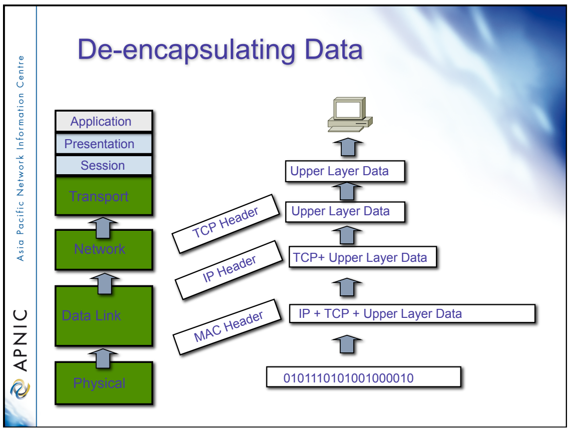
- TCP/IP was created before the OSI model
- It is a layered protocol implementation
- Its layers do not match the OSI model exactly, but the processes defined in the OSI model are contained in the TCP/IP layers

The OSI Model and TCP




Network function of OSI model
Application (Upper) Layers
Application
Presentation
Session
Transport Layer
Network Layer
Data Link
Physical
Data Flow Layers
APNIC
Asia Pacific Network Information Centre


Encapsulating Data
Upper Layer Data
TCP Header
Upper Layer Data
IP Header
Data
MAC Header
Data
FCS
0101110101001000010
Application
Presentation
Session
Transport
Network
Data Link
Physical
PDU
Segment
Packet
Frame
Bits
APNIC
Asia Pacific Network Information Centre


De-encapsulating Data
Application
Presentation
Session
Transport
Network
Data Link
Physical
Upper Layer Data
TCP Header
Upper Layer Data
IP Header
TCP+ Upper Layer Data
MAC Header
IP + TCP + Upper Layer Data
0101110101001000010
APNIC
Asia Pacific Network Information Centre

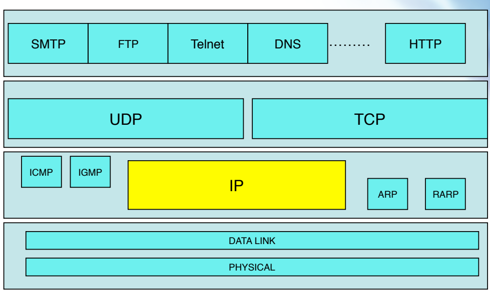
Packets

- A packet then contains a set of data made of the various headers from each layer including the data generated by the application layer.
- The packet is “built” during a sending process when each layer determines the information needed for its tasks, and adds this header information
- The layer will then take this information, with any other data it might have received from a higher layer, and pass it as one set of data to a lower layer.
- This process is then repeated and is called *encapsulation*

Internet Protocol (IP)

- IP is an unreliable, connectionless delivery protocol
  - A best-effort delivery service
  - No error checking or tracking (no guarantees – Post Office)
  - Every packet treated independently
    - Can follow different routes to same destination
  - IP leaves higher level protocols to provide reliability services (if needed)
- IP provides three important definitions:
  - basic unit of data transfer
    - specifying exact format of the headers
  - routing function
    - choosing path over which data will be sent
  - rules about delivery
    - how IP datagrams should be processed
    - how to deal with unusual events (errors)

TCP/IP Protocol Structure



From Forouzan

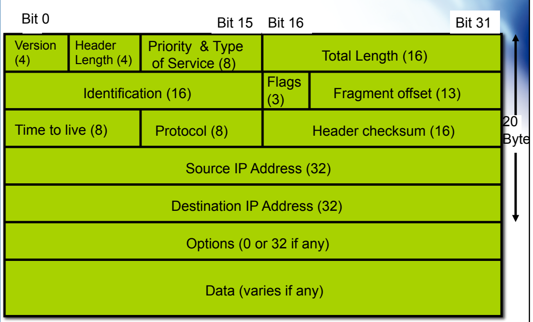
IP Datagram format

- That part of a packet containing the IP headers and the data from the higher layers passed to the IP layer are called ***datagrams***
- IP specifies the header information for the data it requires for its tasks - information needed for routing and delivery
  - eg source and destination IP addresses
- It has nothing to do with higher layer headers or data and can transport arbitrary data

Datagram header	Datagram data area

Asia Pacific Network Information Centre
APNIC

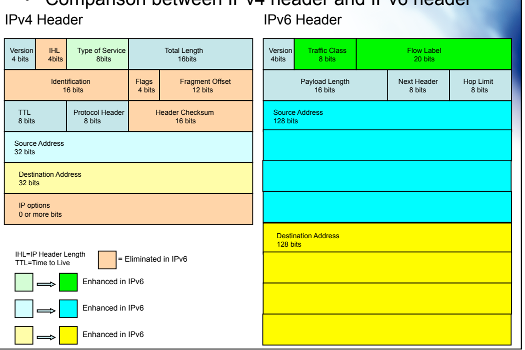
IPv4 Datagram header fields



Asia Pacific Network Information Centre
APNIC

IPv6 header

- Comparison between IPv4 header and IPv6 header



Questions?

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

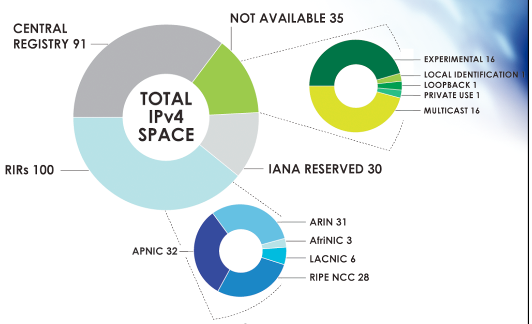
Overview

- IP addressing Issues and solution
- Variable Length Subnet Mask (VLSM)
  - Written exercise : VLSM calculation
- Summarisation of routes
- Classless InterDomain routing (CIDR)
- Internet registry IP management procedure
  - Written exercise : Route summarisation

IP Addressing issues

- Exhaustion of IPv4 addresses
  - Wasted address space in traditional subnetting
  - Limited availability of /8 subnets address
- Internet routing table growth
  - Size of the routing table due to higher number prefix announcement
- Tremendous growth of the Internet

How many IPv4 IANA pool available



Source : Internet Number Resource Report - Number Resource Organization (NRO)

IP addressing solutions

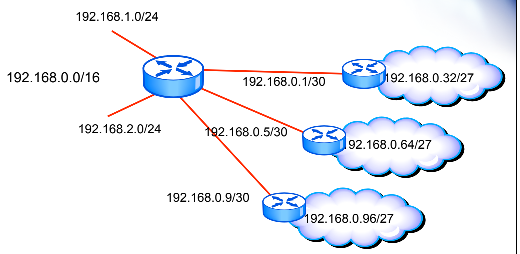
- Subnet masking and summarization
  - Variable-length subnet mask definition
  - Hierarchical addressing
  - Classless InterDomain Routing (CIDR)
  - Routes summarization (RFC 1518)
- Private address usage (RFC 1918)
  - Network address translation (NAT)
- Development of IPv6 address

Variable Length Subnet Mask

- Allows the ability to have more than one subnet mask within a network
- Allows re-subnetting
  - create sub-subnet network address
- Increase the routes capability
  - Addressing hierarchy
  - Summarisation

Calculating VLSM example

- Subnet 192.168.0.0/24 into smaller subnet
  - Subnet mask with /27 and /30 (point-to-point)



Calculating VLSM example (cont.)

- Subnet 192.168.0.0/24 into smaller subnet
  - Subnet mask with /30 (point-to-point)

Description	Decimal	Binary
Network Address	192.168.0.0/30	x.x.x.00000000
1st valid IP	192.168.0.1/30	x.x.x.00000001
2nd valid IP	192.168.0.2/30	x.x.x.00000010
Broadcast address	192.168.0.3/30	x.x.x.00000011

Calculating VLSM example (cont.)

- Subnet 192.168.0.0/24 into smaller subnet
  - Subnet mask with /27

Description	Decimal	Binary
Network Address	192.168.0.32/27	x.x.x.000**00000**
Valid IP range 192.168.0.33 - 192.168.0.62		x.x.x.000**00001**
		x.x.x.000**00010**
Broadcast address	192.168.0.63/30	x.x.x.000**11111**

Calculating VLSM example (cont.)

- Subnet 192.168.0.0/24 into smaller subnet
  - Subnet mask with /27

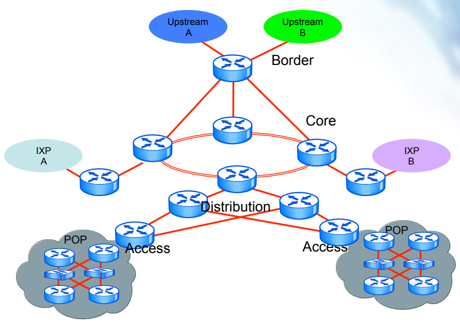
Description	Decimal	VSLM	Host	Host range
1st subnet	192.168.0.0/27	**x.x.x.000**	**00000**	0-31
2nd subnet	192.168.0.32/27	**x.x.x.001**		31-63
3rd subnet	192.168.0.64/27	**x.x.x.010**		64-95
4th subnet	192.168.0.96/27	**x.x.x.011**		96-127

n = 5 (n is the remaining subnet bits)
2n – 5 = 30 host per subnet

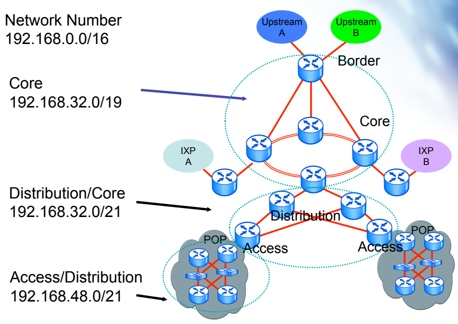
Addressing Hierarchy

- Support for easy troubleshooting, upgrades and manageability of networks
- Performance optimisation
  - Scalable and more stable
  - Less network resources overhead (CPU, memory, buffers, bandwidth)
- Faster routing convergence

Addressing Hierarchy example



Addressing Hierarchical (cont.)



Classful and classless

- Classful *(Obsolete)*
  - Wasteful address architecture
    - network boundaries are fixed at 8, 16 or 24 bits (class A, B, and C)
- **Classless** Best Current Practice
  - Efficient architecture
    - network boundaries may occur at any bit (e.g. /12, /16, /19, /24 etc)
- CIDR
    - Classless Inter Domain Routing architecture
  - Allows *aggregation* of routes within ISPs infrastructure

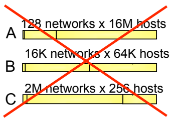
RFC 1517
RFC 1518
RFC 1519

APNIC Asia Pacific Network Information Centre

Classless & classful addressing

Best Current Practice

Classful



Obsolete
- *inefficient*
- *depletion of B space*
- *too many routes from C space*

Classless

Addresses	Prefix	Classful	Net mask
...	...	...	...
8	/29		255.255.255.248
16	/28		255.255.255.240
32	/27		255.255.255.224
64	/26		255.255.255.192
128	/25		255.255.255.128
256	/24	1 C	255.255.255.0
...	...	...	...
4096	/20	16 Cs	255.255.240.0
8192	/19	32 Cs	255.255.224
16384	/18	64 Cs	255.255.192
32768	/17	128 Cs	255.255.128
65536	/16	1 B	255.255.0.0
...	...	...	... *

* See back of slide booklet for complete chart

- Network boundaries may occur at *any* bit

APNIC Asia Pacific Network Information Centre

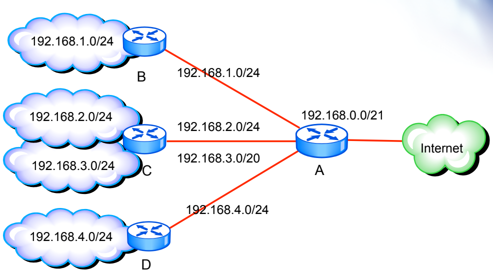
Prefix routing / CIDR

- Prefix routing commonly known as classless inter domain routing (CIDR)
  - It allows prefix routing and summarisation with the routing tables of the Internet
- RFCs that talks about CIDR
  - *RFC 1517* Applicability statement for the implementation of CIDR
  - *RFC 1518* Architecture for IP address allocation with CIDR
  - *RFC 1519* CIDR : an address assignment and aggregation strategy
  - *RFC 1520* Exchanging routing information access provider boundaries in a CIDR environment

APNIC Asia Pacific Network Information Centre

CIDR solution advantage

- CIDR offers the advantages reducing the routing table size of the network by summarising the ISP announcement in a single /21 advertisement

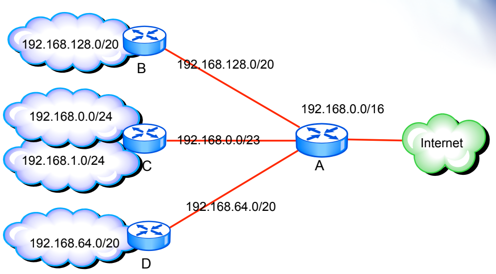


Route summarisation

- Allows the presentation of a series of networks in a single summary address.
- Advantages of summarisation
  - Faster convergence
  - Reducing the size of the routing table
  - Simplification
  - Hiding Network Changes
  - Isolate topology changes

Summarisation example

- Router C summarises its networks (2 x/24) before announcing to its neighbors (routers B and D)
- Router A combined the networks received from B, C, D and announce it as single /16 routing to Internet



Route summarisation

- Subnet 192.168.0.0/24 and 192.168.1.0/24 combining then to become a bigger block of address “/23”

Network	Subnet Mask	Binary
192.168.0.0	255.255.255.0	x.x.0000**0000.x**
192.168.1.0	255.255.255.0	x.x.0000**0001.x**
Summary	192.168.0.0/23	x.x.0000**0000.x**
192.168.0.0	255.255.254.0	x.x.0000**0000.x**

Configuring summarisation

- Manual configuration is required with the use of newer routing protocols
  - Each of the routing protocols deal with it in a slightly different way
- All routing protocols employ some level of automatic summarisation depending on the routing protocol behavior (be cautious about it)

Manual summarisation

- Manual summarisation uses by OSPF are more sophisticated.
  - Sends the subnet mask including the routing update which allows the use of VLSM and summarisation
- Performs a lookup to check the entire database and acts on the longest match

Discontiguous networks

- A network not using routing protocol that support VLSM creates problem
  - Router will not know where to send the traffic
  - Creates routing loop or duplication
- Summarisation is not advisable to network that are discontiguous
  - Turn off summarisation
    - Alternative solution but understand the scaling limitation
    - Find ways to re-address the network
  - Can create disastrous situation

Questions?

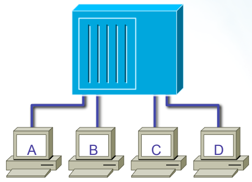
Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

Objectives

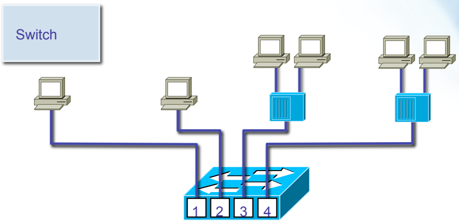
- To be able to gain knowledge about the foundation of the routing protocols
- Classify the difference between a classful and classless routing architecture
- Compare distance vector and link-state protocol operation
- Describe the information written inside the routing table

Routing Fundamental Physical Layer



All workstation will be in the same collision domain
All workstation will be in the same broadcast domain
Workstations will share the total bandwidth

Routing fundamental Data Link Layer

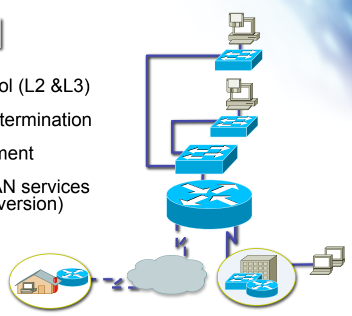


Each port will have its own collision domain
All ports will be in the same broadcast (LAN) domain

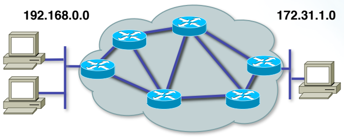
Routing fundamental Network Layer

Router

Broadcast control (L2 &L3)
Optimal path determination
Traffic management
Connects to WAN services (Protocol conversion)

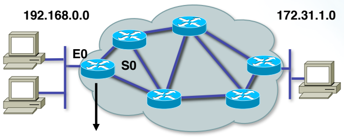


What is Routing?



- To route, a router needs to know:
  - Destination addresses
  - Sources it can learn from
  - Possible routes
  - Best route
  - Maintain and verify routing information

What is Routing? (cont.)



Network Protocol	Destination Network	Exit Interface
Connected	192.168.0.0	E0
Learned	172.31.1.0	S0

Routed Protocol: IP

Routers must learn destinations that are not directly connected

Static and Dynamic Routing

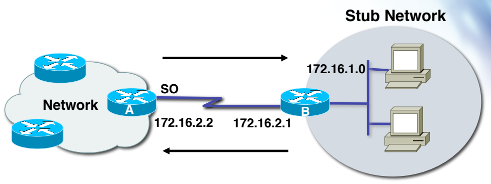
- Static Route

A route that a network administrator enters into the router manually

- Dynamic Route

A route that a network routing protocol adjusts automatically for topology or traffic changes

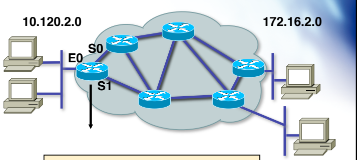
Static Routing



Configure unidirectional static routes to and from a stub network to allow communications to occur.

Dynamic Routing

- Routing protocols are used between routers to determine paths and maintain routing tables.
- Once the path is determined a router can route a routed protocol.



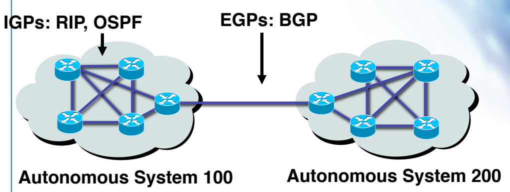
Network Protocol	Destination Network	Exit Interface
Connected	10.120.2.0	E0
RIP	172.16.2.0	S0
IGRP	172.17.3.0	S1

Routed Protocol: IP
Routing protocol: RIP, IGRP

What is a dynamic routing protocol?

- A set of rules defined to facilitate the exchanges of routing information between routers (Layer 3 device) inside networks
- Build routing tables dynamically to let the route find its path in a network having more than one path to a remote network.
- Maintains the devices connectivity within the network about the available network connections.

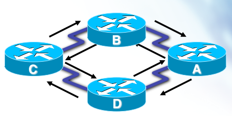
Interior or Exterior Routing Protocols

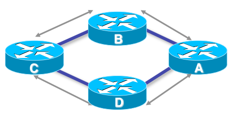


- An autonomous system is a collection of networks under a common administrative domain
- IGPs operate within an autonomous system
- EGPs connect different autonomous systems

Classes of Routing Protocols

Distance Vector





Link State

Routing protocol behavior

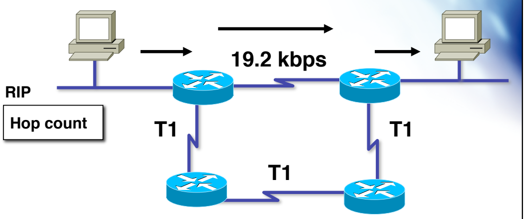
- Mechanism to update Layer 3 routing devices, to route the data across the best path
- Learns participating routers advertised routes to know their neighbors
- Learned routes are stored inside the routing table

Distance Vector Routing Protocol

- Pass periodic copies of routing table to neighbor routers
- Accumulate metric on every router (I.e Hop count)

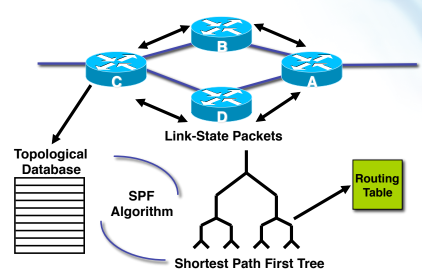
61

Distance Vector—Best Route selection



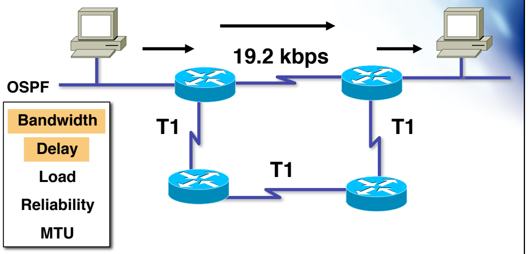
Information used to select the best path for routing

Link-State Routing Protocols



After initial flood, pass small event-triggered link-state updates to all other routers

Link State—Best Route selection



Information used to select the best path for routing

Distinction between *routed* and *routing* protocols

- Routed protocols
  - Layer3 datagram that carry the information required in transporting the data across the network
- Routing protocols
  - Handles the updating requirement of the routers within the network for determining the path of the datagram across the network

Routing and *routed* protocols

Routed protocol	Routing protocol
AppleTalk	RTMP, AURP, EIGRP
IPX	RIP, NLSP, EIGRP
Vines	RTP
DecNet IV	DecNet
IP	RIPv2, OSPF, IS-IS, BGP and (Cisco Systems proprietary) EIGRP,

Metric field

- To determine which path to use if there are multiple paths to the remote network
- Provide the value to select the best path
- But take note of the administrative distance selection process ☺

APNIC Asia Pacific Network Information Centre

Routing protocol metrics

Routing protocol	Metric
RIPv2	Hop count
EIGRP	Bandwidth, delay, load, reliability, MTU
OSPF	Cost (the higher the bandwidth indicates a lowest cost)
IS-IS	Cost

APNIC Asia Pacific Network Information Centre

Administrative distance

- Is the method used for selection of route priority of IP routing protocol, the lowest administrative distance is preferred
  - Manually entered routes are preferred from dynamically learned routes
    - Static routes
    - Default routes
  - Dynamically learned routes depend on the routing protocol metric calculation algorithm and default metrics values the smallest metric value are preferred

APNIC Asia Pacific Network Information Centre

Administrative distance chart (Cisco)

Route sources	Default distance
Connected interface	**0**
Static route out an interface	**0**
Static route to a next hop	**1**
External BGP	**20**
IGRP	**100**
OSPF	**110**
IS-IS	**115**
RIP v1, v2	**120**
EGP	**140**
Internal BGP	**200**
Unknown	**255**

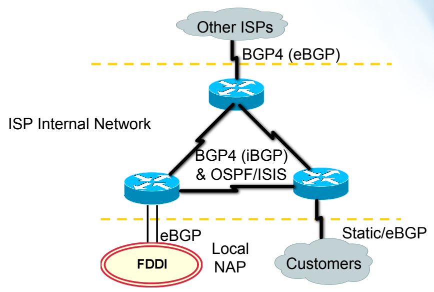
Principles of addressing

- Separate customer & infrastructure address pools
  - Manageability
    - Different personnel manage infrastructure and assignments to customers
  - Scalability
    - Easier renumbering - customers are difficult, infrastructure is relatively easy

Principles of addressing

- Further separate infrastructure
  - 'Static' infrastructure examples
    - RAS server address pools, CMTS
    - Virtual web and content hosting LANs
    - Anything where there is no dynamic route calculation
- Customer networks
    - Carry in iBGP, do not put in IGP
      - No need to aggregate address space carried in iBGP
      - Can carry in excess of 100K prefixes

Hierarchy of routing protocols



Questions?

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

Purpose of naming

- Addresses are used to locate objects
- Names are easier to remember than numbers
- You would like to get to the address or other objects using a name
- DNS provides a mapping from names to resources of several types

Naming History

- 1970's ARPANET
  - Host.txt maintained by the SRI-NIC
  - pulled from a single machine
  - Problems
    - traffic and load
    - Name collisions
    - Consistency
- DNS created in 1983 by Paul Mockapetris (RFCs 1034 and 1035), modified, updated, and enhanced by a myriad of subsequent RFCs

DNS

- A lookup mechanism for translating objects into other objects
- A globally distributed, loosely coherent, scalable, reliable, dynamic database
- Comprised of three components
  - A "name space"
  - Servers making that name space available
  - Resolvers (clients) which query the servers about the name space

DNS Features: Global Distribution

- Data is maintained locally, but retrievable globally
  - No single computer has all DNS data
- DNS lookups can be performed by any device
- Remote DNS data is locally cachable to improve performance

DNS Features: Loose Coherency

- The database is always internally consistent
  - Each version of a subset of the database (a zone) has a serial number
    - The serial number is incremented on each database change
- Changes to the master copy of the database are replicated according to timing set by the zone administrator
- Cached data expires according to timeout set by zone administrator

DNS Features: Scalability

- No limit to the size of the database
  - One server has over 20,000,000 names
    - Not a particularly good idea
- No limit to the number of queries
  - 24,000 queries per second handled easily
- Queries distributed among masters, slaves, and caches

DNS Features: Reliability

- Data is replicated
  - Data from master is copied to multiple slaves
- Clients can query
  - Master server
  - Any of the copies at slave servers
- Clients will typically query local caches

DNS Features: Dynamicity

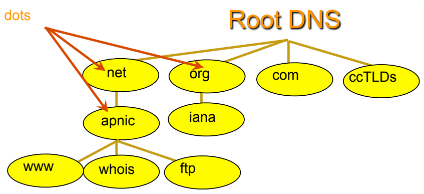
- Database can be updated dynamically
  - Add/delete/modify of any record
- Modification of the master database triggers replication
  - Only master can be dynamically updated
    - Creates a single point of failure

Concept: DNS Names

- How names appear in the DNS
  - Fully Qualified Domain Name (FQDN)
    - `WWW.APNIC.NET.`
  - labels separated by dots
- DNS provides a mapping from FQDNs to resources of several types
- Names are used as a key when fetching data in the DNS

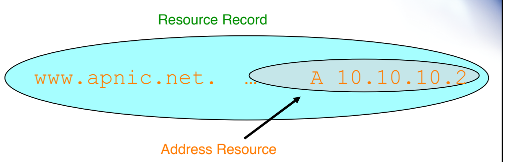
Concept: DNS Names contd.

- Domain names can be mapped to a tree
- New branches at the 'dots'



Concept: Resource Records

- The DNS maps names into data using Resource Records.

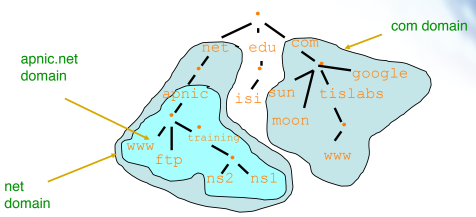


- More detail later

Concept: Domains

- Domains are "namespaces"
- Everything below *.com* is in the com domain
- Everything below *apnic.net* is in the apnic.net domain and in the net domain

Concept: Domains



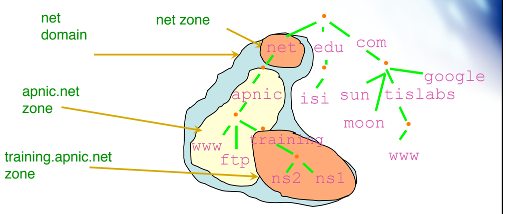
Delegation

- Administrators can create subdomains to group hosts
  - According to geography, organizational affiliation or any other criterion
- An administrator of a domain can delegate responsibility for managing a subdomain to someone else
  - But this isn't required
- The parent domain retains links to the delegated subdomain
  - The parent domain "remembers" who it delegated the subdomain to

Concept: Zones and Delegations

- Zones are "administrative spaces"
- Zone administrators are responsible for portion of a domain's name space
- Authority is delegated from a parent and to a child

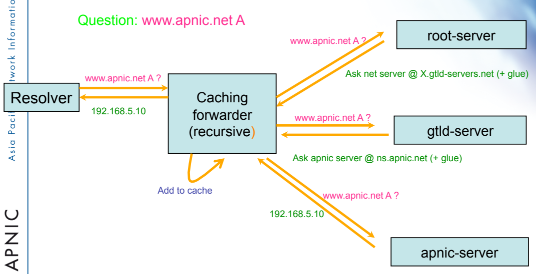
Concept: Zones and Delegations



Concept: Name Servers

- Name servers answer ‘DNS’ questions
- Several types of name servers
  - Authoritative servers
    - master (primary)
    - slave (secondary)
  - (Caching) recursive servers
    - also caching forwarders
  - Mixture of functionality

Concept: Resolving process & Cache



Concept: Resource Records

- Resource records consist of it's name, it's TTL, it's class, it's type and it's RDATA
- TTL is a timing parameter
- IN class is widest used
- There are multiple types of RR records
- Everything behind the type identifier is called rdata

```
www.apnic.net.    3600    IN    A    10.10.10.2
```

Label, ttl, class, type, rdata

Example: RRs in a zone file

```
apnic.net. 7200 IN      SOA     ns.apnic.net. admin.apnic.net.
                (
                        2001061501      ; Serial
                        43200   ; Refresh 12 hours
                        14400   ; Retry 4 hours
                        345600  ; Expire 4 days
                        7200    ; Negative cache 2 hours )

apnic.net.              7200  IN    NS      ns.apnic.net.
apnic.net.              7200  IN    NS      ns.ripe.net.

whois.apnic.net.        3600  IN    A       193.0.1.162

host25.apnic.net.       2600  IN    A       193.0.3.25
```

Label, ttl, class, type, rdata

Resource Record: SOA and NS

- The SOA and NS records are used to provide information about the zone itself
- The NS indicates where information about a given zone can be found

```
apnic.net.  7200  IN    NS      ns.apnic.net.
apnic.net.  7200  IN    NS      ns.ripe.net.
```

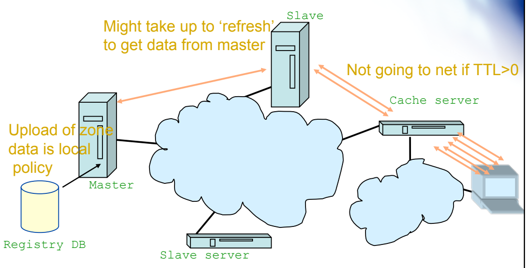
- The SOA record provides information about the start of authority, i.e. the top of the zone, also called the APEX

Concept: TTL and other Timers

- TTL is a timer used in caches
  - An indication for how long the data may be reused
  - Data that is expected to be 'stable' can have high TTLs
- SOA timers are used for maintaining consistency between primary and secondary servers

Places where DNS data lives

- Changes do not propagate instantly



To remember...

- Multiple authoritative servers to distribute load and risk:
  - Put your name servers apart from each other
- Caches to reduce load to authoritative servers and reduce response times
- SOA timers and TTL need to be tuned to needs of zone. Stable data: higher numbers

Performance of DNS

- Server hardware requirements
- OS and the DNS server running
- How many DNS servers?
- How many zones expected to load?
- How large the zones are?
- Zone transfers
- Where the DNS servers are located?
- Bandwidth

Performance of DNS

- Are these servers Multihomed?
- How many interfaces are to be enabled for listening?
- How many queries are expected to receive?
- Recursion
- Dynamic updates?
- DNS notifications

Writing a zone file

- Zone file is written by the zone administrator
- Zone file is read by the master server and it's content is replicated to slave servers
- What is in the zone file will end up in the database
- Because of timing issues it might take some time before the data is actually visible at the client side

First attempt

- The 'header' of the zone file
  - Start with a SOA record
  - Include authoritative name servers and
  - Add other information
- Add other RRs
- Delegate to other zones

APNIC Asia Pacific Network Information Centre

Authoritative NS records and related A records

```
apnic.net.       3600  IN NS   NS1.apnic.net.
apnic.net.       3600  IN NS   NS2.apnic.net.
NS1.apnic.net.   3600  IN A    203.0.0.4
NS2.apnic.net.   3600  IN A    193.0.0.202
```

- NS record for all the authoritative servers
  - They need to carry the zone at the moment you publish
- A records only for "in-zone" name servers
  - Delegating NS records might have glue associated

APNIC Asia Pacific Network Information Centre

Zone file format short cuts nice formatting

```
apnic.net.        3600  IN SOA NS1.apnic.net. admi
n\.email.apnic.net. (
               2002021301    ; serial
               1h          ; refresh
               30M         ; retry
               1W          ; expiry
               3600 )        ; neg. answ. Ttl

apnic.net.     3600 IN NS   NS1.apnic.net.
apnic.net.     3600 IN NS   NS2.apnic.net.
apnic.net.         3600 IN MX   50   mail.apnic.net.
apnic.net.             3600 IN MX   150  mailhost2.apnic.net.

apnic.net.             3600 IN TXT  "Demonstration and test zone"
NS1.apnic.net.     4500 IN A    203.0.0.4
NS2.apnic.net.     3600 IN A    193.0.0.202
localhost.apnic.net.   3600 IN A    127.0.0.1
www.apnic.net.     3600 IN CNAME IN.apnic.net.
```

APNIC Asia Pacific Network Information Centre

Zone file short cuts: repeating last name


```
apnic.net.        3600  IN SOA NS1.apnic.net. admi
n\.email.apnic.net. (
              2002021301 ; serial
              1h      ; refresh
              30M         ; retry
              1W      ; expiry
              3600 )      ; neg. answ. Ttl
               3600 IN NS   NS1.apnic.net.
                   3600 IN NS   NS2.apnic.net.
           3600 IN MX   50   mail.apnic.net.
           3600 IN MX   150  mailhost2.apnic.net.

           3600 IN TXT  "Demonstration and test zone"
NS1.apnic.net.     3600 IN A    203.0.0.4
NS2.apnic.net.     3600 IN A    193.0.0.202

localhost.apnic.net.   4500 IN A    127.0.0.1

NS1.apnic.net.     3600 IN A    203.0.0.4
www.apnic.net.     3600 IN CNAME IN.apnic.net.
```

Zone file short cuts: default TTL


```
$TTL    3600 ; Default TTL directive
apnic.net.      IN SOA NS1.apnic.net. admin\.email.apnic.net. (
               2002021301    ; serial
               1h         ; refresh
               30M        ; retry
               1W         ; expiry
               3600 )     ; neg. answ. Ttl
               IN NS   NS1.apnic.net.
                   IN NS   NS2.apnic.net.
           IN MX   50  mail.apnic.net.
           IN MX   150 mailhost2.apnic.net.

           IN TXT  "Demonstration and test zone"
NS1.apnic.net.     IN A   203.0.0.4
NS2.apnic.net.     IN A   193.0.0.202

localhost.apnic.net. 4500 IN A    127.0.0.1

NS1.apnic.net.         IN A    203.0.0.4
www.apnic.net.     IN CNAME NS1.apnic.net.
```

Zone file short cuts: ORIGIN

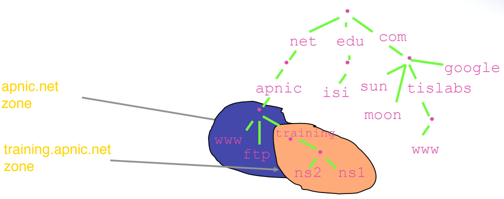

```
$TTL    3600 ; Default TTL directive
$ORIGIN apnic.net.
@           IN SOA NS1 admin\.email.apnic.net. (
               2002021301    ; serial
               1h         ; refresh
               30M        ; retry
               1W         ; expiry
               3600 )     ; neg. answ. Ttl
           IN NS   NS1
               IN NS   NS2
       IN MX   50  mailhost
       IN MX   150 mailhost2


       IN TXT  "Demonstration and test zone"
NS1         IN A    203.0.0.4
NS2         IN A    193.0.0.202

localhost 4500 IN A     127.0.0.1

NS1             IN A    203.0.0.4
www         IN CNAME NS1
```

Zone file short cuts: Eliminate IN

```
$TTL    3600 ; Default TTL directive
$ORIGIN apnic.net.
@           SOA NS1 admin\.email.sanog.org. (
                2002021301    ; serial
                1h         ; refresh
                30M        ; retry
                1W         ; expiry
                3600 )     ; neg. answ. Ttl
            NS   NS1
                NS   NS2
        MX   50  mailhost
        MX   150 mailhost2


        TXT  "Demonstration and test zone"
NS1          A    203.0.0.4
NS2          A    193.0.0.202

localhost 4500  A    127.0.0.1

NS1             A    203.0.0.4
www          CNAME NS1
```

Delegating a zone (becoming a parent)

- Delegate authority for a sub domain to another party (splitting of *training.apnic.net* from *apnic.net*)



Concept: Glue

- Delegation is done by adding NS records:

```
training.apnic.net.     NS ns1.training.apnic.net.
training.apnic.net.     NS ns2.training.apnic.net.
training.apnic.net.     NS ns1.apnic.net.
training.apnic.net.     NS ns2.apnic.net.
```

- How to get to ns1 and ns2… We need the addresses
- Add glue records to so that resolvers can reach ns1 and ns2

```
ns1.training.apnic.net. A 10.0.0.1
ns2.training.apnic.net. A 10.0.0.2
```

APNIC Asia Pacific Network Information Centre

Concept: Glue contd.

- Glue is 'non-authoritative' data
- Don't include glue for servers that are not in sub zones

```
training.apnic.net.     NS    ns1.training.apnic.net.
Training.apnic.net.     NS  ns2.training.apnic.net.

training.apnic.net.     NS    ns2.apnic.net.
training.apnic.net.     NS    ns1.apnic.net.
ns1.training.apnic.net. A      10.0.0.1
Ns2.training.apnic.net. A  10.0.0.2
```

Only this record needs glue

APNIC Asia Pacific Network Information Centre

Delegating training.apnic.net. from apnic.net.

training.apnic.net

Setup minimum two servers
Create zone file with NS records
Add all training.apnic.net data

apnic.net

Add NS records and glue

Make sure there is no other data from the training.apnic.net. zone in the zone file

APNIC Asia Pacific Network Information Centre

Questions?

Reverse DNS

Overview

- Principles
- Creating reverse zones
- Setting up nameservers
- Reverse delegation procedures

What is 'Reverse DNS'?

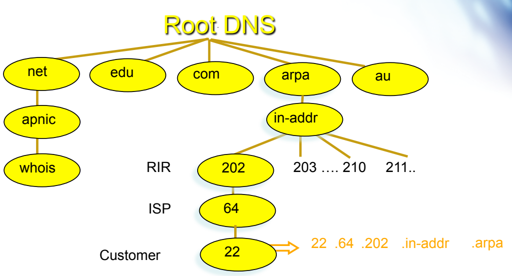
- 'Forward DNS' maps names to numbers
  - svc00.apnic.net -> 202.12.28.131
- 'Reverse DNS' maps numbers to names
  - 202.12.28.131 -> svc00.apnic.net

Reverse DNS - why bother?

- Service denial
  - That only allow access when fully reverse delegated eg. anonymous ftp
- Diagnostics
  - Assisting in trace routes etc
- SPAM identifications
- Registration responsibilities

Principles – DNS tree

- Mapping numbers to names - 'reverse DNS'



Creating reverse zones

- Same as creating a forward zone file
  - SOA and initial NS records are the same as normal zone
  - Main difference
    - need to create additional PTR records
- Can use BIND or other DNS software to create and manage reverse zones
  - Details can be different

Creating reverse zones - contd

- Files involved
  - Zone files
    - Forward zone file
      - e.g. db.domain.net
    - Reverse zone file
      - e.g. db.192.168.254
  - Config files
    - <named.conf>
  - Other
    - Hints files etc.
      - Root.hints

Start of Authority (SOA) record

```
<domain.name.>      CLASS   SOA     <hostname.domain.name.>
 <mailbox.domain.name> (
      <serial-number>
                       <refresh>
                        <retry>
                        <expire>
                       <negative-caching> )
```

253.253.192.in-addr.arpa.

Pointer (PTR) records

- Create pointer (PTR) records for each IP address

```
131.28.12.202.in-addr.arpa. IN PTR svc00.apnic.net.
```

or

```
131          IN      PTR            svc00.apnic.net.
```

A reverse zone example

```
$ORIGIN 1.168.192.in-addr.arpa.
@ 3600  IN SOA test.company.org. (
         sys\.admin.company.org.
         2002021301    ; serial
         1h     ; refresh
         30M       ; retry
         1W     ; expiry
         3600 )        ; neg. answ. ttl

  NS  ns.company.org.
  NS  ns2.company.org.

1  PTR    gw.company.org.
      router.company.org.

2  PTR    ns.company.org.
;auto generate:  65 PTR host65.company.org
$GENERATE 65-127 $ PTR host$.company.org.
```

Setting up the primary nameserver

- Add an entry specifying the primary server to the *named.conf* file

```
zone "<domain-name>" in {
type master;
file "<path-name>"; };
```

- <domain-name>
  - Ex: 28.12.202.in-addr.arpa.
- <type master>
  - Define the name server as the primary
- <path-name>
  - location of the file that contains the zone records

Setting up the secondary nameserver

- Add an entry specifying the primary server to the *named.conf* file

```
zone "<domain-name>" in {
type slave;
file "<path-name>";
Masters { <IP address> ; }; };
```

- <type slave> defines the name server as the secondary
- <ip address> is the IP address of the primary name server
- <domain-name> is same as before
- <path-name> is where the back-up file is

Reverse delegation requirements

- /24 Delegations
  - Address blocks should be assigned/allocated
  - At least two name servers
- /16 Delegations
  - Same as /24 delegations
  - APNIC delegates entire zone to member
  - Recommend APNIC secondary zone
- < /24 Delegations
  - Read "classless in-addr.arpa delegation"

RFC 2317

APNIC & ISPs responsibilities

- APNIC
  - Manage reverse delegations of address block distributed by APNIC
  - Process organisations requests for reverse delegations of network allocations
- Organisations
  - Be familiar with APNIC procedures
  - Ensure that addresses are reverse-mapped
  - Maintain nameservers for allocations
    - Minimise pollution of DNS

Subdomains of in-addr.arpa domain

- Example: an organisation given a /16
  - 192.168.0.0/16 (one zone file and further delegations to downstreams)
  - 168.192.in-addr.arpa zone file should have:

```
0.168.192.in-addr.arpa.    NS ns1.organisation0.com.
0.168.192.in-addr.arpa.    NS ns2.organisation0.com.
1.168.192.in-addr.arpa.    NS ns1.organisation1.com.
1.168.192.in-addr.arpa.    NS ns2.organisation1.com.
2.168.192.in-addr.arpa.    NS ns1.organisation2.com.
2.168.192.in-addr.arpa.    NS ns2.organisation2.com.
        :
        :
```

Subdomains of in-addr.arpa domain

- Example: an organisation given a /20
  - 192.168.0.0/20 (a lot of zone files!) – have to do it per /24)
  - Zone files

0.168.192.in-addr.arpa.
1.168.192.in-addr.arpa.
2.168.192.in-addr.arpa.
:
:
15.168.192.in-addr.arpa.

Reverse delegation procedures

- Standard APNIC database object,
  - can be updated through MyAPNIC, Online form or via email.
- Nameserver/domain set up verified before being submitted to the database.
- Protection by maintainer object
- Zone file updated instantly

Creation of domain objects

- If you opt to create the domain objects yourself
  - Either you can use MyAPNIC
  - Or use web/email templates
- Using web/email templates will result in initial errors
  - As the /8 is hierarchically maintained by MAINT-AP-DNS
  - Contact <helpdesk@apnic.net>

Whois domain object

```
domain:      28.12.202.in-addr.arpa
descr:       in-addr.arpa zone for 28.12.202.in-addr.arpa
admin-c:     DNS3-AP
tech-c:      DNS3-AP
zone-c:      DNS3-AP
nserver:     ns.telstra.net
nserver:     rs.arin.net
nserver:     ns.myapnic.net
nserver:     svc00.apnic.net
nserver:     ns.apnic.net
mnt-by:      MAINT-APNIC-AP
mnt-lower:   MAINT-DNS-AP
changed:     inaddr@apnic.net 19990810
source:      APNIC
```

Reverse Zone

Contacts

Name Servers

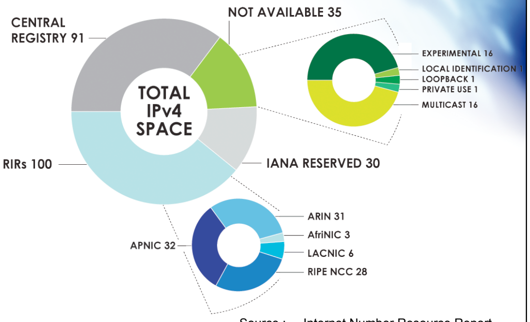
Maintainers (protection)

Questions?

Overview

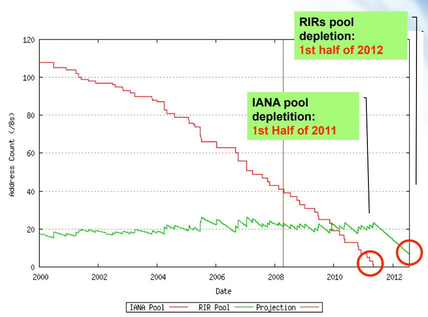
- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

How many IPv4 IANA pool available



Source : Internet Number Resource Report - Number Resource Organization (NRO)

Projected lifetime of remaining IPv4 addresses



137

According to this model

- IANA unallocated address pool will be exhausted
  - 10 May 2010
  - This is the model's predicted date as of 22nd October 2007
  - Tomorrow's prediction will be different

Ref: IPv4 unallocated address space exhaustion by Geoff Huston, Sept 2007

So what will happen after the exhaustion?

- The Internet will not stop but its growth will be impacted
- Who will be impacted?
  – ISPs
    - Sustaining their business models will become more difficult unless you have huge IPv4 address blocks
  – End users
    - Cost of access to the Internet will increase

Some possible scenarios

- So what will happen after the IPv4 unallocated address space exhaustion?
  – Persist in IPv4 networks using more NATs
  – Address markets emerging for IPv4
  – Routing fragmentation
  – IPv6 transition

Ref: IPv4 unallocated address space exhaustion by Geoff Huston, Sept 2007

IPv4 NATs today

- Today NATs are largely externalised costs for ISPs
  – Customers buy and operate NATs
  – Applications are tuned to single-level-NAT traversal
  – Static public addresses typically attract a traffic premium in the real market
    - For retail customers, IP addresses already have a market price!

Ref: IPv4 unallocated address space exhaustion by Geoff Huston, Sept 2007

APNIC Asia Pacific Network Information Centre

The "Just" add more NATs option

- Demand for increasing NAT "intensity"
  - Shift ISP infrastructure to private address realms
  - Multi-level NAT deployment both at the customer edge and within the ISP network
    - This poses issues in terms of application discovery and adaptation to NAT behaviours
  - End cost for static public addresses may increase
- How far can NATs scale?
  - Not well known
  - What are the critical resources here
    - Nat biding capability and state maintenance, NAT packet throughput, private address pool sizes and application complexity

Ref: IPv4 unallocated address space exhaustion by Geoff Huston, Sept 2007

APNIC Asia Pacific Network Information Centre

Recovering unused IPv4 address space

- 46 x /8 (in various prefixes) un-routed address spaces existing
  - APNIC and LACNIC have active reclamation processes
  - However, recovery of such address space is not easy
    - Most of historical address space exist in USA
    - Historical address space: address distributed before the RIR mechanism kicked into the system
    - Reclamation processes are not only likely to be lengthy and difficult, but also expensive
    - Most likely "address market" will emerge
  - Amount of recoverable address space is relatively insignificant
  - Fragmented address blocks
    - Increase injection to the global routing table
- Only provides limited solutions

Ref: APster IIssues 23 – Septemner 2007, "Responses to IPv4 address space consumption" By Paul Wilson

APNIC Asia Pacific Network Information Centre

Reuse of 240/4 address space for private use

- APNIC's Paul Wilson and Geoff Huston submitted an Internet draft recently
  - draft-wilson-class-e
  - Proposes the redesigtation of the IPv4 address block 240/4 from "Future Use" (originally designated to IETF as "Class E") to "Limited Use for Large Private Internet"
- To prepare the future demands of large networks that will be deployed behind NAT
  - Such networks large enough to exceed the exisitng private address space available under RFC1918 (defining IPv4 private address space)
- To allow an extended period of dual stack IPv4 /IPv6 networks

Ref: APster IIssues 23 – Septemner 2007, "Reuse of 240/4 address space for private use"

Transition to IPv6

- But IPv6 is not backward compatible with IPv4 on the wire
- So the plan is that we need to run some form of a “dual stack” transition process
  - running both IPv4 and IPv6 protocol stacks in the host
  - Or dual stack via protocol translating proxies

IPv6 is the only alternative technology mature enough to be successfully deployed

Ref: IPv4 unallocated address space exhaustion by Geoff Huston, Sept 2007

APNIC Asia Pacific Network Information Centre

What is IPv6?

- IPv6 is a new version of the Internet layer protocol (IP) in the TCP/IP suite of protocols.
- It replaces the current Internet protocol layer commonly referred to as IPv4

APNIC Asia Pacific Network Information Centre

MAC layer address resolution

- IPv4
  - ARP (Address Resolution Protocol)
  - Hosts maintain a table of the link-layer addresses corresponding to IP addresses
  - If no corresponding MAC address is found in this table, ARP request will be broadcasted
  - A host who knows the answer will send an ARP reply
  - ARP has some issues: security
    - No guarantee that it has actually come from the correct system
- IPv6 considerably improves host-to-address mapping mechanism
  - Neighbour discovery
  - ICMP Neighbour Discovery is an IP protocol
  - It can be secured by IPsec
  - It includes the link-layer addresses within the body of messages

Ref: IPv6 Network Administration, p9

APNIC Asia Pacific Network Information Centre

ICMPv6

- ICMPv6 is very different from ICMP in IPv4
  - Encompasses the roles filled by ICMP, IGMP (Internet Group Management Protocol) and ARP in the IPv4 world
  - ICMPv6 neighbour discovery packets: two types of packets
    - Neighbour Solicitation
      - Very similar to an ARP request packet
      - Send a request to translate a target IPv6 unicast address into a link-layer address
      - "The owner of this IPv6 address please contact me"
      - Sent via solicited node multicast address (not broadcast)
        - » Reserved address space
        - » Ff02::1:ff00:0/104
    - Neighbour Advertisement
      - Reply to the above query: "I am the MAC address for the IPv6 address you are looking for"
      - Used during Duplicate Address Detection (DAD)

Ref: IPv6 Network Administration, p9

Main IPv6 benefits - summary

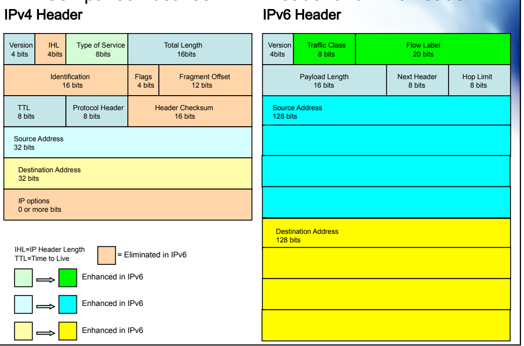
- Expanded addressing capabilities
- Server-less autoconfiguration ("plug-n-play") and reconfiguration
- More efficient and robust mobility mechanisms
- Built-in, strong IP-layer encryption and authentication (but must be configured)
- Streamlined header format and flow identification
- Improved support for options / extensions

RFC2460

- "Internet Protocol Version 6 Specification"
- Changes from IPv4 to IPv6:
  - Expanded addressing capabilities
  - Header format simplification
  - Improved support for extensions and options
  - Flow labeling capability
  - Authentication and privacy capabilities

IPv6 header

- Comparison between IPv4 header and IPv6 header



IPv6 header

- IPv6 header is considerably simpler than IPv4
  - IPv4: 12 fields + options , IPv6: 8 fields + options
- IPv4 header less flexible – cannot exceed 60 bytes
- Eliminated fields in IPv6
  - Header Length
  - Identification
  - Flag
  - Fragmentation Offset
  - Checksum
- Enhanced fields in IPv6
  - TOS =>Traffic Class
  - Time to Live => Hop Limit
  - Protocol => Next header (extension headers)
  - New Flow Label
- Authentication and privacy capabilities

The fields in the IPv6 header

Version	4 bits		Version of the protocol = 6
Traffic class	1 byte		Used to distinguish priorities of IPv6 packets
Flow label	20 bits		Used to label sequences of packets that require the same treatment for more efficient processing on routers.
Payload length	2 bytes		Length of data carried after IPv6 header
Next header	1 byte		Contains a protocol number or a value for an extension header
Hop limit	1 byte		Number of hops. Decremented by one by every router
Source address	16 bytes		
Destination address	16 bytes		

IPv6 Essentials by Silvia Hagen, p19

Extension headers

- The current IPv6 specification defines 6 extension headers:
  - Hop-by-hop options header
  - Routing header
  - Fragment header
  - Destination options header
  - Authentication header
  - Encrypted security payload header
- There can be zero, one, or more than one Extension header in one IPv6 packet
- Are placed between the IPv6 header and the upper-layer protocol header
- Is identified by the Next Header in the preceding header
- Provide flexibility for developing additional Extension Headers in the future if necessary
  - New Extension Headers can be added/used without changing the IPv6 header

IPv6 Essentials by Silvia Hagen, p23

IPv6 fragmentation

- IPv6 manages fragmentation differently to IPv4
- In IPv4 intermediate routers fragment a datagram that is larger than the MTU (maximum transfer unit) of the network over which it must travel
- In IPv6 fragmentation is restricted to the original source - the source machine must perform
- a PATH MTU discovery packet is sent to determine the MTU to use or a default MTU value is used.
- The fragmentation fields (identification, flags and offset value) are therefore contained in an extension header.

IPv6 addressing

- 128 bits of address space
- Divided into eight 16 bit fields, each represented as a 4 digit hexadecimal number.
  - X:X:X:X:X:X:X:X (X=16 bit number, ex: A2FE)
- Example:
  - 2001:DB8:124C:C1A2:BA03:6735:EF1C:683D
  - Abbreviated form of address uses "zero compression"
    - 2001:DB8:0023:0000:0000:036E:1250:2B00

    →2001:DB8:23:0:0:36E:1250:2B00

    →2001:DB8:23::36E:1250:2B00

    Consecutive fields of all zeros can be compressed using ::

    Can be used only once

    Leading zeros can be omitted

IPv6 address prefix

- When you do IPv6 subnetting, you need to think in binary values not in hexadecimal value
- 2001:1::/32
  =2001:0001::/32
  Hex 2001 = Binary 0010 0000 0000 0001
  Hex 0001 = Binary 0000 0000 0000 0001
- 2001:2:3::/48
  =2001:0002:0003::/48
  Hex 2001 = Binary 0010 0000 0000 0001
  Hex 0002 = Binary 0000 0000 0000 0010
  Hex 0003 = Binary 0000 0000 0000 0011
- /64s in 2001:2:3::/48 are
  - 2001:0002:0003:0001::/64
  - 2001:0002:0003:0002::/64
  - 2001:0002:0003:0003::/64
  - Etc.
  - 16 bits of address space
    - You can have 65536 /64s in one /48 IPv6 address
    - Note:: indicates the remaining 64 bits are all zeros and can then be used to identify hosts::

IPv6 address prefix

- Another example:
- 2001:1::/32
  =2001:0001::/32
  Hex 2001 = Binary 0010 0000 0000 0001
  Hex 0001 = Binary 0000 0000 0000 0001
- How about /47s in 2001:1::/32?
  Hex 2001 = Binary 0010 0000 0000 0001 = 16 bits
  Hex 0001 = Binary 0000 0000 0000 0001 = 32
  Hex 0000 = Binary 0000 0000 0000 0000 = 47 (32 bits in prefix –"fixed", 15 bits in subnet)
  So the 15 subnet bits (red) are used to identify the /47s: Subnets numbered using these bits
  Binary 0000 0000 0000 0000 = Hex 0000
  The first /47 is 2001:0001:0000::/47

  Binary 0000 0000 0000 0010 = Hex 0002
  So the second /47 is 2001:0001:0002::/47

  Binary 0000 0000 0000 0100 = Hex 0004
  So the third /47 is 2001:0001:0004::/47

  Binary 0000 0000 0000 0110 = Hex 0006
  So the fourth /47 is 2001:0001:0006::/47

  Binary 0000 0000 0000 1000 = Hex 0008
  So the fifth /47 is 2001:0001:0008 ::/47

Exercise 1: IPv6 addressing

1. Identify the first four /64 address blocks out of 2001:AA:2000::/48
   1. ____________________
   2. ____________________
   3. ____________________
   4. ____________________

Exercise 2: IPv6 addressing

1. Identify the fist four /36 address blocks out of 2001:ABC::/32
   1. ____________________
   2. ____________________
   3. ____________________
   4. ____________________

APNIC Asia Pacific Network Information Centre

Exercise 3: IPv6 addressing

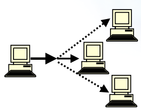
3. Identify the first six /37 address blocks out of 2001:AA::/32
   1. ____________________
   2. ____________________
   3. ____________________
   4. ____________________
   5. ____________________
   6. ____________________

APNIC Asia Pacific Network Information Centre

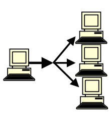
IPv6 addressing type

- **IPv6 Address type** RFC 4291
  - Unicast
    - An identifier for a single interface

  - Anycast
    - An identifier for a set of interfaces

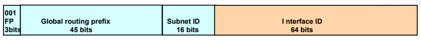


  - Multicast
    - An identifier for a group of nodes



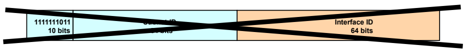
Unicast address

- Address given to interface for communication between host and router
  - Global unicast address currently delegated by IANA



  - Local use unicast address
    - Link-local address (starting with FE80::)

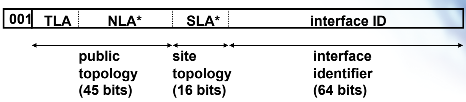
  - Site local address – deprecated (FEC0::)



APNIC Asia Pacific Network Information Centre

Aggregatable global unicast address deprecated

- RFC 2374 – deprecated



- TLA = Top-Level Aggregator
  NLA = Next-Level Aggregator(s)
  SLA = Site-Level Aggregator(s)
- This scheme has been replaced by a coordinated allocation policy defined by RIR.
- You may see them in text books, but remember they are deprecated!

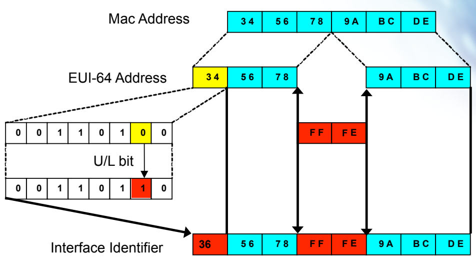
APNIC Asia Pacific Network Information Centre

Interface ID

- The lowest-order 64-bit field addresses may be assigned in several different ways:
  - auto-configured from a 48-bit MAC address expanded into a 64-bit EUI-64
  - assigned via DHCP
  - manually configured
  - auto-generated pseudo-random number (to counter some privacy concerns: RFC 3041)
  - possibly other methods in the future

APNIC Asia Pacific Network Information Centre

EUI-64



U/L bit = 0 if non-unique MAC address (A MAC address may be not unique if the administrator changes the MAC address of the Interface.)
U/L bit = 1 if unique MAC address

Zone IDs for local-use addresses

- Local-use addresses can be reused
  - Link-local addresses are reused on each link (segment)
  - Because of this characteristic, the link-local address is ambiguous
  - To specify the link on which an address is assigned, an additional identifier is needed
    - Zone Identifier – also known as an *interface id*
- The syntax of the zone id
  - Defined by RFC 4007
  - *Address%zone_ID*
    - *Address* = a local use address (a link-local address)
    - *zone-ID* = defined relative to the sending hosts
      - Different hosts can use diferent zone ID values for the same physical zone or segment.
      - E.g., Host A might choose 3 to represent the zone ID of an attached link and Host B might choose 4 to represent the same link
      - This has causes no issues since the zone id is local to the host

http://download.microsoft.com/download/e/9/b/e9bd20d3-cc8d-4162-aa60-3aa3abc2b2e9/IPv6.doc p12

Zone IDs for local-use addresses

- In Windows XP for example:
- Host A:
  - fe80::2abc:d0ff:fee9:4121%4
- Host B:
  - fe80::3123:e0ff:fe12:3001%3
- Ping from Host A to Host B
  - ping fe80::3123:e0ff:fe12:3001%4 (not %3)
    - identifies the interface zone ID on the host which is connected to that segment.

Special addresses

- The unspecified address
  - A value of 0:0:0:0:0:0:0:0 (::)
  - It is comparable to 0.0.0.0 in IPv4
  - Indicates the absence of a valid address
    - Can be used as a source address by a host during the boot process when it sends out a request for address configuration information
    - Should not be statically or dynamically assigned
    - Should not appear as a destination IP address or within an IPv6 routing header

IPv6 Essentials by Silvia Hagen, p44

Special addresses

- The loopback address
  - It is represented as 0:0:0:0:0:0:0:1 (::1)
  - Similar to 127.0.0.1 in IPv4
  - Helpful in troubleshooting and testing the IP stack
    - Can be used to send a packet to the protocol stack without sending it out on the subnet (sending a packet to self)
  - Should never be statically or dynamically assigned

IPv6 Essentials by Silvia Hagen, p44

Anycast address

- One-to-one-of-many communication
  - Delivery to a single interface
- Syntactically the same as a unicast address
- May be assigned to routers only
- Cannot be used as the source address
- Needs more widespread experience in the future

Multicast address

11111111 8 bits	Flag 4 bits	Scope 4bits	Group ID 112 bits

- First 8 bits identifies multicast address
  - 11111111 (FF)
- Flags
  - 0000 = a permanently-assigned (well-known) multicast address
  - 0001 = a non-permanently-assigned (transient) multicast address
- Scope (indicates the scope of the multicast group)
  - 1= node local
  - 2= link local
  - 3= site local
  - 8= organisation local
  - E= global
- Group ID
  - Identifies the multicast group within the specified scope
- Well-known multicast addresses
  - FF02:0:0:0:0:0:0:1 All-nodes address with Link-local scope
  - FF02:0:0:0:0:0:0:2 All-routers address with Link-local scope

Autoconfiguration

IPv6 autoconfiguration

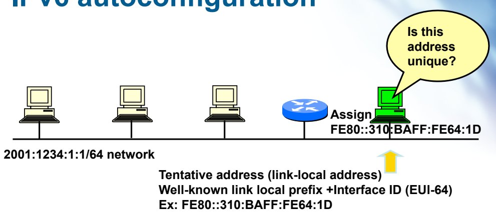
RFC 2462

- Stateless mechanism
  - For a site not concerned with the exact addresses
  - No manual configuration required
  - Minimal configuration of routers
  - No additional servers
- Stateful mechanism
  - For a site that requires tighter control over exact address assignments
  - Needs a DHCP server
    - DHCPv6

Plug and Play

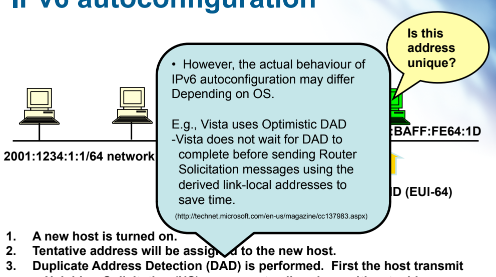
- IPv6 link local address
  - Even if no servers/routers exist to assign an IP address to a device, the device can still auto-generate an IP address
    - Allows interfaces on the same link to communicate with each other
- Stateless
  - No control over information belongs to the interface with an assigned IP address
    - Possible security issues
- Stateful
  - Remember information about interfaces that are assigned IP addresses

IPv6 autoconfiguration



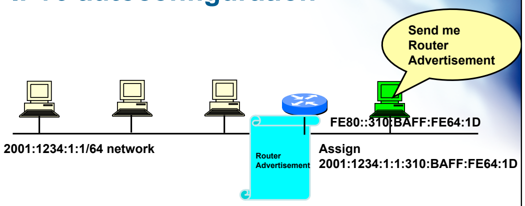
1. A new host is turned on.
2. Tentative address will be assigned to the new host.
3. Duplicate Address Detection (DAD) is performed. First the host transmit a Neighbor Solicitation (NS) message to all-nodes multicast address (FF02::1)
5. If no Neighbor Advertisement (NA) message comes back then the address is unique.
6. FE80::310:BAFF:FE64:1D will be assigned to the new host.

IPv6 autoconfiguration



1. A new host is turned on.
2. Tentative address will be assig... to the new host.
3. Duplicate Address Detection (DAD) is performed. First the host transmit a Neighbor Solicitation (NS) message to all-nodes multicast address (FF02::1)
5. If no Neighbor Advertisement (NA) message comes back then the address is unique.
6. FE80::310:BAFF:FE64:1D will be assigned to the new host.

IPv6 autoconfiguration



1. The new host will send a Router Solicitation (RS) request to the all-routers multicast group (FE02::2).
2. The router will reply with a Routing Advertisement (RA).
3. The new host will learn the network prefix. E.g, 2001:1234:1:1/64
4. The new host will be assigned a new address : network prefix+Interface ID E.g, 2001:1234:1:1:310:BAFF:FE64:1D

IPv6 features – autoconfiguration

- Keeps end user costs down
  - No need for manual configuration
  - In conjunction with the possibility of a low cost network interface
- Helpful when residential networks emerge as an important market
- But the address is not automatically registered into the DNS
- Security issues need to be considered as discussed

Workshop Exercises

- **Exercise 1: IPv6 Host Configuration**

Exercise 1: IPv6 Host Configuration

- Windows XP SP2
- **netsh interface ipv6 install**

- Windows XP
- **ipv6 install**

Exercise 1: IPv6 Host Configuration

Verify your Configuration

- c:\>ipconfig

Exercise 1: IPv6 Host Configuration

Testing your configuration

- **ping fe80::260:97ff:fe02:6ea5%4**

- **Note: the Zone id is YOUR interface index**

Workshop Exercises

- **Exercise 2: IPv6 Subnetting**

Exercise 2: IPv6 Subnetting

Global prefix received: 2001:0df0:000a::/48

Scenario:

This ISP has 6 downstream smaller ISP customers and needs to sub-allocate smaller blocks to these companies. After consideration they decide to allocate /52 blocks.

Exercise 2: IPv6 Subnetting

- Please list all available /52 subnets

Exercise 2: IPv6 Subnetting

All available subnets are:

2001:0DF0:000A:0000::/52
2001:0DF0:000A:1000::/52
2001:0DF0:000A:2000::/52
2001:0DF0:000A:3000::/52
2001:0DF0:000A:4000::/52
2001:0DF0:000A:5000::/52
2001:0DF0:000A:6000::/52
2001:0DF0:000A:7000::/52
2001:0DF0:000A:8000::/52
2001:0DF0:000A:9000::/52
2001:0DF0:000A:A000::/52
2001:0DF0:000A:B000::/52
2001:0DF0:000A:C000::/52
2001:0DF0:000A:D000::/52
2001:0DF0:000A:E000::/52
2001:0DF0:000A:F000::/52

Exercise 2: IPv6 Subnetting

- Take your /52 sub-allocation
- Create /64 subnet
- List first 2 /64 subnet

Exercise 2: IPv6 Subnetting

- ISP1 first 2 /64

2001:0DF0:000A:1000::/64
2001:0DF0:000A:1001::/64

- ISP2 first 2 /64

2001:0DF0:000A:2000::/64
2001:0DF0:000A:2001::/64

- ISP3 first 2 /64

2001:0DF0:000A:3000::/64
2001:0DF0:000A:3001::/64

- ISP4 first 2 /64

2001:0DF0:000A:4000::/64
2001:0DF0:000A:4001::/64

- ISP 5 first 2 /64

2001:0DF0:000A:5000::/64
2001:0DF0:000A:5001::/64

- ISP 6 first 2 /64

2001:0DF0:000A:6000::/64
2001:0DF0:000A:6001::/64

Workshop Exercises

Exercise 3: IOS recap

Exercise 3: IOS recap

IOS version support basic IPv6
- 12.2(2)T

IOS version support OSPF3 (IPv6)
- 12.2(15)T

IOS version support BGP(IPv6)
- 12.2(2)T

IOS version support BGP(4 byte AS Path)
- 12.4(24)T

Exercise 3: IOS recap

Required **global & interface** commands to enable IPv6

Router(Config)#ipv6 unicast-routing
Router(Config)#ipv6 cef (optional)

- Configure IPv6 address on interface

Router(Config-if)#ipv6 address *2001:0df0:00aa::1/64*
Router(Config-if)#ipv6 enable

- Verify IPv6 configuration

Router#sh ipv6 interface fa0/0

- Verify connectivity

Router#ping *2001:0df0:00aa::1*

Exercise 3: IOS recap

- Required **BGP** commands to enable IPv6 routing

Router(config)# router bgp 1
Router(config-router)# neighbor 2001:0df0:00aa::1 remote-as 2 (EBGP)
Router2(config-router)#bgp router-id 10.0.0.1 (if no 32 bit address on any interface)

Router(config-router)#address-family ipv6
Router(config-router-af)# no synchronization
Router(config-router-af)#neighbor 2001:0df0:00aa::1 activate
Router(config-router-af)# network 2001:0df0:00aa::/48

- Verify BGP IPv6 configuration

Router#sh bgp ipv6 unicast summary (summarized neighbor list)
Router#sh bgp ipv6 unicast (BGP database)
Router#sh ipv6 route bgp (BGP routing table)

Exercise 3: IOS recap

Required command to add IX prefix filter

- Create prefix filter in global mode

Router(config)#ipv6 prefix-list AS1 seq 2 permit 2001:0df0:aa::/48

- Apply prefix filter in BGP router configuration mode

Router(config-router)#neighbor 2001:0df0:aa::1 prefix-list AS1 in
Router(config-router)#neighbor 2001:0df0:aa::1 prefix-list AS1 out

Exercise 3: IOS recap

Controlling routing update traffic (Not data traffic)

1. Incoming routing update (Will control outgoing data traffic)
2. Outgoing routing update (Will control incoming data traffic)

Questions?

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

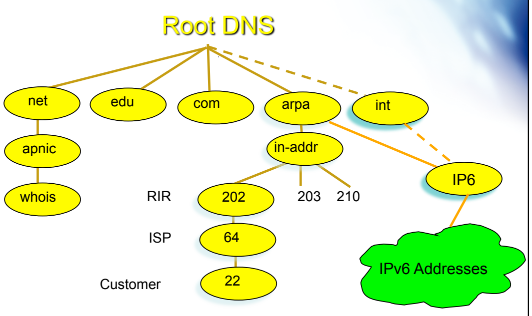
IPv6 representation in the DNS

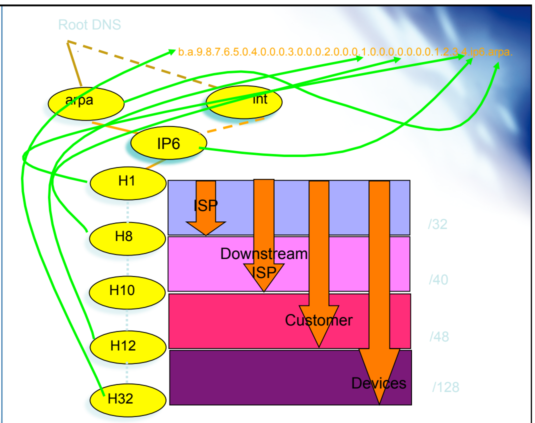
- Forward lookup support: Multiple RR records for name to number
  - AAAA (Similar to A RR for IPv4)
- Reverse lookup support:
  - Reverse nibble format for zone ip6.arpa

IPv6 forward and reverse mappings

- Existing A record will not accommodate IPv6's 128 bit addresses
- BIND expects an A record's record-specific data to be a 32-bit address (in dotted-octet format)
- An address record
  - AAAA (RFC 1886)
- A reverse-mapping domain
  - ip6.arpa

The reverse DNS tree – with IPv6





IPv6 forward lookups

- Multiple addresses possible for any given name
  - Ex: in a multi-homed situation
- Can assign A records and AAAA records to a given name/domain
- Can also assign separate domains for IPv6 and IPv4

Sample forward lookup file

```
;; domain.edu
$TTL          86400
@     IN       SOA      ns1.domain.edu. root.domain.edu. (
      2002093000    ; serial - YYYYMMDDXX
      21600      ; refresh - 6 hours
      1200       ; retry - 20 minutes
      3600000    ; expire - long time
      86400)        ; minimum TTL - 24 hours
;; Nameservers
      IN  NS  ns1.domain.edu.
      IN  NS  ns2.domain.edu.

;; Hosts with just A records
host1      IN  A   1.0.0.1

;; Hosts with both A and AAAA records
host2      IN  A   1.0.0.2
      IN  AAAA   2001:468:100::2
```

IPv6 reverse lookups

- IETF decided to restandardize IPv6 PTR RRs
  - They will be found in the IP6.ARPA namespace
- The ip6.int domains has been deprecated
  - Now using ip6.arpa for reverse

IPv6 reverse lookups - PTR records

- Similar to the in-addr.arpa

```
b.a.9.8.7.6.5.0.4.0.0.0.3.0.0.0.2.0.0.0.1.0.0.0.0.0.0.0.1.2.3.4.ip6.arpa.
    IN    PTR   test.ip6.example.com.
```

- Example: reverse name lookup for a host with address 3ffe:8050:201:1860:42::1

```
$ORIGIN 0.6.8.1.1.0.2.0.0.5.0.8.e.f.f.3.ip6.arpa.

1.0.0.0.0.0.0.0.0.0.0.0.2.4.0.0  14400  IN PTR host.example.com.
```


Sample reverse lookup file

```
;; 0.0.0.0.0.0.1.0.8.6.4.0.1.0.0.2.rev
;; These are reverses for 2001:468:100::/64)
;; File can be used for both ip6.arpa and ip6.int.
$TTL          86400
@    IN      SOA     ns1.domain.edu. root.domain.edu. (
            2002093000       ; serial - YYYYMMDDXX
            21600        ; refresh - 6 hours
            1200         ; retry - 20 minutes
            3600000      ; expire - long time
            86400)       ; minimum TTL - 24 hours
;; Nameservers
        IN   NS   ns1.domain.edu.
        IN   NS   ns2.domain.edu.
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0   IN   PTR host1.ip6.domain.edu
2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0   IN   PTR host2.domain.edu
;;
;; Can delegate to other nameservers in the usual way
;;
```


Questions?

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

Acknowledgement

- "An IPv6 deployment guide" published by The 6NET Consortium" (September 2005) is also referred to in this module.
- The material is available at:
  - http://www.6net.org/book/deployment-guide.pdf
- APNIC very much appreciates 6NET's efforts to share their knowledge with the broader Internet community.

Integration and transition

- Smaller and larger sites have different requirements for smooth IPv6 transition or adoption of IPv6
- However, if planned effectively, the deployment can be done in a phased and controlled manner
- Need to know
  - Your networks' peculiarities and specifics
  - Available solutions
  - How to configure them
  - How to deploy services and accessibility required for contininuity of customer service
  - How to maintain and manage your business and operational needs in new environment

Transition overview

- How to get connectivity from an IPv6 host to the global IPv6 Internet?
  - Via an native connectivity
  - Via IPv6-in-IPv4 tunnelling techniques
- IPv6-only deployments are rare
- Practical reality
  - Sites deploying IPv6 will not transit to IPv6-only, but transit to a state where they support both IPv4 and IPv6 (dual-stack)

http://www.6net.org/book/deployment-guide.pdf p59

APNIC Asia Pacific Network Information Centre

Transition overview

- Three basic ways of transition
  - Dual stack
  - Deploying IPv6 and then implementing IPv6-in-IPv4 tunnelling
  - IPv6 only networking
- Different demands of hosts and networks to be connected to IPv6 networks will determine the best way of transition

APNIC Asia Pacific Network Information Centre

Transition overview

- Dual stack
  - Allow IPv4 and IPv6 to coexist in the same devices and networks
- Tunnelling
  - Allow the transport of IPv6 traffic over the existing IPv4 infrastructure
- Translation
  - Allow IPv6 only nodes to communicate with IPv4 only nodes

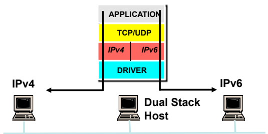
IPv6 essentials by Silvia Hagen, p255

APNIC Asia Pacific Network Information Centre

Dual stack transition

RFC 4213

- Dual stack = TCP/IP protocol stack running both IPv4 and IPv6 protocol stacks simultaneously
  - Application can talk to both
- Useful at the early phase of transition



Dual stack

- A host or a router runs both IPv4 and IPv6 in the protocol TCP/IP stack.
- Each dual stack node is configured with both IPv4 and IPv6 addresses
- Therefore it can both send and receive datagrams belonging to both protocols
- The simplest and the most desirable way for IPv4 and IPv6 to coexist

http://www.6net.org/book/deployment-guide.pdf p60

Dual stack

- Challenges
  - Compatible software
    - Eg. If you use OSPFv2 for your IPv4 network you need to run OSPFv3 in addition to OPSFv2
  - Transparent availability of services
    - Deployment of servers and services
    - Content provision
    - Business processes
    - Traffic monitoring
    - End user deployment

Dual stack and DNS

- DNS is used with both protocol versions to resolve names and IP addresses
  - An dual stack node needs a DNS resolver that is capable of resolving both types of DNS address records
    - DSN A record to resolve IPv4 addresses
    - DNS AAAA record to resolve IPv6 addresses
- Dual stack network
  - Is an infrastructure in which both IPv4 and Ipv6 forwarding is enabled on routers

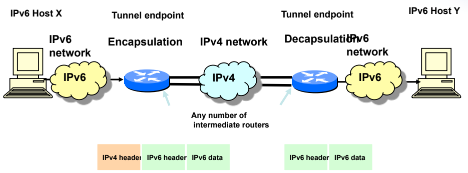
IPv6 essentials by Silvia Hagen, p256

Tunnels

- Part of a network is IPv6 enabled
  - Tunnelling techniques are used on top of sn existing IPv4 infrastructure and uses IPv4 to route the IPv6 packets between IPv6 networks by transporting these encapsulated in IPv4
  - Tunnelling is used by networks not yet capable of offering native IPv6 functionality
  - It is the main mechanism currently being deployed to create global IPv6 connectivity
- Manual, automatic, semi-automatic configured tunnels are available

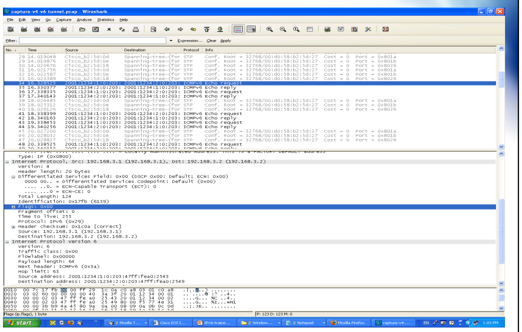
Tunnelling – general concept

- Tunnelling can be used by routers and hosts
  - IPv6-over-IPv4 tunnelling
  - Involves three steps
    - Encapsulation, decapsulation, and tunnel management



Concept is borrowed from Cisco training material "IPv6 Seminar"

Encapsulated IPv6 packets in IPv4



Tunnelling – general concept

- A tunnel can be configured in four different ways:
  - Router to router
    - Spans one hop of the end-to-end path between two hosts. Probably the most common method
  - Host to router
    - Spans the first hop of the end-to-end path between two hosts. Found in the tunnel broker model
  - Host to host
    - Spans the entire end-to-end path between two hosts
  - Router to host
    - Spans the last hop of the end-to-end path between two hosts

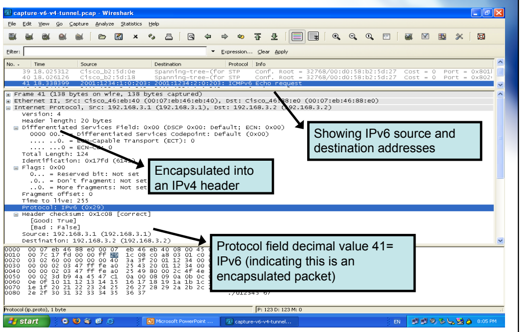
Tunnel encapsulation

The steps for the encapsulation of the IPv6 packet

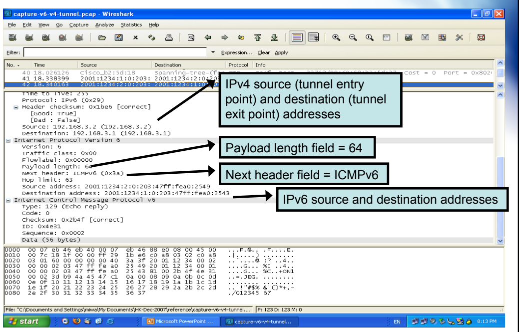
- The entry point of the tunnel decrements the IPv6 hop limit by one
- Encapsulates the packet in an IPv4 header
- Transmits the encapsulated packet through the tunnel
- The exit point of tunnel receives the encapsulated packet
  - If necessary, the IPv4 packet is fragmented
- It checks whether the source of the packet (tunnel entry point) is an acceptable source (according to its configuration)
  - If the packet is fragmented, the exit point reassembles it
- The exit point removes the IPv4 header
- Then it forwards the IPv6 packet to its original destination

IPv6 essentials by Silvia Hagen, p258

Tunnel encapsulation

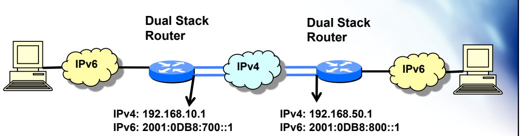


Tunnel encapsulation



Manual configuration

RFC 4213



Manually configured tunnels require:

- Dual stack end points
- Explicit configuration with both IPv4 and IPv6 addresses at each end

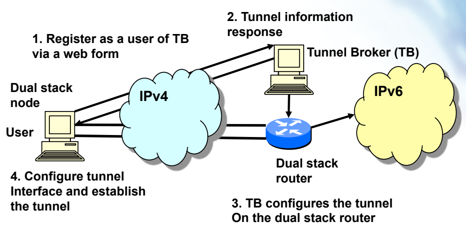
Concept is borrowed from Cisco, Training material "Ipv6 Seminar" delivered at South Asian IPv6 Summit, Jan 2004

Tunnel broker

- Semi-automatic alternative to manual configuration
- Useful when:
  - A dual stack host in an IPv4-only network wishes to gain IPv6 connectivity
- The basic concept of a tunnel broker:
  - A user connects to a web server(the TB)
  - Enters some authentication details
  - Receives back a short script to run
  - The script then establishes an IPv6-in-IPv4 tunnel to the tunnel broker DS router

Tunnel broker

RFC 3053



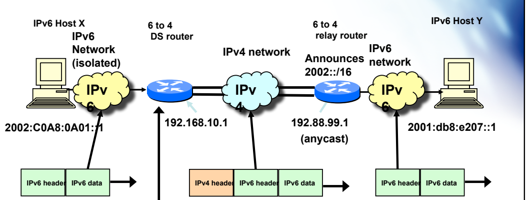
TB is an external system

- Free TB services are available

http://www.sixxs.net/tools/aiccu/brokers/

Automatic tunneling – 6to4

RFC 3056 RFC 3068



S=2002:C0A8:0A01::1
D=2001:db8:e207::1

S (v4)=192.168.10.1
D (v4)=192.88.99.1
S (v6)=2002:C0A8:0A01::1
D (v6)=2001:db8:e207::1

S=2002:C0A8:0A01::1
D=2001:db8:e207::1

Default IPv6 route is 2002:co58:6301::

A destination route to a 2002::/ prefix is encapsulated in IPv4 and bits 17 – 48 used as the next hop. Ie 192.88.99.1 anycast

6to4

- When 6to4 domains communicate with 6to4 domains, things are relatively simpler
  - The IPv4 address of the destination 6to4 router is used in the default IPv6 route of the source router.

If you are an ISP wishing…

- To offer some support for IPv6 clients but you are not ready to do the full dual stack deployment across your entire network:
  - If you all want to do initially is:
    - Move IPv6 packets
    - Support the IPv6 connectivity services

  What are your options?

  What is in the initial shopping list?
- At a minimum one of:
  - A dual stack gateway
  - An IPv6 router
  - IPv6 peers or IPv6 transit services

http://www.potaroo.net/ispcol/2008-02/tui.html

Questions?

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

Transit VS Peering

Two type of traffic exchange between ISPs

- Transit
  - Where ISP will pay to send/receive traffic
  - Downstream ISP will pay upstream ISP for transit service
- Peering
  - ISPs will not pay each other to interchange traffic
  - Works well if win win for both
  - Reduce cost on expensive transit link

IX Peering Model

- **BLPA (Bi-Lateral Peering Agreement)**
  - IX will only provide layer two connection/switch port to ISPs
  - Every ISPs will arrange necessary peering arrangement with others by their mutual business understanding.
- **MLPA (Multi-Lateral Peering Agreement)**
  - IX will provide layer two connection/switch port to ISPs
  - Each ISP will peer with a **route server** on the IX.
  - Route server will collect and distribute directly connected routes to every peers.

IXP Peering Policy

- BLPA is applicable where different categories of ISPs are connected in an IX
  - Large ISPs can choose to peer with large ISPs (base on their traffic volume)
  - Small ISPs will arrange peering with small ISPs
- Would be preferable for large ISPs
  - They will peer with selected large ISPs (Equal traffic interchange)
  - Will not loose business by peering with small ISP

IX Peering Policy

- MLPA model works well to widen the IX scope of operation (i.e national IX).
- Easy to manage peering
  - Peer with the **route server** and get all available local routes.
  - Do not need to arrange peering with every ISPs connected to the IX.
- Unequal traffic condition can create not intersected situation to peer with route server

IX peering Policy

- Both peering model can be available in an IX.
- Member will select peering model i.e either BLPA or MLPA (Route Server Peering)
- IX will provide switch port
- **Mandatory MLPA** model some time not preferred by large ISP (Business Interest)
  - Can create not interested situation to connect to an IX

IX Operating Cost

- Access link
- Link maintenance
- Utility
- Administration

Cost Model

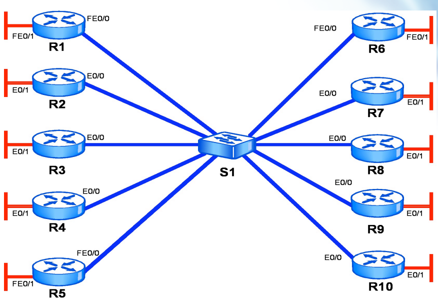
- Not for profit
- Cost sharing
- Membership based
- Commercial IX

Questions?

Overview

- Internet Fundamental
  - Internet Protocols – some revision
  - IP addressing basic
  - IP Routing basic
  - Introduction to DNS & RevDNS
  - IPv6 overview
  - IPv6 RevDNS
  - IPv6 transition technologies
  - IX Policies
  - Exercise on IX and IPv6 tunnelling

IX Network Diagram



Steps to be done

- Determine the IP addressing scheme for the IX and for your ISP LAN network
- Configure the external interfaces of the Routers connecting your ISP to the IX
- Configure an internal LAN for your ISP
- Configure BGP on the Router
- Test this connectivity

IPv6 addressing plan

IX Subnet: 2001:AA::/48

Routers interface IPv6 Address (IX side)

Router 1: 2001:00AA::1/64
Router 2: 2001:00AA::2/64
Router 3: 2001:00AA::3/64
Router 4: 2001:00AA::4/64
Router 5: 2001:00AA::5/64
Router 6: 2001:00AA::6/64
Router 7: 2001:00AA::7/64
Router 8: 2001:00AA::8/64
Router 9: 2001:00AA::9/64
Router 10: 2001:00AA::10/64

IPv6 addressing plan

ISP's Global routing prefix

Router 1: 2001:abc1::/32
Router 2: 2001:abc2::/32
Router 3: 2001:abc3::/32
Router 4: 2001:abc4::/32
Router 5: 2001:abc5::/32
Router 6: 2001:abc6::/32
Router 7: 2001:abc7::/32
Router 8: 2001:abc8::/32
Router 9: 2001:abc9::/32
Router 10: 2001:abca::/32

Configuration steps

- •Configure Router Interface Connected to IX (0/0)
- •Configure Router Interface Connected to LAN (0/1)
- •Try ping others

- •Create EBGP Peering
- •Announce LAN/ISP prefix

Step of IOS command line

Interface mode command:

- *Router(config-if) # ipv6 address 2001:ABC1::1/64*

Enable IPv6 on the interface selected.

- *Router(config-if) # ipv6 enable*

Bring the interface up

Router(config-if) no shutdown

Step of IOS command line

Exit from the interface configuration and enable IPv6 unicast datagram forwarding by typing the command below in the global mode.

- *Router(config) # ipv6 unicast-routing*
  - *Router(config) # ipv6 cef*

Configure BGP with the IPv6 address

Type "Router bgp" with the AS number in the command prompt of the Router global mode to configure the BGP protocol.

- ***Router#configure terminal***
- ***Router(config)#router bgp** <ASN>*
- ***Router(config-router)#no auto summary***
- ***Router(config-router)#no synchronization***
- ***Router (config-router-af)#no synchronization (IPv6 address-family mode)***

Where the AS number is the number of your Router

Configure BGP with the IPv6 address

Configure the peering address of the neighboring AS. Use the point to-point interface IP address for each Router connected to the IX.

NOTE: Each Router will have 9 neighbours

– *Router(config-router)# neighbor <other ASN interface IP> remote-as <other ASN>*

- Example for Router1:

```
Router#configure terminal
Router(config)#router bgp 1
Router(config-router)#no auto-summary
Router(config-router)#no synchronization
  Router(config-router)#neighbor 2001:00AA::2
  remote-as 2 (for peering with Router2)
```

Configure BGP with the IPv6 address

```
Router(config-router)#address-family ipv6
Router(config-router-af)#neighbor 2001:00AA::2 activate
Router(config-router-af)#network 2001:00AA::/64
```

Configure BGP with the IPv6 address

Configure BGP router-id (optional). BGP protocol might ask for "router id" if there's no IPv4 address configured aside from IPv6 address. Each eBGP speaker needs to have a 32 bit integer router ID.

The highest IP address configured on the router will become the router ID.

If a loopback interface address is configured, it will be use as the router ID.

If no IPv4 address is configured, watch out for such error message below.

- *% BGP cannot run because the Router-id is not configured*
- *BGP Router identifier 0.0.0.0, local AS number 1*

Verifying the BGP process

show bgp ipv6 unicast summary (to check the bgp summary table)

Expected output:

- Router6#sh bgp ipv6 unicast summary

- BGP router identifier 192.169.8.1, local AS number 6
- BGP table version is 4, main routing table version 4
- 3 network entries using 447 bytes of memory
- 3 path entries using 228 bytes of memory

- 0 BGP filter-list cache entries using 0 bytes of memory
- BGP using 1787 total bytes of memory
- BGP activity 8/1 prefixes, 14/4 paths, scan interval 60 secs

```
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
– 2001:ABC6::2 4     7   4252    4259      4       0    0   2d22h        0
– 2001:ABC6:0:1::2 4 8   5515    5513      4       0    0   3d19h
```

Asia Pacific Network Information Centre APNIC

Verifying the BGP process

sh bgp ipv6 (to check the routing table for the BGP announcement)

- Expected Output:

```
– Router6#sh bgp ipv6 unicast
– BGP table version is 4, local router ID is 192.169.8.1
– Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
–             r RIB-failure, S Stale
– Origin codes: i - IGP, e - EGP, ? - incomplete

–    Network          Next Hop          Metric LocPrf Weight Path
– *> 2001:ABC6::/32   ::                     0         32768 i
– *> 2001:ABC8::/32   2001:ABC6:0:1::2
–                                         0              0 8 i
– *> 2001:ABC9::/32   2001:ABC6:0:1::2
–                                                        0 8 9 i
```

Asia Pacific Network Information Centre APNIC

Verifying the BGP process

sh ipv6 route (to check the IPv6 routing table)

- Expected Output:

```
• Routerouter#sh ipv6 route
• IPv6 Routing Table - 9 entries
• Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
         U - Per-user Static route
•     I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
•     O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
•     ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
• S   ::/0 [1/0]
•     via ::, Null0
• C   2001:AA::/64 [0/0]
•     via ::, Ethernet0/0
• L   2001:AA::2/128 [0/0]
•     via ::, Ethernet0/0
• C   2001:ABC1::/64 [0/0]
•     via ::, Ethernet0/0
• L   2001:ABC1::2/128 [0/0]
•     via ::, Ethernet0/0
```

Asia Pacific Network Information Centre APNIC

Verifying the BGP process

sh ipv6 route (to check the IPv6 routing table)

Expected Output continue…….

- S 2001:ABC2::/32 [1/0]
- via ::, Null0
- B 2001:ABC3::/32 [20/0]
- via FE80::2E0:1EFF:FE63:2901, Ethernet0/0
- L FE80::/10 [0/0]
- via ::, Null0
- L FF00::/8 [0/0]
- via ::, Null0

Apply IX peering policy

- BLPA
  - Get an IX switch port
  - Arrange separate peering with other participating member
  - Routing updates can be controlled based on individual peer
  - Configuration example:

```
Router(config)#ipv6 prefix-list AS2-IN seq 2 permit 2001:0df0:abc2::/32
Router(config)#ipv6 prefix-list AS3-IN seq 2 permit 2001:0df0:abc3::/32
Router(config)#ipv6 prefix-list MYAS-PREFIX seq 2 permit 2001:0df0:abc1::/32

Router(config-router)# neighbor 2001:0df0:00aa::2 remote-as 2 (EBGP)
Router(config-router)# neighbor 2001:0df0:00aa::3 remote-as 3 (EBGP)

Router(config-router)#neighbor 2001:0df0:aa::2 prefix-list AS2-IN in
Router(config-router)#neighbor 2001:0df0:aa::2 prefix-list MYAS-PREFIX out

Router(config-router)#neighbor 2001:0df0:aa::3 prefix-list AS3-IN in
Router(config-router)#neighbor 2001:0df0:aa::3 prefix-list MYAS-PREFIX out
```

Apply IX peering policy

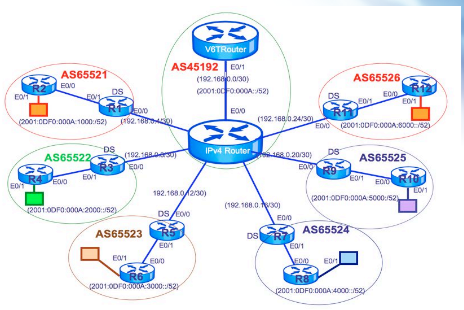
- MLPA
  - Get an IX switch port
  - Arrange a single peering with route server
  - Routing updates can be controlled on individual prefix
  - Configuration example:

```
Router(config)#ipv6 prefix-list RS-IN seq 2 permit 2001:0df0:abc2::/32
Router(config)#ipv6 prefix-list RS-IN seq 3 permit 2001:0df0:abc3::/32
Router(config)#ipv6 prefix-list RS-OUT seq 2 permit 2001:0df0:abc1::/32

Router(config-router)# neighbor 2001:0df0:00aa::e remote-as 100 (EBGP)

Router(config-router)#neighbor 2001:0df0:aa::e prefix-list RS-IN in
Router(config-router)#neighbor 2001:0df0:aa::2 prefix-list RS-OUT out
```

Workshop Exercises

- **Exercise 5: IPv6 ISP Tunneling Topology**

Exercise 5: IPv6 ISP Tunneling Topology



Exercise 5: IPv6 ISP Tunneling Topology

Steps to be done

- Determine the IP addressing scheme for your ISP LAN network
- Determine the IP addressing scheme for the tunnel interface
- Configure the interfaces of the Routers with IPv6 address
- Configure EBGP on Dual Stack (DS) router
- Configure Tunnel in DS router with IPV6 address
- Configure EBGP Peering with IPv6 router
- Configure iBGP peering with ISP router
- Test this connectivity

APNIC Asia Pacific Network Information Centre

Exercise 5: IPv6 ISP Tunneling Topology

- Global prefix received: 2001:0df0:000a::/48

2001:0DF0:000A:0000::/52 (AS45192)
2001:0DF0:000A:1000::/52 (AS65521)
2001:0DF0:000A:2000::/52 (AS65522)
2001:0DF0:000A:3000::/52 (AS65523)
2001:0DF0:000A:4000::/52 (AS65524)
2001:0DF0:000A:5000::/52 (AS65525)
2001:0DF0:000A:6000::/52 (AS65526)

APNIC Asia Pacific Network Information Centre

Exercise 5: IPv6 ISP Tunneling Topology

AS45192 IP distribution

192.168.0.0/30 [IPv6Router(1) -IPv4Router(2)]
2001:0DF0:000A:0000::/52 (AS45192)
2001:0DF0:000A:0000::/64 (IPv6Router-R1 Tunnel0)
2001:0DF0:000A:0001::/64 (IPv6Router-R3 Tunnel0)
2001:0DF0:000A:0002::/64 (IPv6Router-R5 Tunnel0)
2001:0DF0:000A:0003::/64 (IPv6Router-R7 Tunnel0)
2001:0DF0:000A:0004::/64 (IPv6Router-R9 Tunnel0)
2001:0DF0:000A:0005::/64 (IPv6Router-R11 Tunnel0)

APNIC Asia Pacific Network Information Centre

Exercise 5: IPv6 ISP Tunneling Topology

Allocated IPv6 address for different AS

AS65521

192.168.0.4/30 [R1(6) -IPv4Router(5)]
2001:0DF0:000A:1000::/52 (AS65521)
2001:0DF0:000A:1000::/64 (R1-R2)
2001:0DF0:000A:1001::/64 (R1 LAN)
2001:0DF0:000A:0000::2/64 (R1 Tunnel 0)

AS65522

192.168.0.8/30 [R3(10) -IPv4Router(9)]
2001:0DF0:000A:2000::/52 (AS65522)
2001:0DF0:000A:2000::/64 (R3-R4)
2001:0DF0:000A:2001::/64 (R4 LAN)
2001:0DF0:000A:0001::2/64 (R3 Tunnel 0)

AS65523

192.168.0.12/30 [R5(14) -IPv4Router(13)]
2001:0DF0:000A:3000::/52 (AS65523)
2001:0DF0:000A:3000::/64 (R5-R6)
2001:0DF0:000A:3001::/64 (R6 LAN)
2001:0DF0:000A:0002::2/64 (R5 Tunnel 0)

Exercise 5: IPv6 ISP Tunneling Topology

Allocated IPv6 address for different AS

192.168.0.16/30 [R7(18) -IPv4Router(17)]
2001:0DF0:000A:4000::/52 (AS65524)
2001:0DF0:000A:4000::/64 (R7-R8)
2001:0DF0:000A:4001::/64 (R8 LAN)
2001:0DF0:000A:0003::2/64 (R7 Tunnel 0)

AS65524

192.168.0.20/30 [R9(22) -IPv4Router(21)]
2001:0DF0:000A:5000::/52 (AS65525)
2001:0DF0:000A:5000::/64 (R9-R10)
2001:0DF0:000A:5001::/64 (R10 LAN)
2001:0DF0:000A:0004::2/64 (R9 Tunnel 0)

AS65525

192.168.0.24/30 [R11(26) -IPv4Router(25)]
2001:0DF0:000A:6000::/52 (AS65526)
2001:0DF0:000A:6000::/64 (R11-R12)
2001:0DF0:000A:6001::/64 (R12 LAN)
2001:0DF0:000A:0005::2/64 (R11 Tunnel 0)

AS65526

Exercise 5: IPv6 ISP Tunneling Topology

Configuration steps in every AS

- DSRouter(Config)#ipv6 unicast-routing
- DSRouter(Config)#ipv6 cef
- DSRouter(Config-if)#IPv4 address with IPv4Router
- DSRouter(Config)# EBGP with IPv4Router
- DSRouter(Config-if)#6 to 4 Tunnel with IPv6Router
- DSRouter(Config)#EBGP with IPv6 router
- DSRouter(Config-if)#IPv6 address with IPv6 only router
- DSRouter(Config-if)#iBGP peering with IPv6 only router

- IPv6OnlyRouter(Config)#ipv6 unicast-routing
- IPv6OnlyRouter(Config)#ipv6 cef
- IPv6OnlyRouter(Config)#IPv6 address with DSRouter
- IPv6OnlyRouter(Config)#IPv6 address with LAN
- IPv6OnlyRouter(Config)#iBGP Peering with DS router

Exercise 5: IPv6 ISP Tunneling Topology

Verification steps in every AS

- DSRouter#sh bgp ipv6 (unicast) summary
- DSRouter#sh bgp ipv6 (unicast)
- DSRouter#sh ipv6 route (bgp)

- IPv6OnlyRouter#sh bgp ipv6 (unicast) summary
- IPv6OnlyRouterRouter#sh bgp ipv6 (unicast)
- IPv6OnlyRouterRouter#sh ipv6 route (bgp)

APNIC Asia Pacific Network Information Centre

Questions?

APNIC Asia Pacific Network Information Centre

Thank you!