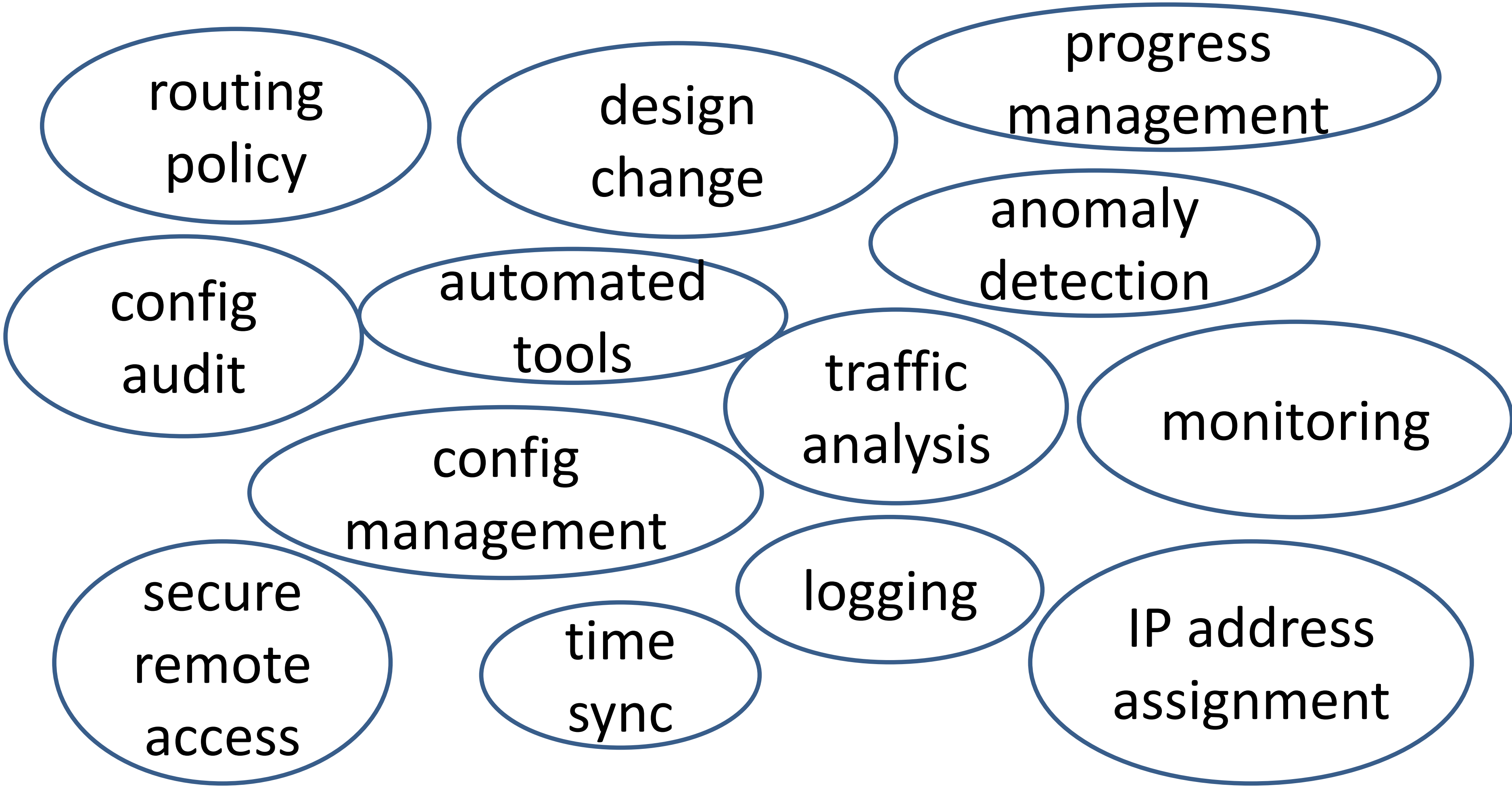
Securing Network

Matsuzaki 'maz' Yoshinobu
<maz@iij.ad.jp>

Goals

- Ensuring Network Availability
- Controlling Routing Policy
- Protecting Information
- Preventing Misuse
- Mitigating Attacks
- Responding to Incidents
- etc.

Operational Design



routing policy

design change

progress management

anomaly detection

config audit

automated tools

traffic analysis

monitoring

config management

logging

secure remote access

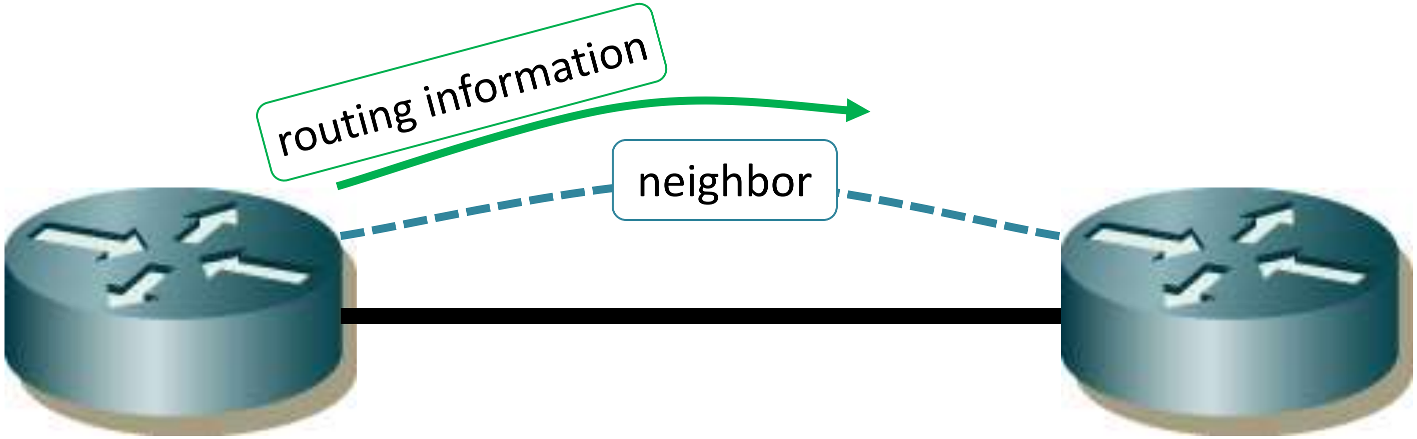
time sync

IP address assignment

Protecting Routing

- To keep your network working
  - as you designed
  - as you configured
- Static Routing
  - mostly depends on design
- Dynamic Routing
  - possibility of remote attack

Routing Protocol



- Routers exchange routing information over a neighboring relationship.

Threat Model for Routing

- Neighboring Relationship
  - Unexpected Neighboring
  - Shutdown by Someone else
  - Spoofed Neighbor
- Routing Information
  - Propagation of Wrong Information
  - Unintended Routing Policy
  - Hit a Hardware Limitation

OSPFv3 Neighbors

- Establishing a relationship among trusted neighbors only
- Disabled by default
  - Especially on a link to other parties (IX,customer)
    - to avoid unexpected neighbors
    - if you have to enable on these links, use ‘passive’ feature
  - Enabled where it is needed like backbone
- Authentication
  - IPsec(RFC4552) or OSPFv3 AT (RFC6506)

OSPFv3 IPsec configuration

cisco

```
interface <interface_name>
 ipv6 enable
 ipv6 ospf <process#> area <area#>
 ipv6 ospf authentication ipsec spi <spi_value> <auth-algorithm> <key>
```

juniper

```
[[edit security ipsec]
security-association <ipsec-sa_name> {
mode transport;
    manual {
        direction bidirectional {
            protocol ah;
            spi <spi_value>;
            authentication {
                algorithm <auth-algorithm>;
                key ascii-text “<key>”;
}}}}
edit protocols ospf3 area <area#>]
interface <interface_name> {
    ipsec-sa <ipsec-sa_name>;
}
```

BGP4 Neighbors

- Protecting TCP sessions
  - md5 authentication
- Peering with other parties
  - possibility of injection
  - needs more attention about routing information

Securing Neighbors

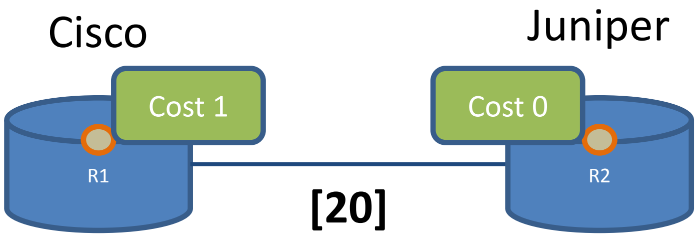
IPv4

- OSPFv2
  - md5 authentication
- BGP4
  - ipv4 for tcp session
  - md5 authentication

IPv6

- OSPFv3
  - ipsec authentication
- BGP4+
  - ipv6 global for tcp session
  - md5 authentication

OSPFv2 loopback cost



```
R1 Loopback:  10.0.0.1/32   Cisco
R2 Loopback:  10.0.0.2/32   Juniper
R1 – R2:      192.168.0.1/30 – 192.168.0.2/30
```

```
R1> sh ip route 10.0.0.2
Routing entry for 10.0.0.2/32
  Known via “ospf 1”, distance 110, metric 20, type intra area
  Last update from 192.168.0.2 on GigabitEthernet1/1 6d21h
ago
  Routing Descriptor Blocks:
  *  10.0.0.2, from 10.0.0.2, 6d21h ago, via GigabitEthernet1/1
       Route metric is 20, traffic share count is 1
```

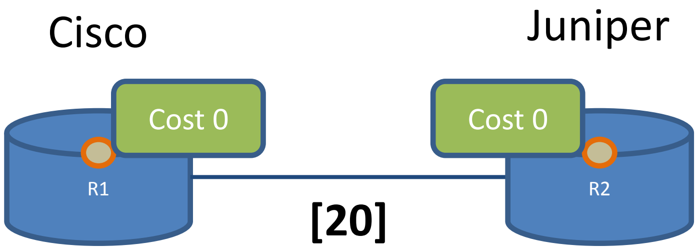
```
R2> > show route 10.0.0.1

inet.0: 36 destinations, 37 routes (36 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.1/32     *[OSPF/10] 6d 21:54:09, metric 21
              > to 192.168.0.1 via ge-0/0/1.0
```

OSPFv3 loopback cost

Prefix length 128
IGP Cost 0



```
R1 Loopback:   2001:db8:10::1/128   Cisco IOS
R2 Loopback:   2001:db8:10::2/128   Juniepr Junos
R1 – R2:       2001:db8::1/64 – 2001:db8::2/64
```

```
R1> show  ipv6 route 2001:db8:10::2
Routing entry for 10.0.0.2/32
   Known via “ospf 1”, distance 110, metric 20, type intra area
   Route count is 1/1, share count  0
Routing paths:
   FE80::217:CBFF:FEDA:625, GigabitEthernet1/1
      Last updated 00:05:19 ago
```

```
R2> > show route 2001:db8::10::1

inet6.0: 24 destinations, 25 routes (24 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:10::1/128
              *[OSPF3/10] 00:04:28, metric 20
               > to fe80::20a:41ff:fe43:d080 via ge-0/0/1.0
```

BGP UPDATE

- Prefixes + Path Attributes
- major attributes
  - AS Path
  - localpreference
  - MED
  - nexthop
  - bgp community

Inbound BGP Prefix filtering

- strict filtering for customer
  - customer should know what they announce
  - ISP can maintain a prefix filter to accept the prefixes , and deny any other prefixes to avoid unexpected routing
- loose filtering for peers/upstreams
  - deny special prefixes, then accept any
  - max-prefix setting to avoid memory exhausts

Prefix filtering for Customer

cisco

```
ipv6 prefix list <prefix-list_name> seq 5 permit 2001:db8::/32
router bgp <as#>
 address-family ipv6
 neighbor <neighbor_ip> prefix-list <prefix-list_name> in
```

juniper

```
[edit]
policy-options {
  prefix-list <prefix-list_name> {
     2001:db8::/32;
}}
  policy-statement <policy_name> {
    term Permit {
            from {
                family inet6;
                prefix-list <prefix-list_name>;
            }
            then next policy;
     }
     then reject;
}}
```

```
[edit protocol bgp]
neighbor <neighbor_ip> {
   import [<policy_name> <others>];
}
```

BGP community

- ISPs use BGP community to tag a prefix so that they can control the prefix later on.
  - to distinguish prefixes that should be announced to other ASes
- Protect your BGP community space
  - overwrite the attribute on the eBGP sessions

BGP NEXT_HOP

- As per RFC4271,now routers use Routing Table (including BGP routes) to resolve BGP NEXT_HOP by recursive route lookup
  - Before Cisco used only IGP to resolve
- Carrying prefixes that is used by recursive route lookup by IGP is not enough
  - What happens if you receive a more specific route from your BGP neighbor

Protecting Recursive route lookup

- Protect routing information that is used to resolve the BGP NEXT_HOP
  - shouldn't receive from outside
- common prefixes that cover BGP NEXT_HOP
  - peering links (IX, PNI)
  - customer links
- filtering these prefixes on eBGP sessions
  - including more specific prefixes
  - most ISPs filter /24 or longer for IPv4
  - ISPs tend to filter /48 or longer for IPv6

Resource Limitations

- CPU load
  - high CPU load causes route flapping
- considerations
  - AS Path Length
    - usually 1~15 without prepending
  - # of prefixes
    - # of IPv6 full routes is around 12000, and still growing
    - 1 x /32 == 4294967295 x /64s

IPv6 forwarding performance

- same as IPv4's if
  - Router has the same architecture for both
  - Capacity depends on traffic volume
- different from IPv4's if
  - router has different architecture like:
    - IPv4 forwarding is done by ASIC
    - IPv6 forwarding is done by CPU
  - CPU load might go high, when IPv6 traffic increases

AS Path length case

- In 2009, an AS announced a prefix with 252 prepends, and the prefix was propagated to the internet. At some points the total AS Path length became 255, and hit router bugs. This caused BGP sessions flapping in the Internet.
- 2 Bugs found
  - Resetting a BGP session if AS Path length > 255
  - Announcing a malformed UPDAT if AS Path length > 256

AS Path length limitation

cisco

```
router bgp <as#>
 bgp maxas-limit <max-as-path-length>
```

juniper

```
[edit]
policy-options {
 as-path <as-path_name> “.{<max-as-path-length>,}”;
 policy-statement <policy_name> {
   term Deny {
         from {
            family inet6;
            as-path <as-path_name>;
         }
         then reject;
   }
    then next policy;
}}
```

```
[edit protocol bgp]
neighbor <neighbor_ip> {
  import [<policy_name> <others>];
}
```

# of prefixes cases

- So many cases
- Mostly caused by mistakes
  - Leaking of internal routes
  - Announcing full routes to its peers/upstreams
  - leaking of test routes
  - re-originating others’ prefixes
    - redistribution mistake like BGP->OSPF->BGP

# of received prefix limitation

cisco

```
router bgp <as#>
 address-family ipv6
 neighbor <neighbor_ip> maximum-prefix <max_limit>
```

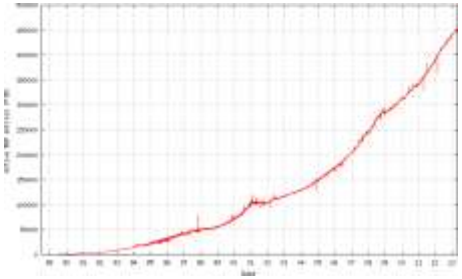
juniper

```
[edit protocol bgp]
neighbor <neighbor_ip> {
 family inet6 {
       unicast {
             prefix-limit {
                  maximum <max_limit>;
                  teardown 75 idle-timeout forever;
 }}}}
```

BGP Advertisement to others

- Aggregated prefixes only
  - In case you received a /32 IPv6 prefix from VNNIC, announce only the /32 to other ISPs
  - You should filter your internal prefixes
- To avoid unnecessary BGP table growth
  - The IPv4 case is terrible, we shouldn't do that on IPv6

IPv4 BGP Table



ripe-532

- Recommendations on IPv6 Route Aggregation
- It is suggested that prefix filters allow for prudent subdivision of an IPv6 allocation. The operator community will ultimately decide what degree of subdivision is supportable, but the majority of ISPs accept prefixes up to a length of /48 within PA space.
- Advertisement of more specific prefixes should not be used unless absolutely necessary and, where sensible, a covering aggregate should also be advertised. Further, LIRs should use BGP methods such as NO_EXPORT [RFC-1997] and NOPEER [RFC-3765] or provider-specific communities, as described in RIPE-399 to limit the propagation of more specific prefixes in the routing table.
- Operators should register appropriate "route6" objects in their preferred routing registry, or ROAs in the RPKI, to reflect any more specific advertisements.

Unauthorized Announcement

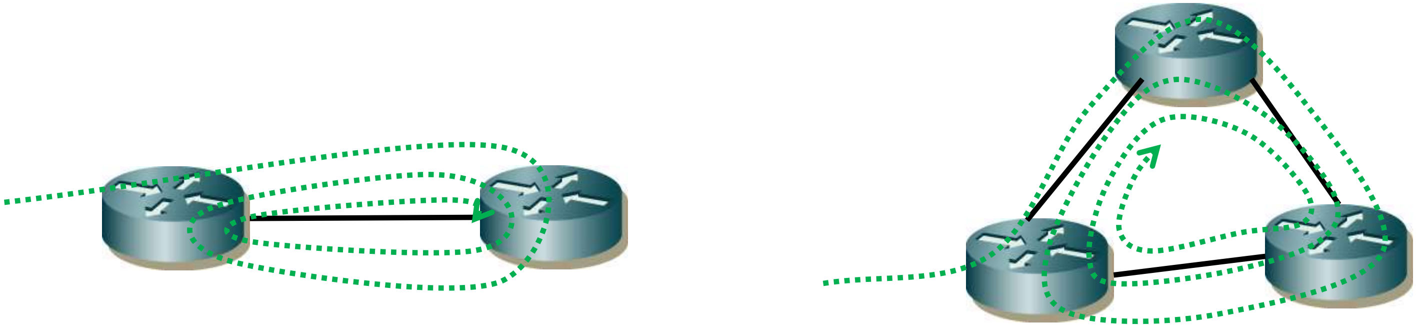
- Someone announces your prefix without your permission
  - This actually happens in the Internet
  - Also called as ‘route hijack’
  - Mostly caused by mistakes
- Solution
  - Maintain a strict prefix filter to accept customer prefix from the customer AS
  - Announce prefixes you needs to announce to other AS
    - Your prefix and Customers’ prefixes

Reactive/Proactive

- Contact the origin AS of the unauthorized announcement
  - whois, IRR and peeringdb.com are useful to find a contact
- Ask communities for help
  - APOPS, NANOG
- Keep your records up-to-date on whois/IRR
  - to show you are the right resource holder

Routing Loops

- A packet goes thorough the same path twice or more



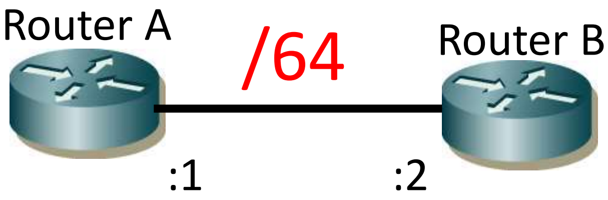
- Mostly caused by a routing issue
- Attackers can use this to amplify traffic
  - about 100 multiplying factor (10Mbps -> 1Gbps)
  - producing a link congestion

Avoiding Routing Loops

- IPv6 has 96bit more address space than IPv4
  - Possibility to create a rooting loop by unused space
- Dynamic Routing
  - Designed to avoid routing loops as their fundamental feature
  - Might create micro-loop during its convergence time, but this is acceptable
- Manually configured routes
  - connected route
  - static route

/64 for an inter-router link

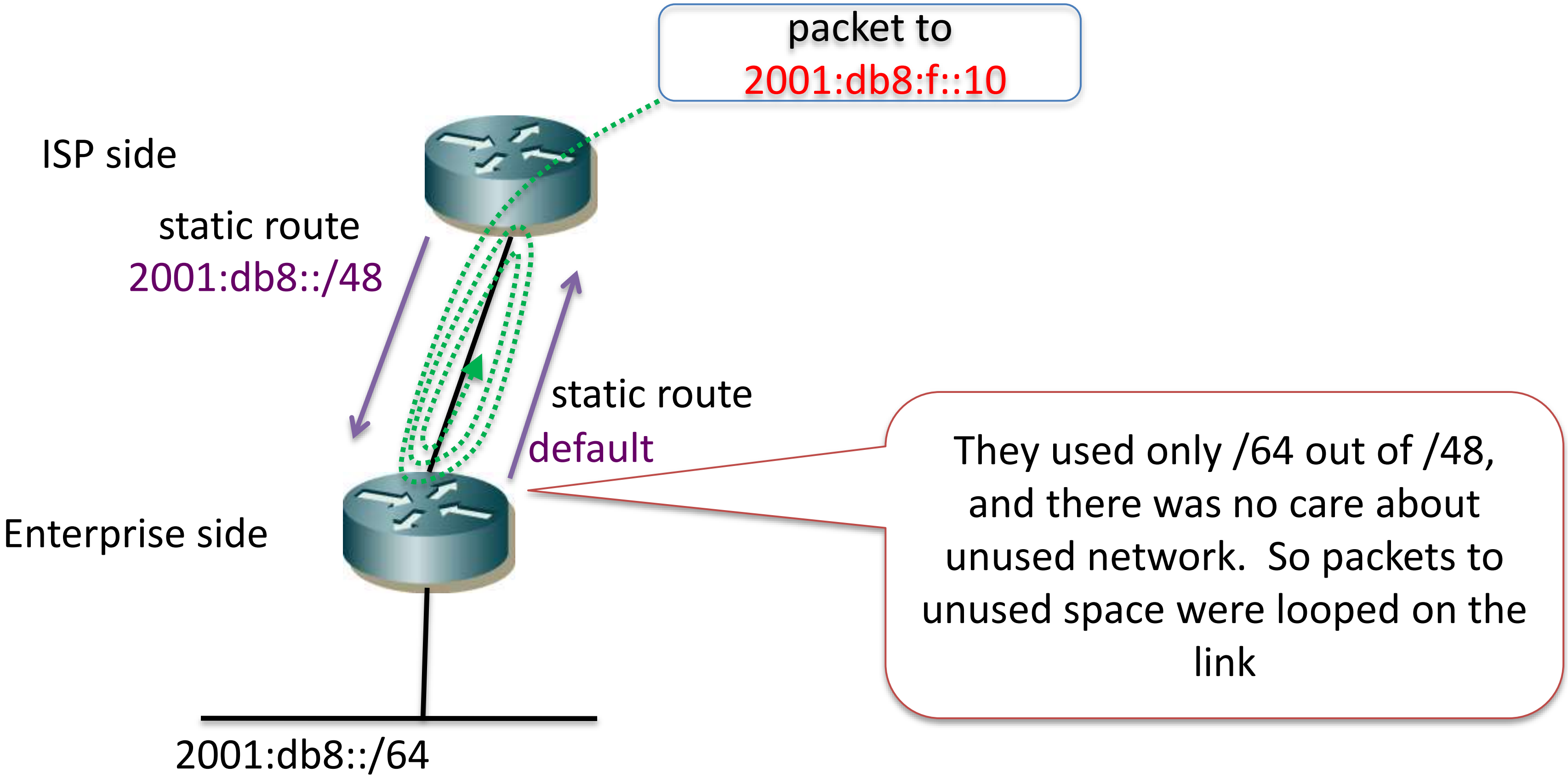
- 2001:db8::/64
  - 2001:db8::0 <- Subnet Router-anycast address
  - 2001:db8::1 <- Router A
  - 2001:db8::2 <- Router B
  - 2001:db8::3-2001:db8::ffff:ffff:ffff:ffff <- unused



point-to-point link

- Typical Medias
  - POS, Serial, tunnel
- no NDP lookup
  - Router forward a packet to neighbor if the destination address is not itself but still on the link
- A packet directed to unused spaces is looped
  - Yes, this is a bug. RFC4443 says router should discard these packets. But it loops on many boxes
  - RFC6164 says you can use /127 to avoid the loop

IPv6 static route



packet to
2001:db8:f::10

ISP side

static route
2001:db8::/48

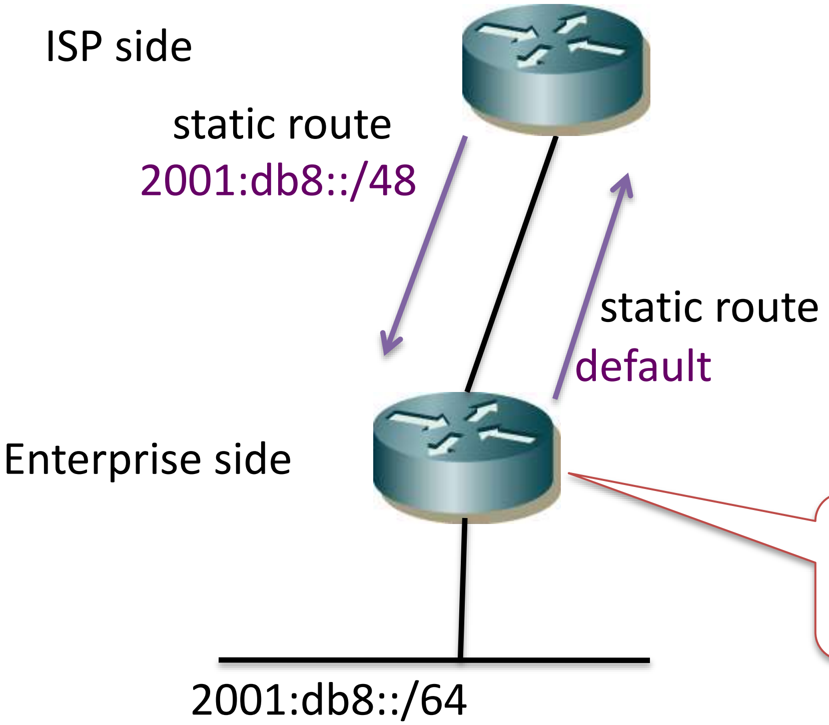
static route
default

Enterprise side

2001:db8::/64

They used only /64 out of /48, and there was no care about unused network. So packets to unused space were looped on the link

Route Termination to avoid loop



ipv6 route 2001:db8::/48 null 0 254

Services on Router

- disable unnecessary services
  - – no ip http servers
- enable minimal services
  - – vty (remote-access)
  - – snmp
  - – Routing Protocol

RA (Route Advertisement)

- Disable it on unnecessary link
  - router to router links
    - IX and peering links
    - backbones
  - statically configured hosts only
    - Data Center
- Enable if needed
  - SLAAC

NOC

- IPv6 capable NOC
  - To check IPv6 reliability
    - ping6 and traceroute6 from your terminal
  - At least monitoring system should be IPv6 capable
- Filtering on devices to allow access from NOC only
  - ACL on Routers

Access control for vty

cisco

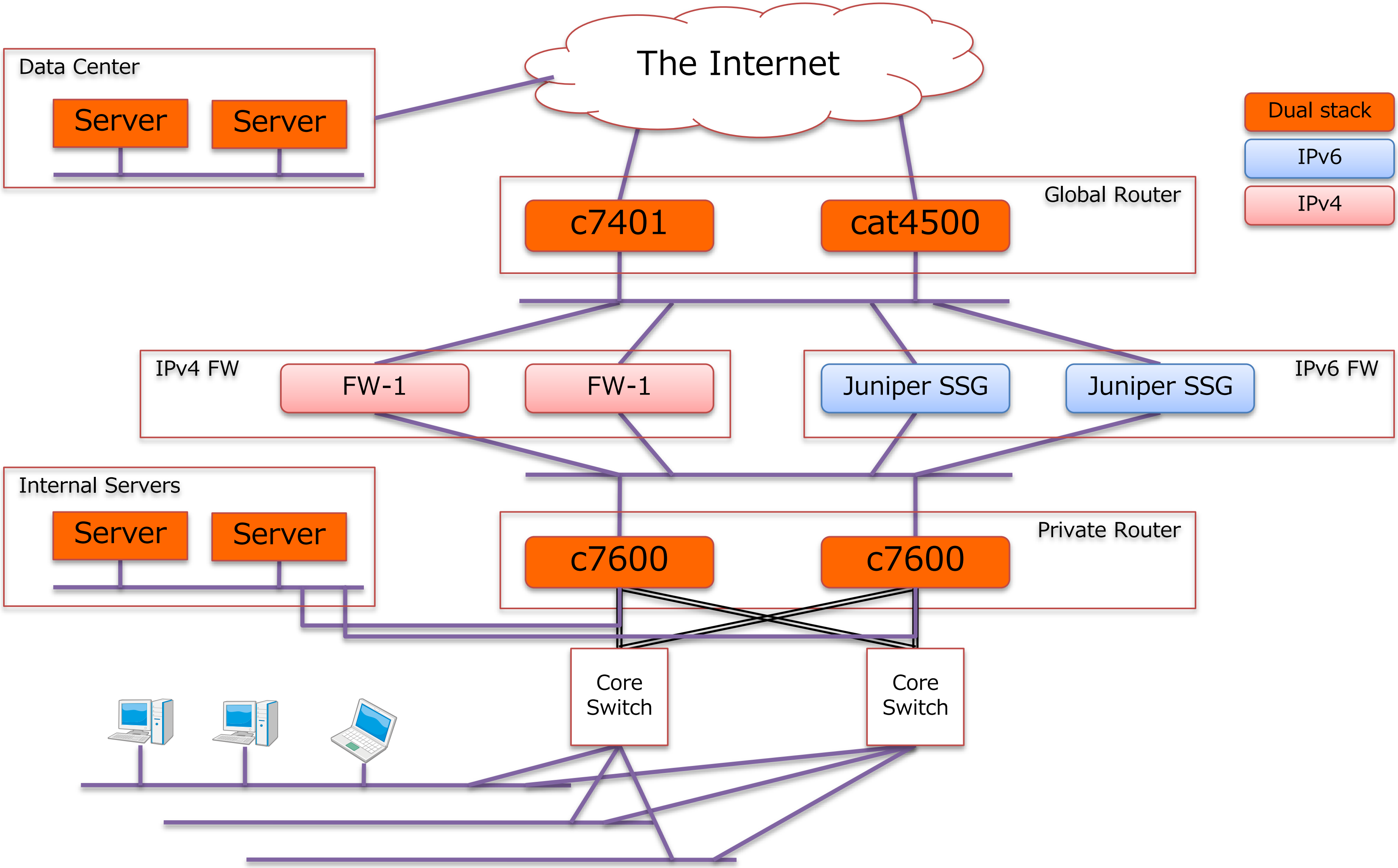
```
ipv6 access-list <list_name>
 permit ipv6 2001:db8:89ab::/48 any
 deny ipv6 any any
line vty 0 4
 ipv6 access-class <list-name> in
```

juniper

```
[edit]
policy-options {
  prefix-list <prefix-list_name> {
     2001:db8::89ab:/48;
| |
interfaces {
   lo0 { unit 0 { family inet6 {
      filter { input <filter_name>;}
}}}}
```

```
firewall {
  family inet6 { filter <filter_name> {
        term Permit {
           from { source-prefix-list { <prefix-list_name> ; }
           next-header tcp; destination-port [ telnet ssh ];
           }
           then accept;
        } term Deny {
           from {
           next-header tcp; destination-port [ telnet ssh ];
           }
           then reject;
        } term Default {
           then accept;
  }}}}
```

IIJ Office Network



Internal Network

- HSRPv2 for Router Redundancy (fe80::1)
- RA from Routers (w/ Other Config Flag)
- DHCPv6 for other configuration
  - default route as fe80::1
  - IPv6 DNS cache servers
  - DNS search list

Security Policy at IIJ

- Mostly same as IPv4's
  - deny incoming TCP connection
  - allow outgoing TCP connection, but only for pre-approved ports
  - more strict policy is applied on Sales team. :p
- Users should apply:
  - latest software update
  - latest anti-virus pattern

Differences

- Different enterprise has different policy:
  - no SLAAC and DHCPv6 addressing
  - no privacy address
- It depends on threat model
  - internal threat

Other Security Practice at IIJ

- Version Checker on Internal web site
  - browser, adobe flash, adobe reader, Java and other major add-ons
- To notify users to use up-to-date version
  - Several internal site don't allow user to access with older version
  - Firewall is not enough to protect clients
    - Attackers somehow lead users to their web site to compromise the client

Enterprise Addressing

- We got /48 for our office, and designed:
  - /49 outside
  - /49 inside
    - /64 for every segment
- Actual
  - No such demands on outside
    - Most servers are at Data Centers
  - Regional offices
    - Each office needs own Internet connectivity

Renumbering

- /52 for head quarter
  - one /56 would be allocated for outside network
  - The rest of the prefix are for internal network
- /54 or /56 for branch offices
  - depending on its size
- This will be our 3rd IPv6 renumbering
  - 1st was from 6bone(3ffe) to PA
  - 2nd was due to restructuring our service addressing policy

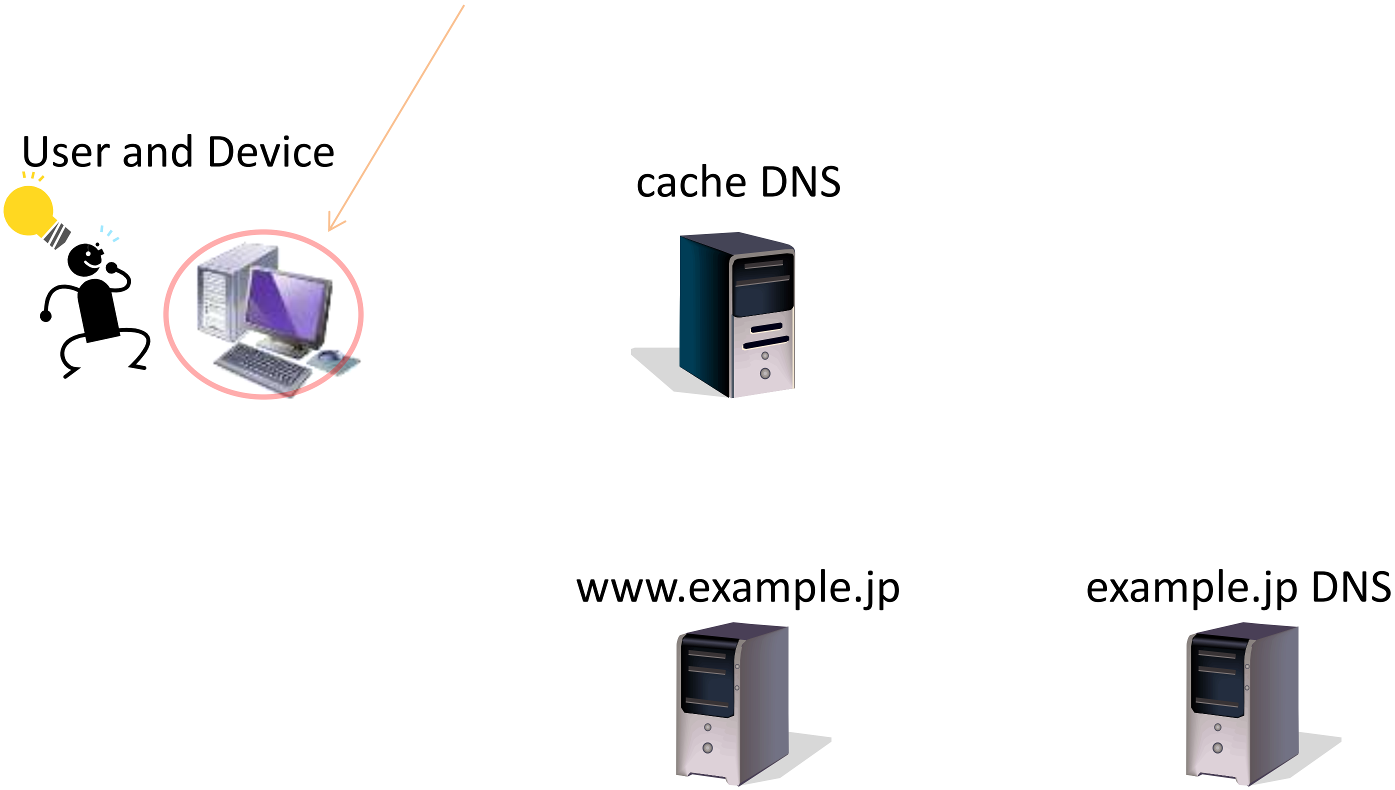
Users and IPv6

- Users usually access a service by indicating FQDN (hostname)
  - Users don't care whether it's IPv4 or IPv6
  - Devices resolve the hostname, and access to the resource

I want access www.example.jp

1. IPv6 capable device

- User’s device supports IPv6



User and Device

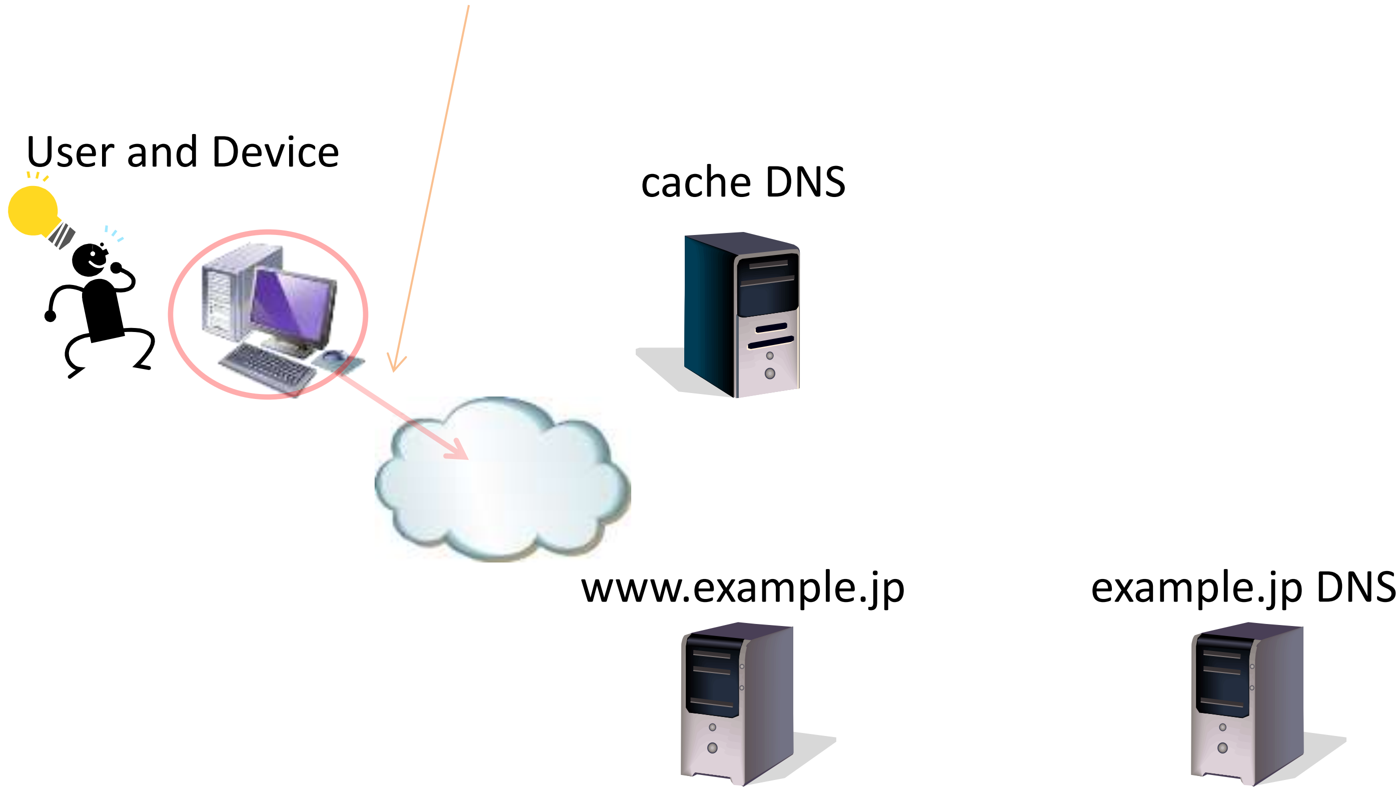
cache DNS

www.example.jp

example.jp DNS

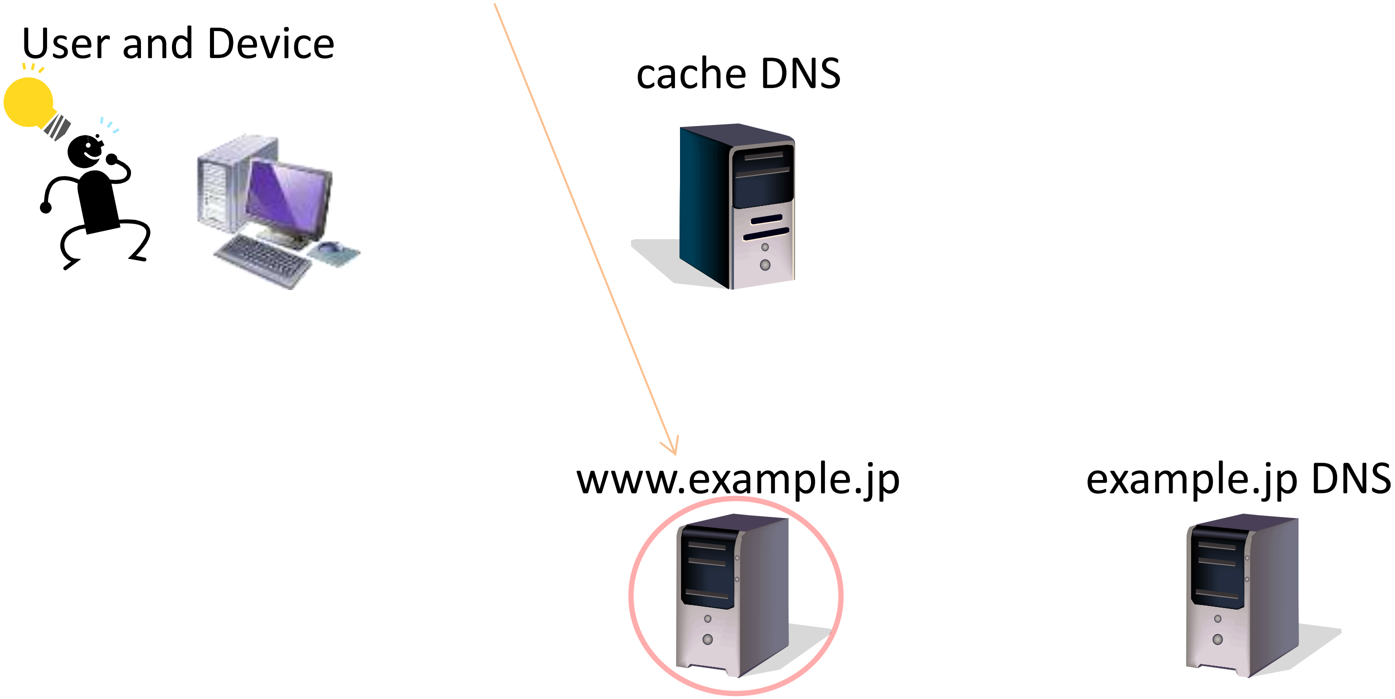
2. IPv6 connectivity

- User has an IPv6 capable connectivity



3. IPv6 capable service

- The server has IPv6 connectivity and IPv6 capable service



User and Device

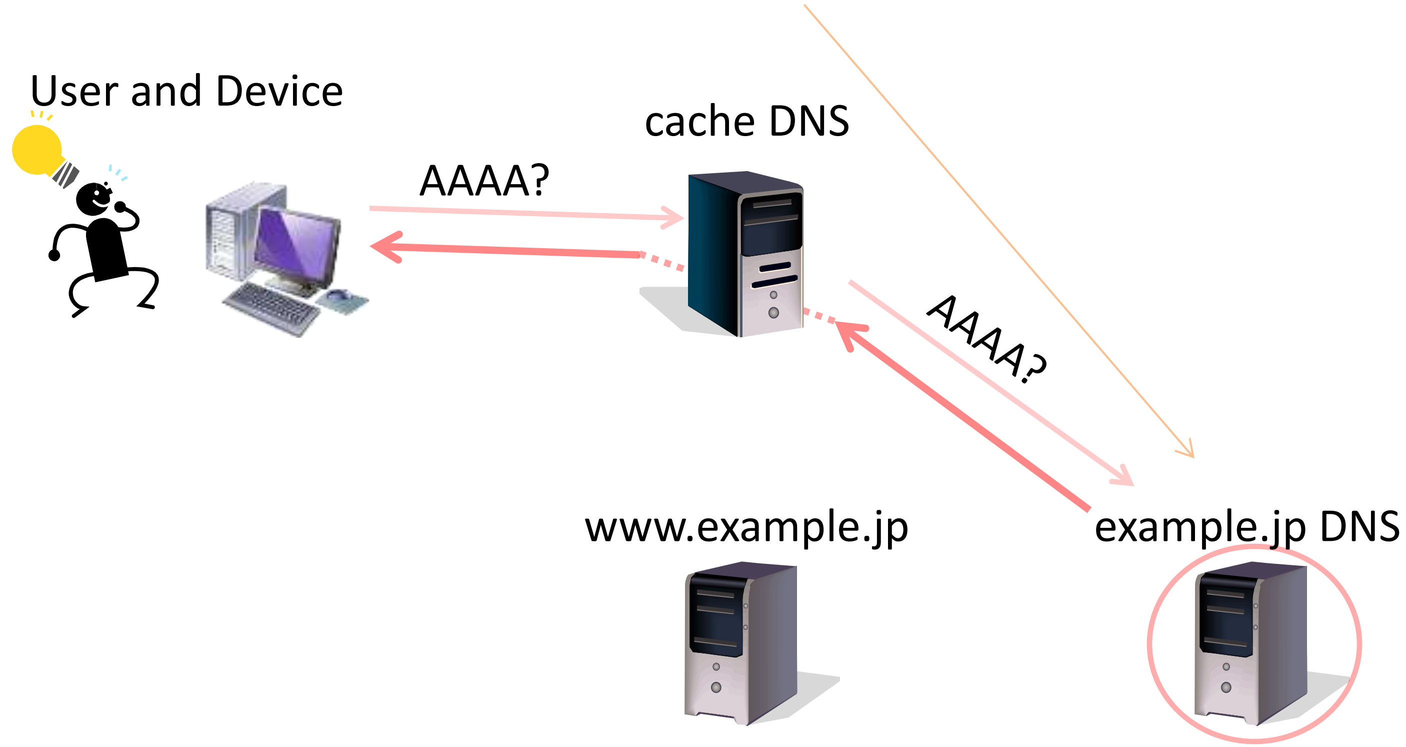
cache DNS

www.example.jp

example.jp DNS

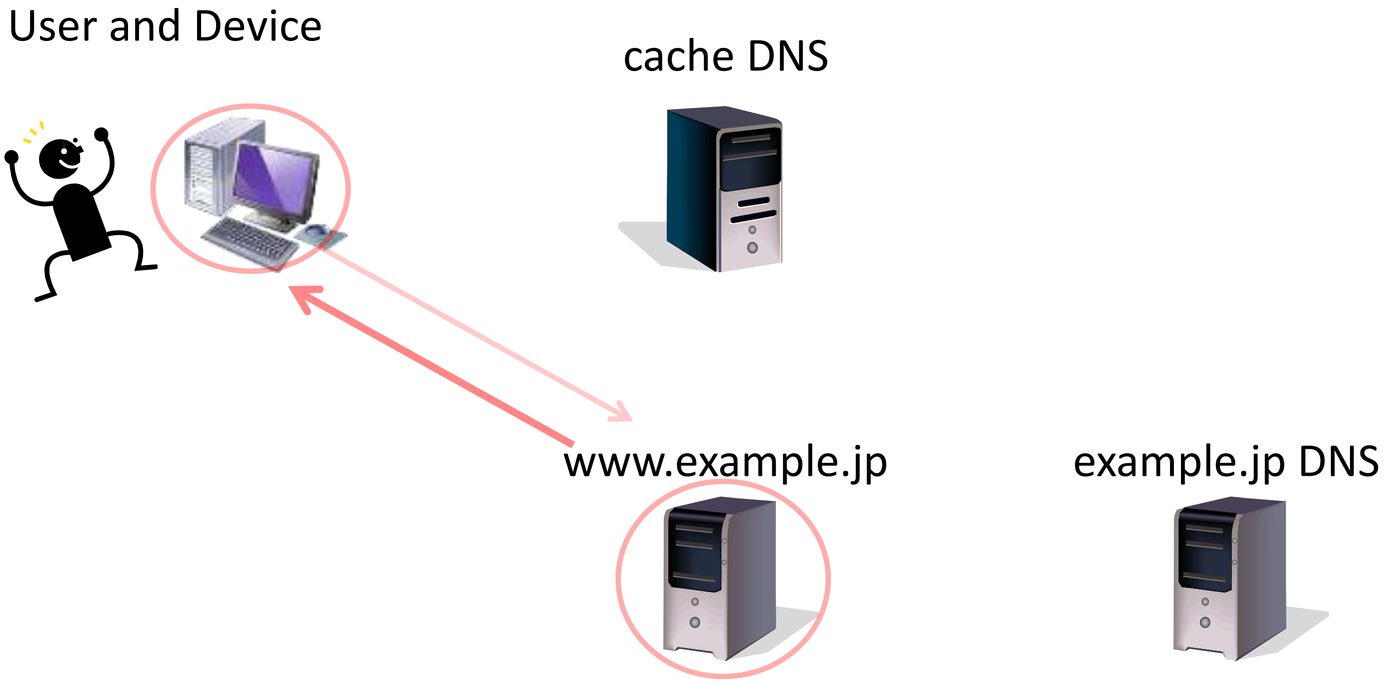
4. AAAA entry for the hostname

- User can resolve AAAA of www.example.jp if the hostname has AAAA



5. IPv6 reachability

- The device tries to connect via IPv6. Everyone is happy if the user can get the content.



Trouble shooting

- User side
  1. IPv6 capable device
  2. IPv6 connectivity
- Service side
  3. IPv6 capable service
  4. AAAA entry for the hostname
- Between User and Service
  5. IPv6 reachability

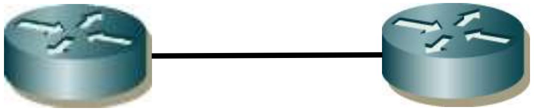
DNS - hostname resolving

- Resource Records
  - IPv4 - A record
  - IPv6 - AAAA record
- Query Transport
  - IPv4
  - IPv6
- A host might send ‘A’ query via IPv6, and vice versa

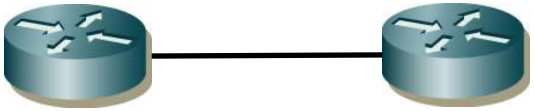
Separated or Dual Stacked design

Separated

IPv4 only



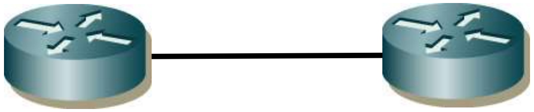
IPv6 only



- no need to touch old equipment

Dual Stacked

IPv6 and IPv4



- efficient bandwidth

Dual Stack

- Users don’t care about IP stack
- Service side might control IPv4 or IPv6 by DNS
  - but they can’t control # of Users
- Traffic just depends on these situation
  - One day traffic might shift IPv4 to IPv6, or ???
- From capacity planning point of view, Dual Stack is the better design.
  - Traffic trend

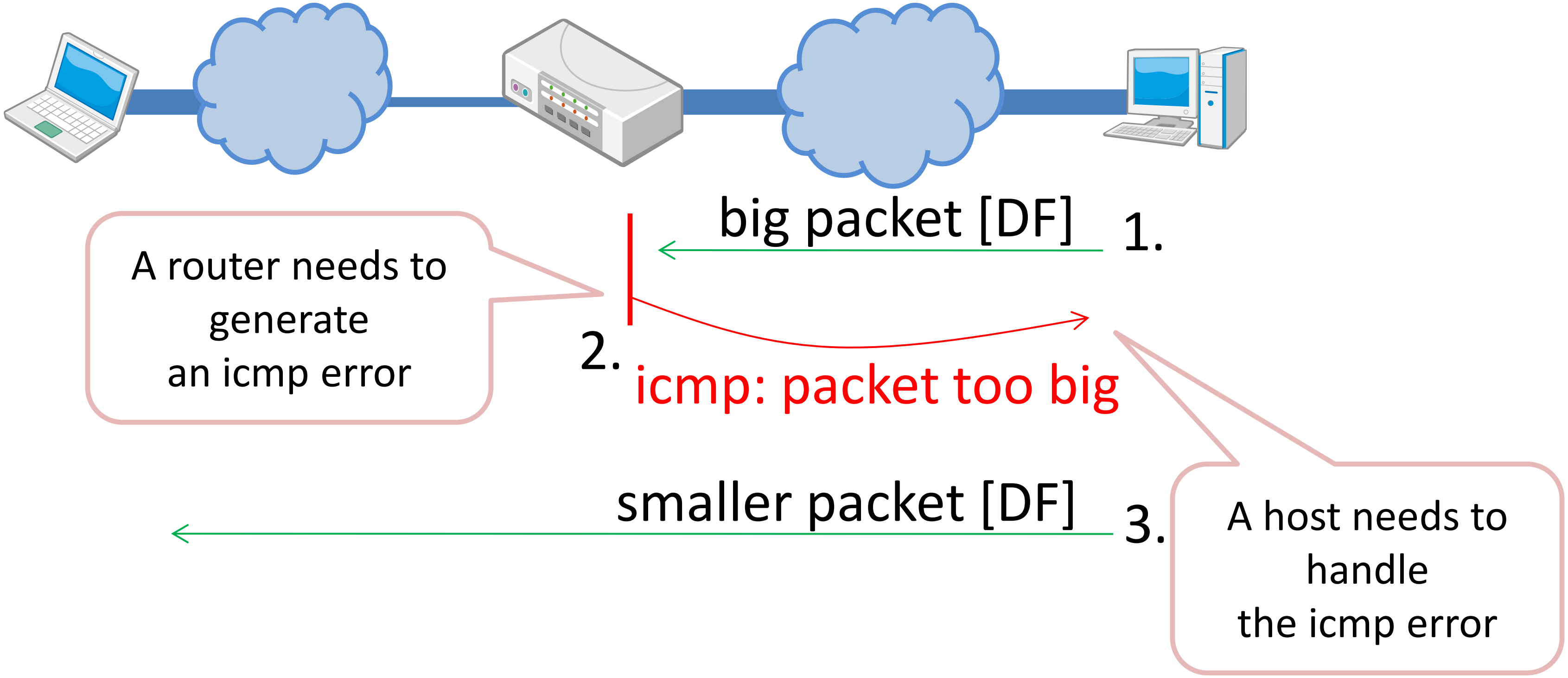
Dual Stack and network equipment

- Today’s routers support IPv6
  - and it works
- Care about IPv4 specialized equipment
  - Layer-2 Switch that too focus on IPv4
  - VLAN implementation
  - Ether-type based tuning

IPv6 and Path MTU Discovery

- Path MTU discovery for IPv6 [RFC1981]
  - IPv6 nodes SHOULD implement Path MTU Discovery in order to discover and take advantage of paths with PMTU greater than the IPv6 minimum link MTU [IPv6-SPEC].
- IPv6 minimum link MTU [IPv6-SPEC] == 1280

path MTU discovery scenario



1. big packet [DF]

A router needs to generate
an icmp error

2. icmp: packet too big

3. smaller packet [DF]

A host needs to handle
the icmp error

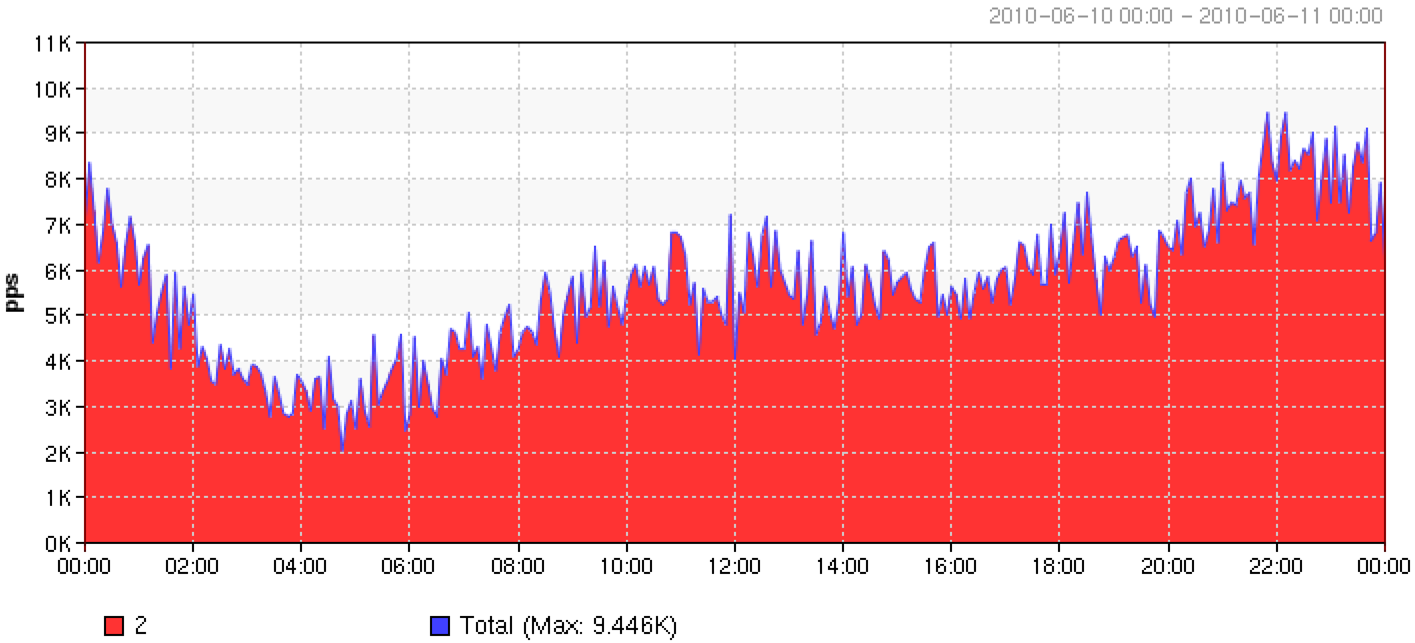
icmp originating-limit

- cisco ios
  - ip icmp rate-limit unreachable 500
    - means icmp errors are limited to one every 500msec
  - ipv6 icmp error-interval 100
    - means icmp errors are limited to one every 100msec
- juniper junos
  - icmpv4-rate-limit {packet-rate 1000;};
    - means max 1000pps for icmp to/from RE
  - icmpv6-rate-limit {packet-rate 1000;};
    - means max 1000pps for icmp to/from RE

summary of failures

- The Path MTU Discovery could fail even if all of given devices support it
  - performance issue
  - icmp message could be discarded
- The Path MTU Discovery is like an “exceptional handling”
  - network ops are usually focused on ‘forwarding performance’ of routers.

IPv4 TCP SYN rate at a consumer aggregation router



learning from IPv4

- Almost of all broadband routers have a TCP MSS hack capability
- It chokes TCP MSS on a tunnel link
  - PPPoE, or whatever the link MTU is less than 1500
  - to avoid unnecessary fallbacks
- The TCP MSS hack works fine
  - No complaint from customers

suggestion for IPv6

- TCP MSS hack by broadband router
  - whatever connectivity MTU is less than 1500
    - PPPoE, or any other tunnel
  - But this works only for TCP
- Avoid to use any Tunnel at backbone/peering
  - tunnel might cause high cpu load
  - tunnel might cause PathMTUd blackhole
  - Use dualstack/native links as possible

Mitigating DoS Attack

- the same techniques are still usable
  - filtering
  - null routing
  - rate-limiting
- And to mitigate huge DoS attack, you need to ask help for upstreams and peers
  - the same as IPv4 world, it’s internet

null routing

- ipv6 route 2001:db8:1234::13:13/128 null0
  - discarding packets to the destination
- if you enable uRPF loose mode at the incoming interface on Cisco
  - discarding packets to the destination, and
  - discarding packets originating from prefix

Summary

- Almost the same
  - IPv4 knowledge are still workable in many cases
- More unused space
  - avoid looping