# IPv6 Tutorial
# DREN Networkers 2010

Ron Broersma

# Introduction

- We've given IPv6 tutorials at all the DREN conferences since 2004, and have been presenting on the topic of IPv6 since 2000
- So, I assume…
  - you are familiar with IPv6
  - you have it enabled on your laptop
    - first did this at the 2004 tutorial
  - you have started to enable IPv6 at your site, or have at least thought about it
  - you are aware of the necessity to transition to IPv6 soon

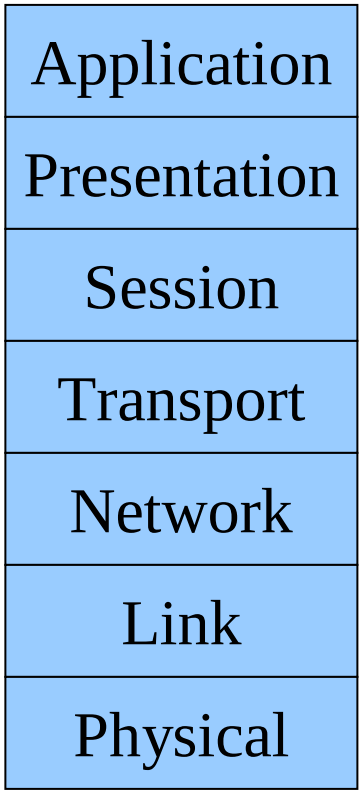
# IPV6 101

# IP version 6

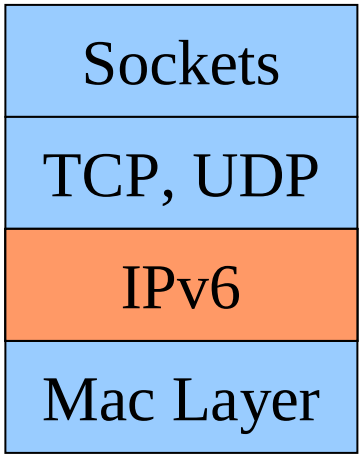
- Next generation Internet protocol
- What we use today is IP version 4
  - In production use for 30+ years, and showing its age
- Features
  - Huge address space (128 bit addresses). Aggregation based hierarchy for route table efficiency.
  - Simplified, fixed length header – better options support
  - Mandatory IPsec (promise for improved security)
  - Autoconfiguration, ease of renumbering
  - Support for QoS

# The piece that has changed

ISO 7 Layer Model



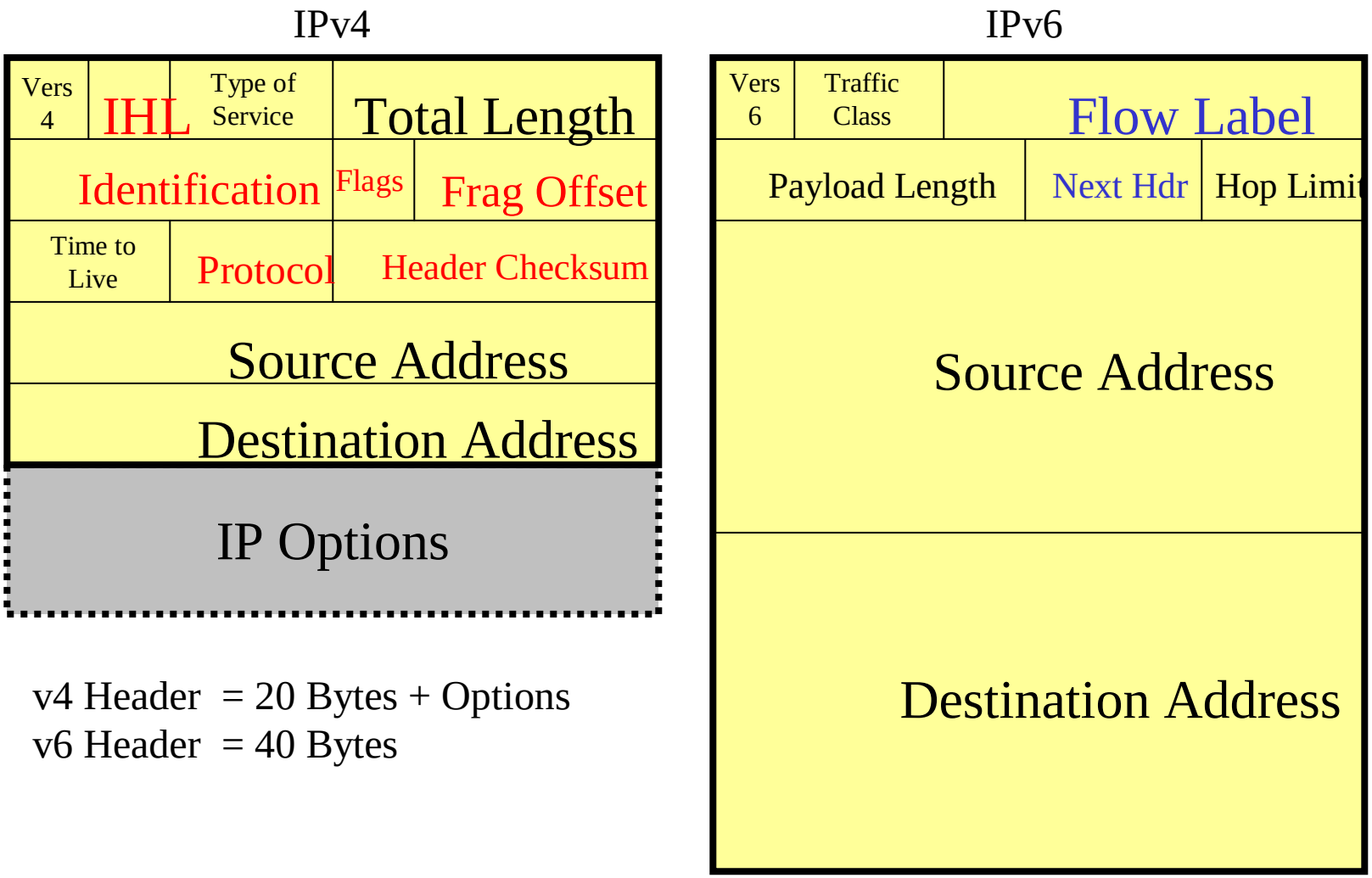

Internet Stack



# IPv6 Specification

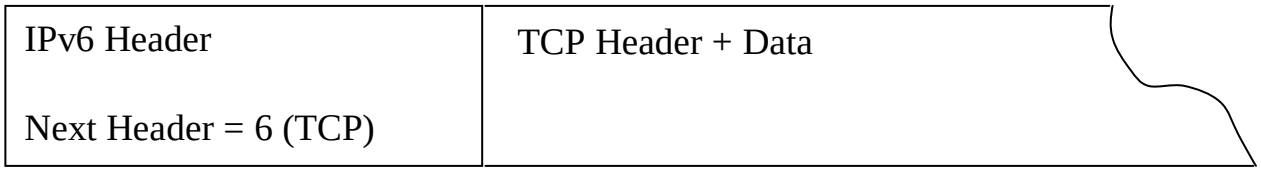
- RFC 2460, Dec '98 (replaces RFC 1883)
- Expanded Addressing
  - 128 bit addresses instead of 32
    - More hierarchy, more nodes
    - Improved multicast (includes scoping)
    - Anycast
  - Simplified IP header
    - Lower processing cost
  - Improved support for extensions and options
  - Flow Labeling capability
  - Security extensions

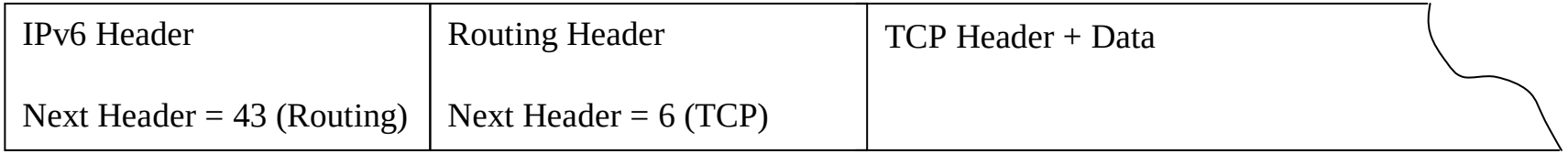
# IP Header Changes

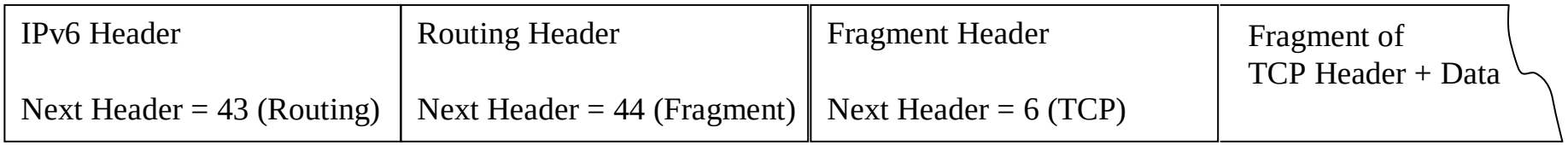



v4 Header = 20 Bytes + Options
v6 Header = 40 Bytes

# Next Header, examples



# Addressing

- 128 bits gets you 340,282,366,920,938,463,463,374,607,431,768,211,456 addresses.
- That's 665,570,793,348,866,943,898,599 addresses per square meter of the surface of the planet Earth.
- So, how do we allocate all that address space?

# IPv6 Address Space

| Allocation | Prefix (binary) | Fraction of Address Space | |
|---|---|---|---|
| Unassigned (see Note 1 below) | 0000 0000 | 1/256 | |
| Unassigned | 0000 0001 | 1/256 | |
| Reserved for NSAP Allocation | 0000 001 | 1/128 | [RFC1888] |
| Unassigned | 0000 01 | 1/64 | |
| Unassigned | 0000 1 | 1/32 | |
| Unassigned | 0001 | 1/16 | |
| Global Unicast | 001 | 1/8 | [RFC2374] |
| Unassigned | 010 | 1/8 | |
| Unassigned | 011 | 1/8 | |
| Unassigned | 100 | 1/8 | |
| Unassigned | 101 | 1/8 | |
| Unassigned | 110 | 1/8 | |
| Unassigned | 1110 | 1/16 | |
| Unassigned | 1111 0 | 1/32 | |
| Unassigned | 1111 10 | 1/64 | |
| Unassigned | 1111 110 | 1/128 | |
| Unassigned | 1111 1110 0 | 1/512 | |
| Link-Local Unicast Addresses | 1111 1110 10 | 1/1024 | |
| Site-Local Unicast Addresses | 1111 1110 11 | 1/1024 | |
| Multicast Addresses | 1111 1111 | 1/256 | |

# IPv6 Address Space

| Allocation | Prefix (binary) | Fraction of Address Space | |
|---|---|---|---|
| Unassigned (see Note 1 below) | 0000 0000 | 1/256 | |
| Unassigned | 0000 0001 | 1/256 | |
| Reserved for NSAP Allocation | 0000 001 | 1/128 | [RFC1888] |
| Unassigned | 0000 01 | 1/64 | |
| Unassigned | 0000 1 | 1/32 | |
| Unassigned | 0001 | 1/16 | |
| Global Unicast | 001 | 1/8 | [RFC2374] |
| Unassigned | 010 | 1/8 | |
| Unassigned | 011 | 1/8 | |
| Unassigned | 100 | 1/8 | |
| Unassigned | 101 | 1/8 | |
| Unassigned | 110 | 1/8 | |
| Unassigned | 1110 | 1/16 | |
| Unassigned | 1111 0 | 1/32 | |
| Unassigned | 1111 10 | 1/64 | |
| Unassigned | 1111 110 | 1/128 | |
| Unassigned | 1111 1110 0 | 1/512 | |
| Link-Local Unicast Addresses | 1111 1110 10 | 1/1024 | |
| Site-Local Unicast Addresses | 1111 1110 11 | 1/1024 | |
| Multicast Addresses | 1111 1111 | 1/256 | |

# Representation of Addresses

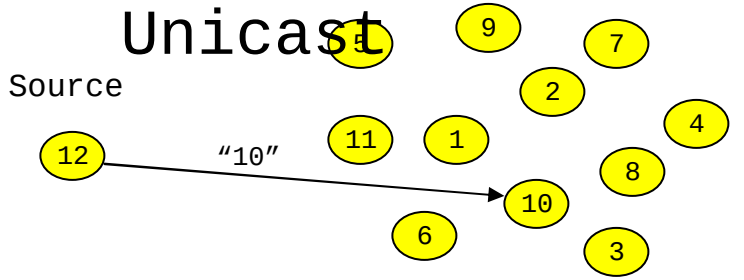
- IPv4 used 8 bit fields, separated by periods, in decimal notation.
  - d.d.d.d
- The IPv6 128 bit address is broken into eight 16 bit fields separated by colons and given in hexadecimal notation.
  - x:x:x:x:x:x:x:x
  - Example: 2FAB:00F7:18C0:298D:0000:0000:FA48:972B
- Leading 0's in any field can be omitted.
- Any single sequence of fields of all 0's can be represented as ::
  - Above example could be 2FAB:F7:18C0:298D::FA48:972b

# Address representation

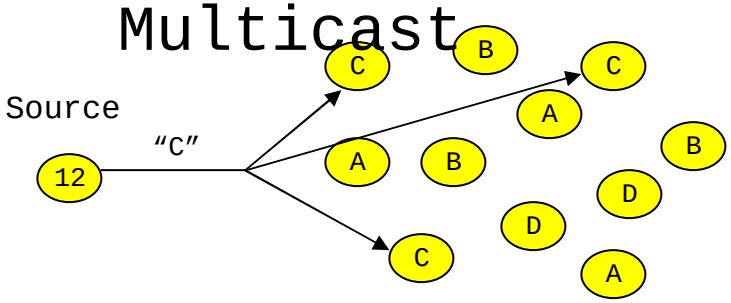
- If an address begins with ::, then 0 fill to the left. If an address ends with ::, then 0 fill to the right.
  - 0:0:0:0:0:0:0:1 can be shown as ::1
  - 0:0:0:0:0:0:0:0 can be shown as ::
    - Note: this is the "unspecified" address.
- The double colon :: can only appear once in an address.
- To specify an address in a URL, put it in square brackets. RFC 2732.
  - http://[2001:480:0110::102:23]:80/index.html

# Different Types of Addresses

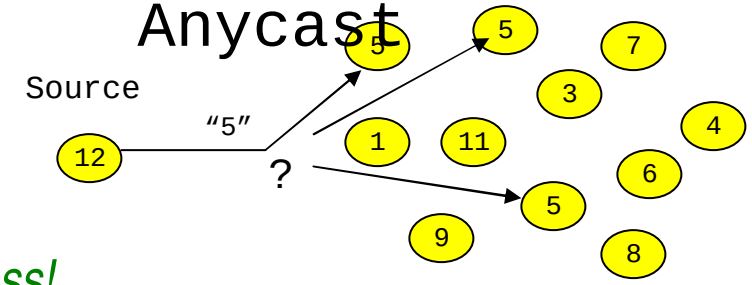
- Unicast: Destination address specifies exactly one target.




- Multicast: Destination address specifies a group that includes multiple targets




- Anycast: Destination address specifies the closest of multiple targets




*No Broadcast address!*

# Unicast Addressing

- Global Unicast – 2000::/3
  - Global scope, globally unique (like IPv4 public addresses).
  - All addresses start with a 2 or a 3 (for now).
    - i.e 2000:: through 3FFF::
    - This is 1/8 of the total IPv6 address space

DREN is 2001:480::

- Loopback address
  - ::1
  - Like 127.0.0.1 in IPv4.
  - Destination is "this node"

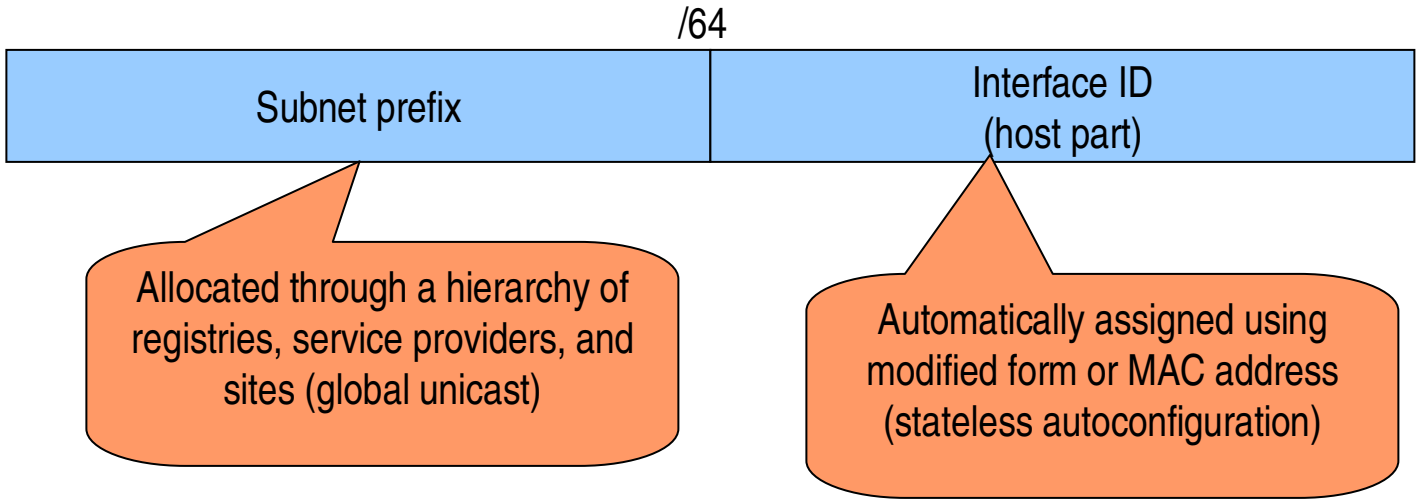
Ping yourself:

```
% ping ::1
```

- Unspecified address.
  - :: (all zeros)
  - Like 0.0.0.0 in IPv4.
  - Use this to indicate "no address"

# Address Structure

- Unicast addresses are structured as a subnetwork prefix and an interface identifier.




*Size of a given (sub)network is effectively not limited by the number of unique host values as was the case in IPv4 where a /24 (Class C) net can only have 254 hosts.*

# Interface ID

- Required to be in modified EUI-64 format
  - (except where prefix starts with binary 000)
- Deriving IID from a 48 bit MAC address
  - Invert the value of the universal/local (7th) bit.
  - Insert 2 octets with value 0xFFFE in middle of MAC address.
  - Example:

48 bit MAC: `080020a8539d`

IID: `::a00:20ff:fea8:539d`

Used in global unicast addr: `2001:480:0:100:a00:20ff:fea8:539d`

# Multicast Addressing

- FF00::/8
  - Anything that starts with "FF" is multicast
- The 2nd byte is used for "flags" and "scope".


Flags
0 – permanently assigned
"well known" (IANA)
1 - transient

Scope
1 – interface local
2 – link local
3 – subnet local
4 – admin local
5 – site local
8 – organization local
E - global

- Predefined multicast addresses
  - FF01::1, FF02::1 – all nodes on (interface, link)
  - FF01::2, FF02::2, FF05::2 – all routers (interface, link, site)
  - FF02::6 – OSPFv3, FF02::9 – RIPng, etc.
  - http://www.iana.org/assignments/ipv6-multicast-addresses

# Interfaces get multiple addresses

- Every interface is assigned multiple addresses:
  - A link-local address
  - Zero or more unicast or anycast addresses (manually or automatically configured).
  - The "all nodes" multicast address.
    - FF02::1
  - The "solicited node" multicast address for each of its unicast and anycast addresses.
  - Multicast addresses for any groups to which it belongs.
- Routers also have:
  - Subnet-router anycast address (<subnet prefix>::0)
  - The "all routers" multicast address
    - FF02::2

# Example: Linux

```
$ /sbin/ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:06:5B:8C:29:2D
          inet addr:128.49.192.82  Bcast:128.49.192.255  Mask:255.255.255.0
          inet6 addr: fe80::206:5bff:fe8c:292d/10 Scope:Link
          inet6 addr: 2001:480:11:1192:206:5bff:fe8c:292d/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:422941335 errors:2 dropped:0 overruns:0 frame:2
          TX packets:451187648 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:4294967295 (4095.9 Mb)  TX bytes:4294967295 (4095.9 Mb)
          Interrupt:28
```

# Example: Cisco Router

```
>sh ipv6 int
FastEthernet3/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::230:85FF:FE37:98F1
  Global unicast address(es):
    2001:480:0:102::1, subnet is 2001:480:0:102::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::5
    FF02::6
    FF02::9
    FF02::1:FF37:98F1
    FF02::1:FF00:1
  MTU is 1500 bytes
```

# Example: Solaris

```
% ifconfig -a6
lo0: flags=2000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv6> mtu 8252 index 1
        inet6 ::1/128
hme0: flags=2000841<UP,RUNNING,MULTICAST,IPv6> mtu 1500 index 2
        inet6 fe80::a00:20ff:fea8:539d/10
hme0:1: flags=2080841<UP,RUNNING,MULTICAST,ADDRCONF,IPv6> mtu 1500 index 2
        inet6 2001:480:10:192:a00:20ff:fea8:539d/64
```

# Mac OS-X

```
% ifconfig en0
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fe80::203:93ff:fe0a:fa2e%en0 prefixlen 64 scopeid 0x4
        inet 128.49.84.105 netmask 0xfffffc00 broadcast 128.49.87.255
        inet6 2001:480::102:203:93ff:fe0a:fa2e prefixlen 64 autoconf
        ether 00:03:93:0a:fa:2e
        media: autoselect (100baseTX <full-duplex>) status: active
```

# Example: WinXP

```
C:\>ipv6 if
Interface 4:
  uses Neighbor Discovery
  uses Router Discovery
  link-level address: 00-04-23-60-47-80
    preferred link-local fe80::204:23ff:fe60:4780, life infinite
    multicast interface-local ff01::1, 1 refs, not reportable
    multicast link-local ff02::1, 1 refs, not reportable
    multicast link-local ff02::1:ff60:4780, 1 refs, last reporter
  link MTU 1500 (true link MTU 1500)
  current hop limit 128
  reachable time 19000ms (base 30000ms)
  retransmission interval 1000ms
  DAD transmits 1
  default site prefix length 48
```

# Example: Vista

```
C:>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::f826:42f9:2e5e:e10%8
   IPv4 Address. . . . . . . . . . . : 10.211.55.3
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.211.55.1
```

# Comparisons

## IPv4

- 32 bit addresses
  - 128.49.16.7
  - 127.0.0.1
- Manual or DHCP
- Single IP on interface
- Get your address space from "the NIC"
- Address plan tries to efficiently use address space

## IPv6

- 128 bit addresses
  - 2001:480:10:16::7
  - ::1
- Manual or plug-n-play
  - DHCPv6 in the future
- Many IP addrs on interface
- Get your address space from your ISP (DREN)
- Address plan chosen for ease of operation and maintenance

# Address Types

| Prefix | Designation and Explanation | IPv4 Equivalent |
|---|---|---|
| ::/128 | **Unspecified**<br>This address may only be used as a source address by an initialising host before it has learned its own address. | 0.0.0.0 |
| ::1/128 | **Loopback**<br>This address is used when a host talks to itself over IPv6. This often happens when one program sends data to another. | 127.0.0.1 |
| ::ffff/96<br><br>Example:<br>::ffff:192.0.2.47 | **IPv4-Mapped**<br>These addresses are used to embed IPv4 addresses in an IPv6 address. One use for this is in a dual stack transition scenario where IPv4 addresses can be mapped into an IPv6 address. See RFC 4038 for more details. | There is no equivalent. However, the mapped IPv4 address can be looked up in the relevant RIR's Whois database. |
| fc00::/7<br><br>Example:<br>fd8f:f53b:82e4::53 | **Unique Local Addresses (ULAs)**<br>These addresses are reserved for local use in home and enterprise environments and are not public address space.<br><br>These addresses might not be unique, and there is no formal address registration. Packets with these addresses in the source or destination fields are not intended to be routed on the public Internet but are intended to be routed within the enterprise or organisation.<br><br>See RFC 4193 for more details. | Private, or RFC 1918 address space:<br><br>10.0.0.0/8<br>172.16.0.0/12<br>192.168.0.0/16 |
| fe80::/10<br><br>Example:<br>fe80::200:5aee:feaa:20a2 | **Link-Local Addresses**<br>These addresses are used on a single link or a non-routed common access network, such as an Ethernet LAN. They do not need to be unique outside of that link.<br><br>Link-local addresses may appear as the source or destination of an IPv6 packet. Routers must not forward IPv6 packets if the source or destination contains a link-local address.<br><br>Link-local addresses may appear as the source or destination of an IPv6 packet. Routers must not forward IPv6 packets if the source or destination contains a link-local address. | 169.254.0.0/16 |

| Prefix | Designation and Explanation | IPv4 Equivalent |
|---|---|---|
| 2001:0000::/32<br><br>Example:<br>2001:0000:4136:e378:<br>8000:63bf:3fff:fdd2 | **Teredo**<br>This is a mapped address allowing IPv6 tunneling through IPv4 NATs. The address is formed using the Teredo prefix, the server's unique IPv4 address, flags describing the type of NAT, the obfuscated client port and the client IPv4 address, which is probably a private address. It is possible to reverse the process and identify the IPv4 address of the relay server, which can then be looked up in the relevant RIR's Whois database.<br><br>You can do this on the following webpage:<br>http://www.potaroo.net/cgi-bin/ipv6addr | No equivalent |
| 2001:0002::/48<br><br>Example:<br>2001:0002:6c::430 | **Benchmarking**<br>These addresses are reserved for use in documentation. They should not be used as source or destination addresses. | 198.18.0.0/15 |
| 2001:0010::/28<br><br>Example:<br>2001:10:240:ab::a | **Orchid**<br>These addresses are used for a fixed-term experiment. They should only be visible on an end-to-end basis and routers should not see packets using them as source or destination addresses. | No equivalent |
| 2002::/16<br><br>Example:<br>2002:cb0a:3cdd:1::1 | **6to4**<br>A 6to4 gateway adds its IPv4 address to the 2002::/16, creating a unique /48 prefix. As the IPv4 address of the gateway router is used to compose the IPv6 prefix, it is possible to reverse the process and identify the IPv4 address, which can then be looked up in the relevant RIR's Whois database.<br><br>You can do this on the following webpage:<br>http://www.potaroo.net/cgi-bin/ipv6addr | There is no equivalent but 192.88.99.0/24 has been reserved as the 6to4 relay anycast address prefix by the IETF. |
| 2001:db8::/32<br><br>Example:<br>2001:db8:8:4::2 | **Documentation**<br>These addresses are used in examples and documentation. They should never be source or destination addresses. | 192.0.2.0/24<br>198.51.100.0/24<br>203.0.113.0/24 |
| 2000::/3 | **Global Unicast**<br>Other than the exceptions documented in this table, the operators of networks using these addresses can be found using the Whois servers of the RIRs listed in the registry at:<br>http://www.iana.org/assignments/ipv6-unicast-address-assignments | No equivalent single block |
| ff00::/8<br><br>Example:<br>ff01:0:0:0:0:0:0:3 | **Multicast**<br>These addresses are used to identify multicast groups. They should only be used as destination addresses, never as source addresses. | 224.0.0.0/4 |

http://www.ripe.net/ipv6-address-types/ipv6-address-types.pdf

# Terminology

- Dual Stack
  - When a network or computer system operates with both IPv4 and IPv6 at the same time.
    - You can use either protocol to communicate
  - This is the most interoperable transition mechanism available
- SLAAC
  - StateLess Address AutoConfiguration
  - The plug-and-play addressing feature of IPv6, where the router tells you the first 64 bits of your address, and your MAC address determines the last 64 bits.

# Autoconfiguration

- Stateless (RFC 2462)
  - "plug 'n play"
  - Interface ID is the Modified EUI-64 address (generated from MAC address)
  - FE80::<IID> is link local
  - <prefix learned from router>::<IID> for global unicast addresses.
- Stateful
  - DHCPv6

Manual configuration is allowed as well, just like IPv4

# Transition mechanisms for IPv4 to IPv6

- Many transition mechanisms proposed
- These fall into 3 major categories:
  - Dual stack host
  - Various Tunneling mechanism (IPv6 packets encapsulated in IPv4 header)
    - Manually configured
    - Automatic (using IPv4-compatible IPv6 addresses)
      - 6to4, ISATAP, Teredo (Home use only!), DSTM, silkroad, etc.
      - Tunnel broker
  - Translation
    - NAT-PT

# IPv6 deployment: How hard is it?

- Easy parts
  - Dual-stacking the nets (WANs, LANs)
  - Enabling IPv6 functionality in modern operating systems
  - Establishing basic IPv6 services (DNS, SMTP, NTP)
  - Enabling IPv6 in some commodity services (HTTP)
- A little more challenging
  - Getting the address plan right
  - Operating and debugging a dual stack environment
  - Multicast (but easier than IPv4)
- Hard parts
  - Creating the security infrastructure where not supported in current products
    - firewalls, IDS, proxys, IDP/IPS, VPNs, ACLs
  - Working around missing or broken functionality
  - DHCPv6
  - Creating incentives to upgrade and try IPv6
  - Getting the vendors to fix bugs or incorporate necessary features
    - Not enough market pressure, so other activities take priority

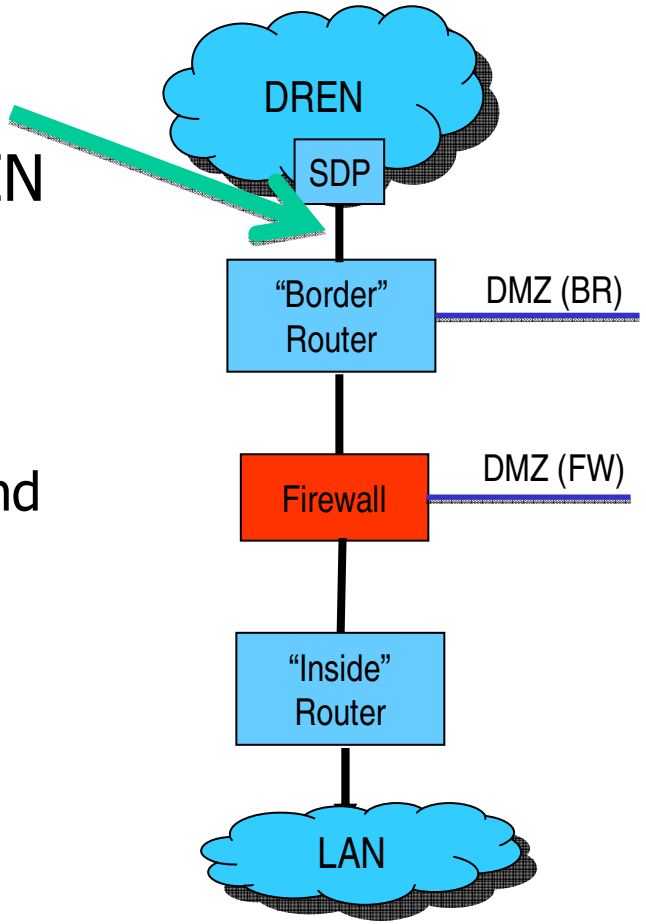
**Just focus on these for now, to get started.**

# Step 1

## GETTING IPV6 CONNECTIVITY TO DREN

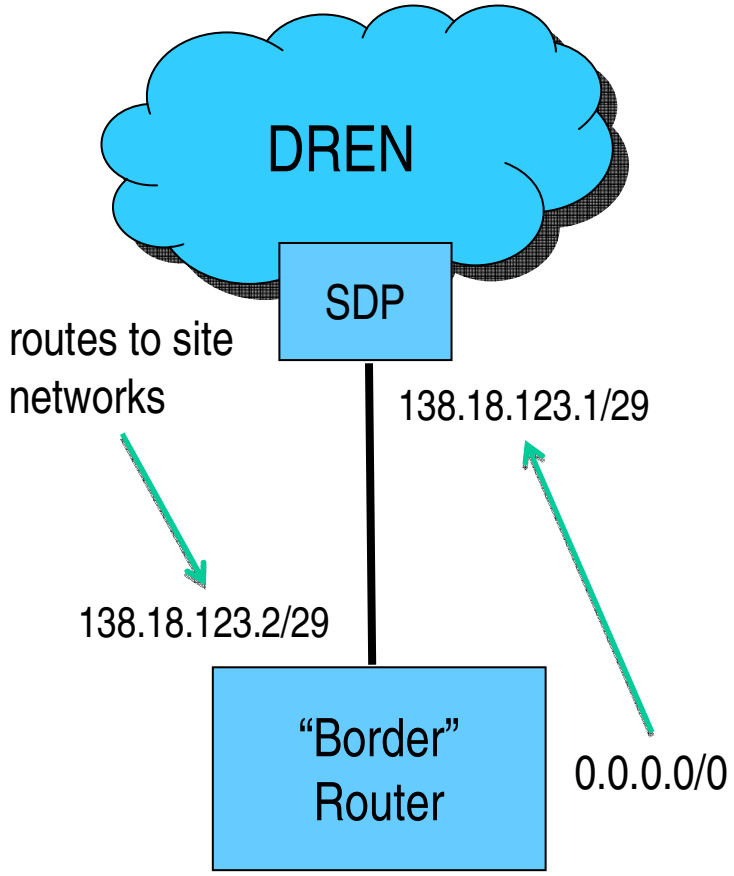
# Typical site

- Your "border router" or "customer premise router" connects to the nearest DREN SDP
  - Link has a 138.18.x.x address
  - Usually a /30 or a /29
  - The SDP might be the ".1" , and you the ".2".
- Task: Make this link "dual stack"



# Dual Stack the customer interface

- Existing connection has IPv4 addresses
- Routing entries support internetworking between the WAN and LAN
- Need to add IPv6 interface addresses
  - NOC will assign



# Request IPv6 address from NOC

```
To: noc@dren.net
Subject: request IPv6 addresses

Dear NOC,

I am at site XXX and the address of my border router is
138.18.xxx.yyy.

1. Please configure my customer interface for IPv6, and tell
me my interface address and default route.

2. Please provide a /48 prefix for use at my site, and route
it statically to the rest of DREN.

Respectfully,
<site POC>
```

# Reply from NOC (Example)

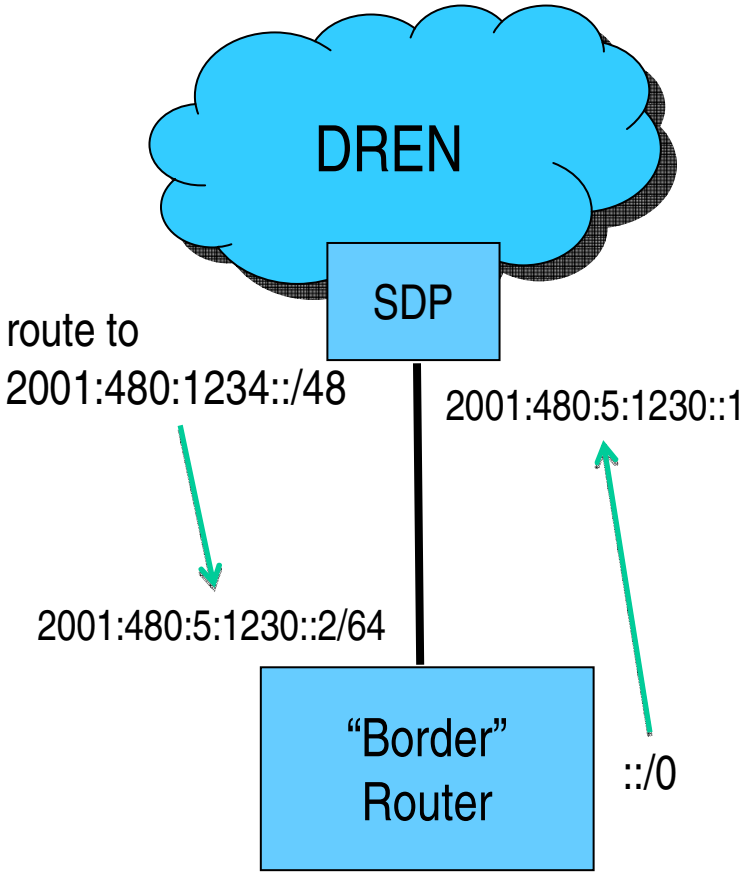
```
From: noc@dren.net
Subject: Re: request IPv6 addresses

Dear <site POC>,

Your interface address will be 2001:480:0:1F0::123:2/64.

The SDP has been configured for IPv6 address
2001:480:0:1F0::123:1 and you should use this as your
IPv6 default route.

Your site prefix will be 2001:480:1234::/48. This is being
statically routed to the rest of DREN.

Respectfully,
DREN NOC
```

# Request IPv6 address from DREN NOC

- Call 866-NOC-DREN and ask for
  - the IPv6 address of your interface to the SDP
  - your site IPv6 prefix
- This has already been pre-assigned and pre-configured by the NOC

# Configuring your Border Router

- Add the new IPv6 addresses at the same place as your IPv4 addresses
- Add a default route for IPv6
  - ::/0



# Example (Juniper) – interface configuration

```
interfaces {
/* Connection to DREN SDP */
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 138.18.123.2/29;
            }
            family inet6 {
                address 2001:480:5:1230::2/64;
            }
        }
    }
}
```

# Example (Juniper) – default route

```
routing-options {
    static {
        route 0.0.0.0/0 next-hop 138.18.123.1
        route ::/0 next-hop 2001:480:5:1230::1
    }
}
```

# Example (Cisco) – interface configuration

```
ipv6 unicast-routing
!
interface FastEthernet1/0
ip address 138.18.123.2
ipv6 address 2001:480:5:1230::2/64
!
ip route 0.0.0.0 0.0.0.0 138.18.123.1
ipv6 route ::/0 2001:480:5:1230::1
```

# Test it

```
admin@border-router> ping 2001:480:5:1230::1
PING6(56=40+8+8 bytes) 2001:480:5:1230::2 --> 2001:480:5:1230::1
16 bytes from 2001:480:5:1230::1, icmp_seq=0 hlim=64 time=1.025 ms
16 bytes from 2001:480:5:1230::1, icmp_seq=1 hlim=64 time=0.845 ms
16 bytes from 2001:480:5:1230::1, icmp_seq=2 hlim=64 time=0.950 ms
16 bytes from 2001:480:5:1230::1, icmp_seq=3 hlim=64 time=0.941 ms
16 bytes from 2001:480:5:1230::1, icmp_seq=4 hlim=64 time=0.987 ms
^C
--- 2001:480:5:1230::1 ping6 statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/std-dev = 0.845/0.950/1.025/0.060 ms
```

# Something more exotic…

```
admin@border-router> traceroute 2001:200:0:8002:203:47ff:fea5:3085
traceroute6 to 2001:200:0:8002:203:47ff:fea5:3085
 1  ge-0-0-0.mayberry.dren.net (2001:480:5:1230::1)  1.050 ms  0.960 ms  0.829 ms
 2  2004:480:0:1::1fd (2004:480:0:1::1fd)  12.635 ms  12.641 ms  12.650 ms
 3  2600:80a:40f::6 (2600:80a:40f::6)  210.554 ms  222.405 ms  213.545 ms
 4  2600:80a:40f::6 (2600:80a:40f::6)  209.689 ms  288.026 ms  211.659 ms
 5  hitachi1.otemachi.wide.ad.jp (2001:200:0:4401::3)  218.663 ms  224.612 ms
    211.057 ms
 6  2001:200:0:1802:20c:dbff:fe1f:7200 (2001:200:0:1802:20c:dbff:fe1f:7200)
    211.027 ms  209.414 ms  213.158 ms
 7  ve42.foundry4.nezu.wide.ad.jp (2001:200:0:11::66)  209.747 ms  217.316 ms
    214.108 ms
 8  ve45.foundry2.yagami.wide.ad.jp (2001:200:0:12::74)  211.289 ms  217.246 ms
    210.245 ms
 9  2001:200:0:4803:212:e2ff:fe28:1ca2 (2001:200:0:4803:212:e2ff:fe28:1ca2)
    216.332 ms  210.990 ms  219.418 ms
10orange.kame.net (2001:200:0:8002:203:47ff:fea5:3085)  213.972 ms  220.126 ms
   216.433 ms
```

# Other considerations

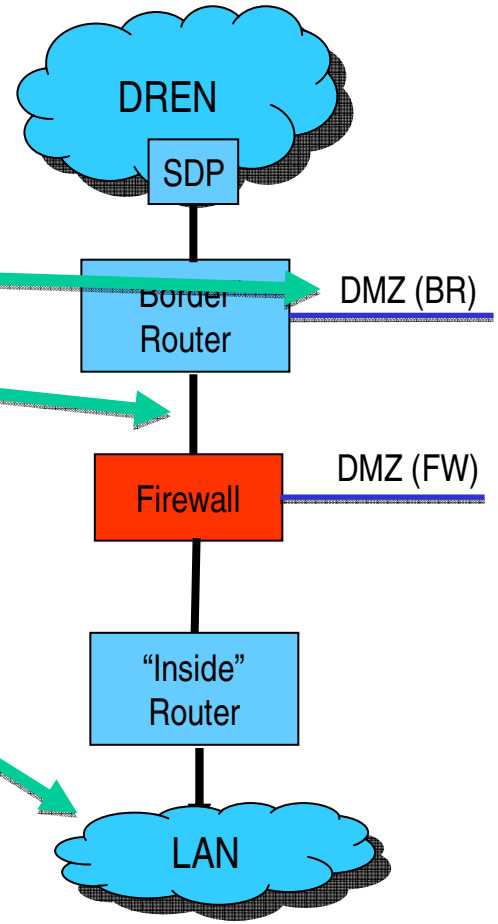
- If you exchange routes with DREN using BGP, your routing configuration will be different than the above examples
  - can send IPv6 routes over IPv4 BGP peer, or create a new IPv6 BGP peer
  - can't send IPv4 routes over an IPv6 BGP peer
- If you have a VERY old router, or are running ancient software on it, it might not support IPv6
  - time to upgrade!

# Step 2

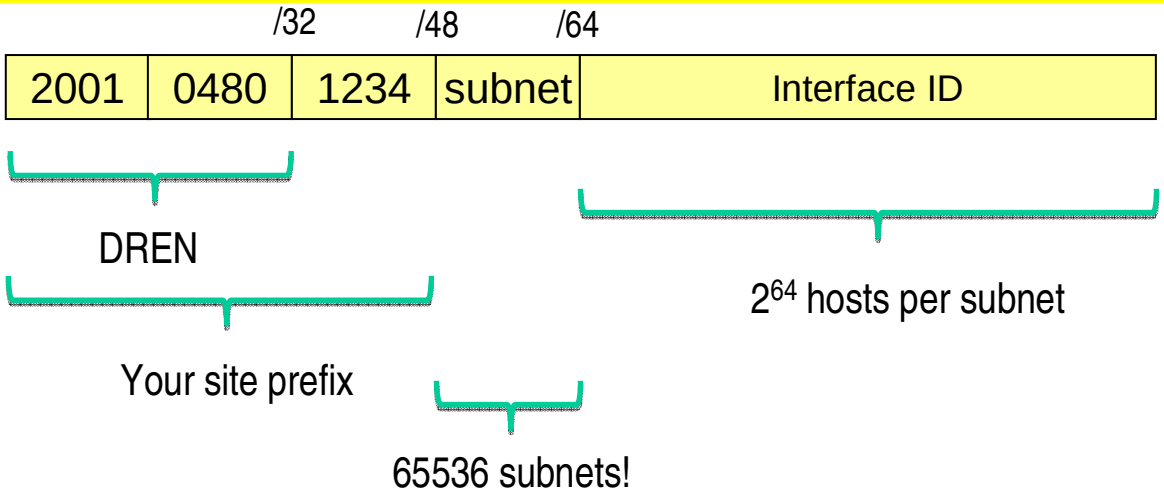
## CONFIGURING YOUR LAN FOR IPV6

# Adding IPv6 addresses to your LAN

- You probably have lots of subnets in your network
  - DMZs
  - Interconnects
  - user subnets
- These subnets all have IPv4 addresses today
- What IPv6 addresses should be assigned to these subnets?

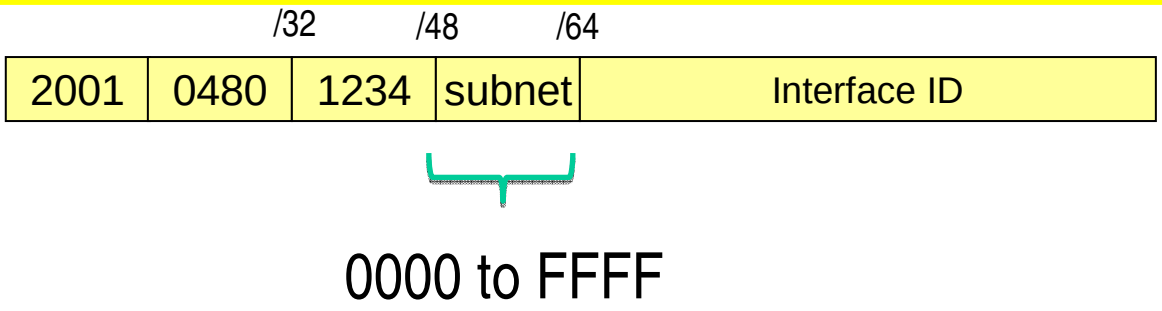


# Syntax of your /48 prefix



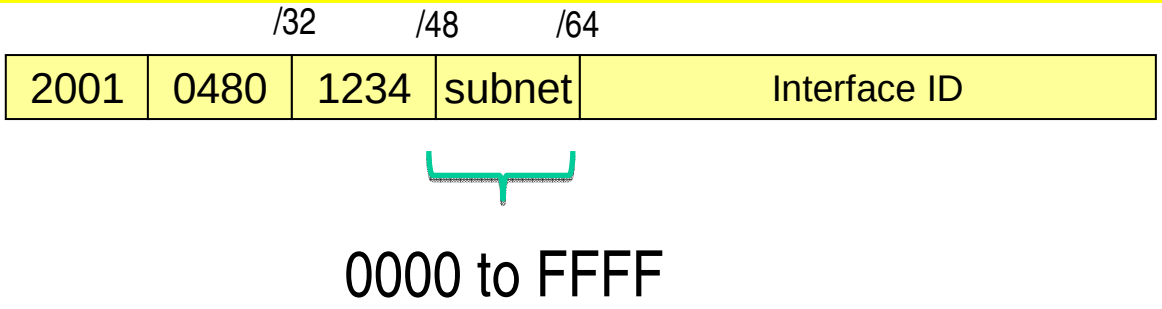

Like having a “Class A” network of incredibly huge subnets

# Your /48 prefix




- You could just assign them incrementally
  - 0, 1, 2, 3, etc
- You could have them match some part of your existing IPv4 subnet numbers
  - Like the 3rd octet of your subnets addresses, if you have a "Class B" and all your subnets are /24's
- You may want to create some hierarchy, if you have separate enclaves or security zones.

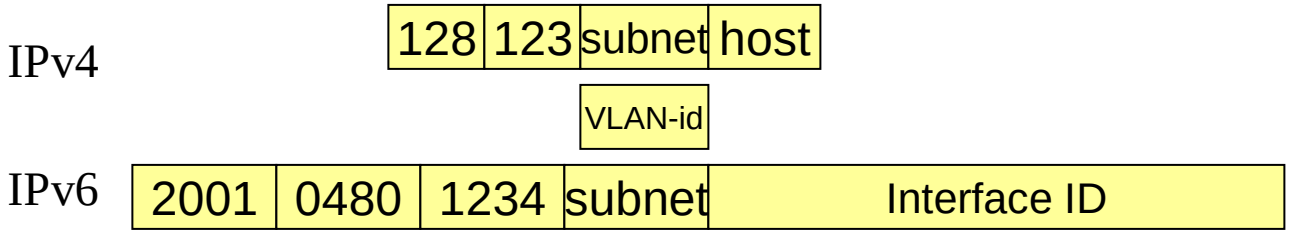
# Hierarchy Example




- Save the top 4 bits of the subnet number for "enclave"
  - That's a /52
- Subnet numbers are then 000 to FFF
  - still plenty of subnet numbers

# Example Addressing Scheme

- Address the network for consistency between protocols
  - Align VLAN number with 3rd octet of IPv4 address
  - Align IPv6 "subnet number" with the above



# IPv6 Addressing Plan

- Inventory your IPv4 subnets (network segments)
- Group by enclave or security zone
- Number your enclaves (examples)
  - 0 – "inside the firewall"
  - 1 – "outside the firewall"
  - 2 – "HPC assets behind separate firewall"
  - etc…
- Map new IPv6 subnet numbers to existing IPv4 subnet numbers

# IPv6 Addressing Plan

- Read my lips
  - forget about being conservative like in IPv4
  - subnets are /64
    - yes, even the point-to-point links
  - don't encode v4 subnet values into bottom 64 bits
  - no NAT

# IPv6 Addressing Plan Example

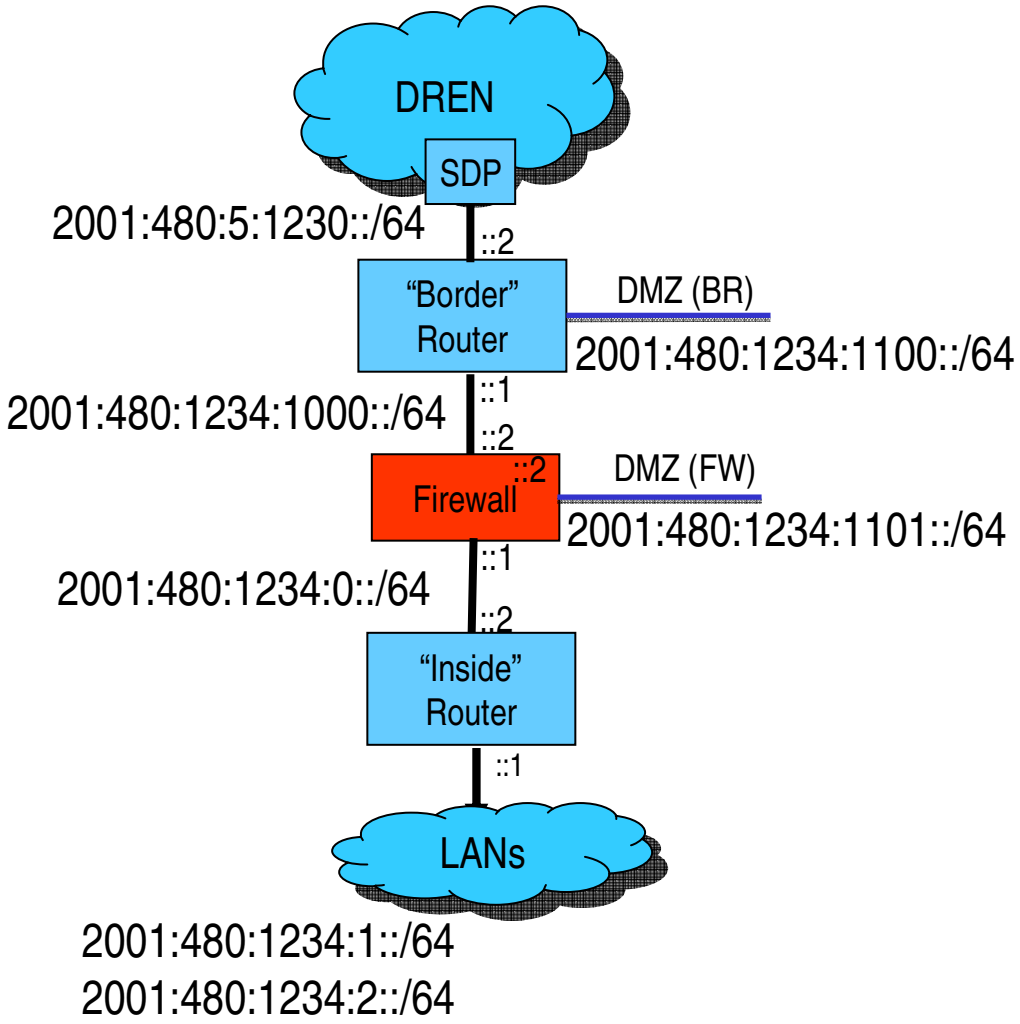
| Subnet | IPv4 | IPv6 |
|---|---|---|
| Offices | 128.123.1.0/24 | 2001:480:1234:1::/64 |
| Computer Room | 128.123.2.0/24 | 2001:480:1234:2::/64 |
| DMZ (BR) | 128.123.100.0/24 | 2001:480:1234:1100::/64 |
| DMZ (FW) | 128.123.101.0/24 | 2001:480:1234:1101::/64 |
| fw-to-br | 128.123.254.0/30 | 2001:480:1234:1000::/64 |
| fw-to-ir | 128.123.254.4/30 | 2001:480:1234:0000::/64 |
| | | |

Notes:

- Used subnet 000 for “infrastructure” links
- /52 used to designate security zone (0 – trust, 1 – untrust)
- IPv4 and IPv6 subnet numbers try to align, where possible (when IPv4 subnets are /24)
- didn’t use /126’s for the point-to-point links

# Assigning the addresses

- See earlier instructions on how to configure your interfaces with IPv6 addresses

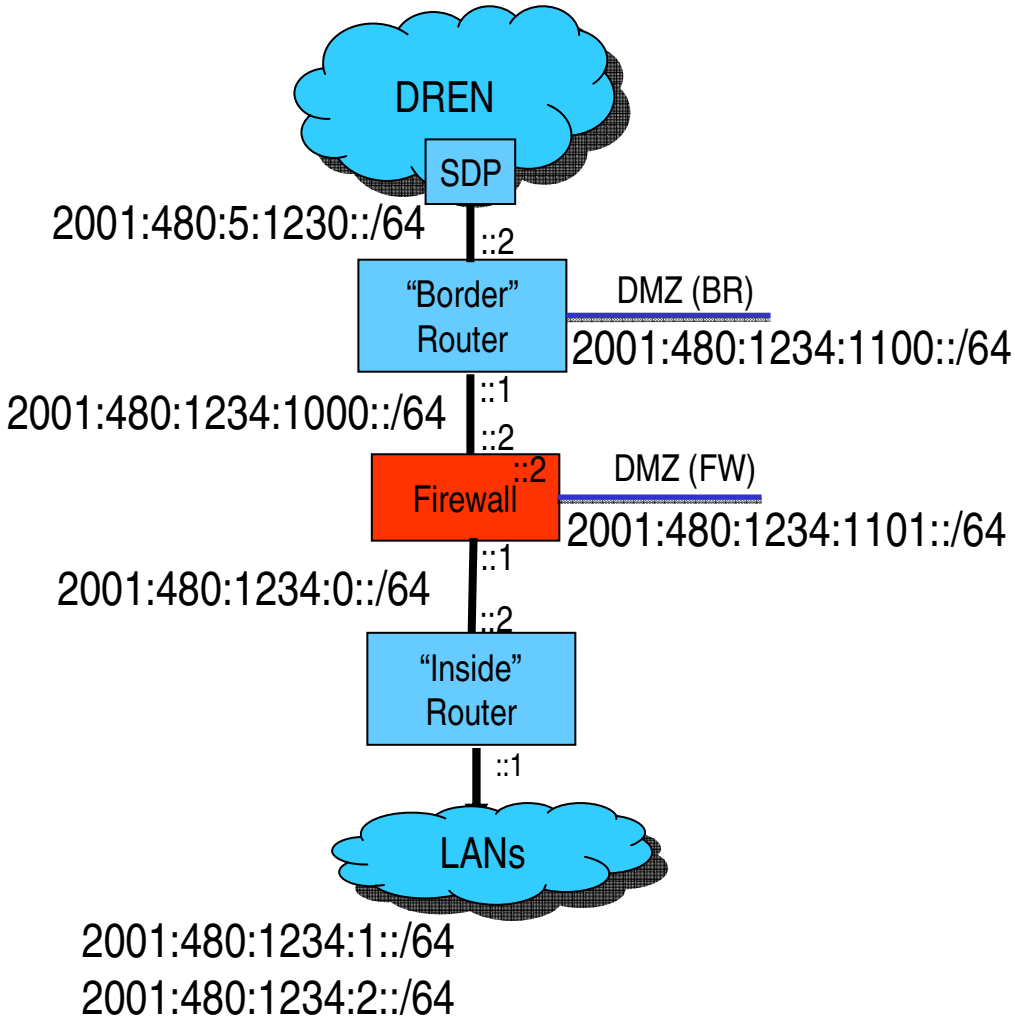


# Adding the routing

2001:480:1234::/48 -> 2001:480:1234:1000::2

::/0 -> 2001:480:1234:1000::1
2001:480:1234:0::/52 -> 2001:480:1234:0::2

::/0 -> 2001:480:1234:0::1



# Potential issues

- If your firewall doesn't support IPv6
  - upgrade to an IPv6-capable firewall
    - Still one of the best: Juniper Netscreen ISG-2000
- If you run an IDS or IPS
  - make sure you upgrade to something that supports IPv6, so that you retain an equivalent security posture
- If you don't establish contiguous connectivity (i.e. you have IPv6 "islands")
  - you will observe timeouts as systems try IPv6 but can't reach the destination
  - Example: LAN is IPv6-enabled, but no IPv6-path to the Internet

# Step 3

## IPV6 ON YOUR COMPUTERS

# How to enable IPv6

- Modern operating systems have it enabled by default:
  - Windows Vista, Windows 7, Windows Server 2008
  - Mac OS X (version 10.2 and later)
  - Modern Linux
- Solaris 8 (and later)
  - Include it when asked during installation
  - Create an empty /etc/hostname6.hme0 file (or whatever your interface name is)
  - Look at /etc/inet/ipnodes and /etc/nsswitch.conf for possible additional configuration entries

# How to Enable IPv6

- Older Linux
  - In /etc/sysconfig/network add the following:

```
NETWORKING_IPV6 = yes
```

- Windows XP
  - Use the "ipv6" command to enable and configure

```
C:\>ipv6 install
```

# Privacy addresses (bad news)

- See RFC 4941
- Windows systems do this by default
- Breaks many things in enterprise environments
  - Forensics
  - Stable DNS entries
  - Automated management tools
- Could fix with DHCPv6, but client not available in important OS's
  - Windows XP, Mac OSX
- Would be nice if RA's could say "don't do this"
- So we have to visit every Windows machine to disable this.
  - Breaks the "plug and play" goal of IPv6 for clients.

```
netsh interface ipv6 set privacy state=disabled store=persistent
netsh interface ipv6 set global randomizeidentifiers=disabled store=persistent
```

# Step 3

**IPV6 ON YOUR INFRASTRUCTURE SERVICES**
**DNS, SMTP, NTP, …**

# DNS Configuration

- Need to resolve IPv6 names and addresses
- "AAAA" resource records used for IPv6 addresses. (RFC 1886)
  - To distinguish from IPv4 addresses ("A" records)
  - Modern DNS implementations support this.

```
mush   IN   A      128.49.192.15
       IN   AAAA   2001:480:10:192:a00:20ff:fea8:539d
```

- "PTR" resource record works for IPv6...

```
$ORIGIN 0.0.8.4.0.1.0.0.2.ip6.int
d.9.3.5.8.a.e.f.f.f.0.2.0.0.a.0.2.9.1.0.0.1.0 IN PTR mush.nosc.mil.
```

# IPv6 for DNS

- 2 major considerations
  - being able to handle IPv6 "AAAA" and "PTR" resource records
  - being able to support IPv6 "transport"
    - doing your actual queries over an IPv6 connection
- Handling IPv6 AAAA and PTR records
  - Supported in "BIND" for the last 10 years. It just works.
- Transport via IPv6
  - not critical to IPv6 operation today
    - Windows XP doesn't even support it as a client
  - IPv6 transport available in BIND 9
  - Turn it on…

```
options {
        listen-on-v6 {any;};
};
```

# IPv6 for SMTP

- This example is for “sendmail”
- In /etc/mail/sendmail.mc, change the smtp port configuration to something like this…

```
DAEMON_OPTIONS(`Port=smtp, Name=MTA, Family=inet6')dnl
```

- Then “make” and restart sendmail.

# IPv6 for NTP

- IPv6 support has been in NTP (software) for a long time
- If you use host names in the config, and if those names have AAAA records, it will try IPv6 first
- It just works
- Hardware NTP appliances with IPv6 support
  - Symmetricom
  - Spectracom

# Infrastructure services, summary

- IPv6 support has been available in all these services for such a long time, that they pretty much work out of the box
- Very little configuration required
- Most of the effort will go towards getting your systems into DNS
  - not absolutely necessary to do this to make things work

# Step 4

## IPV6 ON YOUR PUBLIC FACING SERVICES WWW, MX, AUTHORITATIVE DNS

# IPv6 on your authoritative DNS servers

- See DNS discussion above
- Add "AAAA" records for your authoritative servers
- Example:

```
spawar.navy.mil.
                    IN      NS      ns1.nosc.mil.
                    IN      NS      ns2.nosc.mil.
                    IN      NS      ns3.nosc.mil.
ns1                 IN      A       198.253.48.7
                    IN      AAAA    2001:480:10:1048::7
ns2                 IN      A       198.253.16.35
                    IN      AAAA    2001:480:610:1016::35
ns3                 IN      A       198.253.252.3
                    IN      AAAA    2001:480:980:1252::3
```

# IPv6 for your Mail eXchangers

- See SMTP (sendmail) discussion above
- Add AAAA records in DNS for your MX hosts

```
spawar.navy.mil.
        300     IN      MX      20      mx1
        300     IN      MX      20      mx2
mx1             IN      A       198.253.50.6
                IN      AAAA    2001:480:10:1050::6
mx2             IN      A       198.253.50.7
                IN      AAAA    2001:480:10:1050::7
```

# IPv6 for your public web server

- Example (Apache)
  - Upgrade to Apache 2.0 if you're running an earlier version
  - Listen to all IP addresses (don't restrict to the IPv4 address)

```
Listen 80
```

- Add AAAA records in DNS for your web server

```
www             IN      A       198.253.50.5
                IN      AAAA    2001:480:10:1050::5
```

# Step 5

## OPERATIONS AND MAINTENANCE

# Debugging tools

- ping and traceroute have IPv6 equivalents
  - different for each OS
  - ping6, traceroute6
- modern sniffing tools have no problems dissecting IPv6
  - tcpdump, wireshark
- Make sure your network management tools know about IPv6 addresses (modern versions all do)
  - HP Openview, InMon, CiscoWorks, Ironview

# Tools

- Critical network tools:
  - ifconfig, iip, netstat, ping, traceroute
- Every operating system seems to use different names for each tool

| | | | | |
|---|---|---|---|---|
| Windows | ipconfig | netsh interface ipv6 | ping6 | tracert6 |
| Solaris | ifconfig –a6 | netstat –f inet6 | ping –A inet6 | traceroute –A inet6 |
| Linux | ifconfig, /sbin/ip | netstat --inet6 | ping6 | traceroute6 |
| Mac OSX | ifconfig | netstat | ping6 | traceroute6 |
| Juniper | show interface | show interface, show route, etc | ping inet6 | traceroute inet6 |
| Cisco | show ipv6 int | show ipv6 int, show ipv6 route | ping ipv6 | trace ipv6 |

# WRAP-UP

# How to get started

- Work from outside in, then bottom-up
  - WAN/ISP, border, DMZ, firewall, enclave
  - LAN interfaces, desktops, servers, apps
- Focus first on your public facing services
  - www, DNS, MX
- Establish a corporate culture to include IPv6 in all IT plans and activities
  - from CIO down to all technical staff
- Take the long view
  - get there via normal tech refresh, not forklift upgrade during crisis
- Don’t be afraid to break some glass

# References

- I try to keep various references at:

http://www.v6.dren.net

# Soapbox

- Enabling IPv6 throughout your environment needs to be a cultural thing.
  - Get everyone involved and on-board
  - Include it as part of tech refresh.
- It may seem overwhelming in the beginning, but its really not that hard to get started.
- Don't be afraid to break some glass
- Very important that we focus on making our public facing services dual-stack as soon as possible.
  - otherwise we'll be in translator-hell
  - eventually some clients won't be able to reach you
- IPv6 is an "unfunded mandate", and everyone needs to do their part.
- Need v4/v6 feature parity in products
- Avoid vendors that don't have a good IPv6 story

END

ron@spawar.navy.mil