IPv6

With thanks to Rick Grazianni of
Cabrillo College

Why IPv6?

- Larger address space
- No need for NAT
- Easier aggregation means more efficient routing
- Improved address assignment
  - StateLess Address AutoConfiguration (SLAAC)
- Improved support for mobility
- No broadcasts

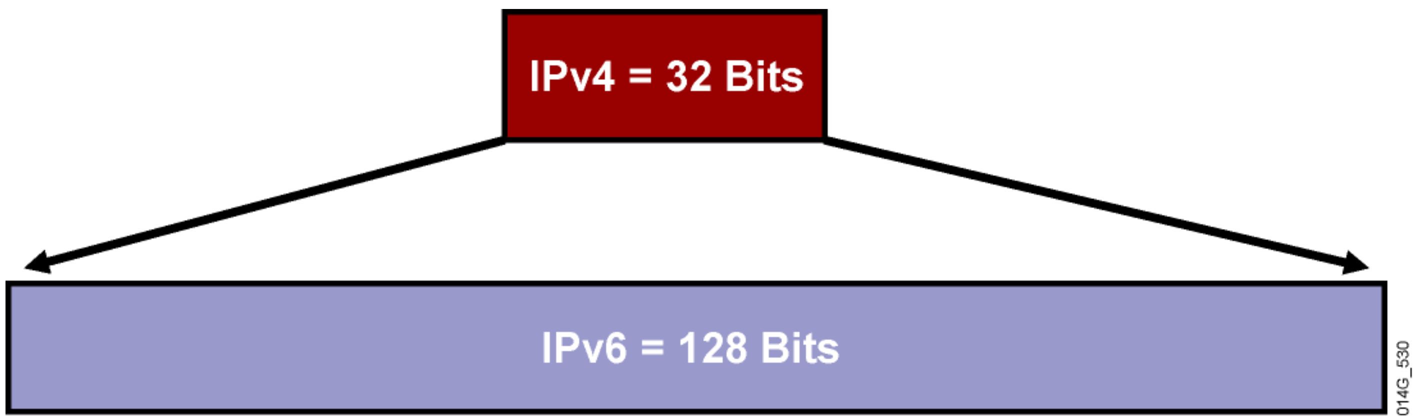
Why IPv6… reality

- Wins
  - Larger address space
  - No need for NAT
  - Improved address assignment
    - StateLess Address AutoConfiguration (SLAAC)

- Not clear (to me:)
  - Easier aggregation means more efficient routing
  - Improved support for mobility
  - No broadcasts

Larger Address Space

- IPv4 = 4,294,967,295 addresses
- IPv6 = 340,282,366,920,938,463,374,607,432,768,211,456 addresses
- 4x in number of bits translates to **huge** increase in address space!



No More IPv4 Addresses

- IPv4 address space in terms of /8's

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	[illegible]	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	[illegible]	151	152	[illegible]	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	[illegible]	199	200	[illegible]	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	[illegible]	229	230	231	[illegible]	[illegible]	[illegible]	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Class A

Class B

Class C

Class D & E

No More IPv4 Addresses

- 24 /8's on January 12, 2010

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

No More IPv4 Addresses

- 20 /8's on April 10, 2010

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

No More IPv4 Addresses

- 13 /8's on May 8, 2010

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

No More IPv4 Addresses

- 7 /8's on November 30th, 2010

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

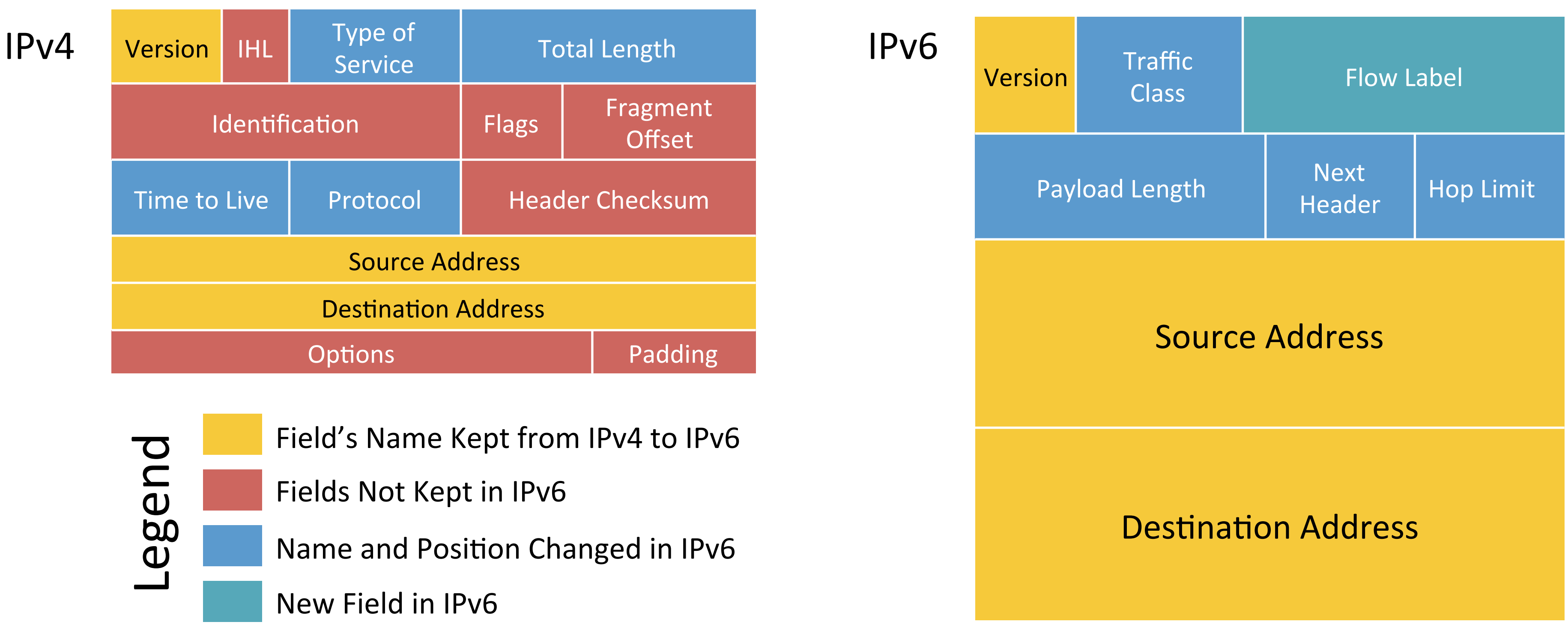
No More IPv4 Addresses

- **0 /8's** on January 31st, 2011!

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Other Significant Protocol Changes

- Increased minimum MTU from 576 to 1280
- No enroute fragmentation… fragmentation only at source
- Header changes



IPv4	IPv6
Addresses are 32 bits (4 bytes) in length.	Addresses are 128 bits (16 bytes) in length
Address (A) resource records in DNS to map host names to IPv4 addresses.	Address (AAAA) resource records in DNS to map host names to IPv6 addresses.
Pointer (PTR) resource records in the IN-ADDR.ARPA DNS domain to map IPv4 addresses to host names.	Pointer (PTR) resource records in the IP6.ARPA DNS domain to map IPv6 addresses to host names.
IPSec is optional and should be supported externally	IPSec support is not optional
Header does not identify packet flow for QoS handling by routers	Header contains Flow Label field, which Identifies packet flow for QoS handling by router.
Both routers and the sending host fragment packets.	Routers do not support packet fragmentation. Sending host fragments packets
Header includes a checksum.	Header does not include a checksum.
Header includes options.	Optional data is supported as extension headers.
ARP uses broadcast ARP request to resolve IP to MAC/Hardware address.	Multicast Neighbor Solicitation messages resolve IP addresses to MAC addresses.
Internet Group Management Protocol (IGMP) manages membership in local subnet groups.	Multicast Listener Discovery (MLD) messages manage membership in local subnet groups.
Broadcast addresses are used to send traffic to all nodes on a subnet.	IPv6 uses a link-local scope all-nodes multicast address.
Configured either manually or through DHCP.	Does not require manual configuration or DHCP.
Must support a 576-byte packet size (possibly fragmented).	Must support a 1280-byte packet size (without fragmentation).

IPv6 Addresses

IPv6 Address Notation

- RFC 5952
- 128-bit IPv6 addresses are represented in:
  - Eight 16-bit segments
  - Hexadecimal (non-case sensitive) between 0000 and FFFF
  - Separated by colons
- Example:
  - `3ffe:1944:0100:000a:0000:00bc:2500:0d0b`
- Two rules for dealing with 0's

One Hex digit = 4 bits

Dec.	Hex.	Binary	Dec.	Hex.	Binary
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	3	0011	11	B	1011
4	4	0100	12	C	1100
5	5	0101	13	D	1101
6	6	0110	14	E	1110
7	7	0111	15	F	1111

0's Rule 1 – Leading 0's

- The leading zeroes in any 16-bit segment do not have to be written.

- Example
  - 3ffe : 1944 : 0100 : 000a : 0000 : 00bc : 2500 : 0d0b
  - 3ffe : 1944 : 100 : a : 0 : bc : 2500 : d0b

3ffe:1944:100:a:0:bc:2500:d0b

0’s Rule 1 – Leading 0’s

- Can only apply to **leading zeros**… otherwise ambiguous results
- Example
  - `3ffe : 1944 :  100 :    a :    0 :   bc : 2500 :  d0b`
- Could be either
  - `3ffe : 1944 : 0100 : 000a : 0000 : 00bc : 2500 : 0d0b`
  - `3ffe : 1944 : 1000 : a000 : 0000 : bc00 : 2500 : d0b0`
  - *Which is correct?*

0's Rule 1 – Leading 0's

- Can only apply to **leading zeros**… otherwise ambiguous results
- Example
  - `3ffe : 1944 :  100 :    a :    0 :   bc : 2500 :  d0b`
- Could be either
  - **`3ffe : 1944 : 0100 : 000a : 0000 : 00bc : 2500 : 0d0b`**
  - `3ffe : 1944 : 1000 : a000 : 0000 : bc00 : 2500 : d0b0`
  - *Which is correct?*

0's Rule 2 – Double Colon

- Any **single, contiguous** string of **16-bit segments** consisting **of all zeroes** can be represented with a **double colon**.

```
ff02 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0005
ff02 :    0 :    0 :    0 :    0 :    0 :    0 :    5
ff02 :                                          :    5
```

ff02::5

0's Rule 2 – Double Colon

- Only a **single** contiguous string of all-zero segments can be represented with a double colon.
- Example:

```
2001 : 0d02 : 0000 : 0000 : 0014 : 0000 : 0000 : 0095
```

- Both of these are correct

```
2001 :  d02 ::                  14 :    0 :    0 :   95
```

OR

```
2001 :  d02 :    0 :    0 :  14 ::                   95
```

0's Rule 2 – Double Colon

- However, using double colon more than once creates ambiguity
- Example

```
2001:d02::14::95
```

2001:0d02:**0000**:**0000**:**0000**:0014:**0000**:0095
2001:0d02:**0000**:**0000**:0014:**0000**:**0000**:0095
2001:0d02:**0000**:0014:**0000**:**0000**:**0000**:0095

Network Prefixes

- In IPv4, network portion of address can by identified by either
  - **Netmask**: `255.255.255.0`
  - **Bitcount**: `/24`

- Only use bitcount with IPv6

`3ffe:1944:100:a::/64`

Special IPv6 Addresses

- Default route: `::/0`
- Unspecified Address: `::/128`
  - Used in SLAAC (coming later)
- Loopback/Local Host: `::1/128`

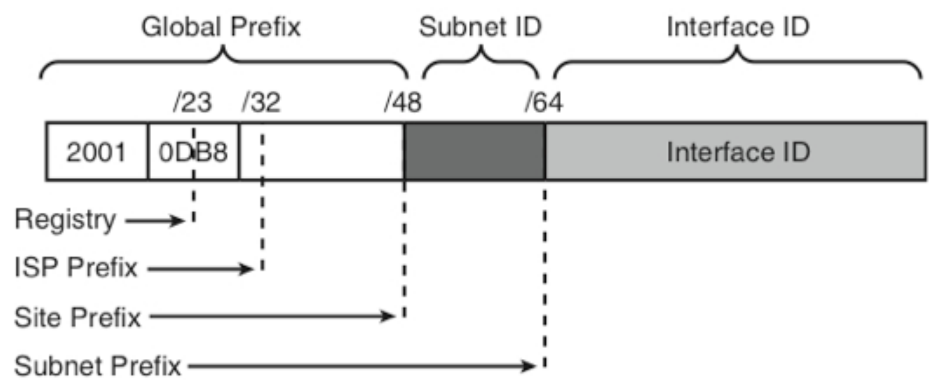
Types of IPv6 Addresses

- RFC 4291– "IPv6 Addressing Architecture"
- **Global Unicast**
  - Globally routable IPv6 addresses
- **Link Local Unicast**
  - Addresses for use on a given subnet
- **Unique Local Unicast**
  - Globally unique address for local communication
- **Multicast**
- **Anycast**
  - A unicast address assigned to interfaces belonging to different nodes

Types of IPv6 Addresses

- RFC 4291– "IPv6 Addressing Architecture"
- **Global Unicast**
  - Globally routable IPv6 addresses
- **Link Local Unicast**
  - Addresses for use on a given subnet
- **Unique Local Unicast**
  - Globally unique address for local communication
- **Multicast**
- **Anycast**
  - A unicast address assigned to interfaces belonging to different nodes

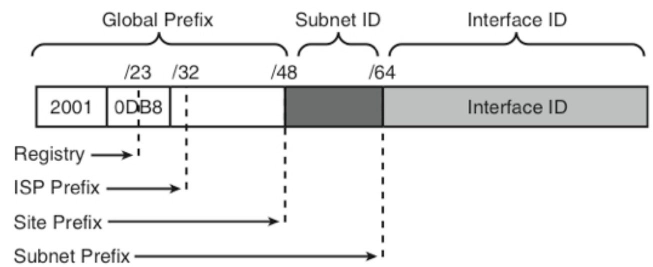
Global Unicast Addresses



- Globally routable addresses
  - RFC 3587
- 3 parts
  - 48 bit **global routing prefix**
    - Hierarchically-structured value assigned to a site
    - Further broken down into Registry, ISP Prefix, and Site Prefix fields
  - 16 bit **Subnet ID**
    - Identifier of a subnet within a site
  - 64(!) bit **Interface ID**
    - Identify an interface on a subnet
    - Motivated by expected use of MAC addresses (IEEE EUI-64 identifiers) in SLAAC…
  - Except GUAs that start with '0 0 0…' binary
    - Used for, e.g., "IPv4-Mapped IPv6 Addresses" (RFC 4308)

Global Unicast Addresses

- Current **ARIN** policy is to assign no longer than /32 to an ISP
  - **A**merican **R**egistry for **I**nternet **N**umbers
  - https://www.arin.net/policy/nrpm.html
  - UCSC allocation is `2607:F5F0::/32`



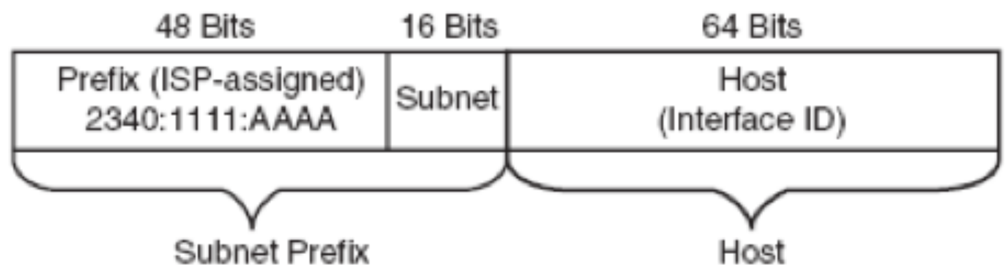
- IANA currently assigning addresses that start with '`001…`' binary
  - `2000::/3`
    - `(2000:: - 3FFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF)`
  - Supports
    - Maximum 2^{29} (536,870,912… 1/8 of an **Internet address space** of) ISPs
    - 2^{45} sites (equivalent to 8,192 **IAS**s of sites!)

- ISP can delegate a minimum of 2^{16}, or 65,535 site prefixes
  - Difference between Global Prefix (48 bits) and ISP Prefix (32 bits)

Subnetting GUAs

- Each site can identify 2^{16} (65,535) subnets

```
2340:1111:AAAA:1::/64
2340:1111:AAAA:2::/64
2340:1111:AAAA:3::/64
2340:1111:AAAA:4::/64
...
```



- Subnet has address space of 2^{64}… an IAS of IASs!

- Can extend the subnet ID into the interface ID portion of the address…
  - Sacrifice ability to use EUI-64 style of SLAAC…
  - Maybe not a bad thing… more later

These are huge numbers!!

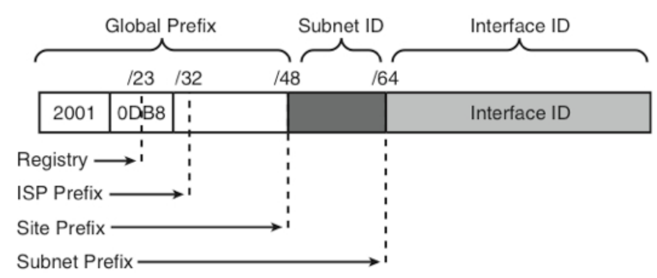
- Assume average /16's allocated to ISPs and /22's allocated to sites in IPv4

IPv6 2000::/3 block

Description	Range	Count	Scale vs IPv4
Total # ISPs	/3 – /32	2^{29} = 512M	**9,362**
Total # Sites	/3 – /48	2^{42} = 4T	**1.2M**
Sites/ISP	/48 – /64	2^{16} = 64K	**1,024**

IPv4 class A, B, and C blocks

Description	Range	Count
Total # ISPs	/16 * 7/8	57K
Total # Sites	/22 * 7/8	3.6M
Sites/ISP	/16 - /22	2^6 = 64



- ***And this keeps assumption of /64 subnets!***

IPv6 Address Space

- **Allocated**
  - 2000::/3 Global Unicast
  - FC00::/7 Unique Local Unicast
  - FE80::/10 Link Local Unicast
  - FF00::/8 Multicast
- ***Accounts for a bit more than 2^{125} of the address space.***

- **Unallocated** (“Reserved by IETF”)
  - /3’s – 4000::, 6000::, 8000::, A000::, C000::
  - /4’s – 1000::, E000::
  - /5’s – 0800::, F000::
  - /6’s – 0400::, F800::
  - /7’s – 0200::
  - /8’s – 0000::, 0100::
  - /9’s – FE00::
  - /10’s – FEC0::
- ***Accounts for a little more than 2^{127}, or more than half, of the address space!!***

http://www.iana.org/assignments/ipv6-address-space/ipv6-address-space.xml

Problem with /64 Subnets

- Scanning a subnet becomes a DoS attack!
  - Creates IPv6 version of 2^{64} ARP entries in routers
  - Exhaust address-translation table space

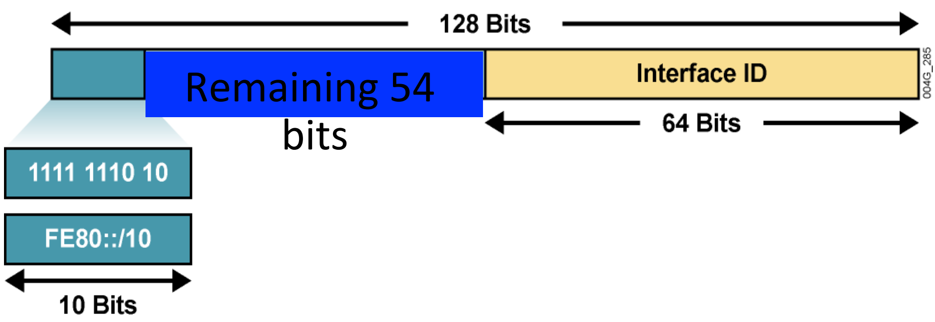
- Solutions
  - RFC 6164 recommends use of /127 to protect router-router links
  - RFC 3756 suggest "clever cache management" to address more generally

Types of IPv6 Addresses

- RFC 4291– "IPv6 Addressing Architecture"
- **Global Unicast**
  - Globally routable IPv6 addresses
- **Link Local Unicast**
  - Addresses for use on a given subnet
- **Unique Local Unicast**
  - Globally unique address for local communication
- **Multicast**
- **Anycast**
  - A unicast address assigned to interfaces belonging to different nodes

Link-Local Addresses

- '11111110 10…' binary (FE80::/10)
  - According to RFC 4291 bits 11-64 should be 0's… so really FE80::/64?
- For use on a single link.
  - Automatic address configuration
  - Neighbor discovery (IPv6 ARP)
  - When no routers are present
  - Routers must not forward



- Addresses "chicken-or-egg" problem… need an address to get an address.
- Address assignment done unilaterally by node (later)
- IPv4 has link-local address (169.254/16, RFC 3927)
  - Only used if no globally routable addresses available

Types of IPv6 Addresses

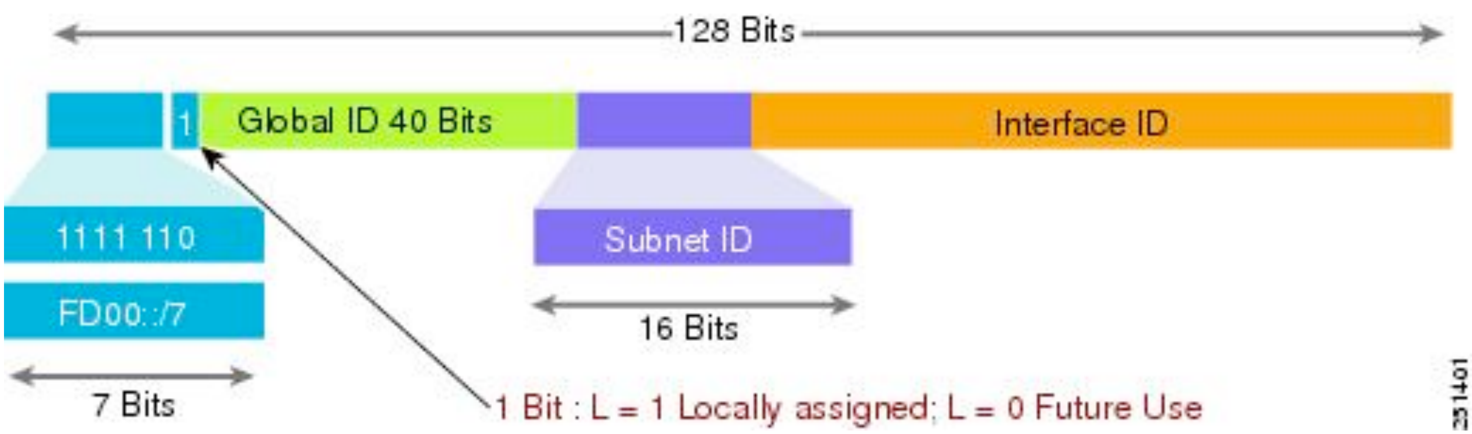
- RFC 4291– "IPv6 Addressing Architecture"
- **Global Unicast**
  - Globally routable IPv6 addresses
- **Link Local Unicast**
  - Addresses for use on a given subnet
- **Unique Local Unicast**
  - Globally unique address for local communication
- **Multicast**
- **Anycast**
  - A unicast address assigned to interfaces belonging to different nodes

Unique Local Addresses

- ‘1111110…’ binary (`FC00::/7`)
- Globally unique addresses intended for local communication
  - IPv6 equivalent of IPv4 RFC 1918 addresses
- Defined in RFC 4193
  - Replace “site local” addresses defined in RFC 1884, deprecated in RFC 3879
- Should not be installed in global DNS
  - Can be installed in “local DNS”

Unique Local Addresses

- 4 parts
  - "**L**" bit always 1
  - **Global ID** (40 bits) randomly generated to enforce the idea that these addresses are not to be globally routed or aggregated
  - **Subnet ID** (16 bits)… same as Globally Unique Subnet ID
  - **Interface ID** (64 bits)… same as Globally Unique Interface ID

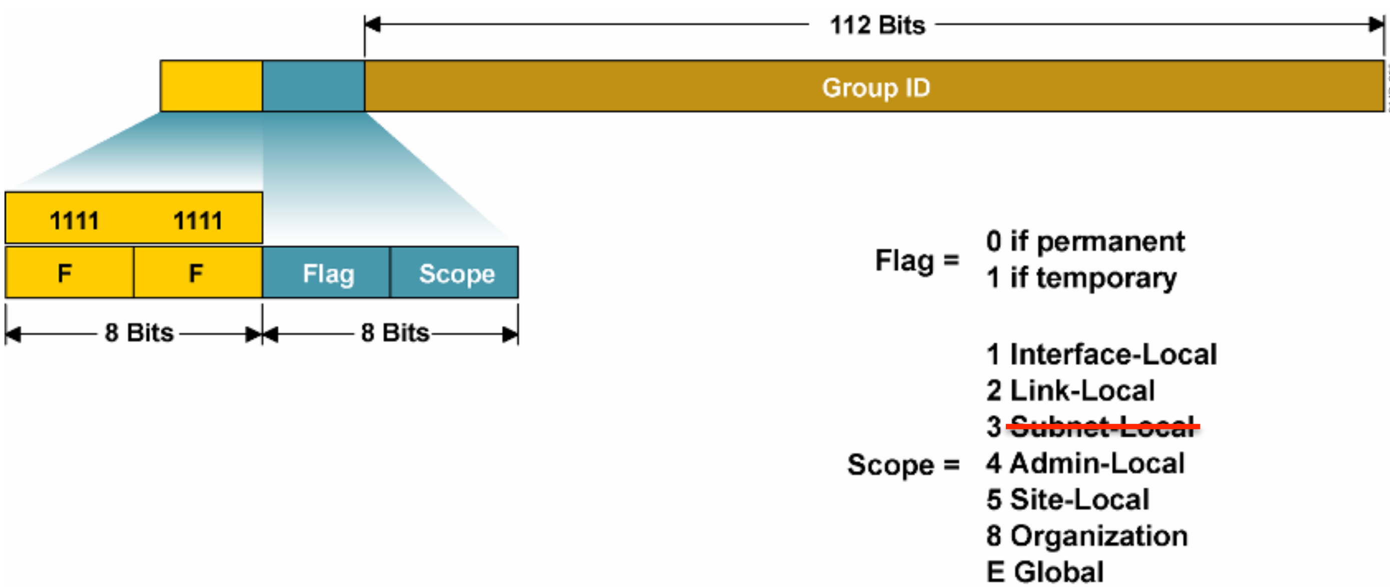


Types of IPv6 Addresses

- RFC 4291– "IPv6 Addressing Architecture"
- **Global Unicast**
  - Globally routable IPv6 addresses
- **Link Local Unicast**
  - Addresses for use on a given subnet
- **Unique Local Unicast**
  - Globally unique address for local communication
- **Multicast**
- **Anycast**
  - A unicast address assigned to interfaces belonging to different nodes

Multicast Addresses

- ‘11111111…’ binary (FF00::/8)
- Equivalent to IPv4 multicast (224.0.0.0/8)
- 3 parts
  - **Flag** (4 bits)
  - **Scope** (4 bits)



Reserved Multicast Addresses

- All nodes
  - **FF01::1** – interface-local; used for loopback multicast transmissions
  - **FF02::1** – link-local; replaces IPv4 broadcast address (all 1's host)
- All routers
  - **FF01::2** (interface-local), **FF02::2** (link-local), **FF05::2** (site-local)
- Solicited-Node multicast
  - Used in Neighbor Discovery Protocol (later)
  - **FF02::FF00:0/104** (FF02::FFXX:XXXX)
  - Construct by replacing 'XX:XXXX' above with low-order 24 bits of a nodes unicast or anycast address
  - Example
    - For unicast address 4037::01:800:200E:8C6C
    - Solicited-Node multicast is FF02::1:FF0E:8C6C

Types of IPv6 Addresses

- RFC 4291– "IPv6 Addressing Architecture"
- **Global Unicast**
  - Globally routable IPv6 addresses
- **Link Local Unicast**
  - Addresses for use on a given subnet
- **Unique Local Unicast**
  - Globally unique address for local communication
- **Multicast**
- **Anycast**
  - A unicast address assigned to interfaces belonging to different nodes

Anycast Addresses

- Allocated from unicast address space
  - Syntactically indistinguishable from unicast addresses
- An address assigned to more than one node
- Anycast traffic routed to the "nearest" host with the anycast address
- Typically used for a service (e.g. local DNS servers)
- Nodes must be configured to know an address is anycast
  - Don't do Duplicate Address Detection
  - Advertise a route?

A Node's Required Addresses

- Link-local address for each interface
- Configured unicast or anycast addresses
- Loopback address
- All-Nodes multicast interface and link addresses
- Solicited-Node multicast for each configured unicast and anycast address
- Multicast addresses for all groups the node is a member of
- Routers must add
  - Subnet-Router anycast address for each interface
    - Subnet prefix with all 0's host part
  - All-Routers multicast address

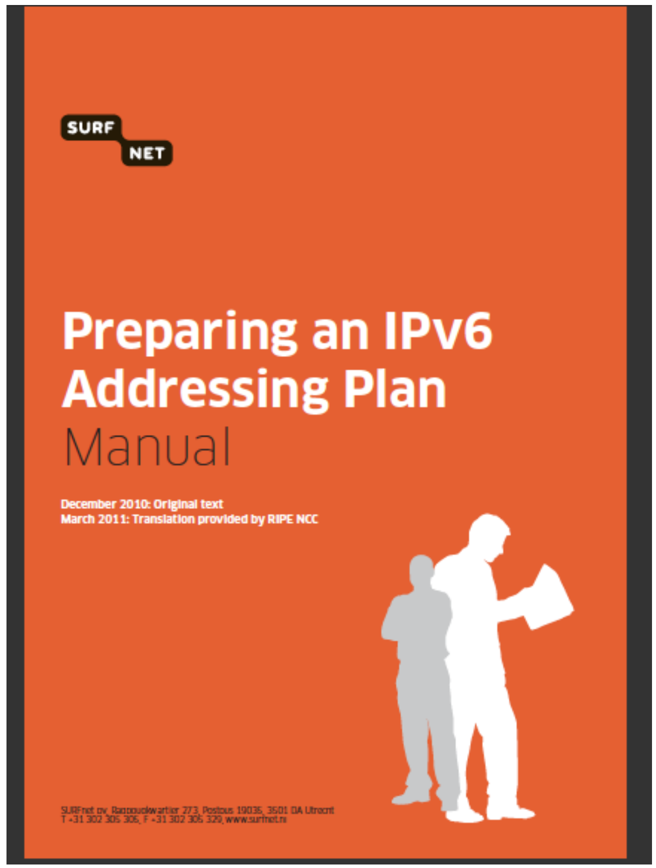
Red = new for IPv6

Question: Will ISPs allocate address blocks to (residential) customers?

Question: Does IPv6 eliminate the need for NAT?

Preparing an IPv6 Addressing Plan

- http://www.ripe.net/lir-services/training/material/IPv6-for-LIRs-Training-Course/IPv6_addr_plan4.pdf



Address Assignment

Assigning Address to Interfaces

- Static (manual) assignment
  - Needed for network equipment
- DHCPv6
  - Needed to track who uses an IP address
- **S**tate**L**ess **A**ddress **A**uto**C**onfiguration (**SLAAC**)
  - New to IPv6
- Describe SLAAC in the following…

SLAAC

- RFC 4862 – IPv6 Stateful Address Autoconfiguration
- Used to assign unicast addresses to interfaces
  - Link-Local Unicast
  - Global Unicast
  - Unique-Local Unicast?
- Goal is to minimize manual configuration
  - No manual configuration of hosts
  - Limited router configuration
  - No additional servers
- Use when “not particularly concerned with the exact addresses hosts use”
  - Otherwise use DHCPv6 (RFC 3315)

SLAAC Building Blocks

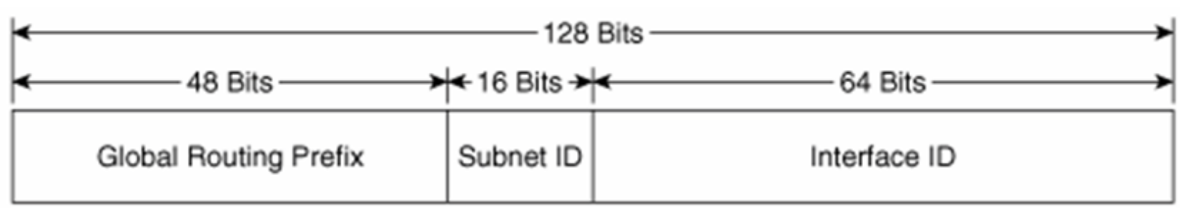
- Interface IDs
- Neighbor Discovery Protocol
- SLAAC Process

SLAAC Building Blocks

- Interface IDs
- Neighbor Discovery Protocol
- SLAAC Process

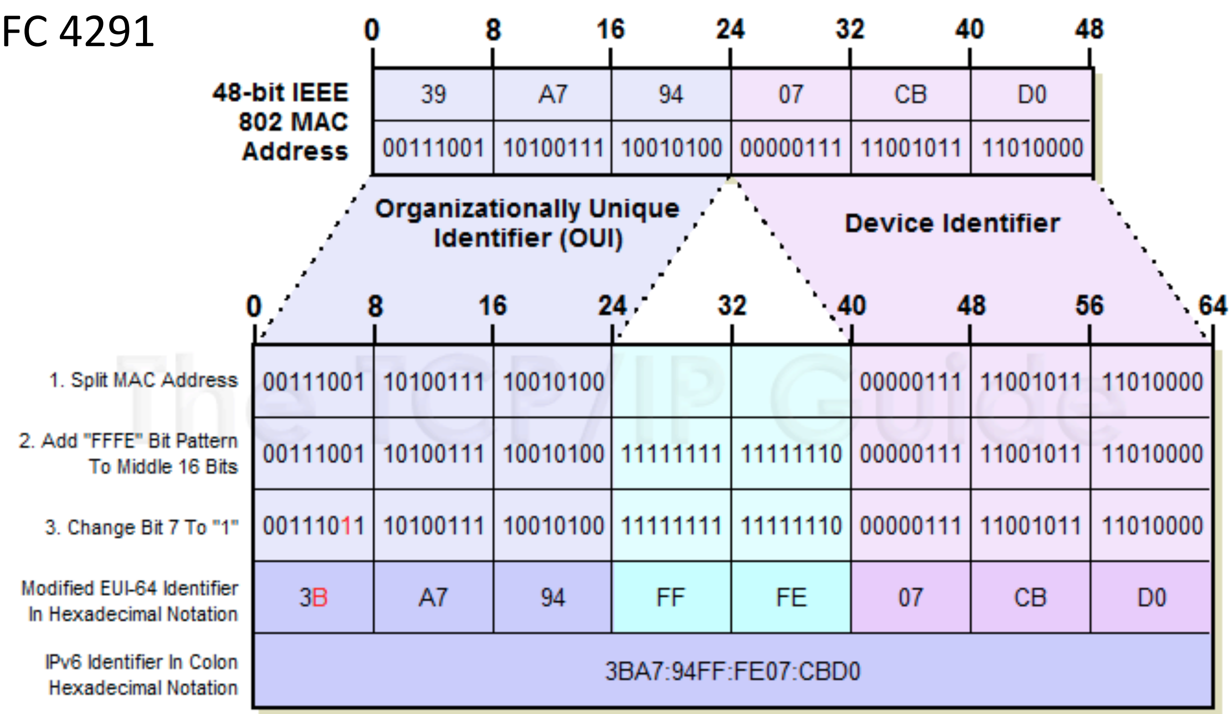
Interface IDs

- Used to identify a unique interface on a link
- Thought of as the "host portion" of an IPv6 address.
- 64 bits: To support both 48 bit and 64 bit IEEE MAC addresses
- Required to be unique on a link
- Subnets using auto addressing must be /64s.
- EUI-64 vs Privacy interface IDs



IEEE EUI-64 Option for Interface ID

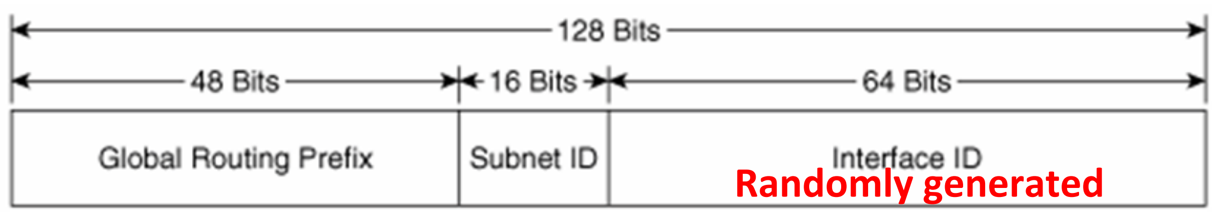
- Use interface MAC address
- Insert FFFE to convert EUI-48 to EUI-64
- FlipUniversal/Local bit to “1”
  - Section 2.5.1 RFC 4291



64-Bit IPv6 Modified EUI-64 Interface Identifier

Privacy Option for Interface ID

- Using MAC uniquely identifies a host… security/privacy concerns!
- Microsoft(!) defined an alternative solution for Interface IDs (RFC 4941)
- Hosts generates a random 64 bit Interface ID



SLAAC Building Blocks

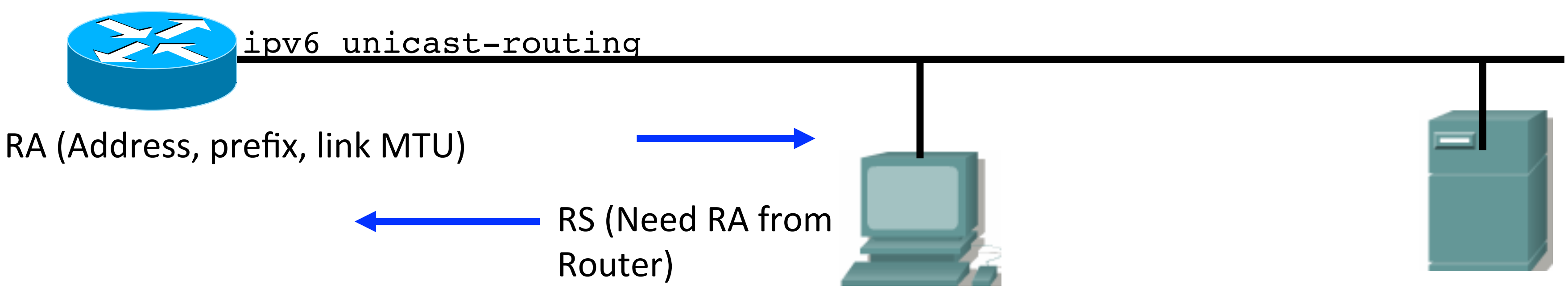
- Interface IDs
- Neighbor Discovery Protocol
- SLAAC Process

NDP

- RFC 4861 – Neighbor Discovery for IPv6
- Used to
  - Determine MAC address for nodes on same subnet
  - Find routers on same subnet
  - Determine subnet prefix and MTU
  - Determine address of local DNS server (RFC 6106)
- Uses 5 ICMPv6 messages
  - **Router Solicitation** (**RS**) – request routers to send RA
  - **Router Advertisement** (**RA**) – router's address and subnet parameters
  - **Neighbor Solicitation** (**NS**) – request neighbor's MAC address (ARP Request)
  - **Neighbor Advertisement** (**NA**) – MAC address for an IPv6 address (ARP Reply)
  - **Redirect** – inform host of a better next hop for a destination

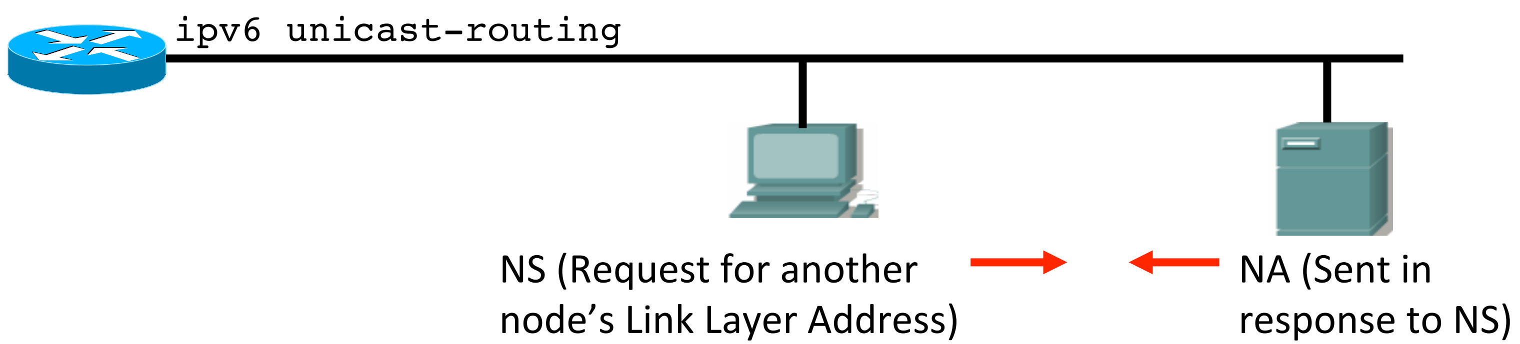
NDP RS & RA

- **Router Solicitation** (**RS**)
  - Originated by hosts to request that a router send an RA
  - Source = unspecified (::) or link-local address,
  - Destination = All-routers multicast (FF02::2)
- **Router Advertisement** (**RA**)
  - Originated by routers to advertise their address and link-specific parameters
  - Sent periodically and in response to Router Solicitation messages
  - Source = link-local address,
  - Destination = All-nodes multicast (FF02::1)



NDP NS & NA

- **Neighbor Solicitation** (**NS**)
  - Request target MAC address while providing ***target*** of source (IPv4 ARP Request)
  - Used to resolve address or verify reachability of neighbor
  - Source = unicast or "::" (DAD)
  - Destination = ***target*** address or solicited-node multicast (`FF02::1:FF:0/104` with last 24 bits of target)
- **Neighbor Advertisement** (**NA**)
  - Advertise MAC address for given IPv6 address (IPv4 Reply)
  - Respond to NS or communicate MAC address change
  - Source = unicast, destination = NS's source or all-nodes multicast (if source "::")



Duplicate Address Detection

- **Duplicate Address Detection** (**DAD**) used to verify address is unique in subnet prior to assigning it to an interface
- **MUST** take place on all unicast addresses, regardless of whether they are obtained through stateful, stateless or manual configuration
- **MUST NOT** be performed on anycast addresses
- Uses Neighbor Solicitation and Neighbor Advertisement messages
- NS sent to solicited-node multicast; if no NA received address is unique

Duplicate Address Detection

My Global Address is

2340:1111:AAAA:1:213:19FF:FE7B:5004

“Tentative”: Need to do Duplicate Address Detection

NS (Neighbor Solicitation)

- **Target Address** = 2340:1111:AAAA:1:213:19FF:FE7B:5004

Destination: Solicited-Node Multicast Address =
FF02::1::FF7B:5004

SLAAC Building Blocks

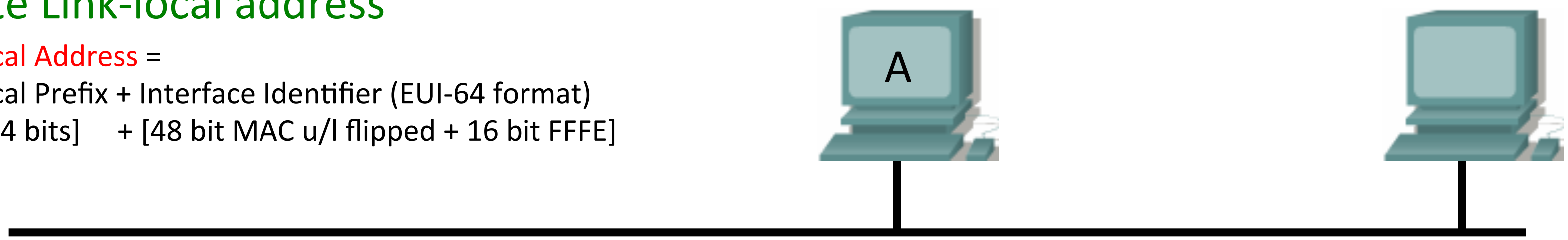
- Interface IDs
- Neighbor Discovery Protocol
- SLAAC Process

SLAAC Steps

- Select link-local address
- Verify “tentative” address not in use by another host with DAD
- Send RS to solicit RAs from routers
- Receive RA with
  - router address,
  - subnet MTU,
  - subnet prefix,
  - local DNS server (RFC 6106)
- Generate global unicast address
- Verify address is not in use by another host with DAD

Create Link-local address

Link-local Address =
Link-local Prefix + Interface Identifier (EUI-64 format)
FE80 [64 bits]   + [48 bit MAC u/l flipped + 16 bit FFFE]



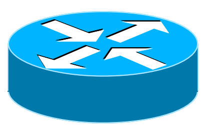
Make sure Link-local address is unique

NS (Neighbor Solicitation)

Make sure Link-local address is unique
DAD: Okay if no NA returned

Destination: Solicited-Node Multicast Address
Target address = Link-local address

Get Network Prefix to create Global unicast address



RS (Router Solicitation)

Get Prefix and other information

RA (Router Advertisement)

Source = Link-local address
Destin = FF02::1 All nodes multicast address
Query   = Prefix, Default Router, MTU, options

IPv6 Address =
Prefix + Interface ID (EUI-64 format)
[64 bits] + [48 bit MAC u/l flipped + 16 bit FFFE]

DAD

NS (Neighbor Solicitation)

Make sure IPv6 Address is unique
Target Address = IPv6 Address
DAD: Okay if no NA returned

Prefix Leases

- Prefix information contained in RA includes lifetime information
  - **Preferred lifetime**: when an address's preferred lifetime expires SHOULD only be used for existing communications
  - **Valid lifetime**: when an address's valid lifetime expires it MUST NOT be used as a source address or accepted as a destination address.

- Unsolicited RAs can reduce prefix lifetime values
  - Can be used to force re-addressing

Question: Is SLAAC really an advantage over DHCPv6?

The End