



# Plan de direccionamiento IPv6

Alejandro Acosta

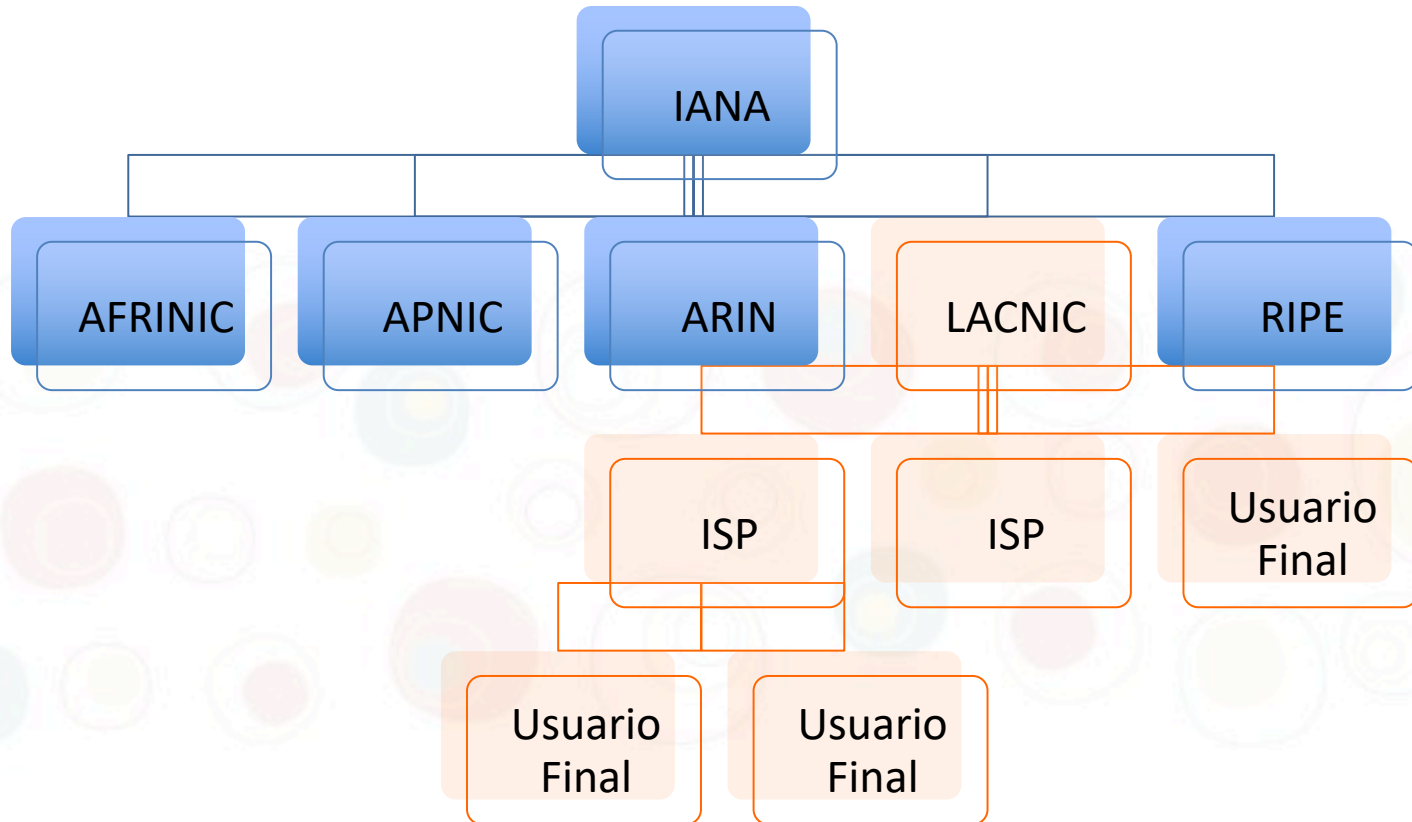
alejandro @ lacnic.net

@ITandNetworking

# ¿Qué es un plan de direccionamiento IP(v4|v6)?

*Se define como el modo, las acciones, el modelo sistemático para llevar a cabo las asignaciones de direcciones IP en una red*

# Recordando...



- Modelo jerárquico de asignación (top-down, árbol invertido)
  - IANA ->RIRs
  - RIRs->NIRs(en algunos países)
  - RIRs/NIRs->ISPs y Usuarios Finales

# Políticas de Asignación

- El espacio mínimo que un RIR recibe de IANA es un /12
- Para operación estimada de al menos 18 meses
- RIR → Sus propias políticas y estrategias de asignación
- RIR es elegible para recibir más espacio IPv6 si tiene menos del 50% de un /12 disponible o menos de 9 meses de operación
- El RIR debe realizar aplicación con justificativos necesarios
- El RIR debe actualizar su sitio web y realizar anuncio de espacio recibido

# ¿Primer paso? IPv6 Address Plan

# Al momento de implementar – Caso Universidad

Ask the upstream providers to allow the prefix & ASN. Please also ask for IPv4

Create an IPv6 Address Plan for the University

Announce the IPv6 prefix to Internet

Configure IPv6 in your network edge

Configure IPv6 in distribution layer

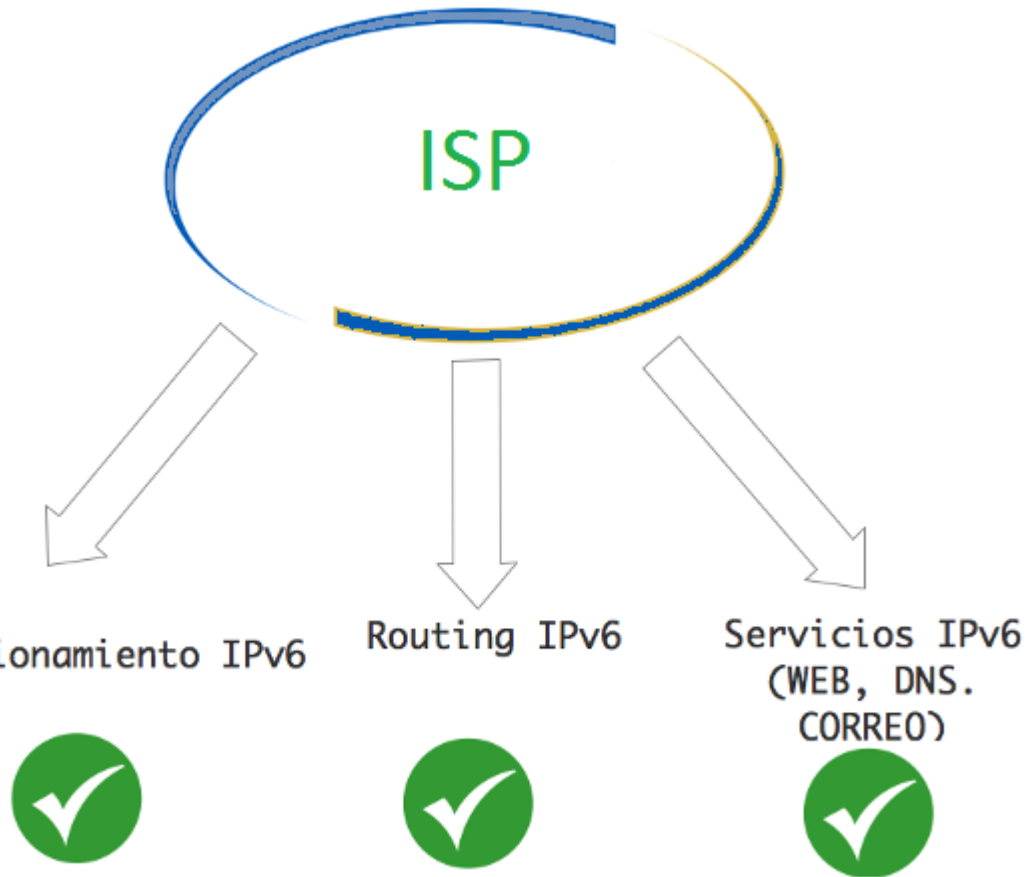
Configure IPv6 in your access layer

Timeline

*¡De esto hablaremos hoy!*

# Al momento de implementar – Caso ISP

**Plan**



¡De esto hablaremos hoy!

# ¿Por qué un plan de direccionamiento? (1/2)

- Eficiencia en la red (tablas de rutas más pequeñas)
- Orden
- Políticas de asignación más fáciles de implementar
- Mantener documentación



# ¿Por qué un plan de direccionamiento? (2/2)

- Troubleshooting
- Facilidad en asignaciones futuras/escalamiento
- Apoyar el crecimiento de la red -ordenado-
- Gerencia de la red mas sencillo

# Como debe ser un Plan de direccionamiento IPv6

- Escalable
- Mejores practicas
  - \* Separar Infraestructura de clientes
- Flexible
- Simple

i Entremos  
en materia !

# ¿Como cambió de IPv4 a IPv6?

- | IPv4:
- | `aaa.bbb.ccc.ddd`
- | 8 bits 8 bits 8 bits 8 bits = 32 bits

# ¿Como cambió de IPv4 a IPv6?

· IPv4:

· `aaa.bbb.ccc.ddd`

· 8 bits 8 bits 8 bits 8 bits = 32 bits



No hay mucho margen  
para hacer cosas ☹️

# ¿Como cambió de IPv4 a IPv6?

· IPv4:

· `aaa.bbb.ccc.ddd`

· 8 bits 8 bits 8 bits 8 bits = 32 bits



No hay mucho margen para hacer cosas ☹️



uffff, menos mal que cambió

# ¿Como cambió de IPv4 a IPv6?

| IPv4:

| `aaa.bbb.ccc.ddd`

| 8 bits 8 bits 8 bits 8 bits = 32 bits



No hay mucho margen para hacer cosas ☹️



uffff, menos mal que cambió

| IPv6:

| `2001:db8:aaaa:bbbb:0:0:0:0:/32`

| 16 bits 16 bits 16 bits + 16 bits+16+16+16 = 128

# ¿Como cambió de IPv4 a IPv6?

- De:
- aaa.bbb.ccc.ddd
- 8 bits 8 bits 8 bits 8 bits = 32 bits

Recordemos que por defecto un ISP recibe un /32. Es decir, cualquier ISP ya tiene al menos el tamaño del Internet actual

2001:db8:aaaa:bbbb:0:0:0:0:/32

16 bits 16 bits 16 bits + 16 bits + 16 + 16 + 16 = 128



# Bordes de nibble

## - Manipular los bloques por nibbles

[\_\_ NET ID \_\_] [Subnet] [Division] [\_Interface ID \_\_\_\_\_]

2001:0db8: 0abc: Ofad: aba:1000:0000:0043

[C1] [C2] [C3] [C4] [C5] [C6] [C7] [C8]

# Bordes de nibble

- Manipular los bloques por nibbles

[\_\_ NET ID \_\_] [\_\_ Net] [Division] [Interface ID]  
2001:0db8:0abc:0fad:aba:1:0000:0043  
[C1] [C2] [C3] [C4] [C5] [C6] [C7] [C8]

*¿Qué es un nibble? [1]  
Nibble, cuado o cuarteto es  
el conjunto de cuatro  
dígitos binarios (bits) o  
medio octeto.*

[1] Tomado de: <https://es.wikipedia.org/wiki/Nibble>

# Bordes de nibble

## - Manipular los bloques por nibbles

[\_\_ NET ID \_\_] [Subnet] [Division] [\_Interface ID \_\_\_\_\_]

2001:0db8: 0abc: Ofad: aba:1000:0000:0043

[C1] [C2] [C3] [C4] [C5] [C6] [C7] [C8]

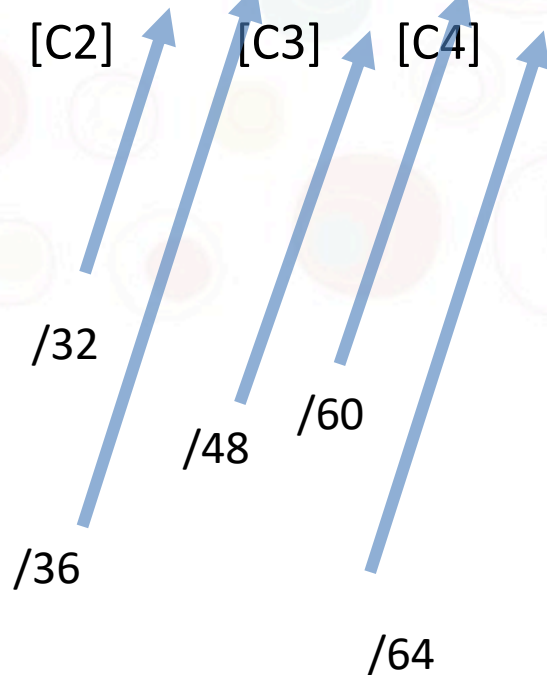
# Bordes de nibble

- Manipular los bloques por nibbles

[\_\_ NET ID \_\_] [Subnet] [Division] [\_Interface ID \_\_\_\_\_]

2001:0db8: 0abc: Ofad: aba:1000:0000:0043

[C1] [C2] [C3] [C4] [C5] [C6] [C7] [C8]



# Modelo general de asignación

## Consideraciones

- Espacio IPv6-> enorme!!
- No asignar bloques y direcciones de manera consecutiva [1]
- Al diseñar Plan de Direccionamiento el objetivo es realizarlo de manera ordenada y siguiendo mejores prácticas.
  - Ej.: /64 o /128 para loopbacks, /64 para LAN, /64 para WAN, /48 para POPs, etc.

[1] Similar al algoritmo Sparse Allocation <https://www.ripe.net/publications/docs/ripe-343#3>

# Consideraciones de Diseño para el Plan de Direccionamiento

- Debe seguir un esquema jerárquico (permite agregación, reduce tablas de ruteo, reduce procesamiento de rutas, incrementa escalabilidad de la red)
- Agregación vs conservación. En IPv6 la agregación es prioridad!
- Usar fronteras “nibble” (para simplificar notación)
- En IPv4 asignamos IPs a los usuarios. En IPv6 asignamos subredes.

# Tamaños de Prefijos Subredes de Acceso

- En IPv6, en gral. todas las subredes tienen longitud de 64 bits (= /64)
- Usar /64 para subredes es un requerimiento de algunos protocolos (ND, SEND, extensiones de privacidad, etc.)
- Igual pueden usarse prefijos más largos para ciertas subredes

# Tamaños de Prefijos

## Enlaces Punto a Punto (Cont.)

- Prefijos de /64, 126 o 127 bits de longitud
  - Análogo a prefijo /30 de IPv4
  - Ventaja:
    - No presenta problemas de seguridad con ND
  - Desventaja:
    - No es simple de usar
    - Si hay que añadir nodos, hay que reenumerar



# Tamaños de Prefijos Loopbacks

- En IPv4 se recomienda /32
- Analógicamente: en IPv6 se recomienda /128
- Se recomienda agrupar todas las loopbacks bajo un mismo /48

# Tamaños de Prefijos

## Usuarios Corporativos (Sitios Finales)

- En el pasado: recomendación de asignar /48, /64
- Sustituida por RFC 6177: Los sitios finales deben recibir una asignación correspondiente a su tamaño y necesidad
- Debe ser sencillo obtener espacio para múltiples subredes
- Que el sitio pueda crecer y evitar problemas de escasez
- Prefijos demasiado pequeños: incrementará costos en el futuro (administración, reenumeración, etc.)
- Considerar operación de DNS reverso y uso de fronteras “nibble”

# Un pequeño consejo (opcional)

Tomar el primer prefijo del cliente/end-user para la WAN. Ejemplo:

- Al cliente se le entrega la 2001:db8:ac::/48
  - El direccionamiento WAN entre el cliente y el ISP podría ser:
    - 2001:db8:ac:0000::/64

# Tamaños de Prefijos

## Usuarios Residenciales (Sitios Finales)

- Redes más pequeñas y sencillas
- Hoy por hoy puede ser suficiente un /56 a /64
- Pero es muy probable que a futuro no sea suficiente y haya que reenumerar
- Asignaciones más comunes:
  - /60 (16 subredes posibles)
  - /56 (256 subredes posibles)
  - /48 (65536 subredes posibles)
- Opción intermedia: asignar /56, pero reservar todo el /48 para posible crecimiento

# Cosas simpáticas que se pueden hacer 1/3:

- 2a03:2880:11:1f04:face:b00c::1
- 2001:db8:0:0:0::1:dead
- 2001:db8::f00d
- 2001:db8::feed:f00d
- 2001:db8::bad::
- 2001:db8::bad:f00d
- 2001:db8::bad:beef
- 2001:db8::f00d:cafe
- 2001:db8::bebe:cafe

# Cosas simpáticas que se pueden hacer 2/3:

Si tu vlan (VID) es 50:

- 2001:db8:**50**::1

- Si tu país es Chile:

- 2001:db8:**56**::1

- Si tu país es Chile y la VLAN es 50

- 2001:db8:**56:50**::1

# Cosas simpáticas que se pueden hacer 3/3:

Red de servidores Web

- 2001:db8:**80**::1

- Red de servidores DNS:

- 2001:db8:**53**::1

- Red de servidores SSH:

- 2001:db8:**22**::1

# Ejemplo de un caso típico

## Asignaremos:

/48 para POPs, /128 para Loopbacks, /64 para LANs y VLANs  
/64 para WANs.

## Procedimiento:

En este caso lo que haremos es jugar con el tercer campo de ceros (Subnet).

Allí tenemos específicamente 16 bits = 65535 subnets

```
[__ NET ID __] [Subnet] [Division] [Interface ID _____]  
2001:0db8: 0000: 0000: aba:1000:0000:0043  
[C1] [C2] [C3] [C4] [C5] [C6] [C7] [C8]
```



IMPORTANTE:

En las siguientes láminas solo  
calcularemos unas cuentas  
/48..., el resto es tarea para la  
casa 😊

# Pasos para realizar los calculos (1/4)

[_ NET ID _]	[Subnet]	[Division]	[_Interface ID ____]			
2001:0db8:	0000:	0000:				
[C1] [C2]	[C3]	[C4]	[C5]	[C6]	[C7]	[C8]

# Pasos para realizar los calculos (2/4)

[\_ NET ID \_] [Subnet] [Division] [\_Interface ID \_\_\_\_]  
2001:0db8: 0000: 0000:  
[C1] [C2] [C3] [C4] [C5] [C6] [C7] [C8]

Dentro de "Subnet"

[\_ NET ID \_] [Subnet]  
2001:0db8: 0000:  
[C1] [C2] 0000 0000 0000 0000 — BINARIO

# Pasos para realizar los calculos (3/4)

Dentro de "Subnet"

```
[_ NET ID _ ] [Subnet]
2001:0db8: 0000:
[C1] [C2] 0000 0000 0000 0000 ← BINARIO
```

Dividimos dentro del mundo binario

```
[_ NET ID _ ] [Subnet]
2001:0db8: 0000:
[C1] [C2] 0000 0000 0000 0000 ----> Red 1
[C1] [C2] 0000 0000 0000 0001 ----> Red 2
[C1] [C2] 0000 0000 0000 0010 ----> Red 3
[C1] [C2] 1010 1100 0001 1110 ----> Red 4
```

# Pasos para realizar los calculos (4/4)

Dividimos dentro del mundo binario

[\_\_ NET ID \_\_] [Subnet]

2001:0db8: 0000:

[C1] [C2] 0000 0000 0000 0000 ----> Red 1

[C1] [C2] 0000 0000 0000 0001 ----> Red 2

[C1] [C2] 0000 0000 0000 0010 ----> Red 3

[C1] [C2] 1010 1100 0001 1110 ----> Red 4

Resultado:

2001:0db8: 0000::/48

2001:0db8: 0001::/48

2001:0db8: 0002::/48

2001:0db8: AC1E::/48

# Preguntas & Comentarios?

Alejandro Acosta  
alejandro @ lacnic.net  
@ITandNetworking

