PENTESTING IPV6 NETWORKS

WHY IPV6 PENTESTING?

- IPv6 infrastructure adds complexity to a pentest if you don't understand the protocol.
- IPv6 provides interesting opportunities if you understand the protocol.(evading defenses, exploiting flaws)
- Many organizations assume that they don't have IPV6 deployed when infact IPV6 is enabled by default.
- Many organizations that deployed/acknowledge IPv6 have poor IPV6 security measures.

WHAT ARE WE COVERING?

- Absolute basics of IPv6
- IPv6 attack surface
- IPv6 tools
- IPv6 pentesting
- Building an IPv6 lab

IPV6 ATTACK SURFACE

- Network recon
- Local network attacks
  - Neighbour Discovery attacks.
  - Router related attacks
  - MLD attacks
- Extension header attacks
- Fragmentation attacks
- Evading defense mechanisms
- Building covert channels

IPV6 TOOLS

- The Hacker Choice's IPv6 Attack Toolkit (aka thc-ipv6)
- The SI6 Networks' IPv6 toolkit
- Chiron - an all-in-one IPv6 penetration testing framework
- Scapy - powerful packet crafting framework
- Nmap, Metasploit, Wireshark, Ping6, traceroute6.

IPV4 IS UNSUSTAINABLE

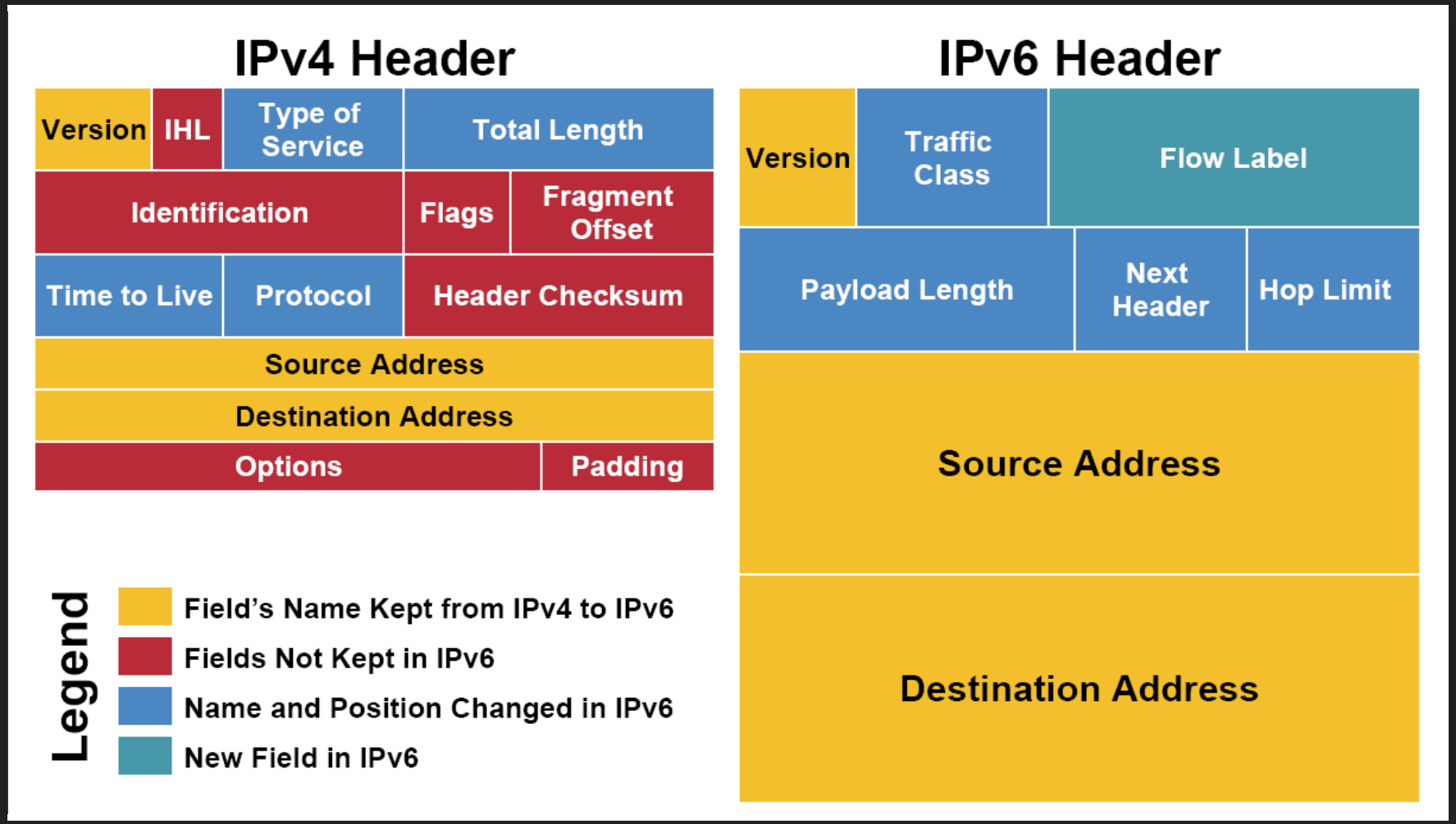
IPV4 IS OLD

IPV4 IS INEFFICIENT

IPV6 IS HERE

WHAT CHANGED IN IPV6?

- More efficient address space allocation
- End-to-end addressing; no NAT anymore!
- Fragmentation only by the source host
- Routers do not calculate header checksum (speedup!)
- Multicasting instead of broadcasting
- Built-in security mechanisms
- Single control protocol (ICMPv6)
- Auto-configuration
- Modular headers structure
- Fixed header length


IPv4 Header
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options
Padding
IPv6 Header
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destination Address
Legend
Field's Name Kept from IPv4 to IPv6
Fields Not Kept in IPv6
Name and Position Changed in IPv6
New Field in IPv6

IPV6 ADDRESS

2001:0DB8:0000:0000:0008:8000:0000:417A

Leading 0s are supressed -> 2001:DB8:0:0:8:8000:0:417A

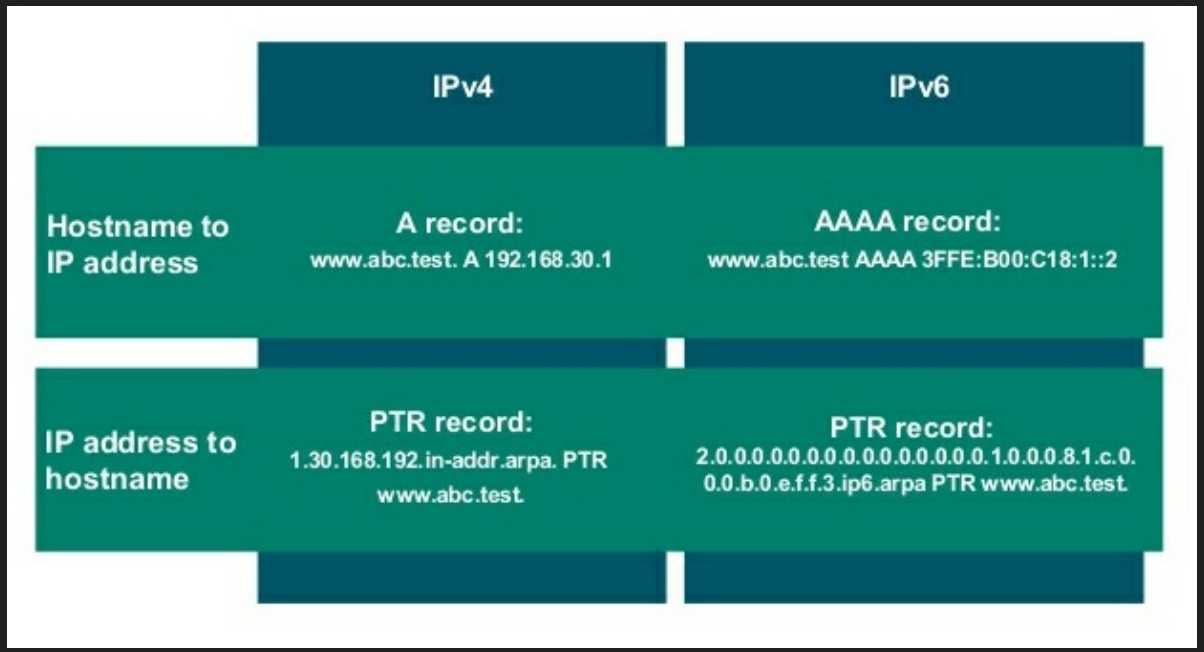
All zero blocks are ommitted, but can be applied only once
2001:DB8::8:8000:0:417A

[RFC 4291]

- 2001:0DB8:0000:0000:0008:8000:0000:417A
- 2001:DB8:0:0:8:8000:0:417A
- 2001:DB8::8:8000:0:417A
- 2001:DB8:0:0:8:8000::417A
- 2001:db8::8:8000:417A

All of them are valid ways of writing the same IP address!

IPV6 DNS


IPv4
IPv6
Hostname to IP address
A record:
www.abc.test. A 192.168.30.1
AAAA record:
www.abc.test AAAA 3FFE:B00:C18:1::2
IP address to hostname
PTR record:
1.30.168.192.in-addr.arpa. PTR www.abc.test.
PTR record:
2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.8.1.c.0.0.0.b.0.e.f.f.3.ip6.arpa PTR www.abc.test.

IPV6 ADDRESS TYPES

- Unicast
  - Global
  - Link local
- Anycast
- Multicast

There are no broadcast addresses in IPv6, special multicast addresses are used instead.

IPV6 ADDRESS CLASSIFICATION

- Address prefix matters in IPv6. Addresses are classified based on the prefix.
- Addresses that start with fe80 are link-local unicast addresses(fe80::/10)
- Addresses that start with ff00 are multicast addresses(ff00::/8)

IPV6 SPECIAL ADDRESSES

Prefix	Purpose
::/128	Unspecified
::1/128	Loopback
2001:db8	Documentation
fe80::/10	Linklocal Unicast
ff00::/8	Multicast

http://www.iana.org/assignments/iana-ipv6-special-registry/iana-ipv6-special-registry.xhtml

IPV6 SPECIAL MULTICAST ADDRESSES

Address	Scope	Use
ff02::1	Link	All nodes
ff02::2	Link	All routers
ff02::5	Link	OSPF routers
ff02::a	Link	EIGRP routers

https://en.wikipedia.org/wiki/Multicast_address#IPv6

WORKING WITH IPV6 ADDRESSES

- *addr6* tool from SI6's IPv6 toolkit comes handy while dealing with IPv6 addresses

```
# Understanding an address
verax@null ~ $ addr6 -a fc00::1024
unicast=unique-local=global=low-byte=unspecified
```

```
# Find all the unique addresses in a file
verax@null ~ $ cat list_of_addresses | addr6 -i -q
2001:db8::8:8000:0:417a
2001:a38::8:8000:0:417a
fe80::e8b:fdff:fef4:916
```

```
# Filter addresses
verax@null ~ $ cat list_of_addresses | addr6 -i --accept fe80::/64
fe80::e8b:fdff:fef4:916
```

HOST DISCOVERY ON IPV6 NETWORKS

[RFC 7707]

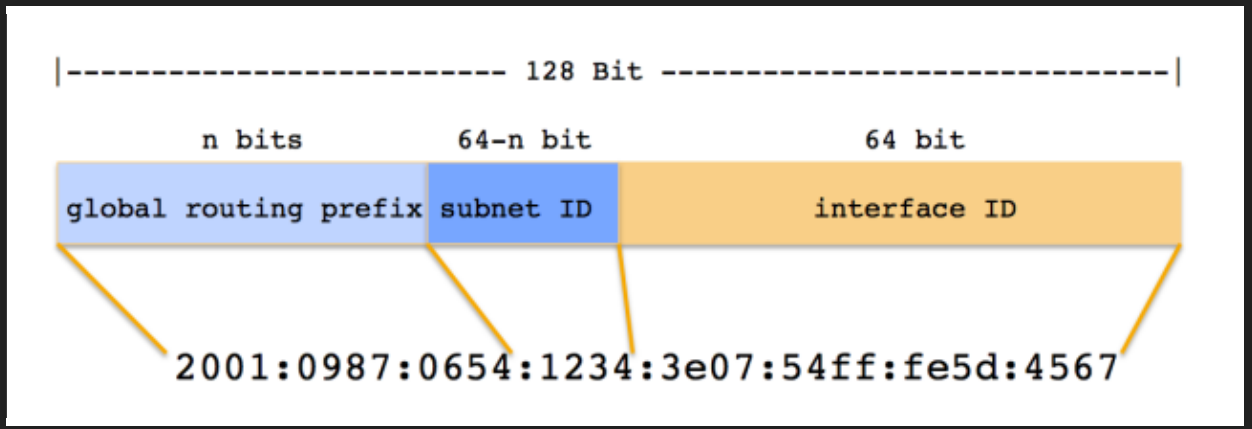
- An IPv6 address is 128 bits long
- If every IP was completely random without a pattern/prefix the search space would be:

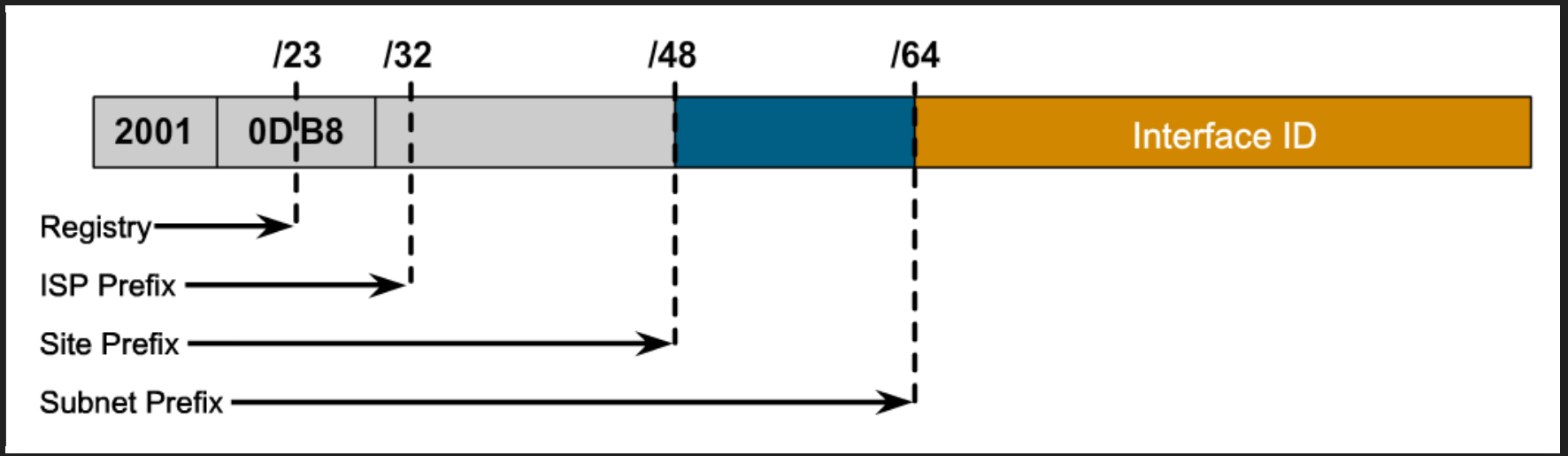
2^128 =
340,282,366,920,938,000,000,000,000,000,000,000,000

say what!!??

But that's not how IPV6 addresses work. IPv6 addresses are logical & hierarchical(even more so than IPv4)

IPV6 ADDRESS STRUCTURE





- Each IPv6 subnet has a fixed size.
- Lower 64 bits of an IPv6 address is the Interface ID(IID).
- The search space at this point equals the maximum number of nodes possible per subnet:

2^64 = 18,446,744,073,709,551,616

Brute force scanning is infeasible, to say the least

If we could find a pattern to the assignment of Interface Identifiers, we could possibly narrow down our search!

INTERFACE IDENTIFIER CONFIGURATION

- Manual configuration
  - Words
  - Last byte
- Autoconfiguration(SLACC)
  - Modified EUI-64
  - Privacy extensions
- DHCPv6

WORDY IIDS

```
verax@null ~ $ host facebook.com
facebook.com has address 157.240.7.35
facebook.com has IPv6 address 2a03:2880:f10c:83:face:b00c:0:25de
facebook.com mail is handled by 10 msgin.vvv.facebook.com.
```

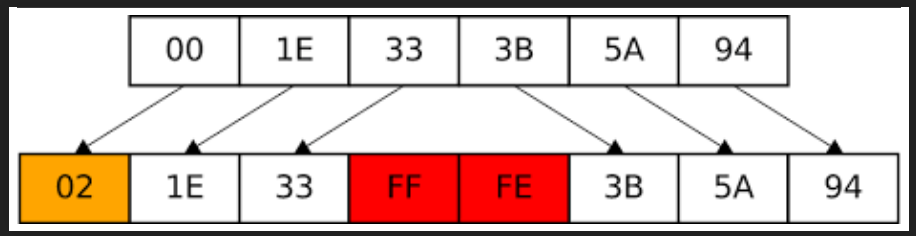
Using words as Interface identifiers

SLACC

Stateless address configuration means that the client picks their own address based on the prefix being advertised on their connected interface(provided by the local router)

EXTENDED UNIQUE IDENTIFIER(EUI-64)

- MAC address is EUI-48.
- An IPv6 address needs 64 bit EUI.



PROBLEM WITH EUI 64 ADDRESSES

- FFFE is fixed, reducing the search space to 2^48.
- OUIs are limited and are publicly available, a clever list of OUIs will reduce the search space to almost 2^24
- Making matters worse, hardware brought togeather tend to have sequential MAC addresses, reducing the search further.

SCANNING EUI 64 ADDRESSES

```
verax@null $ sudo scan6 -i vboxnet0 -d 2001:d:0:1::/64 -V vbox -v
Rate-limiting probe packets to 1000 pps (override with the '-r' op
Target address ranges (1)
2001:d:0:1:a00:27ff:fe00-feff:0-ffff

Alive nodes:
2001:d:0:1:800:27ff:fe00:0
```

```
verax@null $ sudo scan6 -d 2001:d:0:1::/64 -K 'Dell Inc' -v
Rate-limiting probe packets to 1000 pps (override with the '-r' op
Target address ranges (32)
2001:d:0:1:f24d:a2ff:fe00-feff:0-ffff
2001:d:0:1:d6be:d9ff:fe00-feff:0-ffff
2001:d:0:1:d6ae:52ff:fe00-feff:0-ffff

... snipped ...

2001:d:0:1:213:72ff:fe00-feff:0-ffff
2001:d:0:1:212:3fff:fe00-feff:0-ffff
2001:d:0:1:211:43ff:fe00-feff:0-ffff

Alive nodes:
^C
```

BIG BROTHER IS WATCHING YOU!

- MAC addresses are globally unique (mostly)
- SLAAC: Modified EUI-64 Interface ID is derived from MAC Users and when moving between networks, network prefixes are changing but interface ID remains constant over time!
- User can be identified and tracked!

PRIVACY EXTENSIONS FOR SLAAC

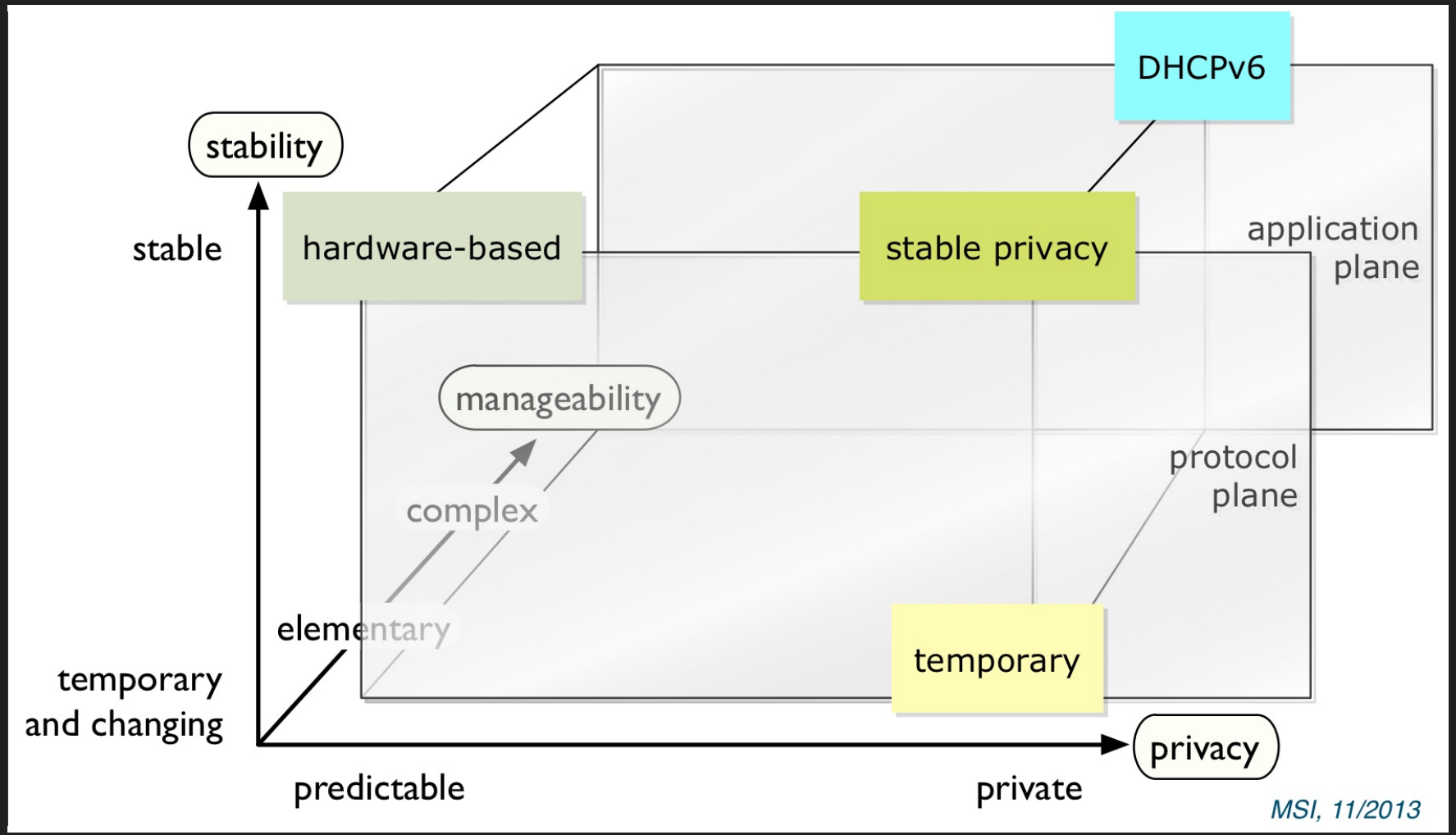
- Task: provide privacy for users
- Approach: Random interface ID that changes over time.
- Availability: Enabled by default on most OSs.

[RFC 4941]

PROBLEMS WITH PRIVACY EXTENSIONS

- Privacy extension addresses are assigned alongside EUI 64.
- EUI-64 is used for server purposes, privacy addresses are used for client needs.
- Constantly changing addresses are a network admin nightmare.

PRIVACY-STABILITY-MANAGEABILITY



To SLACC or not to SLACC

A SOLUTION THAT WORKS

[RFC 7217] A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)

Basically..

Create an IID from network specific data with some crypto which results in an IID that is random, stays the same for a network but changes on a different network.

PING LINK-LOCAL NODES

```
verax@null ~ $ ping6 -I vboxnet0 ff02::1 | cut -d\  -f4
fe80::800:27ff:fe00:0
fe80::800:27ff:fe00:0
fe80::a00:27ff:fef2:eeae
fe80::a00:27ff:fe3f:3acd
... snipped ...
```

PING LINK-LOCAL ROUTERS

```
verax@null ~ $ ping6 -I vboxnet0 ff02::2 | cut -d\  -f4
fe80::a00:27ff:fef2:eeae:
fe80::a00:27ff:fef2:eeae:
... snipped ...
```

EXPLORING NEIGHBOURS WITH **IP** COMMAND

```
verax@null ~ $ ip -6 neigh show dev vboxnet0
fe80::a00:27ff:fe3f:3acd lladdr 08:00:27:3f:3a:cd STALE
2001:d:0:1::1 lladdr 08:00:27:f2:ee:ae router REACHABLE
fe80::a00:27ff:fef2:eeae lladdr 08:00:27:f2:ee:ae router STALE
```

METASPLOIT MULTICAST PING

```
msf auxiliary(ipv6_multicast_ping) > run

[*] Sending multicast pings...
[*] Listening for responses...
[*]    |*| fe80::a00:27ff:fe3f:3acd => 08:00:27:3f:3a:cd
[*]    |*| fe80::a00:27ff:fef2:eeae => 08:00:27:f2:ee:ae
[*] Auxiliary module execution completed
```

Module: auxiliary/scanner/discovery/ipv6_multicast_ping

PORT SCANNING IPV6 DEVICES

METASPLOIT PORT SCAN

```
msf auxiliary(tcp) > run

[*] 2001:d:0:1::1:22 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Module: auxiliary/scanner/portscan/tcp

NMAP SCANNING

```
verax@null ~ $ nmap -6 -sT -T4 -PN -n 2001:d:0:1::0/126

Starting Nmap 6.40 ( http://nmap.org ) at 2016-12-15 19:43 IST

... snipped ...

Nmap scan report for 2001:d:0:1::1
Host is up (0.00033s latency).
Scanned at 2016-12-15 19:42:01 IST for 0s
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap scan report for 2001:d:0:1::2
Host is up (0.092s latency).
All 1000 scanned ports on 2001:d:0:1::2 are filtered
```

BUILDING AN IPV6 LAB

VIRTUALBOX

- VirtualBox supports IPv6 addressing (Host-only, Bridged-wired mode)

TURNING A LINUX BOX INTO IPV6 ROUTER

ROUTER ADVERTISEMENT DAEMON(RADVD)

Used to make Linux/BSD act as IPv6 router. It sends Router Advertisement messages as specified by RFC 2461.

```
# You have to enable IP forwarding
# Uncomment the following line in /etc/sysctl.conf
net.ipv6.conf.all.forwarding=1
```

```
sudo apt-get install radvd      # Install radvd
```

```
# Basic radvd config file /etc/radvd.conf
interface eth0
{
   AdvSendAdvert on;
   prefix 2001:db8:0:2::/64
   {
   };
};
```

```
sudo service radvd start
```

CONFIGURING RADVD

Sample radvd.conf which also advertises DNS servers with RDNSS.

```
interface eth0
{
  AdvSendAdvert on;
  MinRtrAdvInterval 3;
  MaxRtrAdvInterval 10;

  prefix 2001:db8:0:1::/64
  {
  };

  RDNSS 2001:db8:0:1::a 2001:db8:0:1::b
  {
    AdvRDNSSLifetime 10;
  };
};
```

More info at: tldp.org/HOWTO/Linux+IPv6-HOWTO/

ISC DHCP SERVER(DHCPV6)

```
sudo apt-get install isc-dhcp-server
```

```
ddns-update-style none;

default-lease-time 7200;
max-lease-time 86400;

subnet6 2001:db8:0:2::/64 {
  range6
    2001:db8:0:2::1000
    2001:db8:0:2::1fff;

  option dhcp6.name-servers
    2001:db8:0:1::a,
    2001:db8:0:1::b;

  option dhcp6.domain-search
    "koo.fi";
```

```
sudo service isc-dhcp-server6 start
```

http://koo.fi/blog/2013/03/20/linux-ipv6-router-radvd-dhcpv6/

REFERENCES

- http://www.openwall.com/presentations/IPv6/
- njetwork.wordpress.com/2013/11/03/to-slaac-or-not-to-slaac/
- https://tools.ietf.org/rfc/rfc7707.txt
- internetsociety.org/deploy360/resources/privacy-extensions-for-ipv6-slaac
- http://koo.fi/blog/2013/03/20/linux-ipv6-router-radvd-dhcpv6
- https://go6.si/wp-content/uploads/2016/06/Fernando-Gont-IPv6-Security.pdf
- https://www.si6networks.com/tools/ipv6toolkit/

QUESTIONS?

Twitter.com/yamakira_
Github.com/yamakira
http://disruptivelabs.in