Are you ready for IPv6 insecurities ?

George Kargiotakis

kargig@void.gr
ATHCON 2012

whois

$ id

uid=1000(kargig) gid=1000(sysadmin)
groups=1(HELLUG),2(HTFv6),3(Hackerspace.gr),4(DLN.gr)

$ last kargig

GRNET – System Administration

Gennet – Linux-based broadband CPEs (IPv6 capable)

University of Ioannina – System Administration

$ apropos kargig

iloog – Greek gentoo-based livecd

GrRBL – Greek AntiSpam Blacklists

Greek AdBlock Plus filter – self-explanatory

void.gr – My Blog

Moment of truth

How Many of you

- ~~know what IPv6 is ?~~
- have used/are using IPv6 at work/home ?
- **know what SLAAC is ?**
- have used/are using transition mechanisms ?
- **have deployed services over IPv6 ?**
- are using native IPv6 ?
- **are using native IPv6 and have applied IPv6 specific security policies on servers/routers ?**

Topics

- Fast IPv6 crash course (still needed)
- Main Dish

Fast IPv6 crash course

- Old vs New
- IPv6 Header + Header Extensions
- IPv6 Neighbor Discovery
- IPv6 Address Types + Addressing
- IPv6 DNS

Fast IPv6 crash course

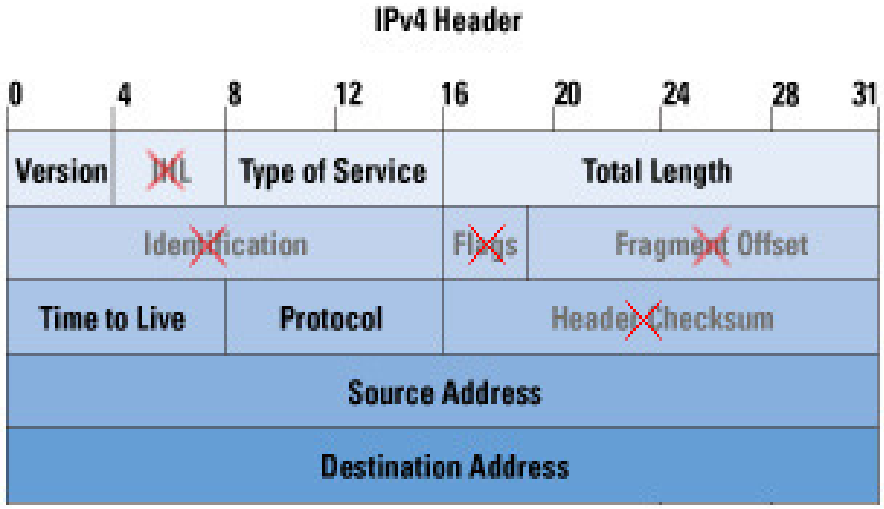
Good ol' times

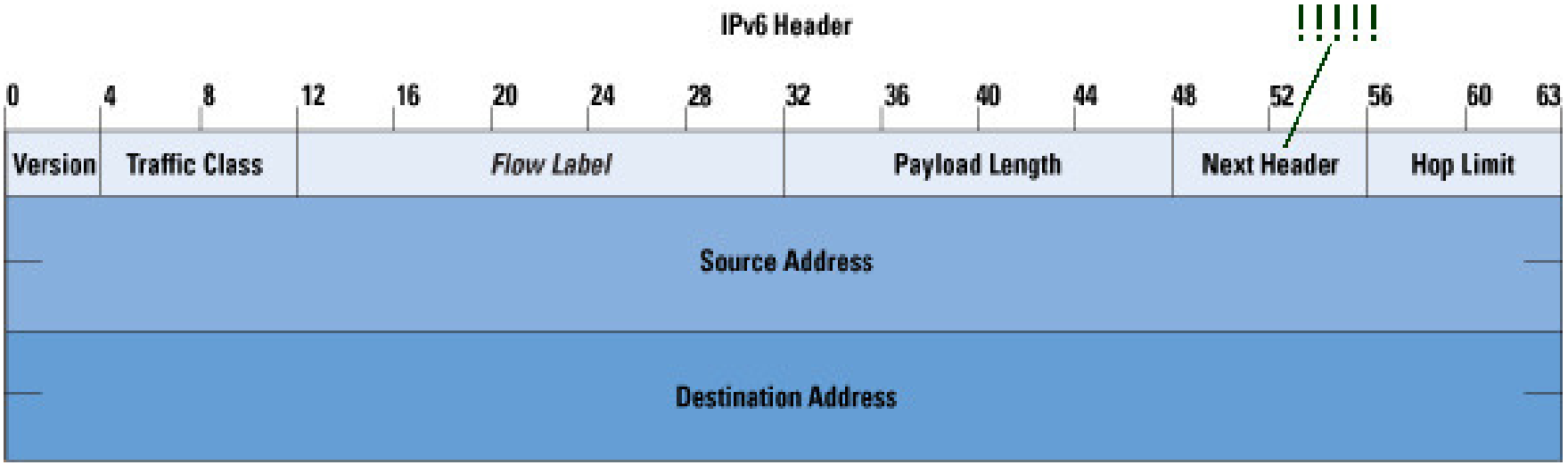
- 32-bit addr - 4.294.967.296 IPs
- Classful → Classless (CIDR)
- Private Addresses + NAT

Embrace the new

- 128-bit addr – 340.282.366.920.938.463.463.374.607.431.768.211.456 IPs
- Hierarchical Address Space
- Multiple IPs (w/ different scopes) per Interface
- Lots of Multicast (no more broadcast!)
- Network Discovery Protocol → Address Auto-configuration
- Simpler Header + Extension Headers + Daisy Chaining

Fast IPv6 crash course

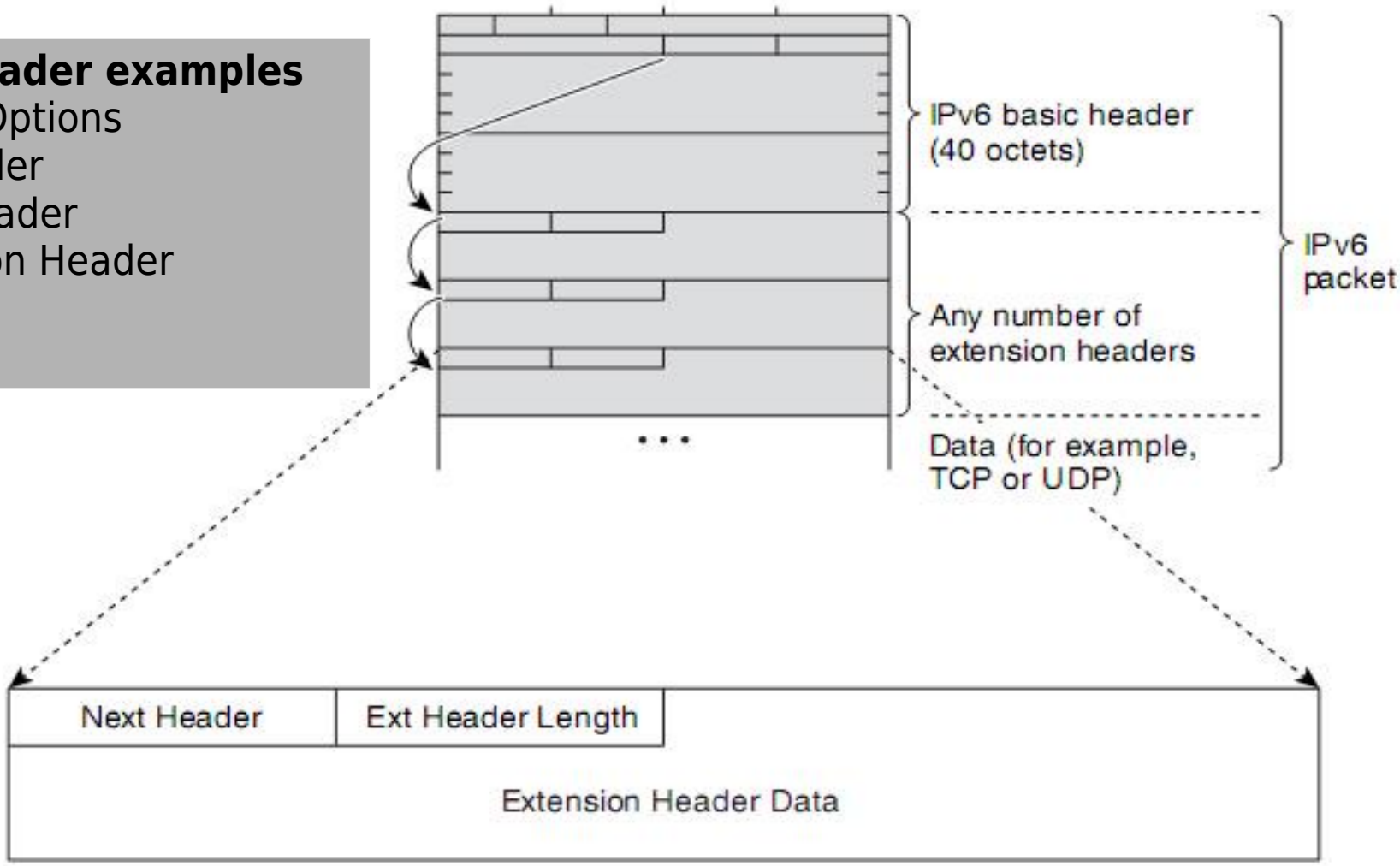




Fast IPv6 crash course

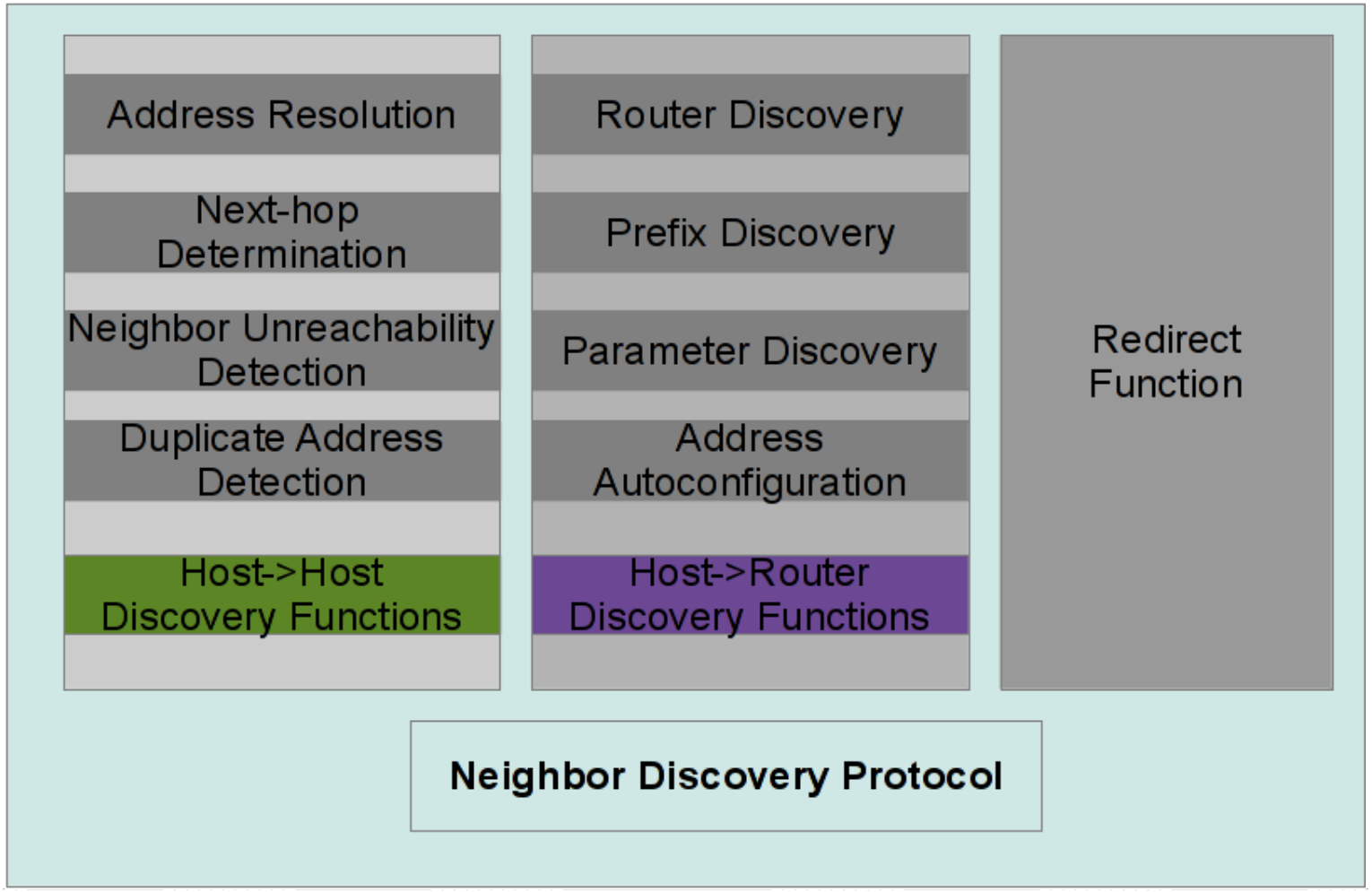
Extension Header Daisy Chaining

Extension header examples
- Destination Options
- Routing Header
- Fragment Header
- Authentication Header
- Mobility



IPv6 Neighbor Discovery Protocol

Based on ICMPv6 → Replaces ARP + ICMP on IPv4



Fast IPv6 crash course

Commonly used ND messages

- **R**outer **A**dvertisement (Type 134)
- **R**outer **S**olicitation (Type 133)
- **N**eighbor **A**dvertisement (Type 136)
- **N**eighbor **S**olicitation (Type 135)

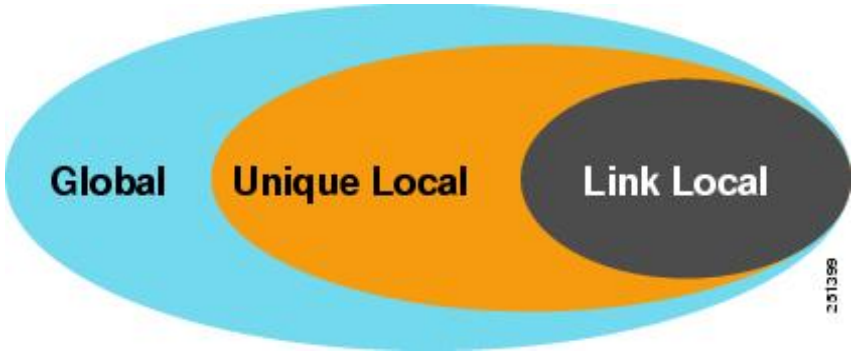
Benefits of IPv6 ND

- Formalize Address Resolution + Router Discovery (Security at layer 3 independent of IPsec → SeND)
- Autoconfiguration
- Dynamic Router Selection
- Multicast

Fast IPv6 crash course

Address Types

- Unicast: Link Local, Unique Local, Global
- Multicast
- Anycast
- Reserved



Address Type	Prefix
Unspecified	:: (or ::/128)
Loopback	::1 (or ::1/128)
Multicast	FF00::/8
LL Unicast	FE80::/10
ULA Unicast	FC00::/7
Global Unicast	All the rest...

Fast IPv6 crash course

IPv6 Address → 2001:db8:5a54:1a3b:1200:b00b:210:98

8 hex groups of 16bit seperated by “:”

2 Transformation Rules

I. Leading 0s within a 16-bit value may be omitted

II. A single occurrence of consecutive groups of 0s within an address may be replaced with a double colon

- Example: 2001:0db8:abcd:cafe:0000:0000:0000:0005

I. 2001:db8:abcd:cafe:0:0:0:5

II. 2001:db8:abcd:cafe::5

Fast IPv6 crash course

Address = **Network ID** + **Interface ID** (64+64 bits)

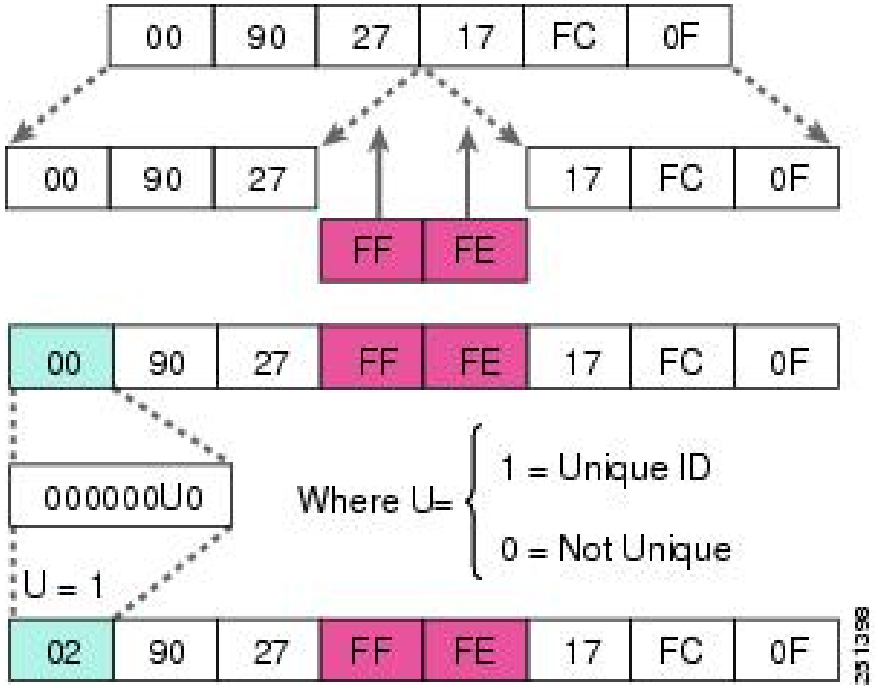
- **Network ID** = ISP Prefix + (random) subnet ID
- **Interface ID** configuration options
    - Auto-configured by MAC address (SLAAC)
    - DHCPv6
    - Manual
    - Pseudo-random (Temp. Addresses)

Getting an IP(v6): **Manually** / **SLAAC** / **DHCPv6**

Fast IPv6 crash course

SLAAC (Stateless Address Autoconfiguration)

- multicast ICMPv6
- EUI-64: Create last 64bits from MAC
- Router Advertisement gives:
  - IPv6 Prefix(es)
  - Default Router
  - MTU
  - Lifetime
  - DNS
  - Other Config (!)



Creating an EUI-64 from a MAC

EUI-64: 00:90:27:17:FC:0F → 0290:27**FF:FE**17:FC0F

Fast IPv6 crash course

The SLAAC problem

- MAC → EUI-64 → last 64bits of address are always the same → IPv6 “super cookie” → **Privacy issues**!

Proposed solution

- **Privacy Extensions** (RFC4941) → Randomize last 64bits of address → **Temp. Addresses**

  Temp. Addresses change every XX mins (e.g. 15)

- **New problem**: How to monitor local users with Temp. Addresses if they keep changing ?

- **New Proposal** (?): Stable but random per Prefix last 64bits (draft-gont-6man-stable-privacy-addresses-00 - Dec 2011)

Fast IPv6 crash course

Stateful DHCPv6

- Model: Client/Server
- Transport: Multicast UDP
- Provides: Addressing, Routing, DNS, SIP, NTP, etc options
- Addresses: Temporary (IA_TA) & non-temporary (IA_NA)
- (NEW) **Prefix Delegation** (IA_PD): Request subnet from WAN to provide addressing for the LAN

Stateless DHCPv6

- Get IP address by SLAAC + OtherConfig Flag “O”=1
- Extra configuration params (e.g. DNS) by DHCPv6

Fast IPv6 crash course

```
# ip address ls dev eth0

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000

   link/ether 00:22:41:1e:a8:d5 brd ff:ff:ff:ff:ff:ff          ← MAC

   inet 192.168.1.94/24 brd 192.168.1.255 scope global eth0    ← IPv4

   inet6 2a02:580:8000:9701:222:41ff:fe1e:a8d5/64 scope global dynamic

      valid_lft 86391sec preferred_lft 3591sec                 ← GLOBAL

   inet6 fdbf:468f:aaa0:474d:222:41ff:fe1e:a8d5/64 scope global dynamic

      valid_lft 86391sec preferred_lft 3591sec                 ← ULA

   inet6 fe80::222:41ff:fe1e:a8d5/64 scope link                ← Link-Local

      valid_lft forever preferred_lft forever
```

Fast IPv6 crash course

DNS is extremely important!

Browsers: http://[2001:1af8:4100:a02c:1::16] ← **Global**

Shell: scp kargig@\[fdbf:468f:aaa0:474d:ab01::ff\]:file.ext localpath/ ← **ULA**

Shell: ssh kargig@fe80::a80c:eaff:feda:b0db%eth0 ← **LL**

AAAA forward record (hostname → IPv6 address)

void.gr. IN AAAA 2001:1af8:4100:a02c:1::16

PTR reverse record (IPv6 address → hostname)

6.1.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.8.f.a.1.1.0.0.2.**ip6.arpa**. IN PTR void.gr

IPv6 Security Considerations

Main Dish

- IPv6 Security Hype
- Common Local Attacks & mitigation
- Remote Network Scanning
- Local Network Scanning
- IDS/Firewalling – OS Support - IPv6 Migration Security
- Scanning IPv6 Internet
- Tools
- Food for thought – IPv6 Security Overview
- (+Bonus Slides)

IPv6 Security Considerations

IPv6 Security Hype

IPsec is mandatory!!111oneoneone

- NOT! IPsec ***support*** is mandatory, not usage!

No more ARP spoofing!!11eleveneleven

- Yeah, they are now called ND (local) attacks...

My fridge/toaster will be accessible from the Internet! Save meeeee!!

- It won't → Stateful firewalls/ACLs

The Fun Begins

IPv6 Common Local Attacks

Address Resolution

- Attacker claims victim's IP address and replies to **N**eighbor **S**olicitation requests

Redirect

- Attacker sends **R**outer **A**dvertisement and redirects traffic heading to an **off-link** host/prefix elsewhere (or himself)

Duplicate Address Detection DoS

- Attacker replies to any victim's DAD requests

IPv6 Common Local Attacks

First-Hop Router Attack

- Attacker sends **RA** and tricks victim into accepting himself as a default router by canceling the previous one (prefix lifetime=0) first. Steals all traffic.

Address Configuration DoS

- Attacker cancels previous default router prefix and sends new (fake) prefix to victim. Victim can't access the network due to spoofed prefix filtering by default router.

DHCPv6 spoofing

ND Attacks Mitigation Techniques

- RAguard (L2 Protection) – RFC 6105
  - Makes switching devices capable of identifying invalid RAs within packets and blocking them
  - Stateless & Stateful modes
- SeND – RFC 3971
  - Crypto approach to Secure ND
  - (Very) Hard to deploy – Needs PKI Infra

IPv6 Common Local Attacks

Other Mitigation Techniques

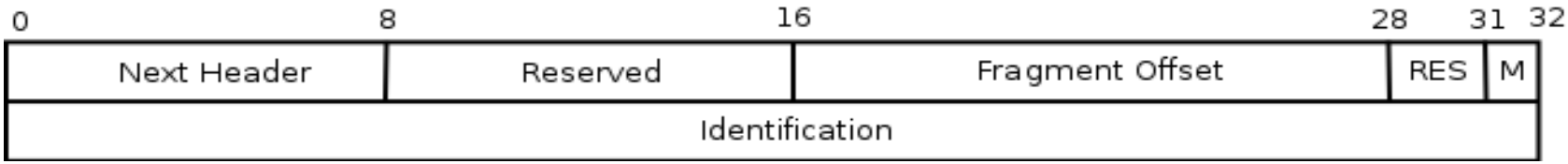
- Firewall/ACL to block **specific** rogue ICMPv6 – RFC 4890
- DHCPv6 filtering/ACLs (UDP port 546/547)
- Monitor ND with NDPMon
- Disable SLAAC (in server environments)

Fragmentation

Fragmentation Issues

IP Fragmentation in IPv6 only happens at hosts!
PMTU is mandatory!

Fragmentation Header (Type 44)



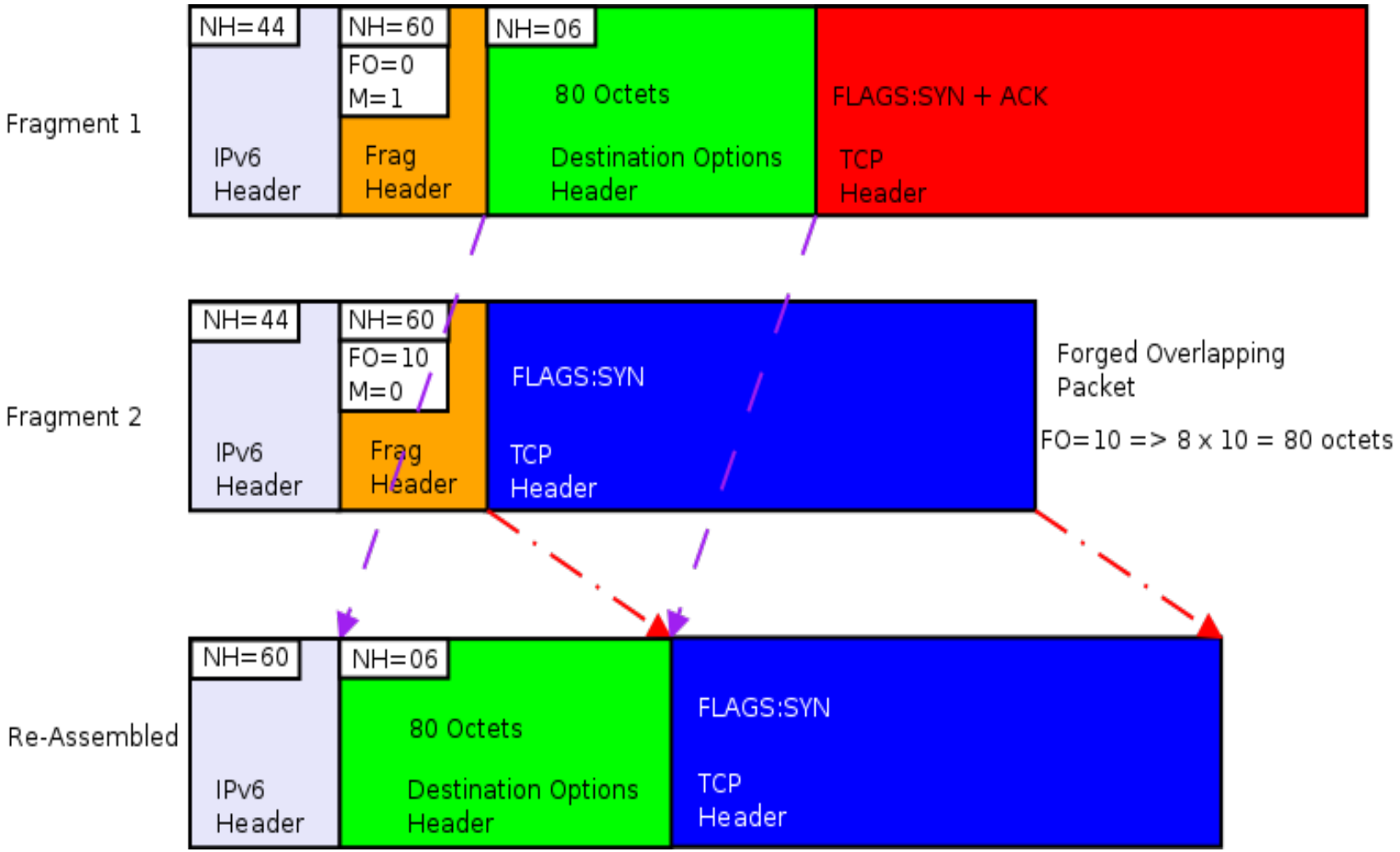
Fragment Offset is integer (x8 octets in size)
M is boolean → More Fragments will follow
Identification is a unique ID per session

Major issues

- Overlapping Fragments (RFC 5722)
- Atomic Fragments + Predictable Fragment Identification values

Fragmentation Issues

(Over-Simplified) Overlapping Fragments example



Fragmentation Issues

- Many OSs use(d) to “merge” overlapping fragments. Any **recent OS** should completely **drop sessions** (not just packets) with overlapping fragments (Linux semi-patched at 11/2010).
- If you can **predict** fragment Identification values and you can send fake packets with overlap. fragments → **DoS** (Linux patched at 07/2011)
- **Atomic** fragments (single packets with a fragm. header) can cause **DoS** to non-fragmented traffic
- (some) RAguard implementations can be evaded using fragments by adding looong or multiple DOH

More on this issue at:
http://blog.si6networks.com/2012/02/ipv6-nids-evasion-and-improvements-in.html

Scanning

Remote Network Scanning

Server LANs (!SLAAC)

- Rely on DNS for prefix extraction
- Start with first 64bits of prefix + ::1 (or ffff::1)
- Try 1-255 as last 16bits of address (or 1-ff)
- Try common last 16bits of addresses:
  - 100,1000,1111,2000,666,etc
  - f00d,cafe,dead,beaf,aaaa,ffff,b0ff,b00b
  - e.g. FB → 2620:0:1cfe:face:b00c::3
- Common router addresses → ::1 or ::2

Recommended Presentation: “Recent Advances In IPv6 Insecurity” by Marc "Van Hauser" Heuse 27C3

Remote Network Scanning

Home/SMB LANs (SLAAC)

- Use Vendor IDs (used in MAC → EUI-64 transformation)

Indirect Methods

- Email Headers
- Parse web logs (watch for 'ff:fe' of SLAAC)
- Client side “attacks”: email to clients → pic on IPv6 only host, parse logs
- Search engines: site:ipv6* site:ip6*

Temp. Addresses make it really hard to remotely scan Home/SMB LANs (but add huge administrative cost)

Remote Network Scanning

Rev DNS Trick - efficiently mapping ip6.arpa

- Reduce queries (from gazzilions to thousands) to find hosts within a /32 (→ that's 2^96 IP addresses)
- Can find hosts within a /32 in minutes-hours
- Go one nibble at a time “backwards” according to responses (does not work with PowerDNS, works with Bind!)
    - Add 0 in front of x.x....x.x.ip6.arpa
    - Do PTR for 0.x.x....x.x.ip6.arpa
    - If response is NXDOMAIN (→ nothing here)
        - Add +1 to that nibble and do a PTR query again
    - If response is NOERROR → a host might be further down → continue with this nibble and add 0 in front of it

Remote Network Scanning

Example: finding host(s) within 2001:1af8::/32

6.1.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.**8.f.a.1.1.0.0.2**.ip6.arpa PTR void.gr

I. 0.8.f.a.1.1.0.0.2.ip6.arpa → NXDOMAIN … **4**.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

II. 0.4.8.f.a.1.1.0.0.2.ip6.arpa → NXDOMAIN … **1**.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

III. **0**.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

IV. **0**.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

V. 0.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NXDOMAIN … **a**.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

VI. **0**.a.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

VII. ….

VIII. ….

IX. 0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NXDOMAIN …
1.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

X. 0.1.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NXDOMAIN …
6.1.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa → NOERROR

Remote Network Scanning

Rev DNS Trick - efficiently mapping ip6.arpa

Tools of the trade

- thc-ipv6: dnsrevenum6 (not included in v1.8)
- ip6-arpa-scan.py
- dns-ip6-arpa-scan.nse

Local Network Scanning

Reveal LL addresses

- **ping6 ff02::1%eth0** - All-Nodes Multicast Address
    - Does not work on Windows Vista/7
- **ping6 ff02::2%eth0** - All-Routers Multicast Address
- thc-ipv6 toolkit:
    - **alive6** (-l): ping + ping with error header (hop by hop opt) (works even on Windows Vista/7)
    - **fake_mld6**: mld discovery

Get Global Addresses from these

- Sniff RAs for advertised prefix & combine LL addresses.

Local Network Scanning

Discover Hosts on a LAN using IPv6 tricks

Rogue RA (control hosts)

- Send RA (lifetime=1) with new prefix & listen for solicitations
  Tool: nmap targets-ipv6-multicast-slaac NSE

Information Leakage from hosts

- mDNS
- scapy + ICMPv6 139/140 - Node Information Query/Response
- dig any void.gr @ff02::1%eth0 (use tcpdump/wireshark and look at the replies)

Be prepared...

IDS / Firewalling

IDS support is still very immature

Bad signs

- Poor logging of IPv6 Addresses (LL!)
- Even poorer actions against attackers
- Few specific IPv6 Rules
- Suffer from fragmentation attacks
- SLAAC enabled on networking interfaces (!?)

Firewalling

- Use dual stack enabled tools (ferm on Linux)
- Don't forget LL, ULA addresses

OS Support

Don't Forget!

Every Major OS has been IPv6 enabled for years now

IPv6 Migration Security

Block transition techniques not in use

- Deny IPv4 protocol 41 forwarding unless that is exactly what is intended (also block 192.88.99.0/24 → 6to4 tunnels)
- Deny UDP 3544 forwarding → Teredo tunneling
- Deny TCP/UDP 3653 → TSP (tunnel setup protocol)

Avoid Dynamic Tunnels (6to4,Teredo,TSP, etc)

- Turn them off if unneeded (Windows 7)
- **Bitch** at your ISP for native IPv6 connectivity

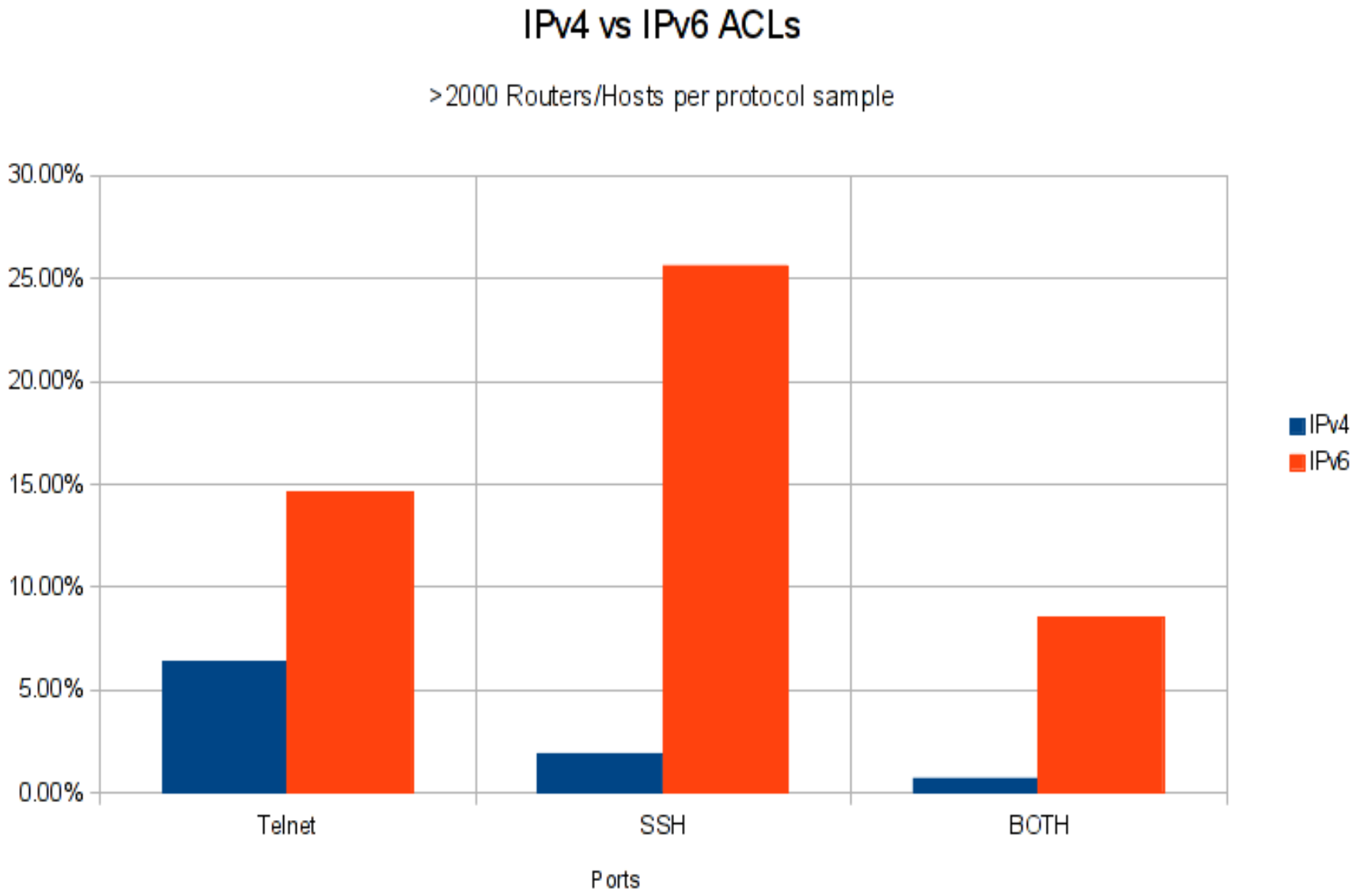
surprise, surprise

Scanning IPv6 Internet

Comparing IPv4 and IPv6 security policies

- Wanted to check status of firewall policies
- Got a few thousand hosts from apache logs on dual stack hosts
- If IPv6-IP {rev DNS → hostname → IPv4-IP }
- Traceroute each IP(v4/v6)
- Connect to SSH/Telnet of every unique intermediate router of the path to IP(v4/v6)
- Log && parse results
- **Surprise!**

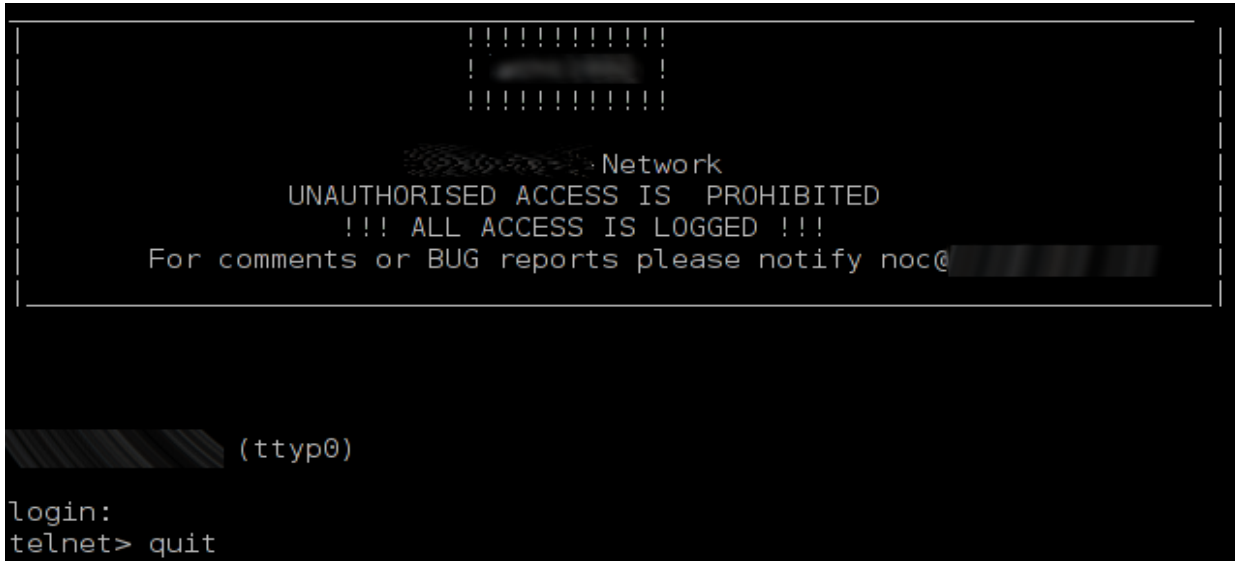
Scanning IPv6 Internet



Scanning IPv6 Internet

Major Internet Carriers/ISPs have either telnet or ssh access over IPv6 unprotected

- Same Router/Host (almost) always has IPv4 ACLs
- If ssh/telnet are open…what about other ports ?
- Only one (Greek) ISP noticed (?) the scans and terminated access over IPv6



Shiny...

Tools

- THC-IPv6
- scapy
- ndisc6
- tcpdump/wireshark (ORLY?)
- nmap (-6) + NSE scripts
- nc6/socat
- 6tunnel
- ndpmon

Food for thought

Things you can play with

- ND flooding/fuzzing ?
- Crashing CPEs with malformed packets ?
- Extension header fuzzing against OSs ?
- Play with IDS/Firewalls + Fragmentation ?
- IPv6 over 3G / Mobile Devices ?
- Mobile IPv6 ?

IPv6 Security Overview

IPv6 is no more or less secure than IPv4

- Experience is the issue – currently there aren't enough experienced engineers, more training needed
- Fewer tools in the wild
- ...but also fewer exploits
- Many topics are just now being standardized
- Many current implementations don't follow standards

IPv6 is ATM a playground for creative Hackers ☺

IPv6 has problems that IPv4 “solved” in the 90s...

IPv6 Security Overview

IPv6 will change traffic patterns (p2p, MIPv6)

IPv6 larger address spaces makes worms and scanning less effective but there are still ways to find hosts (be creative!)

Apply IPsec wherever possible (like you did on IPv4...)

LAN based attacks will become **far** more popular

- Disable SLAAC on IPv4-only networks or when unneeded
- Apply stronger Physical Security, Ethernet-Port Security, NAC, 802.1X, SeND

Links

Interesting Links

- http://lists.si6networks.com/listinfo/ipv6hackers/
- https://www.ietf.org/mailman/listinfo/v6ops
- http://www.packetlevel.ch/html/scapy/scapyipv6.html
- http://www.void.gr/kargig/ipv6/
- http://ipvsix.me/
- http://www.stindustries.net/ipv6-security/

Interesting People

- Marc Hause → http://mh-sec.de/
- Fernando Gont → http://www.si6networks.com/

Thanks fly to...

Thanks fly to

- **Census Labs**
- **@faidonl**
- **@apoikos**

For the endless discussions regarding IPv6 security

Act!

Go and deploy secure IPv6 Services!

The End

Thanks for listening!

(No Angelina Jolies were harmed during the making of this presentation)

Any Questions ?

The End

For any questions or consulting regarding IPv6, you can contact me at

kargig [at] void [dot] gr

pgp 0xE4F4FFE6

https://twitter.com/kargig

Bonus Slides

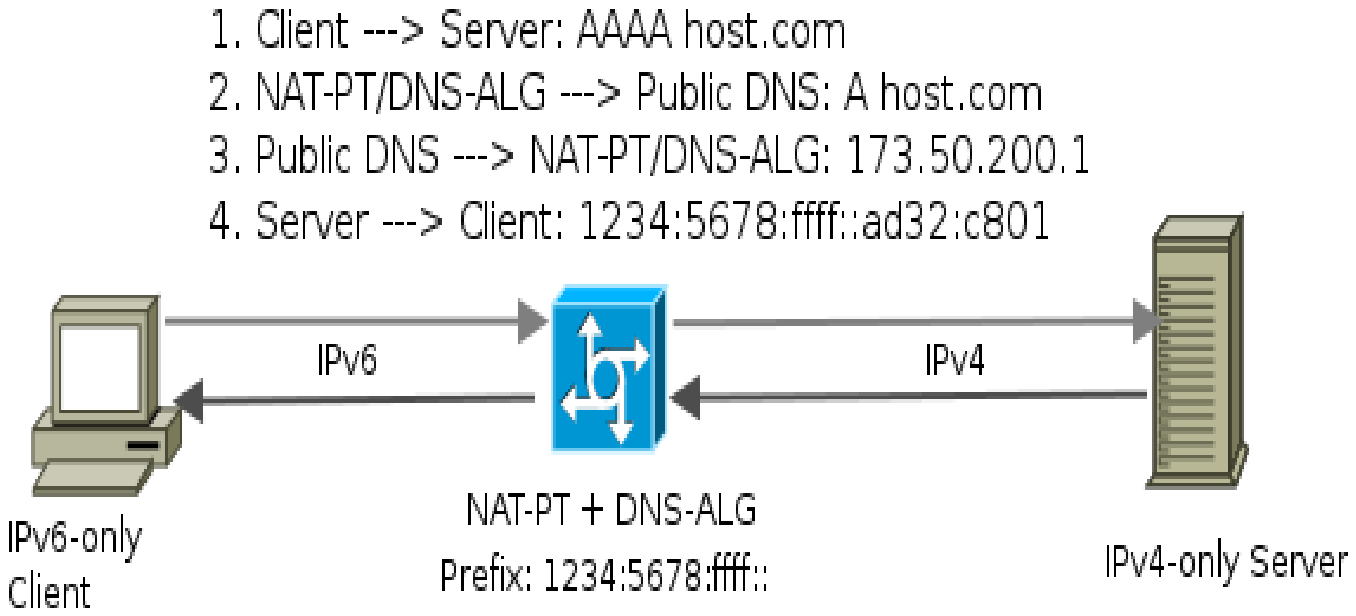
Local Network Protection

GOAL	IPv4	IPv6
Simple Gateway between Internet and Private Network	DHCP	DHCPv6-PD + SLAAC
Simple Security	Filtering side-effect due to lack of translation state	ACL/Firewall
Local Usage Tracking	NAT State Table	Address uniqueness
End-System Privacy	NAT transforms device ID bit in the address	Privacy Extensions
Topology Hiding	NAT transforms subnet bits in the address	Untraceable addresses (IGP host routes/MIPv6 Tunnels)
Addressing Autonomy	Private Address Space	Large Address Space + ULA
Global Address Pool Reservation	Private Address Space	**WHAT ?**
Renumbering/Multihoming	Address translation at border	Lifetime per prefix / Multiple addresses per interface

SLAAC + NAT-PT MITM

NAT-PT (Network Address Translation/Protocol Translation)

- Allows IPv6 hosts to connect to IPv4 hosts through DNS-ALG and NAT manipulation (**deprecated!**)
- Advertise a /96 IPv6 prefix to LAN
- Translates DNS requests and connections



SLAAC + NAT-PT MITM

NAT-PT Router inside an IPv4-only network

- Windows Defaults: SLAAC on + DHCPv6 on
- NAT-PT/DNS-ALG Daemon → DNS/traffic manipulation
- SLAAC + OtherConfig Flag “O”→ Clients get IPv6 prefix served by NAT-PT router + start DHCPv6 Client
- DHCPv6 Client → DNS server pointing to NAT-PT router
- Add them to the mix → Clients can be deceived to use IPv6 to talk to IPv4 Internet hosts

But Luckily…

- NAT-PT daemons don't work that well → NAT64 (?)
- Not clean → Can Break Dual-Stack and v6-only sites