Security considerations for a brave new (IPv6) world

kargig@void.gr
0x375 0x07

Topics

- Intro
- IPv6 fastest crash course ever
- IPv6 Neighbor Discovery Mechanisms
- IPv6 Linux
- IPv6 Security Considerations
- Outro

Intro

```
$ id
uid=1000(kargig) gid=1000(kargig)
groups=1(ILUG),2(HELLUG),3(GR-IPv6 TF)
$ apropos kargig
iloog – Greek gentoo-based livecd
GrRBL – Greek AntiSpam Blacklists
Greek AdBlock plus filter - self-explanatory
```

IPv6 fastest crash course ever

- Old vs New
- Header Comparison - Header Daisy Chaining
- IPv6 Addressing
- IPv6 at home
- IPv6 DNS

Old vs New

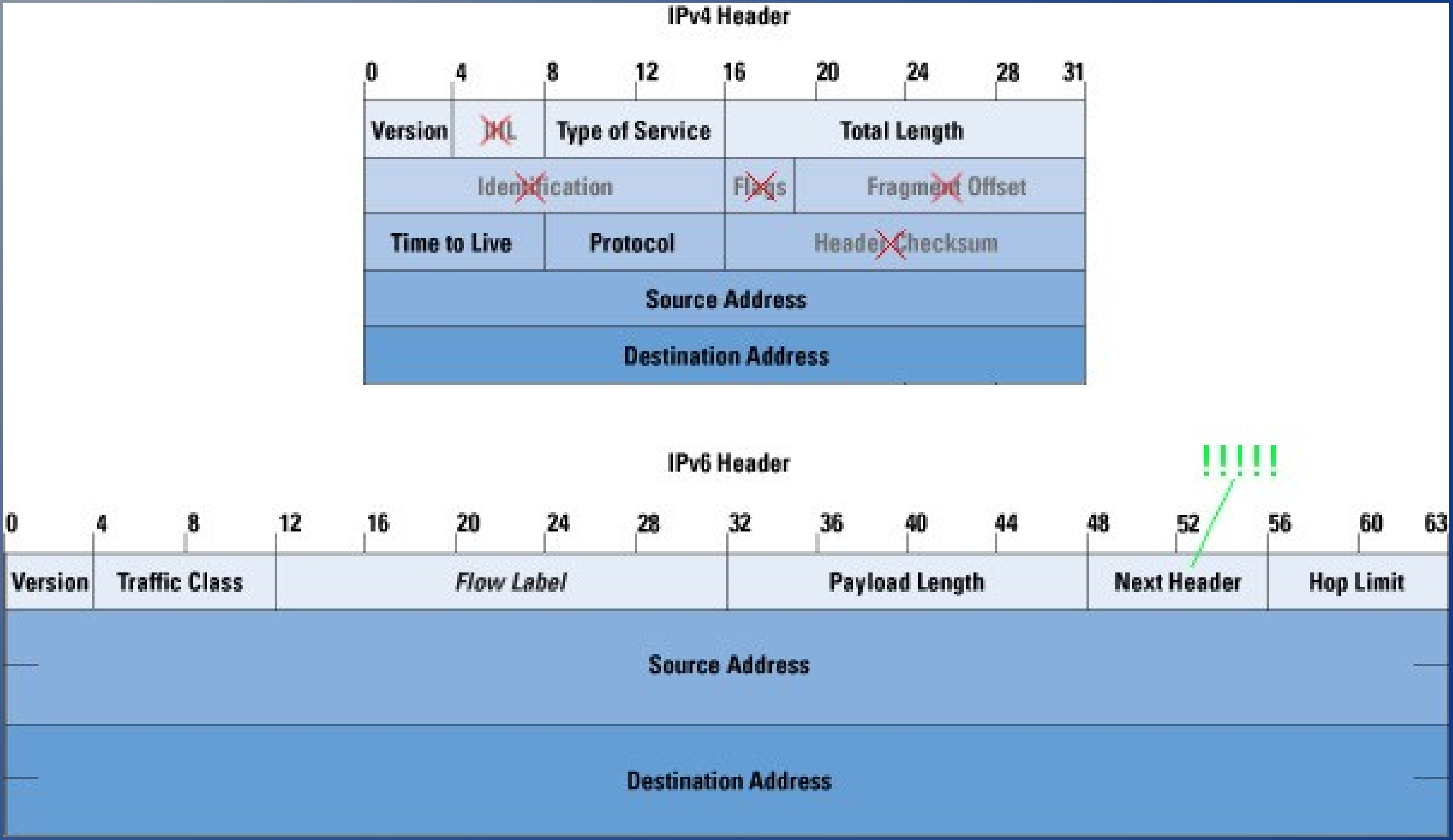
Good ol' times

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

Embrace the new

- IPng→ IPv6 → 128 bits – 340.282.366.920.938.463.463.374.607.431.768.211.456 IPs
- Hierarchical Address Space - Multiple IPs per Interface
- Lots of Multicast (no more broadcast!)
- Network Discovery Protocol - Address Auto-configuration
- Simpler Header (no checksum, no fragmentation) – Header Daisy Chaining

Header Comparison



IPv4 Header

0 4 8 12 16 20 24 28 31

Version | ~~IHL~~ | Type of Service | Total Length

~~Identification~~ | ~~Flags~~ | ~~Fragment Offset~~

Time to Live | Protocol | ~~Header Checksum~~

Source Address

Destination Address

IPv6 Header

!!!!!

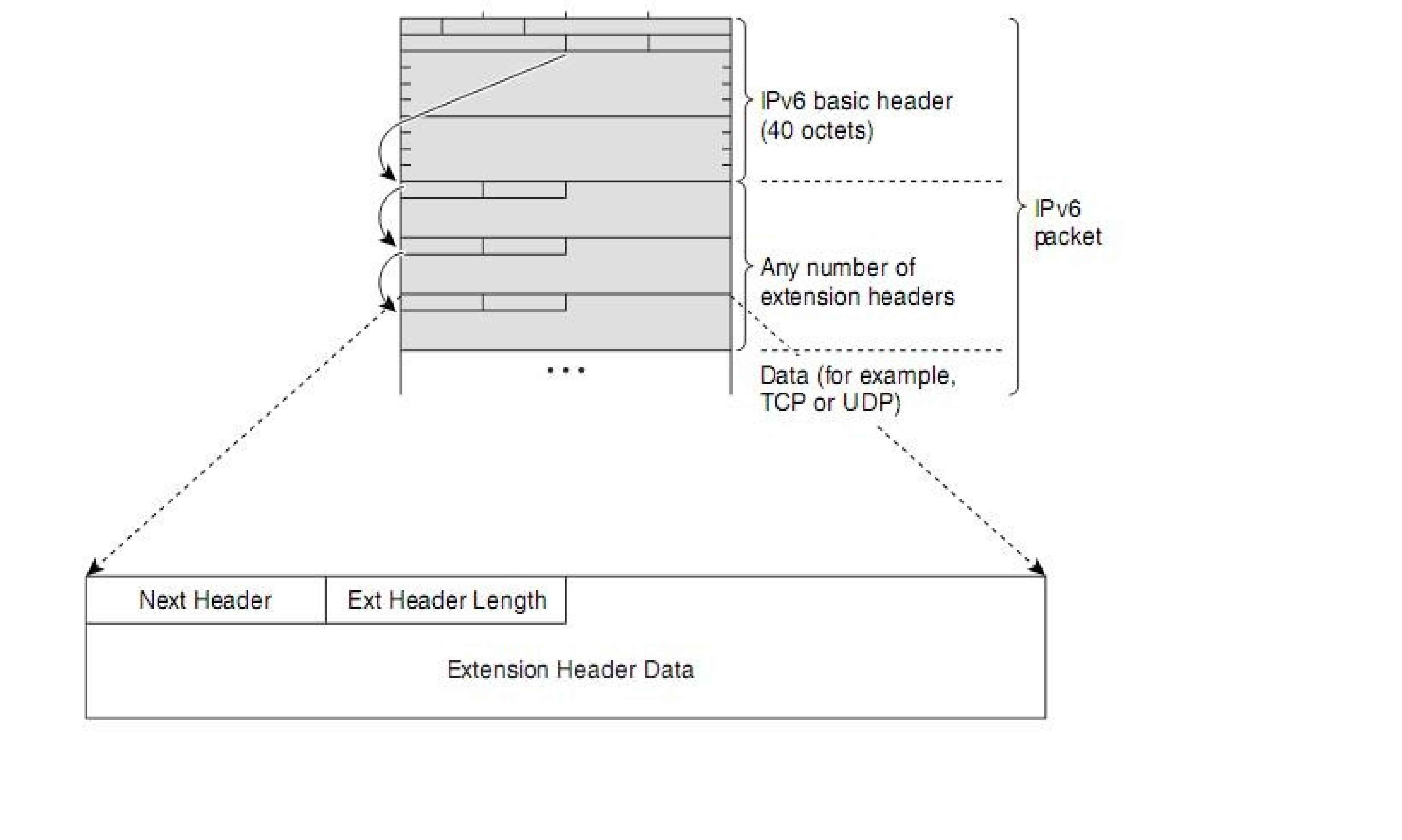
0 4 8 12 16 20 24 28 32 36 40 44 48 52 56 60 63

Version | Traffic Class | *Flow Label* | Payload Length | Next Header | Hop Limit

Source Address

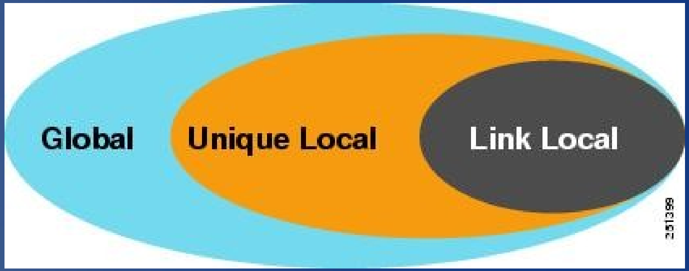
Destination Address

Header Daisy Chaining


IPv6 basic header (40 octets)
Any number of extension headers
Data (for example, TCP or UDP)
IPv6 packet
. . .
Next Header
Ext Header Length
Extension Header Data

IPv6 Addressing (1/2)

- Address Types:
  - Unicast: Link Local (fe80::/10), Unique Local (fc00::/7), Global
  - Multicast (ff00::/8)
  - Anycast
  - Reserved



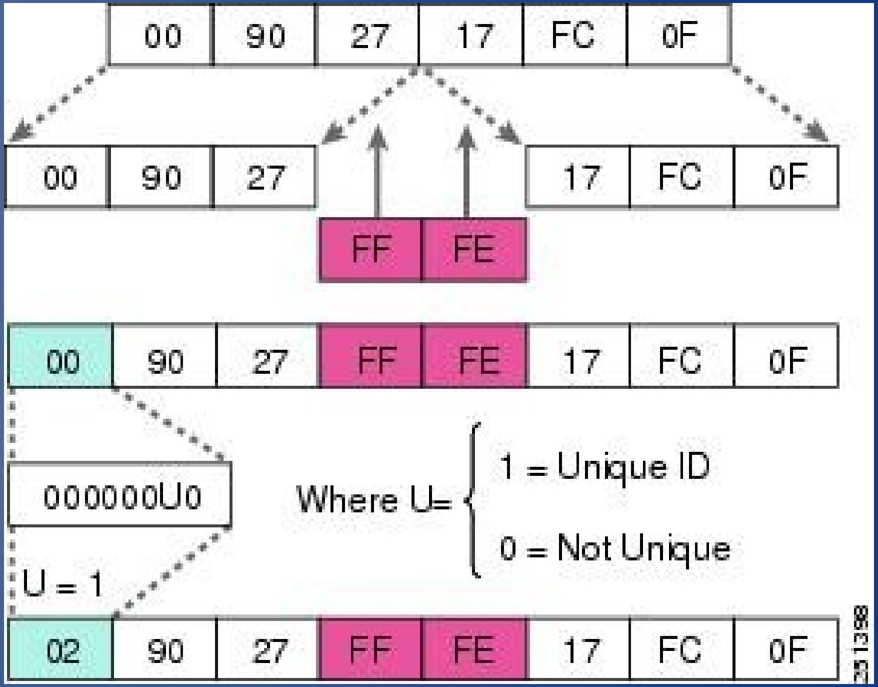
- X:X:X:X:X:X:X (8 hex groups of 16bit) eg 2001:db8:5a54:1a3b:1200:af10:210:98
- 2 Transformation Rules:
  - I. Leading 0 within a 16-bit value may be omitted
  - II. A single occurrence of consecutive groups of 0s within an address may be replaced by a double colon
- Example: 2001:0db8:abcd:cafe:0000:0000:0000:0005
  - I. 2001:db8:abcd:cafe:0:0:0:5
  - II. 2001:db8:abcd:cafe::5

IPv6 Addressing (2/2)

- Unspecified address :: (or ::/128)
- Localhost ::1 (or ::1/128)
- Address = Network ID + Interface ID (64+64 bits)
- Interface ID
  - Auto-configured by MAC address
  - DHCPv6
  - Manual
  - Pseudo-random
- Getting an IP(v6): Manually, SLAAC, DHCPv6

IPv6 Auto-configuration (1/2)

- Stateless (SLAAC) – multicast ICMPv6
  - IPv6 Prefix(es)
  - Default Router
  - MTU
  - Lifetime
  - DNS
  - Other Config (!)



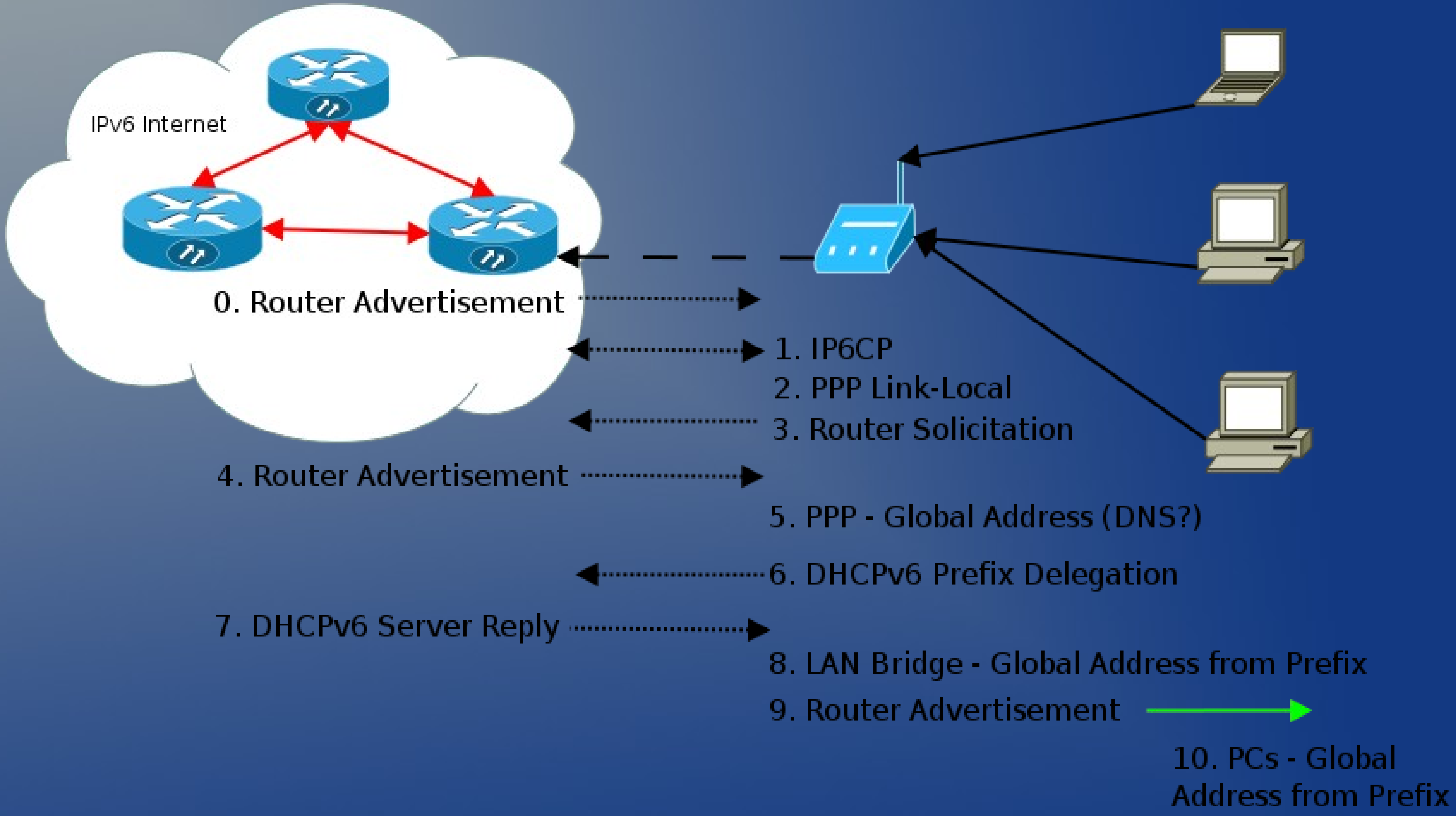
00 | 90 | 27 | 17 | FC | 0F
00 | 90 | 27
17 | FC | 0F
FF | FE
00 | 90 | 27 | FF | FE | 17 | FC | 0F
000000U0
Where U=
1 = Unique ID
0 = Not Unique
U = 1
02 | 90 | 27 | FF | FE | 17 | FC | 0F
231388

- Address (128bit) = Link Prefix (64bit) + EUI-64 (64bit)
- Privacy Extensions

IPv6 Auto-configuration (2/2)

- Stateful DHCPv6
  - Client/Server
  - Multicast UDP
  - DNS (SIP,NTP,etc)
  - Temporary (IA_TA) & non-temporary addresses (IA_NA)
  - Prefix Delegation – IA_PD (!)
- Stateless DHCPv6
  - Get IP by SLAAC - need additional parameters

IPv6 at home (1/2)



0. Router Advertisement

1. IP6CP

2. PPP Link-Local

3. Router Solicitation

4. Router Advertisement

5. PPP - Global Address (DNS?)

6. DHCPv6 Prefix Delegation

7. DHCPv6 Server Reply

8. LAN Bridge - Global Address from Prefix

9. Router Advertisement

10. PCs - Global Address from Prefix

IPv6 at home (2/2)

```
# 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
```

IPv6 DNS

- Extremely important!

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

Shell: scp kargig@\[2001:1af8:4100:a02c:1::16\]:file.ext localpath/

↑ “easy to use, right ??”

- AAAA forward (name→address)

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

- PTR reverse (address→name) ip6.arpa.

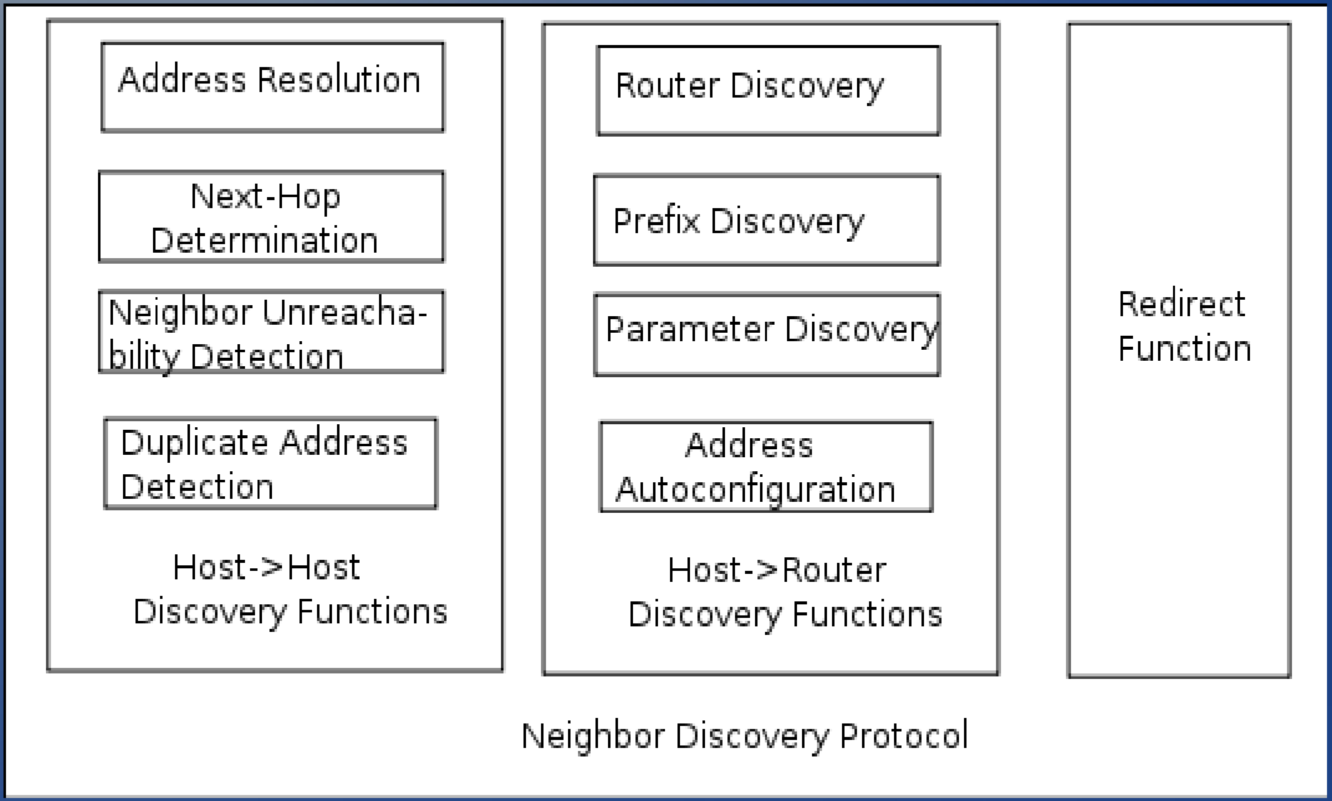
```
6.1.0.0.0.0.0.0.0.0.0.0.1.0.0.0.c.2.0.a.0.0.1.4.8.f.a.1.1.0.0.2.ip6.arpa. IN PTR void.gr
```

IPv6 Neighbor Discovery Mechanisms

- ARP is dead, long live ND
- Host-to-Host
- Host-to-Router

IPv6 ND

- Neighbors = 2 devices on the same local network
- Based on ICMPv6 → Replaces ARP + ICMP on IPv4


Address Resolution
Next-Hop Determination
Neighbor Unreachability Detection
Duplicate Address Detection
Host->Host Discovery Functions
Router Discovery
Prefix Discovery
Parameter Discovery
Address Autoconfiguration
Host->Router Discovery Functions
Redirect Function
Neighbor Discovery Protocol

IPv6 ND Host-to-Host

- **Next-Hop Determination**: The method for looking at an IP datagram's destination address and determining where it should next be sent (Destination Cache).
- **Address Resolution**: The process by which a device determines the layer two address of another device on the local network from that device's layer three (IP) address. Replaces ARP in IPv4 (Neighbor Cache).
- **Neighbor Unreachability Detection**: The process of determining whether or not a neighbor device can be directly contacted.
- **Duplicate Address Detection**: Determining if an address that a device wishes to use already exists on the network.

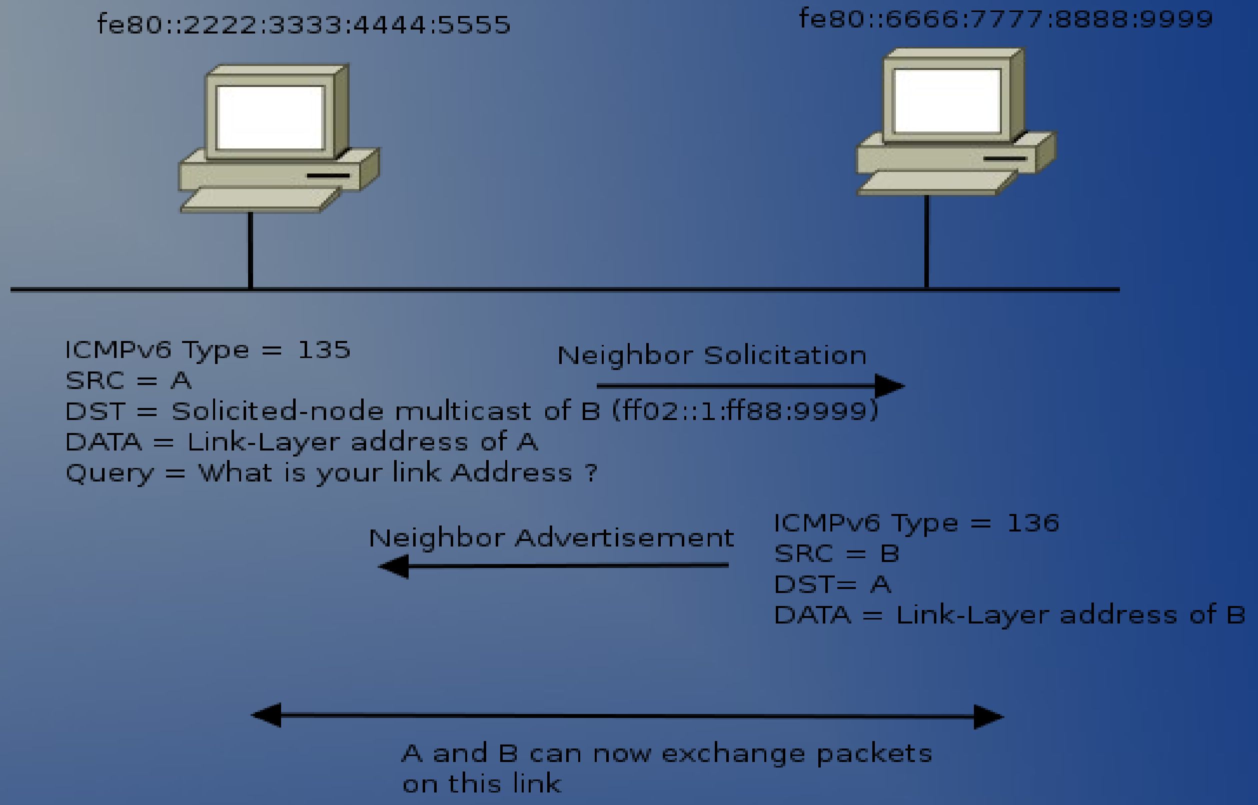
IPv6 Host-to-Router

- **Router Discovery**: The method by which hosts locate routers on their local network.
- **Prefix Discovery**: Hosts use this function to determine what network they are on, which in turn tells them how to differentiate between local and distant destinations and whether to attempt direct or indirect delivery of datagrams (Prefix Cache).
- **Parameter Discovery**: The method by which a host learns important parameters about the local network and/or routers, such as the maximum transmission unit of the local link.
- **Address Autoconfiguration**: Hosts can automatically configure themselves, by information provided by a router.

IPv6 ND Messages

- Commonly used messages:
  - Router Advertisement (Type 134)
  - Router Solicitation (Type 133)
  - Neighbor Advertisement (Type 136)
  - Neighbor Solicitation (Type 135)
  - Redirect
- Benefits:
  - Formalize Address Resolution + Router Discovery (Security at layer 3 independent of IPsec → SeND)
  - Autoconfiguration
  - Dynamic Router Selection
  - Multicast

IPv6 ND Address Resolution



fe80::2222:3333:4444:5555

fe80::6666:7777:8888:9999

ICMPv6 Type = 135
SRC = A
DST = Solicited-node multicast of B (ff02::1:ff88:9999)
DATA = Link-Layer address of A
Query = What is your link Address ?

Neighbor Solicitation

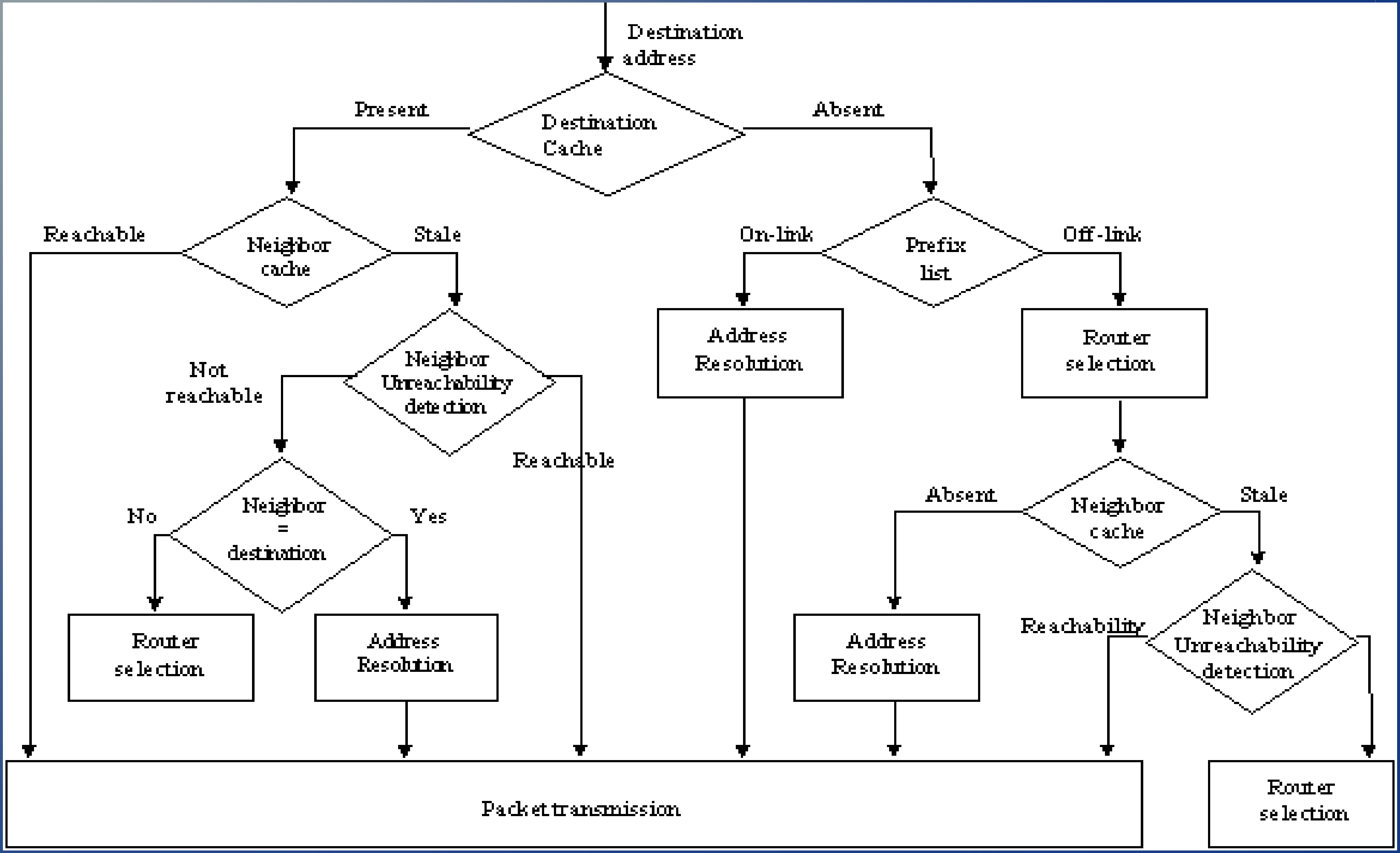
Neighbor Advertisement

ICMPv6 Type = 136
SRC = B
DST= A
DATA = Link-Layer address of B

A and B can now exchange packets
on this link

- Efficiency due to using Solicited-node Multicast Addresses instead of broadcast
- Address Resolution only for “on-link” nodes

IPv6 ND Flow


Destination address
Destination Cache
Present
Absent
Reachable
Neighbor cache
Stale
On-link
Prefix list
Off-link
Not reachable
Neighbor Unreachability detection
Reachable
Address Resolution
Router selection
No
Neighbor = destination
Yes
Absent
Neighbor cache
Stale
Router selection
Address Resolution
Address Resolution
Reachability
Neighbor Unreachability detection
Packet transmission
Router selection

IPv6 Linux

- Show stuff
- Add/Remove stuff
- Show even more stuff
- /proc
- ip6tables
- IPv6 configuration for common software

IPv6 Linux

- Show IPv6 neighbors
  - ip -6 neighbor show
- Show IPv6 addresses
  - ip -6 address
- Show IPv6 routes
  - ip -6 route

IPv6 Linux

- Add neighbor
  - ip neighbor add 2001:db8::2 dev eth0 lladdr 00:11:22:33:44:55
- Add address
  - ip address add 2001:db8::1/64 dev eth0
  - ip address del 2001:db8::1/64 dev eth0
- Add route
  - ip route add 2001:db8::10:1/64 dev eth0
  - ip route del 2001:db8::10:1/64 dev eth0

IPv6 Linux

- Show destination cache
  - ip route show cache
- Show multicast listening addresses
  - ip maddr
- Log routing changes
  - rtmon file /tmp/rtmon.log
  - ip monitor file /tmp/rtmon.log

IPv6 Linux

- /proc/
  - /proc/net/snmp6
  - /proc/sys/net/ipv6/bindv6only
  - /proc/sys/net/ipv6/conf/[all,default,devX]/YYYY
    - accept_ra
    - autoconf
    - forwarding (0,1,2)
    - accept_redirects
    - disable_ipv6 (newer kernels)
    - router_solicitations
    - mtu
    - use_tempaddr (0,1,2)

IPv6 Linux

- ip6tables instead of iptables
- no NAT table! (yet)
- New features:
  - -m eui64
  - -m ipv6header
  - -m rt

IPv6 Linux

- Apache configuration
  - Listen 80
  - Listen [2001:db8::1]:80
  - NameVirtualHost [2001:db8::1]:80
  - <VirtualHost [2001:db8::1]:80>
- vsftpd
  - listen_ipv6=YES
  - sysctl -w net.ipv6.bindv6only=0 (don't forget!)
- Postfix
  - inet_protocols = ipv4, ipv6

IPv6 Security Considerations

- Hype
- Local Network Protection
- Common Local Attacks & mitigation
- Remote Network Scanning
- Local Network Scanning
- IPv6 Migration Security
- Tools
- Food for thought
- Overview

IPv6 Hype

- IPsec is mandatory!!111oneone
- No more ARP spoofing!!11eleveneleven

IPv6 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

IPv6 Common Local Attacks

- Address Resolution
  - Attacker claims victim's IP address
- Redirect
  - Attacker sends RA and redirects traffic heading to an off-link host elsewhere
- DAD (DoS)
  - Attacker replies to any victim's DAD requests

IPv6 Common Local Attacks

- First-Hop Router Attack
  - Attacker tricks victim into accepting itself as a default router canceling the previous one (lifetime=0). Steals all traffic.
- Address Configuration (DoS)
  - Attacker cancels previous default router prefix and sends new prefix to victim. Victim can't access the network due to spoofed prefix filtering by default router.
- DHCPv6 spoofing

IPv6 Common Local Attacks Mitigation techniques

- RAguard (L2 Protection)
- Firewall/ACL to block specific rogue ICMPv6
- DHCPv6 filtering (UDP port 546/547)
- Disable autoconfiguration (when unnecessary)

Remote Network Scanning for IPv6 hosts

- Server LANs (!slaac) vs Home LANs (slaac)
- Start with first 64bits + ::1
- Try 1-255 as last 16bits of address
- 100,1000,2000,666,f00d,cafe,dead,aaaa, ffff, etc → face:b00c
- Vendor IDs
- **DNS** (zone transfers)
- Parse Logs
- Indirect (email to clients--> pic on IPv6 only host)

Local Network Scanning for IPv6 hosts

- ping6 ff02::1%eth0 All-Nodes (demo)
- ping6 ff02::2%eth0 All-Routers
- ICMPv6 139/140 Node Information Query/Response (demo)
- dig any void.gr @FF02::1 (use tcpdump/wireshark and look at the replies) (demo)
- Rogue RA/DHCPv6 (control hosts - demo)

IPv6 Migration Security

- Deny packets for transition techniques not in use
  - Deny IPv4 protocol 41 forwarding unless that is exactly what is intended – unless using 6to4 tunneling
  - Deny UDP 3544 forwarding unless you are using Teredo tunneling
- Avoid Dynamic Tunnels (6to4, Teredo, etc)

Tools

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

Food for thought

- ND flooding/fuzzing
- Unlimited size extension headers daisy-chaining (bypass RAguard example)
- Don't forget Link-Local addresses for firewalls!
- Monitor/IDS support is currently BAD
- Misconfigured firewalls (sync v4/v6 ACLs)
- SLAAC everywhere → same ending 64bits

IPv6 Security Overview

- IPv6 is no more or less secure than IPv4
  - Experience is the issue
  - Fewer tools in the wild
- IPv6 will change traffic patterns (p2p, MIPv6)
- IPv6 larger addresses makes worms and scanning less effective but there are still ways to find hosts (be creative)
- Apply IPsec wherever possible
- LAN based attacks → Stronger Physical Security, Ethernet-Port Security, NAC, 802.1X, SeND

Links

- http://www.packetlevel.ch/html/scapy/scapyipv6.html
- http://www.stindustries.net/ipv6-security/
- http://www.void.gr/kargig/ipv6/

The End

Thanks!

Any Questions ?