

IPv6 Security

János Mohácsi
NIIF/HUNGARNET

Outline of the presentation

- Threats against IPv6– comparing with IPv4
 - Scanning
 - Unauthorised access – IPv6 firewalls review
 - Fragmentation attacks
 - Spoofing
 - Host initialisation attacks
 - Broadcast amplification attacks
 - Other types of attacks
- Specific IPv6 related problems
- IPv6 Security infrastructure

Threats

Scanning

Scanning in IPv6

- Subnet Size is much larger
 - Default subnets in IPv6 have 2^{64} addresses (approx. 18×10^{18}). Exhaustive scan on every address on a subnet is no longer reasonable (if 1 000 000 address per second then $> 500\,000$ year to scan)
 - NMAP doesn't even support for IPv6 network scanning

Scanning in IPv6 /2

- IPv6 Scanning methods are likely to change
 - Public servers will still need to be DNS reachable giving attacker some hosts to attack – this is not new!
 - Administrators may adopt easy to remember addresses (::1, ::2, ::53, or simply IPv4 last octet)
 - EUI-64 address has “fixed part”
 - Ethernet card vendors guess
 - New techniques to harvest addresses – e.g. from DNS zones, logs
 - Deny DNS zone transfer
 - By compromising routers at key transit points in a network, an attacker can learn new addresses to scan
- Other possible network hiding: DNS splitting

Scanning in IPv6 / 3

- New attack vectors “All node/router addresses”
- New Multicast Addresses - IPv6 supports new multicast addresses that can enable an attacker to identify key resources on a network and attack them
- For example, all nodes (FF02::1), all routers (FF05::2) and all DHCP servers (FF05::5)
- These addresses must be filtered at the border in order to make them unreachable from the outside – this is the default if no IPv6 multicasting enabled.

Security of IPv6 addresses

- Private addresses as defined RFC 3041
 - prevents device/user tracking from
 - makes accountability harder
- New privacy extended IPv6 addresses generated CGA (cryptographically generated addresses)
 - maintains privacy
 - accountability possible by link administrators
- New feature: Host ID could be a token to access to a network. – additional security possible

Threats

Unauthorized Access

Unauthorized Access control in IPv6

- Policy implementation in IPv6 with Layer 3 and Layer 4 is still done in firewalls
- Some design considerations! – see next slides

Action	Src	Dst	Src port	Dst port
permit	a:b:c:d::e	x:y:z:w::v	any	ssh
deny	any	any		

Unauthorized Access control in IPv6

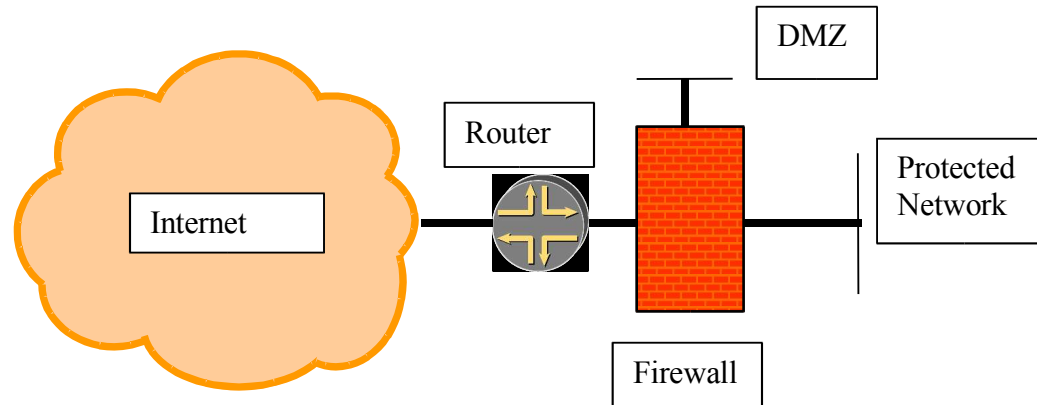
- non-routable + bogon address filtering slightly different
 - in IPv4 easier deny non-routable + bogon
 - in IPv6 easier to permit legitimate (almost)

Action	Src	Dst	Src port	Dst port
deny	2001:db8::/32	host/net		
permit	2001::/16	host/net	any	service
permit	2002::/16	host/net	any	service
permit	2003::/16	host/net	any	service
permit	3ffe::/16	host/net	any	service
deny	any	any		

IPv6 Firewalls

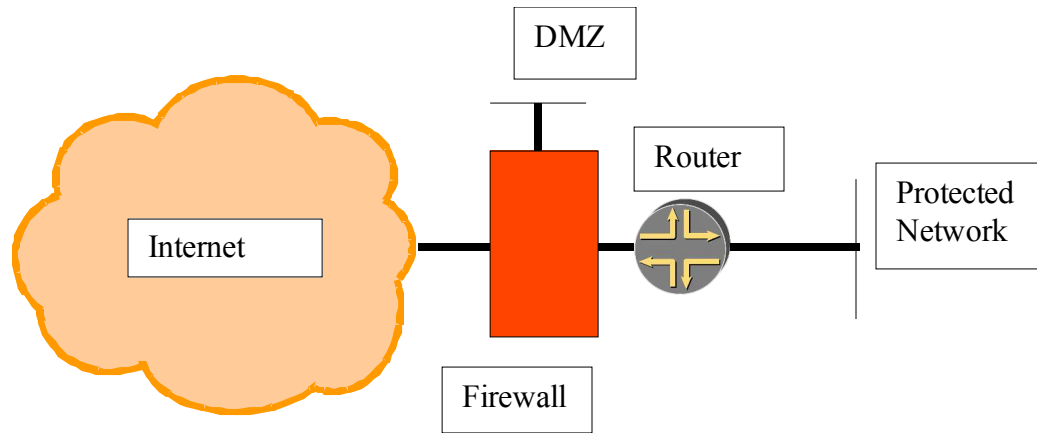
- IPv6 architecture and firewall - requirements
 - No need to NAT – same level of security with IPv6 possible as with IPv4 (security and privacy) – even better: e2e security with IPSec
 - Weaknesses of the packet filtering cannot be made hidden by NAT
 - Support for IPv6 header chaining
 - Support for IPv4/IPv6 transition and coexistence
 - Not breaking IPv4 security

IPv6 firewall setup - method 1



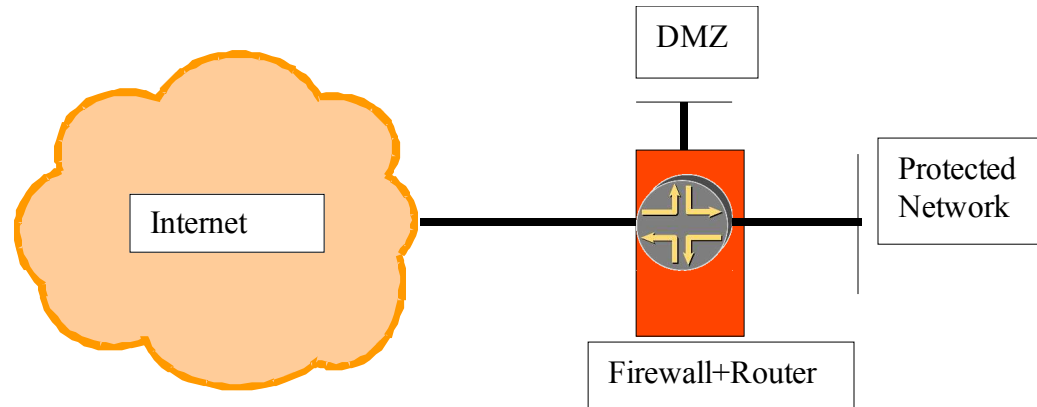
- Internet ↔ router ↔ firewall ↔ net architecture
- Requirements:
 - Firewall must support/recognise ND/NA filtering
 - Firewall must support RS/RA if SLAAC is used
 - Firewall must support MLD messages if multicast is required

IPv6 firewall setup - method2



- Internet ↔ firewall ↔ router ↔ net architecture
- Requirements:
 - Firewall must support ND/NA
 - Firewall should support filtering dynamic routing protocol
 - Firewall should have large variety of interface types

IPv6 firewall setup - method3



- Internet ↔ firewall/router(edge device) ↔ net architecture
- Requirements
 - Can be powerful - one point for routing and security policy – very common in SOHO (DSL/cable) routers
 - Must support what usually router AND firewall do

Firewall setup

- No blind ICMPv6 filtering possible:

	Echo request/reply	Debug
	No route to destination	Debug – better error indication
	TTL exceeded	Error report
	Parameter problem	Error report
IPv6 specific	NS/NA	Required for normal operation – except static ND entry
	RS/RA	For Stateless Address Autoconfiguration
	Packet too big	Path MTU discovery
	MLD	Requirements in for multicast in architecture 1

Firewall setup 2

- No blind IP options (→ extension Header) filtering possible:

Hop-by-hop header	What to do with jumbograms or router alert option? – probably log and discard – what about multicast join messages?
Routing header	Source routing – in IPv4 it is considered harmful, but required for IPv6 mobility – log and discard if you don't support MIPv6, otherwise enable only for Home Agent of MIPv6
ESP header	Process according to the security policy
AH header	Process according to the security policy
Fragment header	All but last fragments should be bigger than 1280 octets

IPv6 firewalls and addresses

address management/2

- Abusing/virus infected host should be tracked back
 - SLAAC
 - possible from IP(InterfaceID) →MAC address if RFC 3041 (privacy addresses) not used! – think twice allowing RFC3041!
 - Firewall could break global connection if IP(InterfaceID) ↔ MAC address
 - DHCPv6
 - You have the DHCPv6 lease database – do you have lease data for last Monday 2:00 AM?
 - Static
 - Be careful with filtering
 - Where Firewall could help – detect inconsistent ND cache – MAC changes – more like NIDS – this kind of protection for IPv4 - existing on Cisco switches

Interoperability of filtered applications

- FTP:
 - Very complex: PORT, LPRT, EPRT, PSV, EPSV, LPSV (RFC 1639, RFC 2428)
 - virtually no support in IPv6 firewalls
 - HTTP seems to be the next generation file transfer protocol with WEBDAV and DELTA
- Other non trivially proxy-able protocol:
 - no support

Evaluation of IPv6 firewalls:

IPfilter 4.1

- clean architecture, powerful filtering, quite portable (*BSD/Solaris/HP-UX/IRIX/True64 Unix)
 - problems:
 - ICMPv6 support is rudimentary- RS/RA and NS/NA pairing is ok (no support for IPv6 defined error conditions); PSV/EPSV ftp proxy support but only for IPv4; FreeBSD contains it by default (stable only)
 - good things:
 - IPv6 extension header matching support; fault tolerant setup possible; checksum checking; TCP sequence randomization possible; quite complete architecture; well documented, performance degradation very acceptable, throttling possible with PPS

Evaluation of IPv6 firewalls: pf 3.5

- Ideas from ipfilter: clean architecture, more powerful filtering, not so portable (*BSD)
 - problems:
 - No fragment handling –blocked!, PSV/EPSV ftp proxy support but only for IPv4; OpenBSD/post FreeBSD5.2 contains it by default
 - good things:
 - Macro support for easier readable rules; one command antispoofing; ICMPv6 support almost OK – RS/RA and NS/NA pairing, ICMPv6 code specified numerically/symbolically (however no support for IPv6 defined error conditions); IPv6 extension header matching support; checksum checking; TCP sequence/IP ID randomization possible; quite complete architecture; well documented, performance degradation acceptable, QoS possible with ALTQ

Evaluation of IPv6 firewalls:

IP6fw

- clean architecture, good filtering, medium portability (*BSD only)
 - problems:
 - architecture not too modern (however IPv4 ipfw2 has some novelties over ip-filter/pf e.g. dumynet), no proxy support at all, antispoofing setup tedious (not in ipfw2), UDP/ICMPv6 is weakly supported
 - good:
 - IPv6 extension header (not extensive), autoconfiguration is supported but needs polishing, some fragment support, *BSD contain them with predefined filtering rules – even against some transition abuses

Evaluation of IPv6 firewalls:

Netfilter

- complex architecture, good filtering, weak portability (Linux only)
 - problems:
 - development version (available only in patch-o-matic not in patch-o-matic-ng), stateless filtering only for IPv6 – connection tracking under development, proxy only via extra kernel programming,
 - good:
 - extensive ICMPv6 support, Autoconfigured EUI-64 address usage check, some autoconfiguration check possible, IPv6 extension header support (even AH/ESP SPI), Rather comprehensive routing/fragment header support, extensive development, good extensible architecture

Evaluation of IPv6 firewalls:

Cisco access list

- good architecture, good filtering, Cisco only (as of IOS 12.3(7)T):
 - two sets: standard ACL, extended ACL
- problems:
 - Could filter only the next header value – content not, only 6 bit from Traffic class, only named ACLs, IPv4 ACL and an IPv6 ACL cannot share the same name, uRPF is not available universally
- good:
 - Implicit deny, SLAAC and ND supported, some extension header support, some fragment support, possible disabling redirect, unknown transport support, ICMPv6 support, reflexive ACL support, time-based ACL, commercially supported, performance can be different on many platform, FTP inspection is available since 12.3(11)T
- Note: IPv6 ACL has implicit? `permit icmp any any nd-na,`
`permit icmp any any nd-ns, and deny ipv6 any any`

Evaluation of IPv6 firewalls:

Cisco PIX firewall

- good architecture, good filtering, Cisco only (not available yet) – will be available in PIX 7.0
- Supported features:
 - Recognition of IPv6 traffic
 - Dual-stack (IPv6/4 combined), IPv6 only, IPv4 only
 - Extended IP ACLs
 - interface, SOURCE ip/port, DEST ip/port, protocol
 - Stateful inspection
 - Application awareness: HTTP, FTP, telnet, SMTP, TCP, SSH, UDP
 - URPF and Frag Guard
 - IPv6 header security checks
 - Reassembly in v6 packets?
 - Static routing
 - ICMPv6 support (with Neighbor discovery support)

Evaluation of IPv6 firewalls:

Juniper firewall

- good architecture, good filtering, Juniper only (as JunOS 6.2):
- problems:
 - Could filter only the next header value, IPv4 filter and an IPv6 filter cannot share the same name, no match for TCP flags in IPv6 filters
- good:
 - No performance impact to enable IPv6 firewall, Some ND supported, some extension header support, some fragment support, ICMPv6 support, commercially supported – what will NetScreen provide?

Evaluation of IPv6 firewalls:

Juniper NetScreen firewall

- good architecture, good filtering, Juniper only – available since ScreenOS 5.0
- Supported features:
 - Dual-stack (IPv6/4 combined), IPv6 only, IPv4 only
 - PPPoEv6 support
 - Extended IP firewall
 - interface, SOURCE ip/port, DEST ip/port, protocol
 - Stateful inspection
 - DoS protection
 - IPv6 IPsec VPN support
 - IPv6 header security checks
 - Static routing and RIPng (OSPFv3 and BGP soon)
 - ICMPv6 support (with Neighbor discovery support)
 - Tunneling and NAT-PT support
 - Traffic prioritisation

Evaluation of IPv6 firewalls:

Windows XP SP2 firewall

- average architecture, mediocre filtering, Windows only
- problems:
 - Host only firewall. Cannot filter on the next header value, no ICMPv6 code filtering – to support ND, no match for TCP flags in IPv6 filters, no stateful capability, incoming filtering only
- good:
 - IPv4 filter and an IPv6 filter shares the same rule, ICMPv6 support, Important separate rule for path MTU discovery, per process configuration possible, commercially supported, graphical interface is there but not everything configurable graphically

Threats

Fragmentation and header handling

Header Manipulation and Fragmentation Best Practices

- Deny IPv6 fragments destined to an internetworking device - Used as a DOS vector to attack the infrastructure
- Ensure adequate IPv6 fragmentation filtering capabilities. For example, drop all packets with the routing header if you don't have MIPv6
- Potentially drop all fragments with less than 1280 octets (except the last fragment)
- All fragment should be delivered in 60 seconds otherwise drop

Threats

L3-L4 spoofing

L3- L4 Spoofing in IPv6

- While L4 spoofing remains the same, IPv6 address are globally aggregated making spoof mitigation at aggregation points easy to deploy
- Can be done easier since IPv6 address is hierarchical
- However host part of the address is not protected
 - You need IPv6 \leftrightarrow MAC address (user) mapping for accountability!

Threats

IPv4 ARP and DHCP attacks -
Subverting host initialization

Autoconfiguration/Neighbor Discovery

- Neighbor Discovery ~ security ~ Address Resolution Protocol
 - No attack tools – arp cache poisoning
 - No prevention tools – dhcp snooping
- Better solution with SEND
 - based on CGA: $\text{token1} = \text{hash}(\text{modifier}, \text{prefix}, \text{publickey}, \text{collision-count})$
 - IPR issues with Ericsson and Microsoft...
- DHCPv6 with authentication is possible
- ND with IPSec also possible

Threats

Broadcast amplification

Amplification (DDoS) Attacks

- There are no broadcast addresses in IPv6
 - This would stop any type of amplification/"Smurf" attacks that send ICMP packets to the broadcast address
 - Global multicast addresses for special groups of devices, e.g. link-local addresses, site-local addresses, all site-local routers, etc.
- IPv6 specifications forbid the generation of ICMPv6 packets in response to messages to global multicast addresses (exception Packet too big message – it is questionable practice).
 - Many popular operating systems follow the specification
 - Still uncertain on the danger of ICMP packets with global multicast source addresses

Mitigation of IPv6 amplification

- Be sure that your host implementation follow the RFC 2463
- Implement RFC 2827 ingress filtering
- Implement ingress filtering of IPv6 packets with IPv6 multicast source address

Other threats

- IPv6 Routing Attack
 - Use traditional authentication mechanisms for BGP and IS-IS.
 - Use IPsec to secure protocols such as OSPFv3 and RIPng
- Viruses and Worms
- Sniffing
 - Without IPsec, IPv6 is no more or less likely to fall victim to a sniffing attack than IPv4
- Application Layer Attacks
 - Even with IPsec, the majority of vulnerabilities on the Internet today are at the application layer, something that IPsec will do nothing to prevent
- Man-in-the-Middle Attacks (MITM)
 - Without IPsec, any attacks utilizing MITM will have the same likelihood in IPv6 as in IPv4
- Flooding
 - Flooding attacks are identical between IPv4 and IPv6

Threats are there – there are possible attacker tools

- Sniffers/packet capture: Snort, TCPdump, Sun Solaris snoop, COLD, Ethereal, Analyzer, Windump, WinPcap, NetPeek, Sniffer Pro
- Worms: Slapper – DDos module only
- Scanners: IPv6 Security Scanner, Halfscan6, Nmap, Strobe, Netcat, Nessus
- DoS Tools: 6tunneldos, 4to6ddos, Imps6-tools
- Packet forgers: SendIP, Packit, Spak6, netwib/netwox

Specific IPv6 related problems

Specific IPv6 related threats

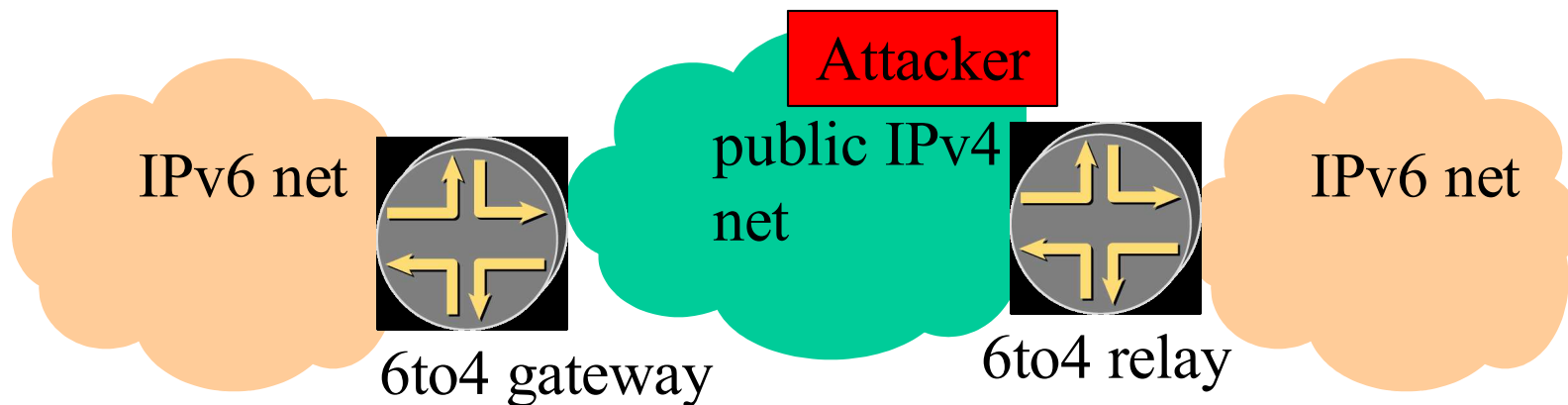
Transition Mechanisms

IPv6 transition mechanisms

- ~15 methods possible in combination
- Dual stack:
 - enable the same security for both protocol
- Tunnels:
 - ip tunnel – punching the firewall (protocol 41)
 - gre tunnel – probable more acceptable

L3 – L4 Spoofing in IPv4 with 6to4

- For example, via 6to4 tunneling spoofed traffic can be injected from IPv4 into IPv6.
 - IPv4 Src: Spoofed IPv4 Address
 - IPv4 Dst: 6to4 Relay Anycast (192.88.99.1)
 - IPv6 Src: 2002:: Spoofed Source
 - IPv6 Dst: Valid Destination



Mixed IPv4/IPv6 environments

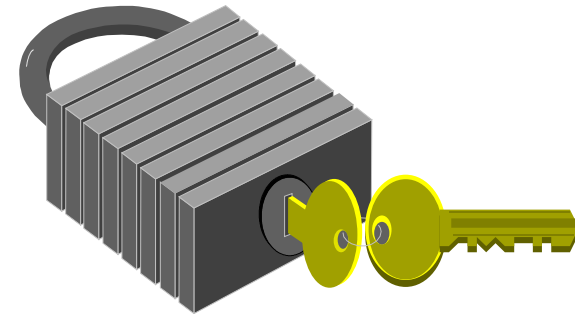
- There are security issues with the transition mechanisms
 - Tunnels are extensively used to interconnect networks over areas supporting the “wrong” version of protocol
 - Tunnel traffic many times has not been anticipated by the security policies. It may pass through firewall systems due to their inability check two protocols in the same time
- Do not operate completely automated tunnels
 - Avoid “translation” mechanisms between IPv4 and IPv6, use dual stack instead
 - Only authorized systems should be allowed as tunnel endpoints
 - Automatic tunnels can be secured by IPsec

IPv6 security infrastructure

- IPsec
- Firewalls
- AAA
 - Radius only -> Diameter?
 - TACACS+ - no plan

IPv6 Security infrastructure

IPSec



- general IP Security mechanisms
- provides
 - authentication
 - confidentiality
 - key management - requires a PKI infrastructure (IKE) – new simplified and unified IKEv2 will be available soon.
- applicable to use over LANs, across public & private WANs, & for the Internet
- IPSec is not a single protocol. Instead, IPSec provides a set of security algorithms plus a general framework that allows a pair of communicating entities to use whichever algorithms provide security appropriate for the communication.

IPSec

- Not a panacea at all, but could be, protection at IP level,
- Very flexible, broad range of algorithms could be used
- Theoretically good, most of the application already solved the problem in a different level (ssl/tls, ssh, etc.)
- Most popular applications are VPNs in a tightly controlled environment
- IPSec is mandated in IPv6 – you can rely on for e2e security



IPv6 Security infrastructure

Firewalls

Cisco IOS IPv6 Access Control Lists

Slides from Patrick Grossetete
Cisco IOS IPv6 Product Manager
Pgrosset@cisco.com

Cisco IOS IPv6 Standard Access

Control Lists

- Cisco IOS IPv6 access-lists are used to filter traffic and restrict access to the router. IPv6 prefix-lists are used to filter routing protocol updates.
- IPv6 Standard ACL (Permit/Deny)
 - IPv6 source/destination addresses
 - IPv6 prefix-lists
 - On Inbound and Outbound interfaces
- Minimum Cisco IOS releases
 - Cisco IOS 12.2(2)T or 12.3(1)M
 - Cisco IOS 12.0(21)ST1 and Cisco 12.0(22)S on Cisco 12000 series only
 - Cisco 12.2(14)S

Cisco IOS IPv6 Extended ACL

- Adds support for IPv6 option header and upper layer filtering
- Only named access-lists are supported for IPv6
- IPv6 and IPv4 ACL functionality
 - Implicit deny any any as final rule in each ACL.
 - A reference to an empty ACL will permit any any.
 - ACLs are NEVER applied to self-originated traffic.
- Minimum Cisco IOS releases
 - Cisco IOS 12.2(13)T or 12.3(1)M
 - Cisco 12.0(23)S on Cisco 12000 series only, 12.0(25)S adds hardware assisted ACL on Engine 3
 - Cisco 12.2(14)S

Cisco IOS IPv6 Extended ACL overview

- CLI mirrors IPv4 extended ACL CLI
- Implicit permit rules, enable neighbor discovery
- ULP, DSCP, flow-label,... matches
- Logging
- Time-based
- Reflexive
- CEFv6 and dCEFv6 ACL feature support
- Extended ACL can apply even if option headers are in a packet

Cisco IOS IPv6 ACL Implicit Rules

- Implicit permit rules, enable neighbor discovery
 - The following implicit rules exist at the end of each IPv6 ACL to allow ICMPv6 neighbor discovery:
 - permit icmp any any nd-na
 - permit icmp any any nd-ns
 - deny ipv6 any any
 - Be careful, when you add “deny ipv6 any any log” at the end

Cisco IOS IPv6 Extended ACL Match

- TCP/UDP/SCTP and ports (eq, lt, gt, neq, range)
- ICMPv6 code and type
- Fragments
- Routing Header
- Undetermined transport
 - The first unknown NH can be matched against (numerically rather than by name).
 - Since an unknown NH cannot be traversed, the ULP cannot be determined.

Cisco IOS IPv6 Extended ACL

- Logging
 - (conf-ipv6-acl)# permit tcp any any log-input
 - (conf-ipv6-acl)# permit ipv6 any any log
- Time based
 - (conf)# time-range bar
 - (conf-trange)# periodic daily 10:00 to 13:00
 - (conf-trange)# ipv6 access-list tin
 - (conf-ipv6-acl)# deny tcp any any eq www time-range bar
 - (conf-ipv6-acl)# permit ipv6 any any

Cisco IOS IPv6 ACL Reflexive

- Reflect
 - A reflexive ACL is created dynamically, when traffic matches a permit entry containing the reflect keyword.
- Evaluate
 - Apply the packet against a reflexive ACL.
 - The implicit deny any any rule does not apply at the end of a reflexive ACL; matching continues after the evaluate in this case.

Cisco IOS IPv6 ACL CLI (1)

- Entering address-family sub-mode
 - [no] ipv6 access-list <name>
 - Add or delete an ACL.
- IPv6 address-family sub-mode
 - [no] permit | deny ipv6 | <protocol> any | host <src> | src/len [sport] any | host <dest> | dest/len [dport] [reflect <name> [timeout <secs>]] [fragments] [routing] [dscp <val>] [flow-label <val>][time-range <name>] [log | log-input] [sequence <num>]
 - Permit or deny rule defining the acl entry. Individual entries can be inserted or removed by specifying the sequence number.
 - Protocol is one of TCP, UDP, SCTP, ICMPv6 or NH value.

Cisco IOS IPv6 ACL CLI (2)

- [no] evaluate
- Evaluate the dynamically created acl via the permit reflect keyword.
- [no] remark
- User description of an ACL.
- Leaving the sub-mode
 - exit
- Showing the IPv6 ACL configuration
 - # show ipv6 access-list [name]
 - # show access-list [name]
- Clearing the IPv6 ACL match count
 - # clear ipv6 access-list [name]
 - # clear access-list [name]

Cisco IOS IPv6 ACL CLI (3)

- Applying an ACL to an interface
 - (config-int)# ipv6 traffic-filter <acl_name> in | out
- Restricting access to the router
 - (config-access-class)# ipv6 access-class <acl_name> in | out
- Applying an ACL to filter debug traffic
 - (Router)# debug ipv6 packet [access-list <acl_name>] [detail]

End

- Further information:
 - <http://www.cisco.com/go/ipv6>

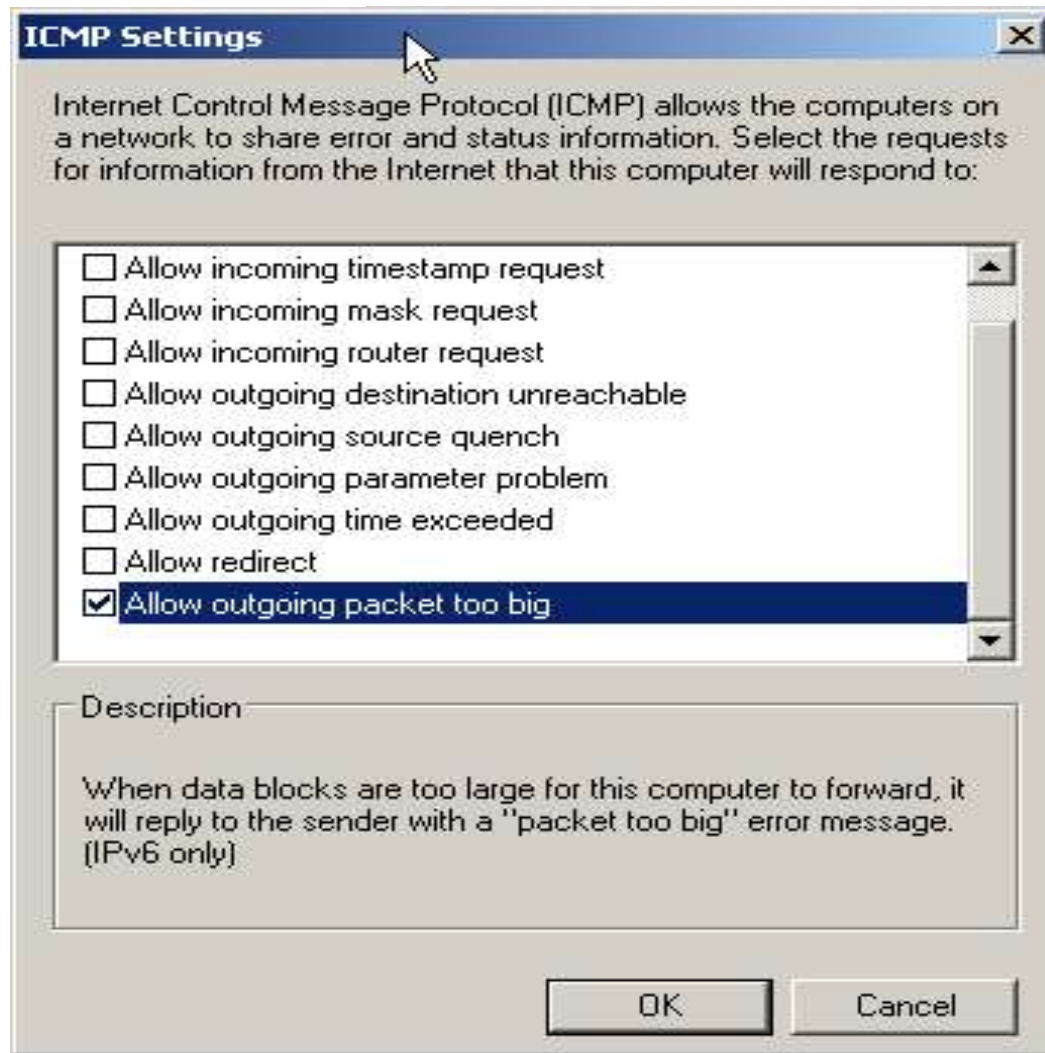
Configuring IPv6 Firewalls with Windows XP SP2

János Mohácsi
NIIF/HUNGARNET

Windows XP Firewall configuration

- Windows XP ICF – same rules for IPv4 and IPv6
 - Show configuration:
 - `netsh firewall show globalport`
 - `netsh firewall show adapter`
 - Set configuration
 - `set globalport [port#=enable|disable] [name=name]`
 - `set adapter [name] [icmp type#=enable|disable] [port port#=enable|disable [name=name] [protocol=tcp|udp]]`
 - `set logging [filelocation=<location>] [filesize=integer]`
 - `[droppedpackets=enable|disable]`
 - `[successfulconnections=enable|disable]`
- After SP2
 - in the firewall you can configure Path MTU discovery support
 - per process configuration possible

Windows SP2 ICF



References

- 6NET D3.5.1: Secure IPv6 Operation: Lessons learned from 6NET
- J. Mohacsi, “IPv6 firewalls”, presentation on the 5th TF-NGN meeting, October 2001 available at http://skye.ki.iif.hu/~mohacsi/athens_tf_ngn_ipv6_firewalls.pdf
- J.Mohacsi, “Security of IPv6 from firewalls point of view”, presentation on TNC2004 conference, June 2004, available at http://www.terena.nl/conferences/tnc2004/programme/presentations/show.php?pres_id=115
- 6NET D6.2.2: Operational procedures for secured management with transition mechanisms
- S. Convery, D Miller, IPv6 and IPv4 Threat Comparison and Best-Practice Evaluation (v1.0)", presentation at the 17th NANOG, May 24, 2004

Thank you!

- Acknowledgement to Patrick Grossetete, Stig Veenas, Ladislav Lhotka, Jerome Durand, Tim Chown, Gunter van de Velde and Eric Marin for their comments.
- Further informations:
 - <http://www.6net.org> And <http://6net.niif.hu>
- Questions: mohacsi@niif.hu