

# **IPv6 Security from point of view firewalls**

**János Mohácsi**

**09/June/2004**

**János Mohácsi, Research Associate, Network Engineer  
NIIF/HUNGARNET**

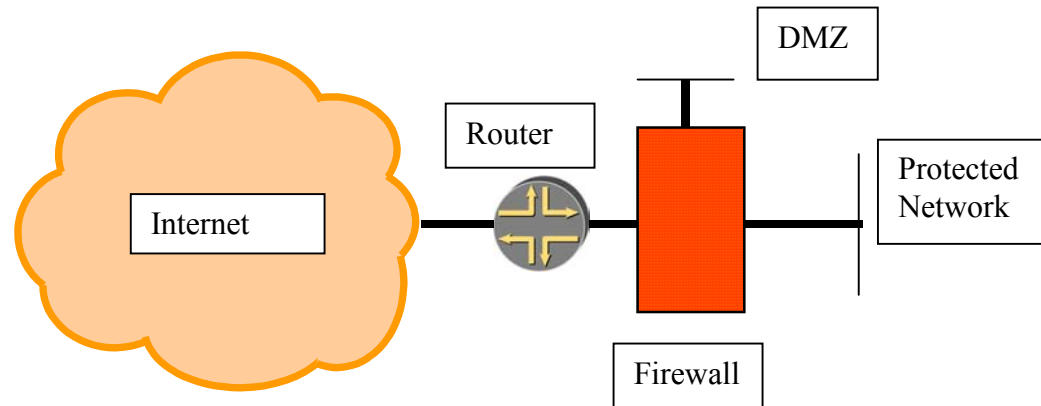
# Contents

- Requirements
- IPv6 firewall architectures
- Firewalls and addresses
- Application support in firewalls
- Survey of IPv6 firewalls
- On going work

# IPv6 Firewalls

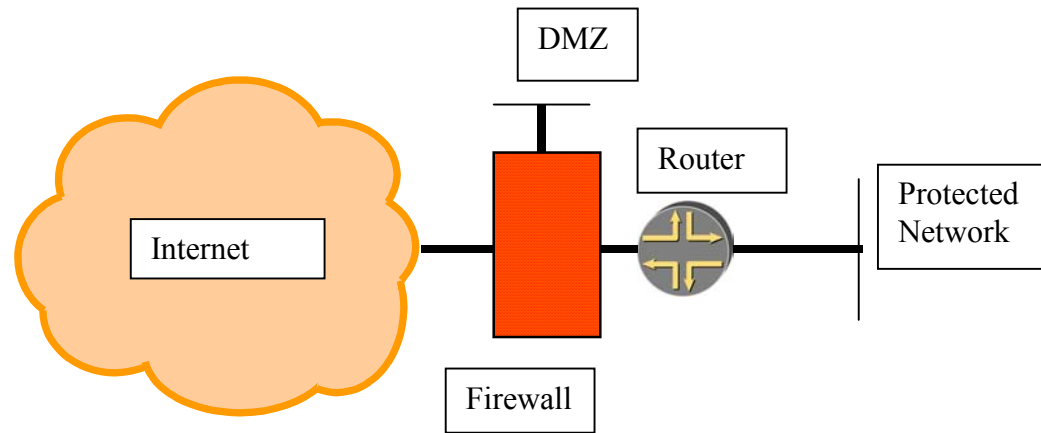
- Next generation Internet:
  - Hype or Security should be better than currently
- IPv6 architecture and firewall
  - No need to NAT
  - Network scanning virtually not possible (/64)
    - Deny DNS zone transfer
  - Other possible network hiding: DNS splitting
  - Weaknesses of the packet filtering cannot be make hidden by NAT
  - Support for IPv6 header chaining
  - Support for IPv4/IPv6 transition and coexistence
  - Not breaking IPv4 security

# IPv6 firewall setup - method 1



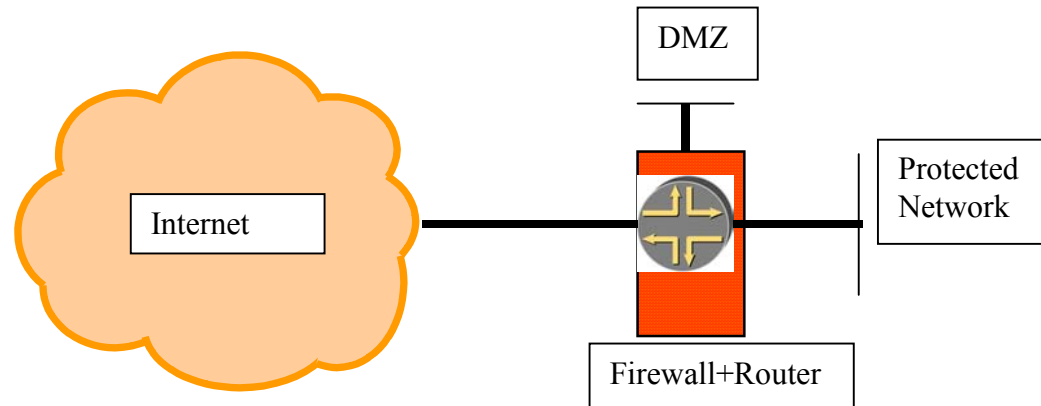
- Internet ↔ router ↔ firewall ↔ net architecture
- Requirements:
  - Firewall must support/recognise ND/NA filtering
  - Firewall must support RS/RA if SLAAC is used
  - Firewall must support MLD messages if multicast is required

# 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	required	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
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



## Firewall setup 3

- RFC2827 ingress filtering – same as IPv4 – can be done easier since IPv6 address is hierarchical
- Broadcast amplification ? Smurf ? – RFC2463 explicitly disallow this
- If architecture 2 used with dynamic routing with IPSEC– routing updates cannot be filtered – potential attack?

# IPv6 firewalls and addresses

- Current practice of address usage:
  - global addresses – global communication
  - link local addresses – link communication
  - NO site local addresses – but globally unique local addresses is under study
- General rule:
  - local addresses - allow - (supposing routers are operating correctly)
  - filter according to the security policy for global addresses – make reachable only that is necessary

# 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 not used
    - 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

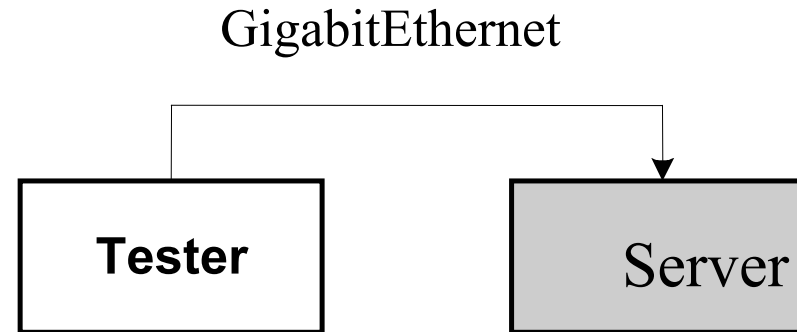
- FTP:
  - Very complex: PORT, LPRT, EPRT, PSV, LPSV, EPSV
  - virtually no support in IPv6 firewalls
  - HTTP seems to be the next generation file transfer protocol with WEBDAV and DELTA
- Other non trivially proxy-able protocol:
  - no support

# Evaluation of IPv6 firewalls:

## IPfilter 4.1

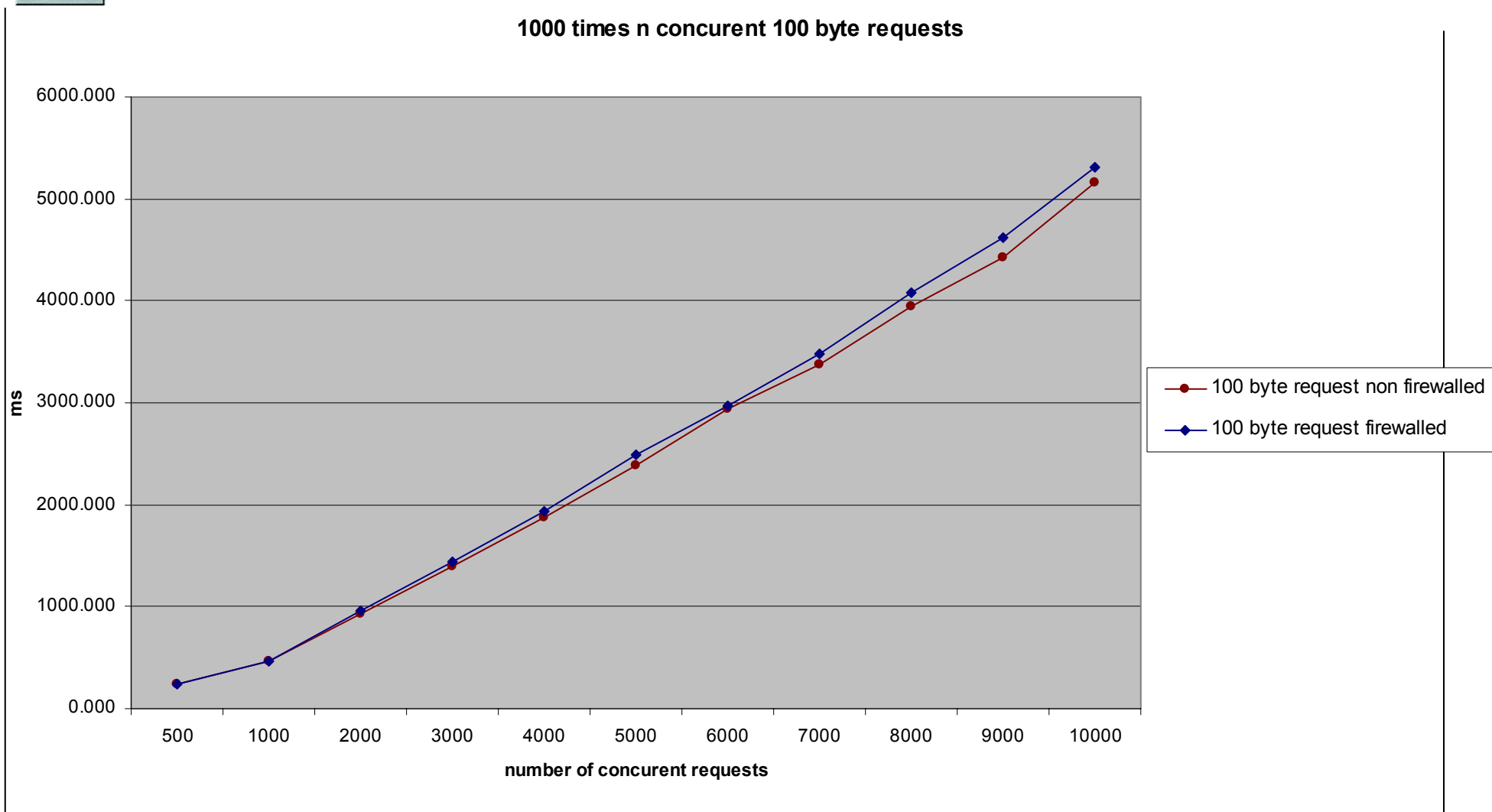
- clean architecture, powerful filtering, quite portable (\*BSD/Solaris/HP-UX/IRIX/True64 Unix)
  - problems:
    - ICMPv6 support is rudimentary- RS/RA and NS/NA pairing is ok (no support for IPv6 defined error conditions); PSV/EPSV ftp proxy support but only for IPv4; FreeBSD contains it by default (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

# Performance of IPfilter

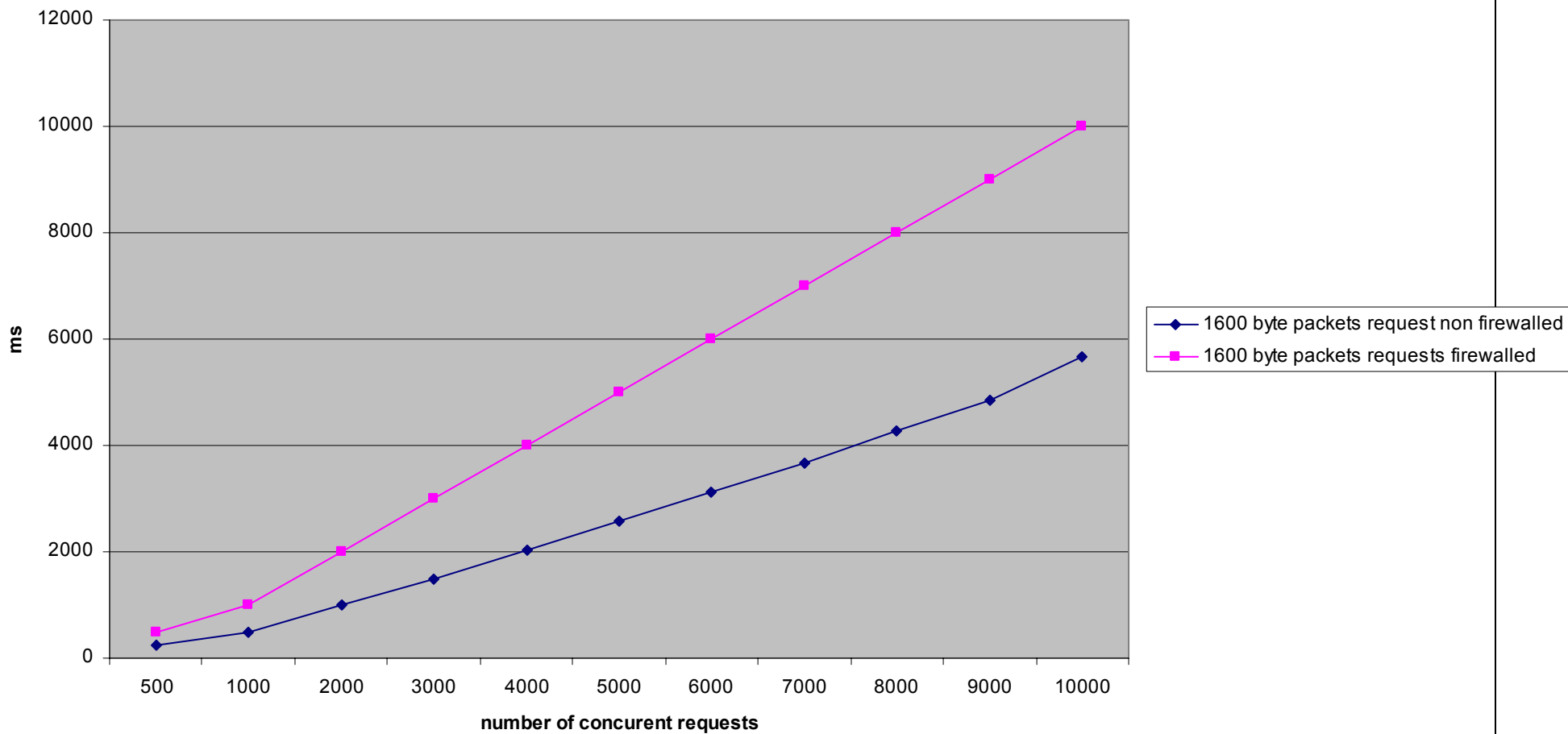


- Server system: FreeBSD 4.10PRE –only MAXUSERS tuned to 1024 – thttpd server
- Tester: Linux – heavily tuned – max openfiles hack – apachebench
- TCP stateful filtering with different number of concurrent request – to test capabilities of IPv6 stateful filtering

### 1000 times n concurrent 100 byte requests



### 1000 times n concurrent 1600 byte packets requests





# 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 antispoofting; 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

- complex 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, 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,
- 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:

## Juniper firewall

- complex 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?

# Conclusion + Future

- IPv6 firewalls are existing
- They are start to be rather usable
- They can be used for firewalling an IPv6 network
- Commercial support is available
- Transition problems – some work in 6NET WP2
- Mobile IPv6 – can be problematical.