

# EMPEROR : ICMPv6 P2P communication without 3<sup>rd</sup> party

Nikolaos Tsapakis, George Tselos

# Whoami

**Nikolaos Tsapakis** is a reverse engineering enthusiast and poetry lover from Greece. He is working as a Security Engineer. He has been writing papers or presented for Virus Bulletin, 2600, LeHack, Symantec, Hakin9, Athcon, DeepSec.

**George Tselos** is a computer science tutor who lives and works in Athens, Greece. He is interested in embedded systems, microcontrollers and peripheral device development.

# The problem

- P2P communication needs a 3rd party service to initiate like a stun server which provides NAT traversal
- Routers and NATs in network infrastructure block direct access to ports exposed on the internet
- Example of services that use 3rd party server are Skype, Zoom, Viber
- Why having intermediates ?
- <https://www.cyberyodha.org/2023/04/what-is-stun-protocol.html>

# The idea

- (RFC 4890) Recommendations for Filtering ICMPv6 Messages in Firewalls
- Traffic that must not be dropped
- Do manufacturers for routers and firewalls respect by default that RFC ?

# ICMPv6

- The unique IPv6 address makes the need for NAT traversal obsolete
- Protocol does not use ports
- Enough space for the payload
- Python 3 support

# Traffic that must not be dropped

Davies & Mohacsi

Informational

[Page 13]

[RFC 4890](#)

ICMPv6 Filtering Recommendations

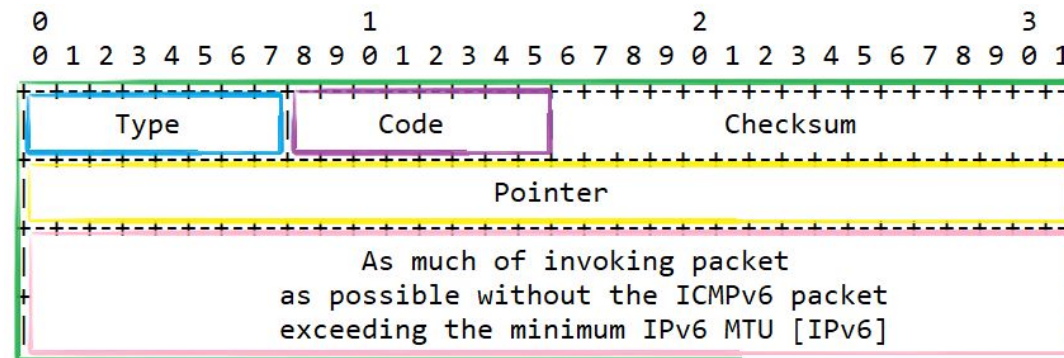
May 2007

## **4.3.1. Traffic That Must Not Be Dropped**

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

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

# Message format (RFC)



IPv6 Fields:

Destination Address

Copied from the Source Address field of the invoking packet.

ICMPv6 Fields:

Type	4
Code	<ul style="list-style-type: none"> <li>0 - Erroneous header field encountered</li> <li>1 - Unrecognized Next Header type encountered</li> <li>2 - Unrecognized IPv6 option encountered</li> </ul>
Pointer	Identifies the octet offset within the invoking packet where the error was detected.

## Message format (python)

```
# IPv6 packet
IPv6_icmp = IPv6()
IPv6_icmp.src = src
IPv6_icmp.dst = dst

# ICMPv6 following
IPv6_icmp.nh = 58

# ICMPv6 ParamProblem message type
ICMPv6 ParamProblem = ICMPv6ParamProblem()
ICMPv6 ParamProblem.type = 4
ICMPv6 ParamProblem.code = 1
ICMPv6 ParamProblem.ptr = 0

# Set verbose to False to avoid printing information messages
packet = IPv6_icmp, ICMPv6 ParamProblem, Raw(load=data)
send(packet, count=1, verbose=False)
```



# Message format (network)

> Frame 171: 112 bytes on wire (896 bits), 112 bytes captured (896 bits) on interface \Device\NPF\_{1ACA4E8A-5909-48D9-AA6E-FB5EC24FA1AC}

> Ethernet II, Src: TPLink\_28:e3:92 (7c:c2:c6:28:e3:92), Dst: HonHaiPrecis\_a4:a4:41 (48:e2:44:a4:a4:41)

Internet Protocol Version 6, Src: 2a02:586:a502:32b3:369f:810:cfe7:fb2a, Dst: 2a02:586:a502:32b3:467d:2fe1:df03:3c9

0110 .... = Version: 6

> .... 0000 0000 .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)

... 0000 0000 0000 0000 0000 = Flow Label: 0x000000

Payload Length: 58

Next Header: ICMPv6 (58)

Hop Limit: 64

> Source Address: 2a02:586:a502:32b3:369f:810:cfe7:fb2a

> Destination Address: 2a02:586:a502:32b3:467d:2fe1:df03:3c9

[Stream index: 1]

Internet Control Message Protocol v6

Type: Parameter Problem (4)

Code: 1 (unrecognized Next Header type encountered)

Checksum: 0x8682 [correct]

[Checksum Status: Good]

Pointer: 0

Data (50 bytes)

Data: beef92ff5b7be2935cc5420153544c2d2b304d8fdbfd8d97750a7cc4575da7ce5ab00e431604491ddc7348f102a9e2eb8ef9

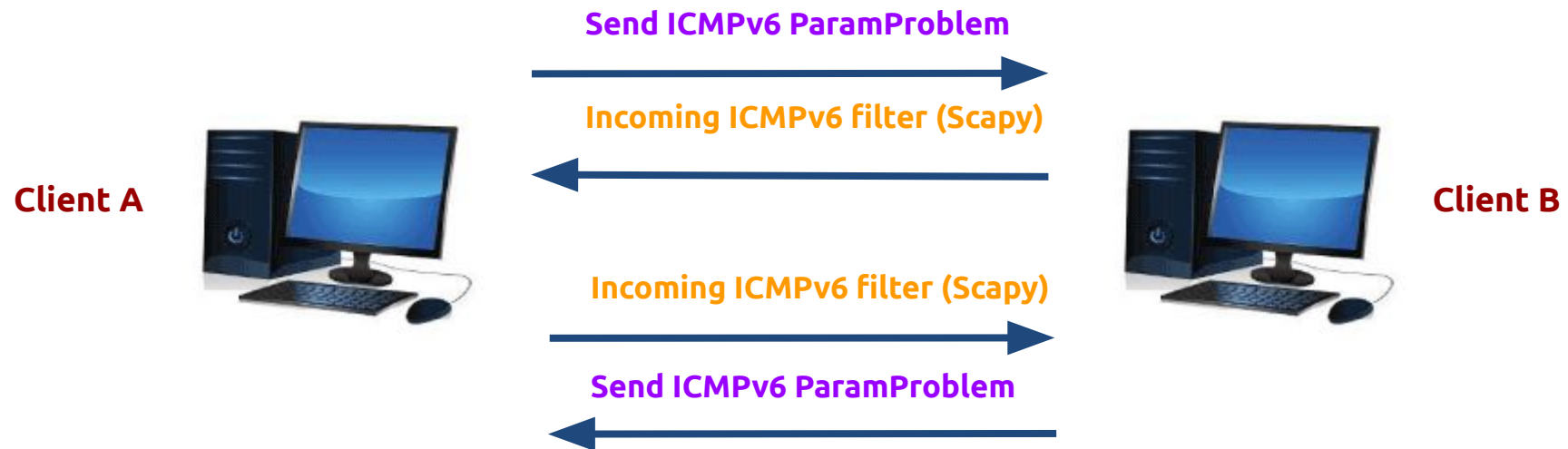
[Length: 50]

```

0000 48 e2 44 a4 a4 41 7c c2 c6 28 e3 92 86 dd 60 00  H·D··A|··(·(·(·(·
0010 00 00 00 3a 3a 40 2a 02 05 86 a5 02 32 b3 36 9f  ····:·@*· ····2·6·
0020 08 10 cf e7 fb 2a 2a 02 05 86 a5 02 32 b3 46 7d  ·····**· ····2·F}
0030 2f e1 df 03 03 c9 04 01 86 82 00 00 00 00 be ef  /·(·(·(·(·(·(·(·(·
0040 92 ff 5b 7b e2 93 5c c5 42 01 53 54 4c 2d 2b 30  ··[{·\· B·STL·+0
0050 4d 8f db fd 8d 97 75 0a 7c c4 57 5d a7 ce 5a b0  M·(·(·u· |·W}·Z·
0060 0e 43 16 04 49 1d dc 73 48 f1 02 a9 e2 eb 8e f9  ·C·(·I·*·s·H·(·(·(·

```

# P2P client



# P2P client (sender)

```
def sender(src, dst, data):

    # IPv6 packet
    IPv6_icmp = IPv6()
    IPv6_icmp.src = src
    IPv6_icmp.dst = dst
    # IPv6_icmp.hlim = 1

    # ICMPv6 following
    IPv6_icmp.nh = 58

    # ICMPv6 ParamProblem message type
    ICMPv6_ParamProblem = ICMPv6ParamProblem()
    ICMPv6_ParamProblem.type = 4
    ICMPv6_ParamProblem.code = 1
    ICMPv6_ParamProblem.ptr = 0

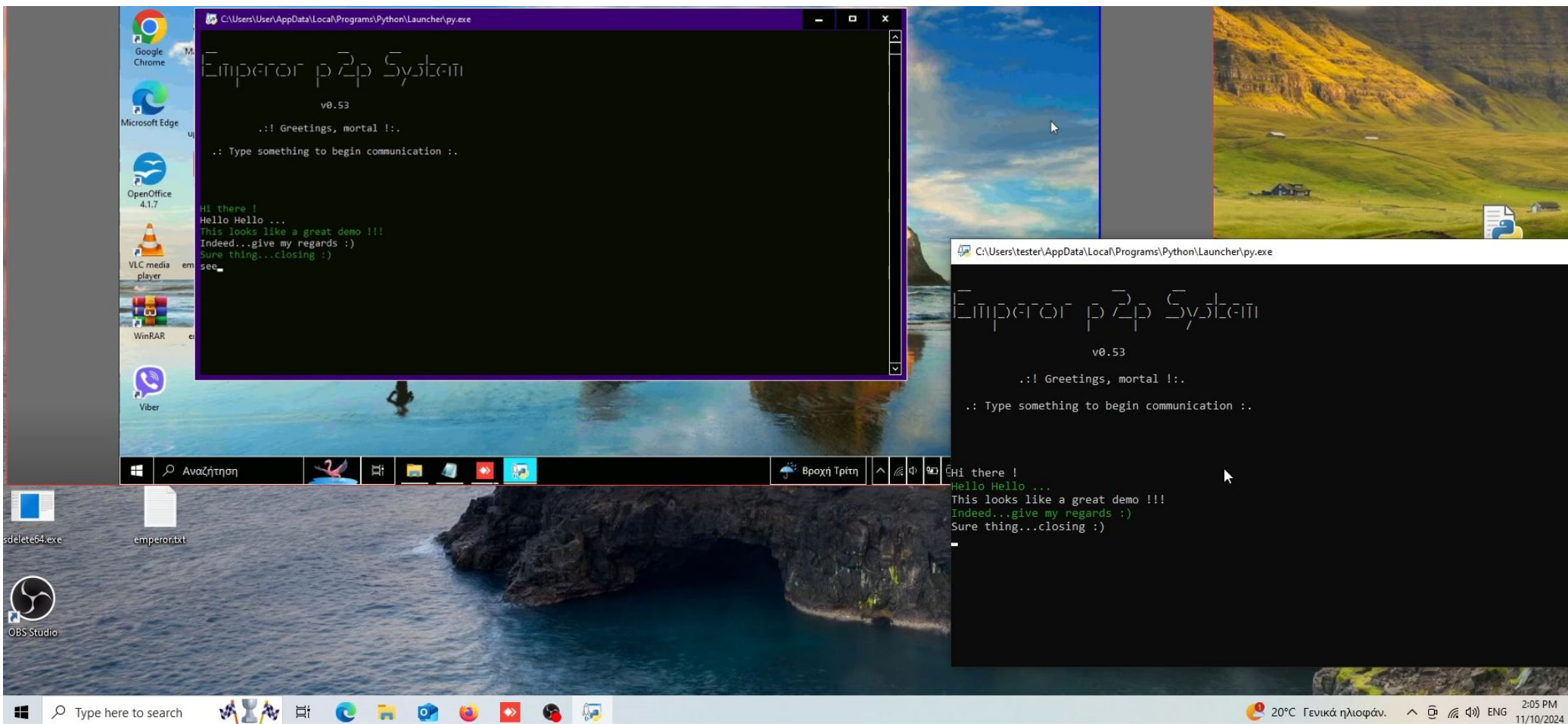
    # Set verbose to False to avoid printing information messages
    packet = IPv6_icmp/ICMPv6_ParamProblem/Raw(load=data)
    send(packet, count=1, verbose=False)
```

# P2P client (receiver)

```
# pip install scapy
from scapy.all import *

def receiver(dst, rc6, iv_data):
    while True:
        # Report only specific packet from specific ipv6 source address
        capture = sniff(filter="icmp6 && icmp6[0] = 4 && icmp6[1] = 1 && ip6 dst " + str(dst), count=1)
        packet = capture[0]
        packet = bytes(packet)
        # print(packet)
        data = packet[62:]
```

# P2P client (demo)



# OS firewall (in)

Windows Defender Firewall with Inbound Rules

- Inbound Rules
- Outbound Rules
- Connection Security Rules
- Monitoring
  - Firewall
  - Connection Security Rules
  - Security Associations

Name	Group	Profile	Enabled	Action	Override	Program	Local Address	Remote Address	Protocol	Local Port	Remote Port
Core Networking - Multicast Listener Done (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Multicast Listener Query (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Multicast Listener Report (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Multicast Listener Report v2 (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Neighbor Discovery Advertisement (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Neighbor Discovery Solicitation (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Packet Too Big (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Parameter Problem (ICMPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	System	Any	fe80::/64	ICMPv6	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	%System...	Any	Any	UDP	Edge Trave...	Any
Core Networking	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Private...	No	Allow	No	System	Any	Local subnet	ICMPv4	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Domain	No	Allow	No	System	Any	Local subnet	ICMPv4	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Private...	No	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Domain	No	Allow	No	System	Any	Any	ICMPv6	Any	Any
Delivery Optimization	Delivery Optimization	All	Yes	Allow	No	%System...	Any	Any	TCP	7680	Any
Delivery Optimization	Delivery Optimization	All	Yes	Allow	No	%System...	Any	Any	UDP	7680	Any
DIAL protocol server	DIAL protocol server	Private	Yes	Allow	No	System	Any	Local subnet	TCP	10247	Any
DIAL protocol server	DIAL protocol server	Domain	Yes	Allow	No	System	Any	Any	TCP	10247	Any
Distributed Transaction Co...	Distributed Transaction Co...	Domain	No	Allow	No	%System...	Any	Any	TCP	RPC Dyna...	Any
Distributed Transaction Co...	Distributed Transaction Co...	Private...	No	Allow	No	%System...	Any	Local subnet	TCP	RPC Dyna...	Any
Distributed Transaction Co...	Distributed Transaction Co...	Private...	No	Allow	No	%System...	Any	Local subnet	TCP	RPC Endp...	Any
Distributed Transaction Co...	Distributed Transaction Co...	Domain	No	Allow	No	%System...	Any	Any	TCP	RPC Endp...	Any
Distributed Transaction Co...	Distributed Transaction Co...	Private...	No	Allow	No	%System...	Any	Local subnet	TCP	Any	Any
Distributed Transaction Co...	Distributed Transaction Co...	Domain	No	Allow	No	%System...	Any	Any	TCP	Any	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	ICMPv4	Any	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	ICMPv4	Any	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	ICMPv6	Any	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
File and Printer Sharing	File and Printer Sharing	All	No	Allow	No	%System...	Any	Local subnet	UDP	5355	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	UDP	138	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	UDP	138	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	UDP	137	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	UDP	137	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	TCP	139	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	TCP	139	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	TCP	445	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	TCP	445	Any

Core Networking - Parameter Problem (ICMPv6-In) Properties

This is a predefined rule and some of its properties cannot be modified.

**General**

Name: Core Networking - Parameter Problem (ICMPv6-In)

Description: Parameter Problem error messages are sent by nodes as a result of incorrectly generated packets.

Enabled

**Action**

Allow the connection

Allow the connection if it is secure

Block the connection

OK Cancel Apply

# OS firewall (out)

- Windows Defender Firewall with Advanced Security
- Inbound Rules
- Outbound Rules
- Connection Security Rules
- Monitoring
- Firewall
- Connection Security Rules
- Security Associations

Name	Group	Profile	Enabled	Action	Override	Program	Local Address	Remote Address	Protocol	Local Port	Remote Port
Core Networking - Internet Group Management Protocol (IGMP-Out)	Core Networking	All	Yes	Allow	No	System	Any	Any	IGMP	Any	Any
Core Networking - IPHTTPS (TCP-Out)	Core Networking	All	Yes	Allow	No	%System...	Any	Any	TCP	Any	IPHTTPS
Core Networking - IPv6 (IPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Any	IPv6	Any	Any
Core Networking - Multicast Listener Done (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Multicast Listener Query (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Multicast Listener Report (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Multicast Listener Report v2 (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking - Neighbor Discovery Advertisement (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Neighbor Discovery Solicitation (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Packet Too Big (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Parameter Problem (ICMPv6-Out)	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking - Parameter Problem (ICMPv6-Out) Properties	Core Networking	All	Yes	Allow	No	System	fe80::/64	Local subnet, ff0...	ICMPv6	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	System	Any	Local subnet, ff0...	ICMPv6	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	%System...	Any	Any	UDP	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Private...	No	Allow	No	System	Any	Local subnet	ICMPv4	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Domain	No	Allow	No	System	Any	Any	ICMPv4	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Private...	No	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
Core Networking Diagnostics	Core Networking Diagnostics	Domain	No	Allow	No	System	Any	Any	ICMPv6	Any	Any
DiagTrack	DiagTrack	All	Yes	Allow	No	%System...	Any	Any	TCP	Any	443
Distributed Transaction Co...	Distributed Transaction Co...	Domain	No	Allow	No	%System...	Any	Any	TCP	Any	Any
Distributed Transaction Co...	Distributed Transaction Co...	Private...	No	Allow	No	%System...	Any	Local subnet	TCP	Any	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	ICMPv4	Any	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	ICMPv4	Any	Any
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	ICMPv6	Any	Any
File and Printer Sharing	File and Printer Sharing	All	No	Allow	No	%System...	Any	Local subnet	UDP	Any	5355
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	UDP	Any	138
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	UDP	Any	138
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	UDP	Any	137
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	UDP	Any	137
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	TCP	Any	139
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	TCP	Any	139
File and Printer Sharing	File and Printer Sharing	Domain	No	Allow	No	System	Any	Any	TCP	Any	445
File and Printer Sharing	File and Printer Sharing	Private...	No	Allow	No	System	Any	Local subnet	TCP	Any	445
HomeGroup	HomeGroup	Private	No	Allow	No	%system...	Any	Local subnet	TCP	Any	3587
HomeGroup	HomeGroup	Private	No	Allow	No	%system...	Any	Local subnet	UDP	Any	3540
SCSI Service	SCSI Service	Private...	No	Allow	No	%System...	Any	Local subnet	TCP	Any	Any
SCSI Service	SCSI Service	Domain	No	Allow	No	%System...	Any	Any	TCP	Any	Any
mDNS	mDNS	Private	Yes	Allow	No	%System...	Any	Local subnet	UDP	Any	5353
mDNS	mDNS	Public	Yes	Allow	No	%System...	Any	Local subnet	UDP	Any	5353
mDNS	mDNS	Domain	Yes	Allow	No	%System...	Any	Any	UDP	Any	5353

Core Networking - Parameter Problem (ICMPv6-Out) Properties

This is a predefined rule and some of its properties cannot be modified.

General

Name: Core Networking - Parameter Problem (ICMPv6-Out)

Description: Parameter Problem error messages are sent by nodes as a result of incorrectly generated packets.

Enabled

Action

Allow the connection

Allow the connection if it is secure

Block the connection

OK Cancel Apply

# Router firewall

Connection

Features

Advanced Settings

Routing

- Static Routing
- IPv6 Static Routing
- RIP

Security

- IP Filter
- IPv6 Filter**
- DMZ
- ALG
- E-mail abuse detection

Settings for the IPv6 Filter

[What is IPv6 filter?](#)

Enable

Protocol TCP

Name TCP

Incoming Traffic ICMPV6

Outgoing Traffic Internet\_ADSL

Start Source IPv6 Address

End Source IPv6 Address

Start Destination IPv6 Address

End Destination IPv6 Address

Start Source port  (1-65535)

End Source port  (1-65535)

Start Destination port  (1-65535)

End Destination port  (1-65535)

Mode Discard

Save

Enable	Name	Start Source IPv6 Address	Start Source port	Start Destination IPv6 Address	Start Destination p
	Protocol mode	End Source IPv6 Address	End Source port	End Destination IPv6 Address	End Destination p

Security status

- ✓ Firewall active
- ✓ WLAN encrypted



# Client functions

- At least 1 destination IP address should be known
- Usual router/firewall setups would be bypassed
- RC6 encryption
- Peer communicates other Peer (P2P)
- No intermediate system assists in establishing communication

## Future work

- P2P IP list sharing among Peers
- Tested on home routers and mobile phone hotspots
- RC6 key exchange using RSA
- Different Message Types

# Demo

- Users on different geolocation behind different home routers
- Demo, presentation, paper, code (Open Source)
- <https://github.com/nitsa>

# Q & A

Any questions ?

Thank you !