Deliverable FI3-D2.1.4
Study of Middle-box Behavior on Network Layer Protocols

Seppo Hätönen, Yonghao Li, Markku Kojo

Tivit Future Internet Program
(Tivit FI)

Period: 1.4.2011 – 30.4.2012

Tivit, Strategisen huippuosaamisen keskittymän tutkimusohjelma

Rahoituspäätös 1171/10, 30.12.2010, Dnro 2790/31/2010

www.futureinternet.fi

www.tivit.fi

This work was supported by TEKES as part of the Future Internet programme of TIVIT (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT).

Executive summary / Internal release

Title: Study of Middle-box Behavior on Network Layer Protocols. Technical Report C-2012-3, University of Helsinki. June 13, 2012

Abstract:

The home gateways typically act as middle-boxes between the internal network of a home user or small enterprise. These middle-boxes often perform various higher-layer functions such as traffic filtering, network address translation (NAT), advanced application layer operations and act as dynamic host configuration protocol (DHCP) server. While some of these functions such as DHCP are well standardized, some functions such as NAT have only been defined on a more abstract level and the exact operations have not been standardized. These more loosely defined functions are known to have undesired effects on normal protocol functions and hinder the development of new protocols and applications. Therefore, it is important to understand the various characteristics of different middle-boxes deployed all around the world to allow network engineers design protocols that can be deployed in realistic environments that typically include middle-boxes. In this paper, we perform an experimental study on a number of different home gateways focusing on the network layer (Layer 3) functionality, which is responsible for packet forwarding and routing. The experiments include tests on how the home gateways treat various IPv4 packet header fields, including IP options, Explicit Congestion Notification (ECN) and Differentiated Services Code Point (DSCP) fields, and how they treat broadcasted packets.

Contact info: Seppo Hätönen (shatonen@cs.helsinki.fi), Yonghao Li (ygli@cs.helsinki.fi), Markku Kojo (kojo@cs.helsinki.fi)

University of Helsinki
Department of Computer Science
Series of Publications C, No. C-2012-3

Study of Middle-box Behavior on Network Layer Protocols

Seppo Hätönen, Yonghao Li, Markku Kojo

Helsinki, June 13, 2012

Technical Report C-2012-3

University of Helsinki
Department of Computer Science
P. O. Box 68 (Gustaf Hällströmin katu 2b)
FIN-00014 University of Helsinki, FINLAND

Study of Middle-box Behavior on Network Layer Protocols

Seppo Hätönen, Yonghao Li, Markku Kojo
Department of Computer Science, University of Helsinki
Technical Report C-2012-3
June 13, 2012
14 pages

Abstract. The home gateways typically act as middle-boxes between the internal network of a home user or small enterprise. These middle-boxes often perform various higher-layer functions such as traffic filtering, network address translation (NAT), advanced application layer operations and act as dynamic host configuration protocol (DHCP) server. While some of these functions such as DHCP are well standardized, some functions such as NAT have only been defined on a more abstract level and the exact operations have not been standardized. These more loosely defined functions are known to have undesired effects on normal protocol functions and hinder the development of new protocols and applications. Therefore, it is important to understand the various characteristics of different middle-boxes deployed all around the world to allow network engineers design protocols that can be deployed in realistic environments that typically include middle-boxes. In this paper, we perform an experimental study on a number of different home gateways focusing on the network layer (Layer 3) functionality, which is responsible for packet forwarding and routing. The experiments include tests on how the home gateways treat various IPv4 packet header fields, including IP options, Explicit Congestion Notification (ECN) and Differentiated Services Code Point (DSCP) fields, and how they treat broadcasted packets.

Contents

1 Introduction

Nowadays, home gateways such as wireless access points and Cable or DSL modems, are widely deployed for residential and Small Office/Home Office (SOHO) customers to access Internet services. The home gateways typically act as middle-boxes performing various higher-layer functions, such as network address translation (NAT) [6], traffic filtering or advanced application layer operations.

The NAT was first proposed by the Internet Engineering Task Force in 1994 to help with the foreseen IPv4 address shortage before IPv6 was designed. Since then, while the last available IPv4 address pool was assigned by Internet Assigned Numbers Authority (IANA) in February 2011, the IPv6 is still not widely deployed, at least not in the small companies and homes. Only a small percent of Internet Service Providers (ISP) offer IPv6 routing to normal home users and the ISPs see the NAT properties as a kind of firewall which masks the internal side of the NAT device from the Internet or public side of the NAT, and the ISPs are not in a hurry to move forward with the change.

Unfortunately, the IETF only defined the basic properties of the NAT and left the implementation open. This has led to many different NAT implementations over the years and many of them cause considerable problems with different Internet protocols. The main goal of this study is to dig deeper into the characteristics of different home gateway devices that implement NAT functionality. We develop a number of tests to provide information on how NAT devices treat the IP traffic that traverses through the devices.

The experiments extend the earlier study [3] that focused more on the NAT binding timeouts, TCP throughput, etc. In this study we investigate more specific characteristics of the NAT devices, focusing on various network layer characteristics. The tests probe how NAT devices behave when the devices encounter packets with header fields set to values that are either not specified or need more attention from the NAT devices than normal packets. For example, how IPv4 packets with unknown options and flags or fragmented packets messages are handled. The tests also address reported leaks of broadcast messages either from Wide Area Network (WAN) to Local Area Network (LAN) or to the opposite direction.

The rest of the report is organized as follows. In Section 2 we describe the testbed we use to run the experiments. Section 3 describes functionality of the tests and presents the results. Section 4 concludes the findings.

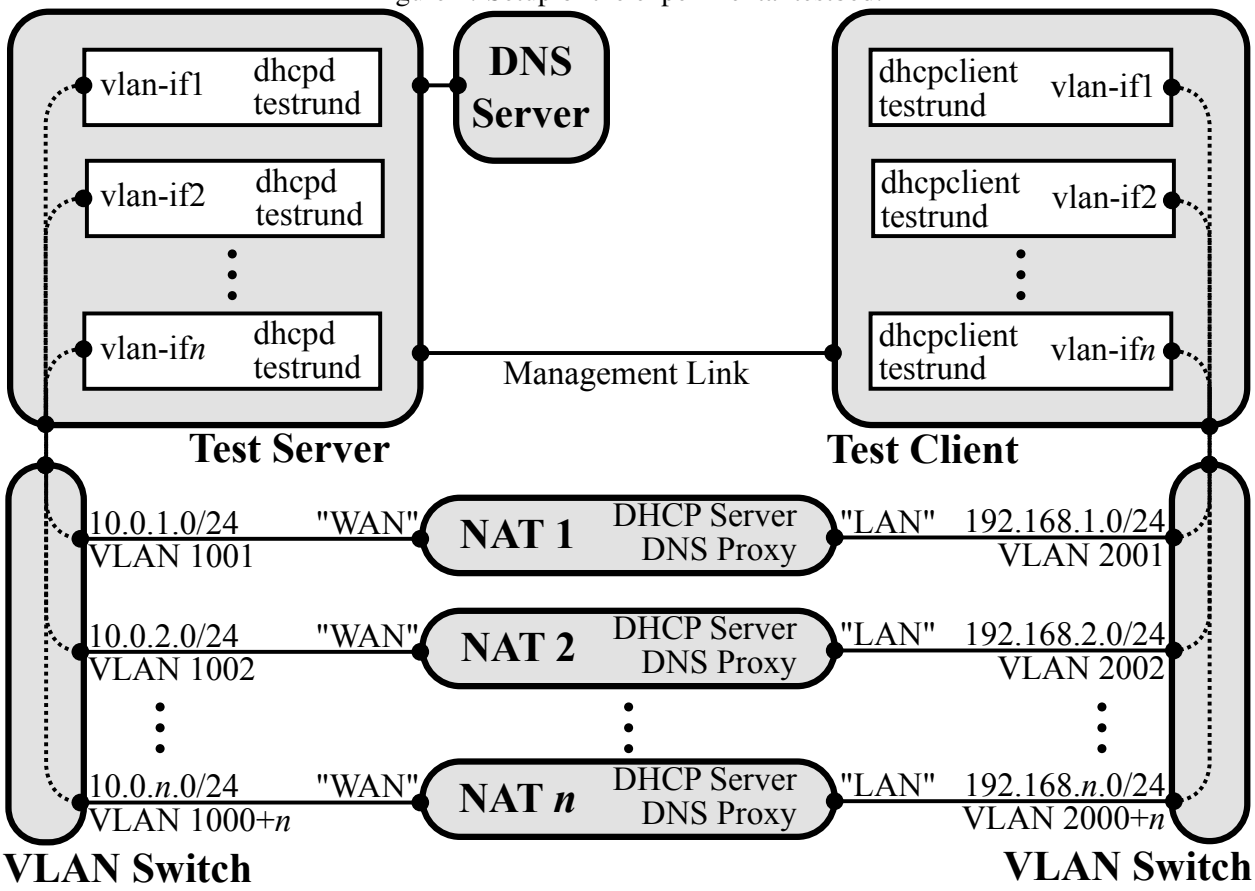
2 Testbed Description

The testbed used in the experiments is shown in the Figure 1. The testbed consists of several servers, a HP 5412 zl switch and 42 NAT devices as listed in the Table 1. Over half of the devices were donated to the University of Helsinki to give a broader view of different home gateway devices abroad. Rest of the devices were bought in spring 2010 to get a picture of current devices that were available at that time and to get a picture what the consumers buy. The test servers are running Linux 2.6.32 kernels

Unfortunately, due to the age of the devices and the fact that the devices are consumer grade hardware, some of the devices failed during the testing and are not reported here. This lowered the number of devices from 48 to 42. The failed hardware included both system hardware and power supplies.

The test servers are divided into two categories, the ”Internet” servers outside of the NAT devices and internal client servers inside the NAT. Both the test servers and the client servers run their instances of

Figure 1: Setup of the experimental testbed.



runtestd which is responsible of setting up test runs, capturing the traffic for analysis using *tcpdump* in both hosts and logging. All tests can either be run in parallel or serially depending on what kind of load the tests generate on the testbed. For example a test that explores the treatment of various header fields can be run in parallel on all devices but a throughput or similar test that requires notable amount of resources must be run serially as the traffic may overload the switches, servers or network interfaces and affect the test results.

Each of the NAT devices are connected to the ”Internet” servers using its “WAN” uplink port through a managed switch where each switch port has been configured to use its own separate virtual LAN (VLAN). The VLAN’s are used to keep the different NAT devices separate from each other. Each of the ”Internet” servers have two IP addresses per VLAN, which enables us to simulate multiple destinations. For some of the tests we enable a second network interface on the client servers to provide second connection to the NAT devices.

The management server is running a DHCP service [2] that provides each VLAN a separate private address block 10.0.x.0, where x is the unique number assigned for each NAT device in the testbed. The NAT devices use these to configure their ”WAN” interface and DNS proxies. The DHCP servers of the NAT devices are configured to distribute private address to clients from 192.168.x.0 blocks. The management server is also running a NTP server that provides synchronised time to both test and client servers.

Table 1: Home gateway models included in the study, with the shorthand "tags" used throughout this report

Vendor	Model	Firmware	Tag
A-Link	WNAP	e2.0.9A	*al*
Apple	Airport Express	7.4.2	*ap*
Asus	RT-N15	2.0.1.1	*as1*
	WL-500G Premium V2	3.0.3.5	*as2*
Belkin	Wireless N Router	F5D8236-4_WW_3.00.02	*be1*
	Enhanced N150	F6D4230-4_WW_1.00.03	*be2*
	Wireless G Router	F:3.00.03 H: F5D7234-4 v3 (01)	*be3*
	Wireless G Plus MIMO Router F5D9230-4 ver. 3000	3.02.76	*be4*
Buffalo	WZR-AGL300NH	R1.06/B1.05	*bu1*
D-Link	DIR-300	1.03	*dl1*
	DIR-300	1.04	*dl2*
	DI-524up	v1.06	*dl3*
	DI-524	v2.0.4	*dl4*
	DIR-100	v1.12	*dl5*
	DIR-600	v2.01	*dl6*
	DIR-615	v4.00	*dl7*
	DIR-635	v2.33EU	*dl8*
	WBR-1310	1.04	*dl11*
Edimax	6104WG	2.63	*ed*
Jensen	Air:Link 59300	1.15	*je*
Linksys	BEFSR41c2	1.45.11	*ls1*
	W54G	v7.00.1	*ls2*
	WRT54GL v1.1	v4.30.7	*ls3*
	WRT54GL-EU	v4.30.7	*ls5*
	WRT54G	OpenWRT RC5	*owrt*
	WRT54GL v1.1	tomato 1.27	*to*
Netgear	RP614 v4	V1.0.2_06.29	*ng1*
	WGR614 v7	(1.0.13_1.0.13)	*ng2*
	WGR614 v9	V1.2.6_18.0.17	*ng3*
	WNR2000-100PES	v.1.0.0.34_29.0.45	*ng4*
	WGR614 v6	V1.0.11_1.0.7	*ng6*
	WGR614	V1.40 Feb 18 2004	*ng7*
	WGT624 v4	V2.0.6_2.0.6NA	*ng8*
	WGT624 v3	v2.0.25_1.0.1NA	*ng9*
	MR314	V3.30(CF.0)	*ng10*
	RP114	V3.26(cd.0)	*ng11*
Netwjork	54M	Ver 1.2.6	*nw1*
SMC Barricade	SMC7004VBR	R1.07	*smc*
Telewell	TW-3G	V7.04b3	*te*
Unicom	WEP-72104G rev. 2	v4.2.3.18.1e	*un1*
Webee	Wireless N Router	e2.0.9D	*we*
ZyXel	P-335U	V3.60(AMB.2)C0	*zy1*

The majority of the NAT devices in the testbed provide a switch capability and one device, *ap*, has only a wireless interface in addition to its "WAN" interface. Each of the NAT devices is connected to the test client through one of the "LAN" interfaces. (The *ap* is connected to the client host through a separate USB WLAN dongle.) The client hosts have a separate DHCP client listening on each separate VLAN and sets the interface with the information that the NAT device provides via its DHCP server. The DHCP client was modified to configure only the interface-specific routes and to not set up the default route.

Table 2: Summary of the Time-To-Live set to 1. •: TTL not decreased, °: Dropped and TTL exceeded returned, †: dropped

	al	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	um1	we	zy1
ping-ttl1	°	°	°	†	•	°	°	°	°	°	•	•	°	•	•	•	•	°	°	•	°	°	°	°	°	°	°	°	°	°	•	°	°	•	°	°	°	°	°	°

3 Experiments

We implement a set of experiments to determine how the NAT devices handle packets in the network layer. The tests focus on the NAT device behavior when the devices encounter packets with header fields set to values that are either not used or need more attention from the NAT devices than normal packets. The tests also address reported leaks of broadcast messages either from Wide Area Network (WAN) to Local Area Network (LAN) or vice versa.

3.1 IP1: Time-To-Live Set to 1

3.1.1 Test Description

Several NAT traversal methods and network mapping tools such as *traceroute* use the Time-To-Live value in the IP header to map the network. The NAT devices handle the TTL value differently from device to device. The possible ways to handle the TTL value is to either decrease the value by one, which is the specified behavior for a network router, or not decrease it. Latter behavior would be correct for a complete transparent NAT, i.e. for both endpoints of the packet flow the NAT device would be completely invisible.

This test is done by using command *ping -c1 -t1 10.0.x.1* to ping the test server through each of the NAT devices. The TTL value of the ping packets is set to one. If the NAT device does not decrease the TTL value and drop the packet, the ICMP Echo message reaches the server and it should return an ICMP Echo Reply message. Otherwise, if the NAT device decreases the TTL value, it should drop the packet and return TTL Exceeded ICMP message back to the sender.

3.1.2 Results

The handling of the Time-To-Live value is important to some NAT traversal methods. The results of test are listed in Table 2. One quarter of the devices do not decrease TTL, resulting in an ICMP ECHO REPLY message to be returned from the server. The rest of the devices decreased the TTL value and most of them return TTL Exceeded ICMP message. Only a single device, *be1*, decreased the TTL value and dropped the packet without sending the TTL Exceeded ICMP message.

3.2 IP2: IPv4 Options

3.2.1 Test Description

The test is important since new proposed standards might use new IP options or extensions. Depending on the NAT, the packets containing these options might be dropped, the options removed or the packets might

Table 3: Summary of the IPv4 Options 0-15 test. •: No change, °: changed, †: dropped

	al	ap	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	un1	we	zy1
opt-1-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†
opt-1-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	†
opt-1-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	†
opt-1-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	†
opt-2-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-2-0-2	†	†	•	•	•	•	•	•	•	•	•	†	•	•	•	†	†	†	†	†	•	†	†	†	•	†	†	†	†	†	†	†	†	†	•	•	•	†	†	•	•
opt-2-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-2-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-3-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-3-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-3-1-0	†	†	†	†	•	•	•	†	†	†	•	†	•	•	•	†	†	°	†	†	•	†	†	†	•	†	†	†	†	†	†	†	†	†	†	†	†	†	†	†	•
opt-3-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-4-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-4-0-2	°	†	°	°	•	•	•	°	°	°	•	°	•	°	•	°	°	†	°	°	•	†	°	°	•	†	†	†	†	†	†	†	†	†	•	°	°	°	°	°	•
opt-4-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-4-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-5-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-5-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-5-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-5-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-6-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-6-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	†	†	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	†	•	•	•
opt-6-1-0	†	†	•	•	•	•	•	•	•	•	•	•	•	•	•	†	†	°	†	†	•	†	†	†	•	†	†	†	†	†	†	†	†	†	†	†	†	†	†	†	•
opt-6-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-7-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-7-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-7-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-7-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-8-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-8-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-8-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-8-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-9-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-9-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-9-1-0	†	†	†	†	†	†	•	†	†	†	•	†	•	†	•	†	†	°	†	†	•	†	†	†	•	†	†	†	†	†	†	†	†	†	†	†	†	†	†	†	•
opt-9-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-10-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-10-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-10-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-10-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-11-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-11-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-11-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-11-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-12-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-12-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-12-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-12-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-13-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-13-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-13-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-13-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-14-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-14-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-14-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-14-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-15-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-15-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-15-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	†	†	†	†	•	•	•	•	•	•	•
opt-15-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	†	•	•	•	†	†	†	†	†	†	†	†	†	•	•	•	•	•	•	•

be translated improperly.

The test is performed by creating a UDP packet with an IP option set in the IP header and sending the packet through each of the NAT devices. We then compare all fields of each packet that arrived at the server to the original packet that was send and determine if the IP option was still present, changed or removed when the packet was translated in the NAT device. This is repeated for all of the IP options, both assigned and unassigned.

3.2.2 Results

The testing was done for both options that are assigned and also for those that are not. The results for option number 0 to 15 are shown in Table 3 and option numbers 16 to 31 are shown in Table 4. Of the

Table 4: Summary of the IPv4 Options 16-31 test. •: No change, °: changed, †: dropped

	al	ap	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	un1	we	zy1
opt-16-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-16-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-16-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-16-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-17-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-17-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-17-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-17-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-18-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-18-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-18-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-18-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-19-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-19-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-19-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-19-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-20-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-20-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-20-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	•	•	•	•	†	†	•	•	•	•	•	•	•
opt-20-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-21-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-21-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-21-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-21-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-22-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-22-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-22-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-22-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-23-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-23-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-23-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-23-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-24-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-24-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-24-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-24-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-25-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-25-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-25-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-25-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-26-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-26-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-26-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-26-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-27-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-27-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-27-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-27-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-28-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-28-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-28-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-28-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-29-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-29-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-29-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-29-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-30-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-30-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-30-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-30-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-31-0-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-31-0-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-31-1-0	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•
opt-31-1-2	•	†	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	†	†	†	•	•	†	†	•	•	•	•	•	•	•

tested options, options from 0 to 25 and 30 are defined by the IANA. The opt-X-Y-Z denote the different options, where X is the number of the option, Y is the copy flag (if 1, the option is included in all possible IP fragments) and Z denotes the option class (0 for control, 2 for debug and measurement.) While some of the options use more than one combination of these numbers, most are defined with single option number, copy and class combination.

The tests were inconclusive for certain options, mainly Loose Source Route (3) and Strict Source Route (9). The reason for this was that the test software was not able to properly generate the messages. The Record Route option (7) seems to crash many of the Netgear devices. Other results indicate that Netgear devices and the Apple device *ap* are more prone to drop the packets containing IP options as five out of ten

Table 5: Summary of the IP fragmentation test. •: Received, †: Dropped

	al	ap	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	un1	we	zy1
i0-FIFO	•	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	†
i0-Reverse	•	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
i0-FILO	•	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•
i1-FIFO	•	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	†
i1-Reverse	•	•	•	•	•	•	†	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
i1-FILO	•	•	•	•	•	•	†	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
i2-FIFO	•	•	•	•	•	•	•	•	•	•	†	•	†	•	†	†	•	†	•	•	•	•	•	•	†	•	•	•	†	•	†	•	•	•	†	•	•	•	•	•	†
i2-Reverse	•	†	•	•	•	•	†	•	•	•	†	•	†	†	†	•	•	†	†	•	•	•	•	•	†	•	•	•	†	•	†	•	•	•	†	•	•	•	•	•	•
i2-FILO	•	†	•	•	•	•	†	•	•	†	†	•	†	•	†	•	•	†	†	•	•	•	•	†	†	•	•	•	†	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i0-FIFO	•	•	•	•	•	•	•	•	•	•	†	•	†	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i0-Reverse	•	•	•	•	•	•	†	•	•	•	†	•	†	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i0-FILO	•	•	•	•	•	•	†	•	•	•	†	•	†	•	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i1-FIFO	•	•	•	•	•	•	•	•	•	•	†	•	†	•	•	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i1-Reverse	•	•	•	•	•	•	†	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i1-FILO	•	•	•	•	•	•	†	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	•	†	•	•	•	•	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i2-FIFO	•	•	•	•	•	•	•	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	•	†	•	•	•	†	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i2-Reverse	•	•	•	•	•	•	†	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	•	†	•	•	•	†	•	†	•	•	•	†	•	•	•	•	•	•
WtoL-i2-FILO	•	•	•	•	•	•	†	•	•	•	†	•	†	•	†	•	•	†	•	•	•	•	•	†	•	•	•	•	†	•	†	•	•	•	†	•	•	•	•	•	•

Netgear devices (*ng2, ng3, ng4, ng8* and *ng9*) constantly dropped the packets. The most surprising result is from NAT device *dl8* which actually removed the IP option from the packet with most of the tested options. Overall, the devices passed the packets containing the options much better than expected and the Netgear devices were mainly responsible for dropped options with some exceptions.

3.3 IP3: Fragmentation

3.3.1 Test Description

Since the maximum transfer unit (MTU) can vary in the Internet, the NAT devices need to be able to handle fragmented packets. Fragmentation test consists of three different scenarios. The first scenario is to create an UDP packet, fragment it and then send the fragments in order with no delay between fragments. The second scenario is to send the fragments in reverse order, i.e. the last fragment is sent first and first fragment is sent last. The third scenario is to send the fragments in order except the first fragment, which is send last. This is done to determine if the NAT device keeps fragments in a buffer or just sends them forward. We also test all scenarios with different intervals between the fragments. First we set the interval between fragments to zero, then to one second and lastly to two seconds.

The test is done by creating an UDP packet and then fragmenting the packet to four pieces. The fragmented packet is send both from LAN to WAN and also from WAN to LAN. The LAN to WAN -test is done to see if the NAT device compiles the packet or just sends the fragments to the destination address. The WAN to LAN -test needs the client inside the NAT to first create a NAT binding. After the binding is created, the server outside the LAN sends the fragmented packet to the client.

3.3.2 Results

The results are shown in Table 5. The i0, i1 and i2 mark the different intervals in seconds between the fragments. The FIFO means that the fragments are send in order. Reverse means that the fragments are send in reverse order. The FILO denotes the last scenario, when the first fragment was sent last while the fragments from second to third were send in order.

Table 6: Summary of the reserved bit test. •: No change, °: changed, †: dropped

	al	ap	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	un1	we	zy1
reserved-WtoL	•	°	•	•	°	°	•	•	•	•	•	•	•	†	•	•	•	°	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	†	•	•	•	•	•	•
reserved	•	°	•	•	°	°	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•	•	•	•	•	°	•	•	•	•	•	•	•	•	•

The results seem to indicate that roughly half of the devices can handle fragmented packets from LAN to WAN. Only three devices do not handle the fragmented packets at all and 20 devices have problems with at least some of the test scenarios. It would seem that if the first fragment arrives later than the rest of the fragments, the devices seem to be more prone to drop fragments. The *be3* seems to handle the fragments that arrive in right order fine, but if the fragments arrive in reverse order, the *be3* drops the packets. The devices handle the fragmented packets coming from LAN to WAN slightly better than the packets coming from WAN to LAN. Since the fragmentation is much more likely to happen in the Internet than inside the LAN, the NAT devices should handle the packets from the "Internet" much better.

3.4 IP4: Reserved Bit in IPv4 Header

3.4.1 Test Description

This test experiments with the only bit in the IPv4 header that has no assigned use, i.e. the most significant bit in the Flags field [5]. The main purpose for this test is to explore if the unused bit can be used for real purposes, i.e. the packet goes through the NAT properly. The test is done by creating an UDP packet and setting the bit to 1. We then send the packet through all NAT devices and determine if the packet came through and if the bit was still set.

In an April 1. joke RFC [1], this bit was designated as "The Evil bit" and it was used to denote the "intent" of the packet. If the bit is set, the intention of the packet is evil and should be dropped by the firewalls and routers. Unfortunately, if the NAT devices drop the packet, we cannot be sure whether the NAT device follows RFC and drops the malicious packet or it just dropped the packet because it was not able to properly translate packet.

3.4.2 Results

The results for the Reserved Bit test are shown in Table 6. The results show that only five devices unset the reserved bit and the rest of the devices do not touch it. Two devices, *dl4* and *nw1* have problems with the packet coming from WAN to LAN. The device *nw1* had problems with creating the binding to be used in the WAN to LAN test and *dl4* dropped the packet containing the reserved bit.

3.5 IP5,6: UDP Broadcast Leaking Through NAT from LAN to WAN

3.5.1 Test Description

It has been reported that some devices may leak broadcast packets from WAN to LAN. Also, since the DHCP (Dynamic Host Configuration Protocol) packets may leak from LAN to WAN, it is necessary to

Table 7: UDP broadcast leaking test. °: Leak, •: Noleak, †: Exception

box	ng2	ls5	owrt	nw1	dl4	ls1	ng1	ls2	dl1	zy1	ed	ng3	dl3	al	as1	ls3	to	bu1	smc	dl2	dl5
WAN-10	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
WAN-255	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
LAN-192	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
LAN-255	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

box	we	dl7	je	dl6	dl8	ap	ng4	be1	be2	te	as2	ng6	ng7	ng10	be3	dl11	ng8	un1	be4	ng11	ng9
WAN-10	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•
WAN-255	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•
LAN-192	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•
LAN-255	•	•	•	•	•	•	•	•	•	†	•	•	•	•	•	•	•	•	•	•	•

detect whether broadcast packets leak through the NAT devices or not. To achieve this goal, we designed first two tests to figure out the broadcast packets leaking problem from LAN to WAN.

In the IP5 test, each internal client behind the NAT device sends UDP broadcast packets to the LAN broadcast address *192.168.x.255* using a randomly selected destination port. The external server listens on the corresponding port.

In addition, the selection of the destination port may affect the leaking behavior. Therefore, this IP5 test attempts to test different ports randomly picked from the well-known, registered and dynamic range.

In the IP6 test, the destination address of the UDP packet is changed to the address *255.255.255.255* in order to explore whether any broadcast packets leak through the NAT device or not. Except the broadcasting address, the rest of settings are similar to the IP5 test.

3.5.2 Results

In the UDP LAN broadcast leaking test (IP5), none of the NAT devices leak the messages from LAN to WAN as indicated in the row "LAN-192" (the packet was sent to the LAN broadcast address *192.168.x.255*) in Table 7. However, the results show that the NAT device *te* has an abnormal behavior. When the internal client sends LAN broadcast packets with LAN broadcast port *69* (TFTP), the NAT device *te* will send one UDP packet with the partial broadcast packets payload back to the internal client.

Moreover, we can observe from the IP6 test result that none of the UDP broadcast packets leaked for all NAT devices as indicated on the row "LAN-255" (the packet was sent to the broadcast address *255.255.255.255*) in Table 7 as well. However, the abnormal case of NAT device *te* is also found in this experiment. Additionally, this strange behavior of the NAT device *te* will be taken a great consideration in the following IP7,8 tests and we will test these experiments on more NAT devices in the future.

3.6 IP7,8: UDP Broadcast Leaking Through NAT from WAN to LAN

3.6.1 Test Description

In these two tests, we try to determine if any broadcasted messages leak from WAN to LAN. In the IP7 test, the server sends a broadcast message using destination address *10.0.x.255* and client attempts to receive any broadcast messages which leak out of NAT from the external server. The WAN broadcast port selection in the tests is the same as in the IP5 and IP6 tests. The only difference between IP7 and IP8 is that the server changes the WAN broadcast address to the address *255.255.255.255* in the IP8 test.

Table 8: Summary of the TCP ECN test. •: No change, °: changed, †: dropped

	al	ap	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	un1	we	zy1
ECT0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
ECT1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
CE	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

3.6.2 Results

In the case of WAN to LAN broadcast leaking tests, we still do not discover any packet leaking through NAT to the internal client from both the row "WAN-10" (the packet was sent to the broadcast address *10.0.x.255*) and the row "WAN-255" (the packet was sent to the broadcast address *255.255.255.255*) in Table 7. This strongly indicates that none of the NAT devices in our testbed leak broadcasted messages. However, the strange behavior of the NAT device *te* in IP5,6 tests also occurred in IP7,8 tests when the broadcast packet is sent to port *69*. Hence, the NAT device *te* may have a special behavior when the TFTP service is triggered.

3.7 ECN1: Can ECN Be Negotiated Through the NAT Device

3.7.1 Test Description

The Explicit Congestion Notification (ECN) is an important tool to detect and help with network congestion. The conventional method of indicating congestion is to drop packets when the network routers become congested. If both endpoints of a connection indicate, that they support ECN and are willing to use it, an ECN aware router can then use the ECN to signal the endpoints of an impending congestion. The ECN uses the two least significant bits in the DiffServ field in the IP header to indicate ECN capable transports (ECT(0) to 10 or ECT(1) to 01) or non ECN-capable (00) and if congestion is encountered, setting the bits to CE (11, congestion encountered). When an ECN capable router detects the impending congestion, the router sets ECN bits to CE. Due to the nature of the ECN protocol, the basic IPv4 protocol won't benefit from the ECN but the protocols above layer3 can benefit from the congestion notification and change their sending rates etc. accordingly. Depending on the NAT devices, some devices might ignore fields and some might drop the packets if the ECN bits in the IP header are set to other value than 00.

This test checks if the NAT devices either forward the TCP SYN packet with ECT set or is the packet dropped or ECT bits set to zero. The test is only done from LAN to WAN as TCP SYN packets cannot reach hosts behind NATs unless there is a static NAT forward set.

3.7.2 Results

The results in Table 8 show that all devices forwarded the SYN packet to the test server without modifying the ECN field in the IP header.

Table 9: Summary of the UDP ECN test. •: No change, °: changed, †: dropped

	al	ap	as1	as2	be1	be2	be3	be4	bu1	dl1	dl11	dl2	dl3	dl4	dl5	dl6	dl7	dl8	ed	je	ls1	ls2	ls3	ls5	ng1	ng10	ng11	ng2	ng3	ng4	ng6	ng7	ng8	ng9	nw1	owrt	smc	te	to	un1	we	zy1
ECT0-WtoL	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
ECT0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
ECT1-WtoL	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
ECT1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
CE-WtoL	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
CE	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

3.8 ECN2: ECN for UDP Traffic

3.8.1 Test Description

This test is a follow-up on the previous test to determine if ECN can be used with UDP packets or with protocols that are implemented on UDP. While the UDP itself cannot handle congestion, the protocols above it could benefit from the congestion information. The test is done by sending UDP packets with the two least significant bits set in the DiffServ field in the IPv4 header and checking if the packet go through the NAT device and if the bits are changed by the NAT device.

3.8.2 Results

Since none of the devices dropped the UDP packets containing the ECN bits (ECT and CE) either from LAN to WAN or from WAN to LAN (WtoL in Table 10), the results indicate that the ECN can be used with UDP or other protocols implemented on UDP.

3.9 DSCP1: Is DSCP field overwritten

3.9.1 Test Description

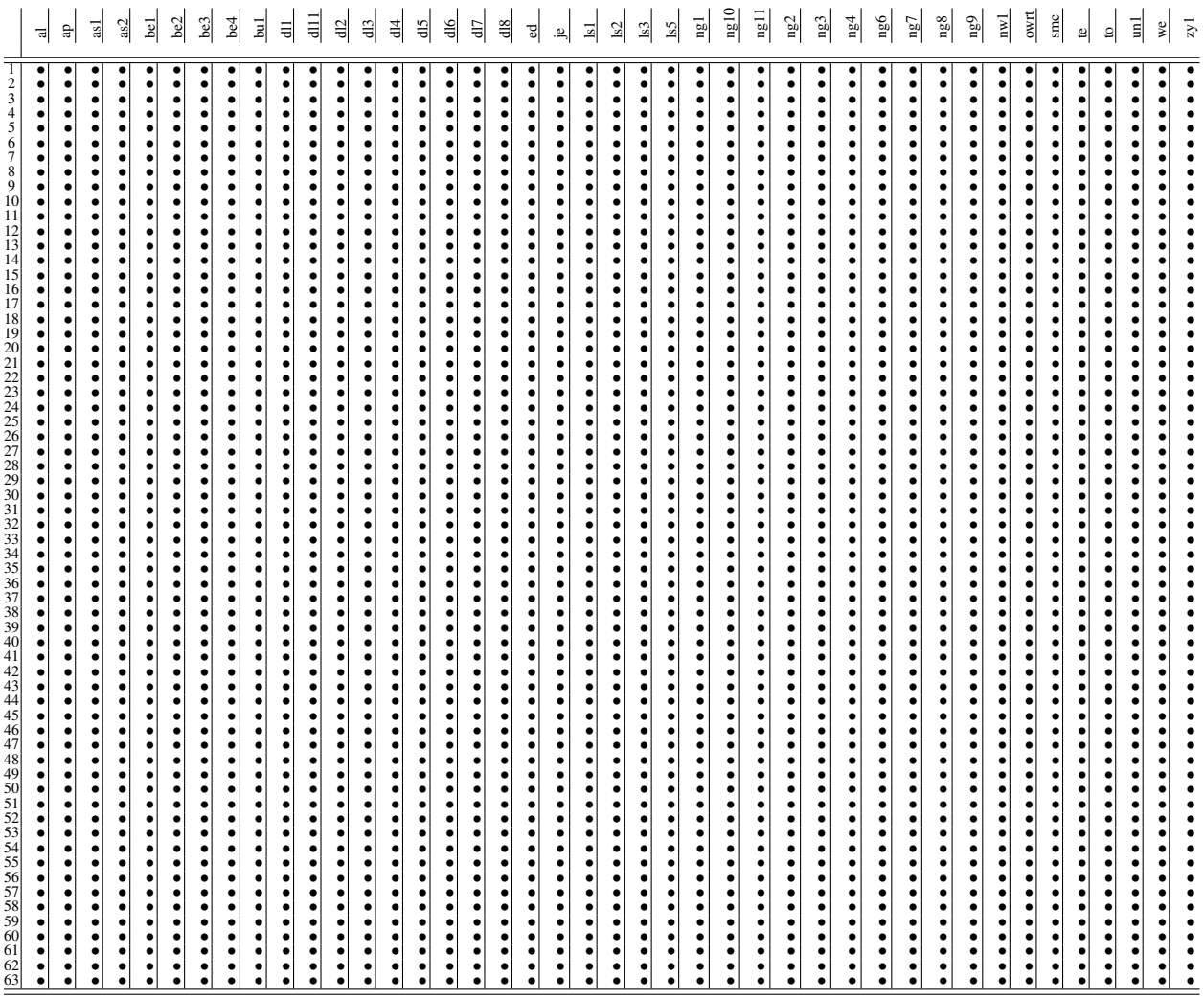
The Differentiated Services Code Point (DSCP) field in the IPv4 header was originally used as Type of Service field. The RFC 2474 [4] redefined the six first bits in the TOS field to be used for Differentiated Services. These services include streaming realtime music and video, which need traffic management and Quality of Service (QoS). The DSCP field provides a coarse grained QoS for the traffic.

This test includes both the defined and undefined values for the DSCP field. The main aim of the test is to explore whether packets with DSCP set are dropped or is the field modified. The test is done by first creating a UDP packet and setting the DSCP field to all possible values.

3.9.2 Results

In this test we created an UDP packet and set the DiffServ field in the IPv4 header to all possible values, as shown in Table 10. The main goal was to untangle if the packets with DSCP set are forwarded by the NAT devices and if so, whether the value of the DSCP field remains unchanged. The results show that surprisingly all NAT devices passed all values unmodified. This result indicates that the DSCP field can probably be used without troubles.

Table 10: Summary of the DSCP1 test. •: No change, °: changed, †: dropped



4 Conclusions

In this report we extended the earlier experimental study [3] with new tests. In this study we focused on the network layer (layer 3) behavior of the home gateway devices and tried to find out how different IPv4 packets were treated. The NAT devices showed many different behavior during the testing. The results were partly surprising as while some tests showed considerable variance between the devices, other tests showed very uniform behavior among the NAT devices.

The tests that focused on the IPv4 handling showed that even between the devices in our testbed have differences and that these differences are not limited to a single manufacturer. The tests actually show that there are differences even between the devices from same manufacturer. Some tests such as the TTL handling show that very common operations are not uniformly handled while some tests, such as passing ECN bits unaltered through the NAT devices are handled almost completely same way in our NAT devices.

Also, the broadcast leaking tests fortunately present that the broadcast leaking problem did not occur in any of these devices in our testbed. One of the more surprising results is the treatment of IPv4 options; many of the options were forwarded through the devices properly and only one device actually removed the option. Still, there were several popular models (some of the Netgears and the single Apple device) that did not forward packets with IP options. While the number of devices is not enough to be conclusive, the result would indicate that extending the IPv4 options may result in operational problems.

5 Acknowledgements

This work was supported by TEKES as part of the Future Internet programme of TIVIT (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT).

The authors would like to thank Lars Eggert and Pasi Sarolahti for their suggestions and fruitful discussions during the study. We would also like to thank all individuals who donated their home gateway hardware used as a part of the testbed. In addition, we thank CSC - IT Center for Science, for donating the testbed servers to the University of Helsinki. The servers were originally a part of the CSC Sepeli cluster.

References

[1] S. Bellovin. The Security Flag in the IPv4 Header. *Internet RFCs, ISSN 2070-1721*, RFC 3514, Apr. 2003.

[2] M. Chatel. Classical versus Transparent IP Proxies. *Internet RFCs, ISSN 2070-1721*, RFC 1919, Mar. 1996.

[3] S. Hätönen, A. Nyrhinen, L. Eggert, S. Strowes, P. Sarolahti, and M. Kojo. An Experimental Study of Home Gateway Characteristics. In *Proceedings of the 10th annual conference on Internet measurement*, IMC '10, pages 260–266. ACM, 2010.

[4] K. Nichols, S. Blake, F. Baker, and D. Black. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. *Internet RFCs, ISSN 2070-1721*, RFC 2474, Dec. 1998.

[5] J. Postel. Internet Protocol. *Internet RFCs, ISSN 2070-1721*, RFC 0791, Sept. 1981.

[6] P. Srisuresh and M. Holdrege. IP Network Address Translator (NAT) Terminology and Considerations. *Internet RFCs, ISSN 2070-1721*, RFC 2663, Aug. 1999.