Practical IPv6 Monitoring on Campus

Best Practice Document

Produced by the CESNET-led Working Group
on Network Monitoring
(CESNET BPD 132)

Authors: Matěj Grégr, Petr Matoušek,
Tomáš Podermański, Miroslav Švéda

May 2011

Document No: GN3-NA3-T4-CBPD132
Version / date: May 2011
Original language : English
Original title: “Practical IPv6 Monitoring at Campus Network”
Original version / date: Version 1.0; April 2011
Contact: tpoder@cis.vutbr.cz

CESNET is responsible for the contents of this document. The document was developed by a CESNET-led working group on network monitoring as part of a joint-venture project within the higher-education sector in the Czech Republic.

The production of the original document received funding from the research project supported by the Czech Ministry of Education, grant MSM 0021630528: Security-Oriented Research in Information Technology, from the ESF project CZ.1.07/2.3.00/09.0067 “TeamIT—Building Competitive Research Teams in IT” and from a BUT FIT grant, FIT-S-10-2.

The research leading to the update and revision of this report has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 238875, relating to the project 'Multi-Gigabit European Research and Education Network and Associated Services (GN3)'.

Table of Contents

Executive Summary

Network monitoring is an essential task of network management. Information obtained by monitoring devices provides a real picture of the network in production, including transmitted data volumes, top hosts and a list of frequently used applications. In-depth analysis of data collected by monitoring can reveal network attacks and detect misuse of network services. The IPv6 protocol creates new challenges for network administrators. Unlike IPv4, an IPv6 address no longer identifies a user or PC uniquely, because an IPv6 address can be randomly generated and keeps changing. Personal Computers (PCs) with an IPv6 stack can also communicate via pre-defined tunnels over the IPv4 infrastructure. This tunnelled traffic usually bypasses network security implemented by firewalls. This best practice document discusses the major monitoring issues of IPv6 connectivity. A practical solution for monitoring both IPv4 and IPv6 traffic is proposed, based on SNMP and Netflow data that help to identify a user. The solution requires an extension of the monitoring data collected from network devices. A new data structure based on extended Netflow records is presented. These data are stored in the central monitoring system deployed at the Brno University of Technology (BUT) campus network. The document discusses the results of this traffic monitoring and future challenges.

1 Introduction

IPv6 (Internet Protocol version 6) is a new version of the fundamental Internet Protocol and has been developed to provide larger address space. Today, the Internet is actively deploying not only IPv4, but also IPv6. IPv6 support is available for operating systems such as Unix, Mac OS and Windows. Moreover, operating systems like Windows 7 and Vista not only support IPv6, but also provide IPv6 connectivity by default. This exposes network users and organisations to additional vulnerabilities. In addition, the auto-configuration of IPv6 addresses creates a new challenge for network administrators. A host in a Local Area Network (LAN) cannot be identified easily by its IP address, since there can be several temporary IPv6 addresses in use. Unfortunately, many users are not aware of such intricacy. They unconsciously violate security policy when they bypass standard IPv4 firewall rules and standard IPv4 addressing policy. Moreover, unique mapping between an IP address and a MAC address cannot be built as easily for IPv6 from a DHCP (Dynamic Host Configuration Protocol) log as was the case for IPv4.

One solution is to apply a strict security policy on all IPv6 tunnels. Since many operating systems use IPv6 tunnelling techniques by default, the blocking of IPv6 tunnels that bypass security policy can cause an interruption of services. Also services running on both IPv4 and IPv6 (dual stack) would be affected. Procedures for the detection of failed tunnel connectivity are not included in the tunnel protocol specifications. Therefore, the only indication of a blocked tunnel is a timeout during socket initialisation, which is obviously not acceptable. In addition, IPv6 tunnels are considered as a transitional technique to native IPv6 connectivity, so they should be treated as legitimate traffic. In fact, transition to native IPv6 may last for months or years, so the monitoring of such traffic is essential.

Traditional monitoring approaches are usually not appropriate for IPv6 traffic because of such issues as temporary addresses, different types of encapsulation of IPv6 over IPv4, and non-unique mapping between the link address and the IP address. Therefore, new techniques using current tools need to be deployed. This best practice document shows the current issues of monitoring of IPv6 traffic and practical approaches to the solution of these issues, as implemented in the Brno University of Technology (BUT) network.

2 State of the Art

There are very few papers or studies that discuss the practical monitoring issues of the IPv6 protocol. A very good overview of the key security issues and challenges is given in the papers [2] and [8]. Report [8] seems to be the most complete study of the security aspects of the practical deployment of IPv6 for network administrators. There are several academic papers that deal with IPv6 monitoring. In the paper [11], a novel architecture of IPv6 monitoring is introduced, using SNMP encapsulated in IPv6 extension headers. The aim of this approach is to insert monitoring data into IPv6 packets, and then, to process that information on border routers. The approach presents an interesting idea, but practical implementation does not appear to be viable because every router would need to recalculate the MTU size, the number of hops and other criteria for every packet. The paper [4] proposes an interesting approach, using IPFIX templates to transmit monitoring information about IPv6, ICMPv6, etc. A similar approach is incorporated in the standards of Netflow v9. In our approach, we extend this method, using data from more sources.

3 Contribution

The contribution of this best practice document includes two parts. First, two major IPv6 monitoring issues are discussed: (i) automatically created tunnels of IPv6 traffic over an IPv4 network that bypass standard security techniques, and (ii) randomly generated IPv6 addresses with temporary validity. These features of IPv6 traffic create a new challenge for network monitoring. User identification based on the IP address is not only required for accounting purposes, but is also a legal obligation of Internet Service Providers (ISPs), under by the EU Data Retention Act[1]. The document explains what kind of information is needed for the successful identification of a user in an IPv4/IPv6 network and how these data can be obtained by combining ARP entries, SNMP data, Netflow records and RADIUS logs. In the second part of this document, a workable solution for IPv6 monitoring is presented. The solution was designed, implemented and deployed on the campus network at BUT. The scheme of the Central Monitoring System at BUT is shown. The experience there demonstrates that the proposed technique is viable and meets most of the security requirements.

[1] Directive 2006/24/EC of the retention of data generated or processed in connection with the provision of publicly available electronic communications services

4 IPv6 Monitoring Issues

This section describes major monitoring issues related to IPv6 connectivity. The discussion is focused on traffic tunnelling and address distribution. From the point of view of network management, IPv6 configured hosts on an IPv4 network can bypass defined security policy or hide their identity using temporary IPv6 addresses. These practical security issues have been the main motivation for the proposed solution.

4.1 Tunnelling IPv6 over IPv4

Tunnelling is a transition technique that, in this case, connects IPv6 sites over the IPv4 infrastructure. Routers, firewalls and security devices at the edge of the enterprise network may not be technically capable of inspecting an IPv6 payload entering or exiting the network that is encapsulated within IPv4 packets. Network administrators should understand that IPv6 tunnelling requires their attention. Three tunnelling approaches are frequently applied: 6to4 tunnels, Teredo and ISATAP. All of these free tunnelling mechanisms are implicitly enabled on Windows 7 and Vista. IPv4 can tunnel IPv6 traffic without security controls, so normal access control filtering is violated. In effect, IPv6 over an IPv4 tunnel may become a backdoor into the network.

1. *6to4 tunnelling* provides IPv6 site-to-site connectivity across an IPv4 network by embedding IPv4 addresses in IPv6 prefixes [3]. If a host has a public IPv4 address, the 6to4 tunnel is the first transition technique to be used for communication. The 6to4 tunnel wraps an IPv6 datagram into an IPv4 protocol with protocol number 41. The wrapped packets are sent to the first available 6to4 relay router. The receiving 6to4 router unwraps the datagram and sends it through the native IPv6 interface to its destination. More information about 6to4 security can be found in RFC 3964 [16].

2. *Teredo (*Tunnelling IPv6 over UDP [12]) has similar functionality to 6to4. All IPv6 traffic from the client is transmitted using UDP packets with port 3544 and directed to the Teredo server via the closest Teredo relays. This technique was designed as an interim transition mechanism and should not be used any longer than necessary. However, there are many PCs that unknowingly use Teredo and Teredo creates a major security hole in local networks. Outgoing traffic that is expected to be filtered out by the IPv4 firewall (e.g., TCP traffic on port 25), is, in fact, transmitted via a UDP packet on 3544 and is decapsulated outside the network. Unlike 6to4 or ISATAP tunnelling, both of which exploit IP encapsulation with protocol 41, Teredo encapsulates IPv6 into UDP. There is no effective method of disabling Teredo or filtering all Teredo traffic. The initial communication between the Teredo server and UDP port 3544 is easily recognisable, but the port assignment can easily be changed. A list of Teredo security threats can be found in [10].

3. *ISATAP* [7] connects dual-stack (IPv6/IPv4) nodes over IPv4 networks via a virtual Non-Broadcast Multi Access (NBMA) data link. ISATAP hosts communicate by tunnelling IPv6 packets over IPv4 using protocol 41. The IPv4 addresses are encoded in the low-order bits of the IPv6 addresses, allowing automated tunnelling.

The monitoring of IPv6 tunnels includes (i) the detection of the tunnel based on encapsulation in IP (6to4, ISATAP) or in UDP (Teredo), and (ii) the analysis of encapsulated packets, i.e., the detection of encapsulated IP addresses and ports.

4.2 IPv6 Addressing Issues

1. *Temporary IPv6 Addresses:* Auto-configuration is a new IPv6 feature that allows a node to automatically generate an IPv6 address on its own. This behaviour is different from IPv4 address configuration, in which the IP address is configured either manually or using DHCP. An IPv6 node can be configured through either stateless or stateful autoconfiguration. The basic stateless configuration [18] combines a network prefix obtained from the router with the IEEE EUI-64 identifier based on the MAC address. This allows keeping the link between an address and user/host, but the host part of the address can easily be tracked all over the Internet.

   Because of user privacy, IPv6 addresses with randomly generated 64-bits interface identifiers are preferred instead of IEEE EUI-64. The RFC 4941 standard [13] defines a way to generate and change temporary addresses. The important requirement is that the sequence of temporarily generated addresses on the interface must be totally unpredictable.

   However, this requirement contradicts the need to identify a malevolent user. Private, temporary addresses hinder the unique identification of users/hosts connecting to a service. This affects logging and prevents administrators from effectively tracking which users are accessing what services.

   Many internal resources require the ability to track the end user's use of services. IPv6 address tracking (or data retention) is also a legal obligation of ISPs, required by governments. If a local security policy requires better control, either fixed IPv6 addresses must be centrally assigned and logged [8], or stateless configuration using DHCPv6 [15] has to be deployed. If stateless auto-configuration is deployed, a new monitoring system is required.

2. *Stateless Auto-configuration and Neighbor Discovery:* In the IPv6 protocol, discovery of a link address from the IP address is not done using a stand-alone protocol such as the Address Resolution Protocol (ARP), but via the built-in Neighbour Discovery protocol (ND) [18] together with router advertisement, redirection, or detection of duplicate addresses as defined in [19]. This new approach creates new security issues.

   ND communication is sensitive to spoofing, cache poisoning, and denial-of-service attacks. To deal with these threats as described in [14], a Secure Neighbor Discovery protocol (SEND) was proposed [1]. SEND uses Cryptographically Generated Addresses (CGA), RSA signatures and certification paths. Unfortunately, this protocol has not yet been supported in any widespread operating systems, such as Windows 7, Linux or Mac OS.

3. *Router advertisement and 6to4 tunnelling:* If a host has an assigned public IPv4 address from its ISP, it can act as a 6to4 router. This situation occurs when Internet Connection Sharing (ICS) is enabled on a host

with Windows 7 or Vista. The host then advertises the 6to4 prefix via router advertisement (RA) messages. Using these messages, the host learns its configuration (network prefix, gateway).

This behaviour is similar to a rogue DHCP server in IPv4; there rogue DHCP servers are eliminated using DHCP snooping. For IPv6, deployment of SEND on all L2 devices can solve the problem. However, as discussed above, SEND has not yet been deployed. Another solution is proposed by the IETF draft IPv6 RA-Guard [6]. The RA-Guard function is almost the same as RA snooping but is rarely implemented on L2 devices.

5 Central System for IPv6 Monitoring

This chapter describes how the issues of IPv4 and IPv6 monitoring discussed above are solved at the BUT campus network. The BUT campus network includes 134 active routing devices on the backbone and thousands of connected users (especially students). The chapter presents the data and data sources required for monitoring and how they are obtained. Some results and statistics about IPv4 and IPv6 traffic are given at the end of this chapter.

5.1 Current IPv4 Monitoring

Today, ISPs identify their hosts based on the hosts' IPv4 addresses. Usually the ISP has a central system for user registration with the users' MAC addresses. The user registers his MAC address in the system. The MAC address is used in the DHCP configuration to assign a corresponding IPv4 address. Registered MAC addresses, together with system logs of DHCPv4 servers and data from RADIUS servers, are sufficient to uniquely identify the user, based on the IPv4 address. For a long-term history, Netflow data are gathered using special hw-accelerated Netflow probes, working on 10 Gb/s links. DHCP logs, RADIUS logs and Netflow records are stored at the central monitoring system, where the users' activity can be looked up, as required by the Data Retention Act.

5.2 Practical Configuration of IPv6 Addresses at BUT

User monitoring of IPv6 traffic is more complicated. The IPv6 address is no longer a unique identifier, as in the case of an IPv4 address. This is mainly because of temporary addresses, as described above. There are two ways to assign IPv6 addresses. Practical experience at BUT indicates that stateful configuration using DHCPv6 does not work properly, so only stateless configuration can be deployed.

1. *Stateful IPv6 configuration:* The DHCP Unique Identifier (DUID), as introduced in [15], can be used to identify a user in an IPv6 network. However, DUID has several disadvantages. Its value is not easily searchable, since every client stores its values at different places on the local disk. The value is changed whenever the operating system is reinstalled. Experience at BUT shows that using stateful configuration for address assignment is extremely difficult. First of all, even if an IPv6 address is assigned to a host with Windows 7 or Vista using DHCPv6, the host will not use this address for communication but will use a temporary address instead. Secondly, the DHCPv6-client is not supported in Windows XP, which is still widely in use.

2. *Stateless IPv6 configuration:* The first part of the IPv6 address – the network prefix - is assigned using RA messages as described in the previous chapter. RA messages do not provide any type of unique identifiers that could be used to identify the host. The second part of the IPv6 address – the interface ID - is generated using EUI-64 or privacy extensions. EUI-64 could be used for host identification since its value is derived from the MAC address. However, Windows 7 and Vista use randomly generated interface ID's instead, by default. Thus, neither stateful nor stateless configuration provides the unique ID needed for user identification. More information has to be obtained, as discussed in the following section.

5.3 IPv6 Host Identification

As explained in the previous sections, a new, unique identifier is needed to identify a host in an IPv6 network. One solution is to collect various sorts of data obtained from devices on the network. The information is described in Table 1.

OSI layer	Data	Source Information
L2	RADIUS log (using 802.1x)	Login, MAC address
L2	Switching table	Switch port, MAC address
L3	Router Neighbor Cache	IPv6 address, MAC address
L3	Router ARP table	IPv4 address, MAC address
L4-7	Netflow records	IPv4/IPv6 addresses, ports

Table 1: Data input for the central monitoring system

All of these pieces of information, together, provide a complex view of the network and can help to identify a host. A tuple *(IPv6 address, MAC address, Login name)* is sufficient to identify a host/user. In practice, an extended tuple is built: *Timestamp, IPv6 address, MAC address, Switch port, Login*, as shown in Figure. 1.

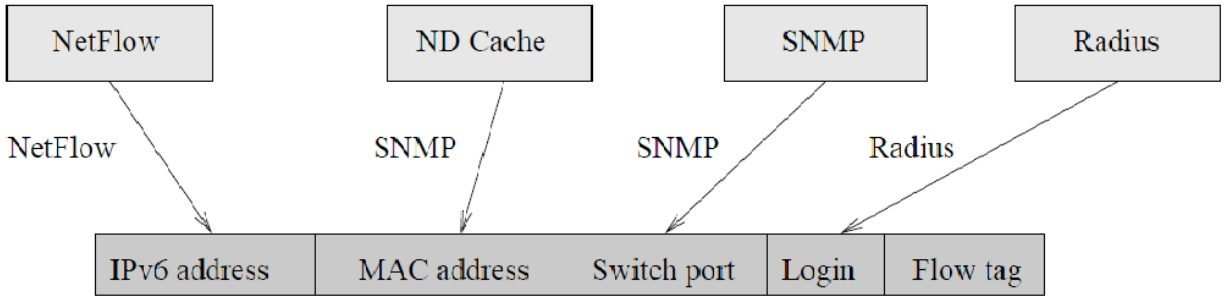

Figure 1: Data collection for the central monitoring system

Timestamp is added to provide a history of communication. Switch port is necessary if the user is blocked or if an unregistered MAC address is used on some port. In addition to these values, the VLAN number and interface statistics are stored; however, these data are not necessary for host identification.

It is important to note that this data structure is not created at once, but it is filled in when data are available. For instance, Netflow data are taken from the Netflow probe when they are sent to the collector. However, there is no information about MAC addresses that are downloaded later from the switch's ND cache. Login data from RADIUS can also be added. However, RADIUS data are not available for every user - only for those who are connected using 802.1x authentication. For other users, only the IPv6 address and the switch port number are used for identification.

5.4 Collecting Monitoring Data

Data are collected using the SNMP protocol and stored in the central database where the network administrator can search data using the IPv6, IPv4 or MAC addresses as keys. SNMP pools the data from switches every fifteen minutes. The mapping between the IPv6 address and its corresponding MAC address is downloaded from the router's neighbour cache. Port, VLAN number and other information comes from the switch's FDB (Forwarding Database) table[2]. Traffic statistics are obtained from Netflow. Netflow records alone are not sufficient for user surveillance and activity tracking because of the temporary IPv6 addresses. Therefore, Netflow records are extended by additional information called *flow tags*. The flow tag is added to a flow record after its creation, usually when the information is received and stored at the main database. The tag is a unique identifier of the user, because Netflow records are generated for every single connection of the user, even with different IPv6 addresses. Flow tags can be used as keys to identify the activities of any user stored in the system. This is necessary because not all data are available immediately in the central monitoring system, for example, due to a delay caused by SNMP pooling. When flow tags are added, more complex statistics based on the flow data is created, for example, top-N users.

The time dependency of the gathering of different data is crucial when accessing the ND Cache. This temporary memory at the router stores information needed to build the link between the IPv6 address and the MAC address. Because IPv6 addresses change in time and have limited validity, if the ND entry is lost, there is no way to link the IPv6 address and the user/host. The ND Cache is accessed via SNMP. To ensure that all information is stored properly in the monitoring system, the SNMP polling interval has to be shorter than the timeout of the ND Cache. Otherwise, some entries in the ND Cache could expire without being downloaded into the central system. Typical timeouts for collecting data are five minutes for Netflow, and fifteen minutes for RADIUS and SNMP polling. The ND Cache expiration is more than one hour.

The combination of Netflow records with SNMP and RADIUS data fills the gap and makes the central monitoring system usable for both IPv4 and IPv6. The system is depicted in Figure 2.

[2] Because older devices support different MIB standards, the ipNetToPhysical table [17] is used to obtain these data.

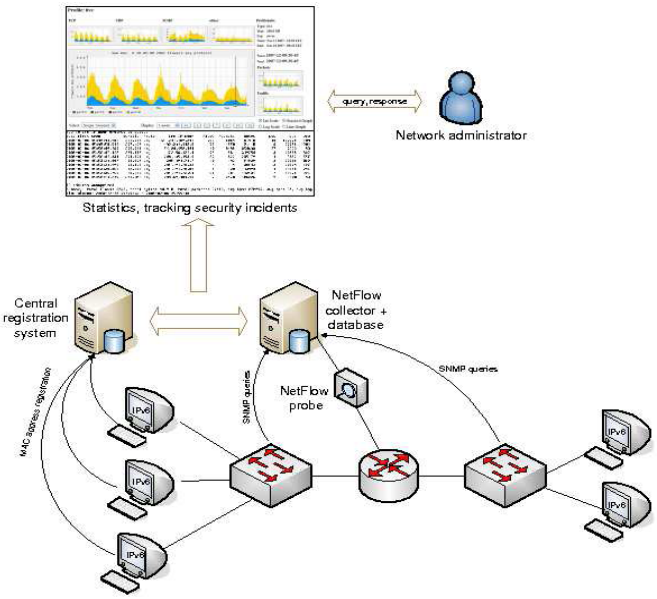

Figure 2: The Central Monitoring System for IPv4 and IPv6 at BUT

There is a huge amount of monitoring data. Daily Netflow data reaches 9 GB of compressed data (18 GB of uncompressed data).Therefore, the size of database grows rapidly. For example, data from routers at student dormitories are taken every fifteen minutes. Because there are several thousands of students at six dormitories, about 600,000 entries need to be added to the database every week.

5.5 Results

The following statistics cover the BUT university campus network, with 2,500 staff users and more than 23,000 students. The top utilisation is at student dormitories where more than 6,000 students are connected via 100 Mb/s and 1Gb/s links. The core of the BUT network is based on 10 Gb/s technology and external connection to CESNET (the Czech academic network). The IPv6 connectivity on campus is implemented according to the Internet Transition Plan [5]. Some parts of university already provide native IPv6 connectivity. However, when a user connects his PC with Windows 7 to an IPv4 segment, tunnelling is used. Figure 3 shows the ratio of IPv4, IPv6 and tunnelled traffic. IPv6 and tunnels account for only a small proportion of the traffic.

The more interesting result is the number of communications via unique IPv4 and IPv6 addresses in Figure 4. The statistics are measured every hour, eliminating discrepancies created by temporary IPv6 addresses. The number of hosts using native connectivity does not exceed 142. The number of hosts using IPv6 via tunnels is almost three times higher. This number will decrease in the future, through the growing number of users with native IPv6 connectivity.

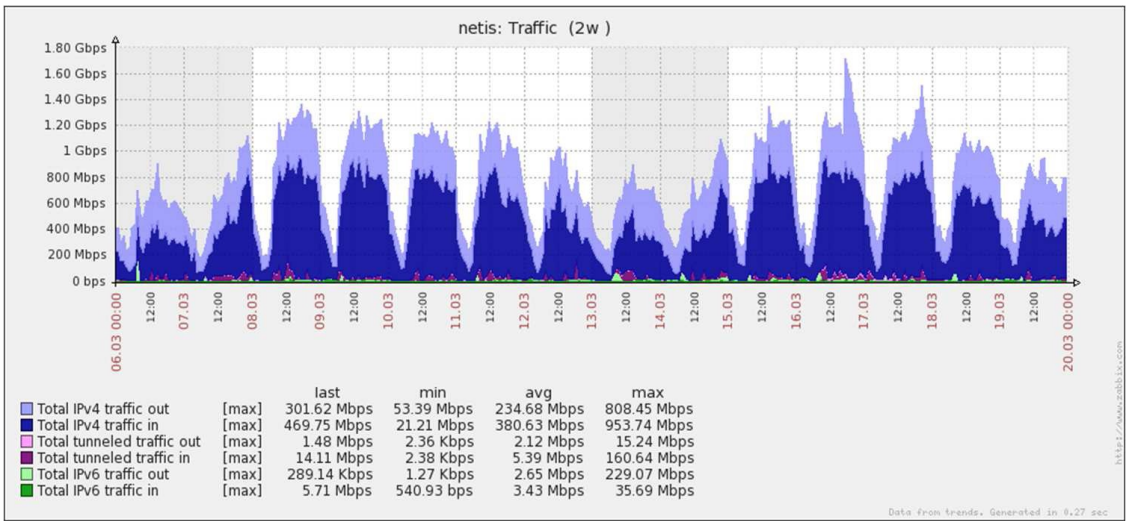

Figure 3: IPv4 and IPv6 traffic

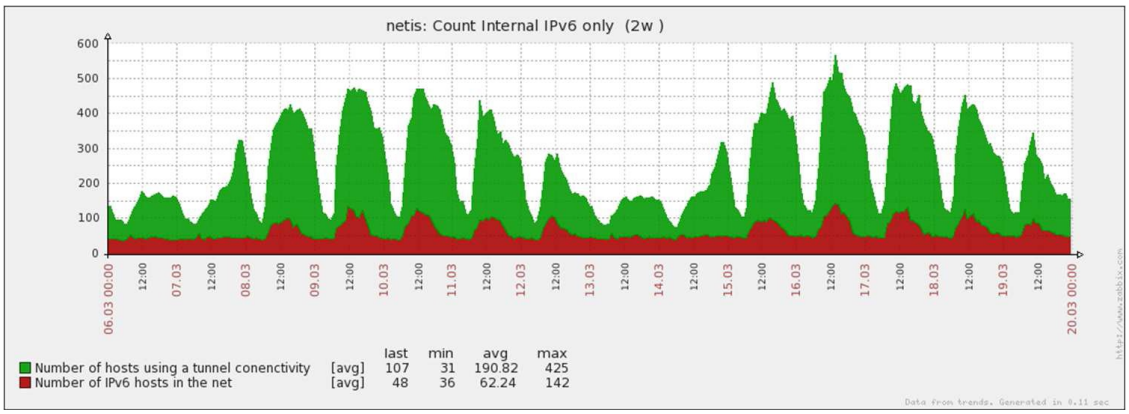

Figure 4: IPv6 tunnelled and native traffic

For comparison, the number of unique IPv4 addresses used during one hour is, on average, around 70,000. An IPv4 address is considered to be unique if it sends more than three packets, which is considered sufficient to eliminate port scanning.

Table 2 shows the proportion of tunnelling protocols on the BUT network. Native IPv6 traffic is greater than Teredo (UDP, port 3544) or 6to4, ISATAP (IP, protocol 41). The volume of native IPv6 traffic is higher than the tunnelled traffic. This is because native IPv6 servers produce more connectivity than individual users with tunnelled connections. These statistics also differ from those of Google [9], where tunnelling traffic is greater than native IPv6 traffic. This is because:

- BUT offers native IPv6 connectivity that has higher priority in operating systems than tunnelled connectivity.
- 6to4 tunnelling traffic is more frequent than Teredo traffic because every node in the BUT network has a public IPv4 address, so that NAT is not needed. As discussed above, the 6to4 tunnelling mechanism is used as the first option.

Protocol	Bytes sent	%	Packets sent	%
protocol 41	13.918 GB	0.21	27.074 M	0.3
udp 3544	32.087 MB	0.00047	0.4 M	0.0045
native IPv6	32.087 MB	1.83	131.82 M	1.45
IPv4	6450.225 GB	97.478	8.910 G	98.23
total	6617.079 GB	100	9.070 G	100

Table 2: IPV4, IPV6 native and tunnelling connections

The Netflow records contain about 63 GB of compressed Netflow data per week (125 GB of uncompressed data). Even the small number of IPv6 hosts creates several thousands of entries.

6 Conclusion

This document presents monitoring issues related to the IPv6 protocol, the solution implemented at the BUT campus network, and preliminary results of this implementation. IPv6 presents new monitoring challenges due to temporary addresses and tunnelling. The main issue is how to identify a host/user. The solution presented here is based on Netflow, SNMP and other data records. Preliminary results from the BUT network demonstrate the viability of this approach for monitoring larger networks. However, there are still open issues that are challenges for IPv6 monitoring. One is the reliability of transmission of monitoring data, because Netflow and SNMP use UDP. When a network is under attack, important data can be lost and monitoring statistics will be incomplete. There is also the challenge to build a new Netflow collector that is optimised for high-volume data. Current systems based on MySQL databases are more suited to small networks. Effective algorithms and data structures for fast lookup are needed for data processing and information retrieval.

7 References

[1] J. Arkko, J. Kempf, B. Zill, and P. Nikander. SEcure Neighbor Discovery (SEND) . RFC 3971, March 2005.

[2] C. E. Caicedo, J. B. Joshi, and S. R. Tuladhar. Ipv6 security challenges. Computer, 42:36–42, 2009.

[3] B. Carpenter and K. Moore. Connection of IPv6 Domains via IPv4 Clouds. RFC 3056, February 2001.

[4] N. Choi, H. Son, Y. Lee, and Y. Choi. Experiences with IPFIX-based Traffic Measurement for IPv6 Networks. In Proc. of IPv6. ACM, 2007.

[5] J. Curran. An Internet Transition Plan. RFC 5211, July 2008.

[6] E.Levy-Abegnoli, G. de Velde, C. Ppoviciu, and J. Mohacsi. IPv6 RAGuard. draft-ietf-v6ops-ra-guard-04, November 2009.

[7] D. T. F. Templin, T. Gleeson. Intra-Site Automatic Tunnel Addressing Protocol (ISATAP). RFC 5214, March 2008.

[8] S. Frankel, R. Graveman, and J. Pearce. Guidelines for the secure deployment of ipv6 (draft). Technical Report 800-119, National Institute of Standards and Technology, 2010.

[9] S. H. Gunderson. Global IPv6 statistics. Measuring the current state of IPv6 for ordinary users. In Proc. of 73 IETF, 2008.

[10] J. Hoagland, S. Krishnan, and D. Thaler. Security concerns with ip tunneling. Internet-Draft draft-ietf-v6ops-tunnel-security-concerns-02, IETF, March 2010.

[11] E. Hofig and H. Coskun. Intrinsic Monitoring Using Behaviour Models in IPv6 Networks. LNCS, 5844:86–99, 2009.

[12] C. Huitema. Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs). RFC 4380, February 2006.

[13] T. Narten, R. Draves, and S. Krishnan. Privacy Extensions for Stateless Address Autoconfiguration in IPv6. RFC 4941, September 2007.

[14] P. Nikander, J. Kempf, and E. Nordmark. IPv6 Neighbor Discovery (ND) Trust Models and Threats. RFC 3756, May 2004.

[15] R.Droms, J.Bound, B.Volz, T.Lemon, and C.Perkins. Dynamic Host Configuration Protocol for IPv6 (DHCPv6) . RFC 3315, July 2003.

[16] P. Savola and C. Patel. Security Considerations for 6to4 . RFC 3964, December 2004.

[17] S.Routhier. Management Information Base for the Internet Protocol (IP). RFC 4293, April 2006.

[18] S.Thomson, T.Narten, and T.Jinmei. IPv6 Stateless Address Autoconfiguration. RFC 4862, September 2007.

[19] T.Narten, E.Nordmark, W.Simpson, and H.Soliman. Neighbor Discovery for IP version 6 (IPv6). RFC 4861, September 2007.