An Investigation Into Teredo and 6to4 Transition Mechanisms: Traffic Analysis

Martin Elich*, Petr Velan†‡, Tomas Jirsik‡, Pavel Celeda‡

*Faculty of Informatics, Masaryk University, Brno, Czech Republic
elich@mail.muni.cz
†CESNET z.s.p.o., Prague, Czech Republic
petr.velan@cesnet.cz
‡Institute of Computer Science, Masaryk University, Brno, Czech Republic
{velan|jirsik|celeda}@ics.muni.cz

***Abstract*—The exhaustion of IPv4 address space increases pressure on network operators and content providers to continue the transition to IPv6. The IPv6 transition mechanisms such as Teredo and 6to4 allow IPv4 hosts to connect to IPv6 hosts. On the other hand, they increase network complexity and render ineffective many methods to observe IP traffic. In this paper, we modified our flow-based measurement system to involve transition mechanisms information to provide full IPv6 visibility. Our traffic analysis focuses on IPv6 tunneled traffic and uses data collected over one week in the Czech national research and education network. The results expose various traffic characteristics of native and tunneled IPv6 traffic, among others the TTL and HOP limit distribution, geolocation aspect of the traffic, and list of Teredo servers used in the network. Furthermore, we show how the traffic of IPv6 transition mechanisms has evolved since 2010.**

***Index Terms*—Teredo, 6to4, IPv6, Transition Mechanisms.**

I. Introduction

Despite IPv6 being the standard for several years its adoption is still in process [5]. There are several ways of getting IPv6 connectivity, the dual-stack being the preferred one. Most IPv6 studies deal with native IPv6. However, there are other globally used options known as transition mechanisms. They can provide IPv6 connectivity on networks without native IPv6 connectivity enabled or without an IPv6 ready infrastructure.

The transition mechanisms tunnel IPv6 traffic through IPv4 network. Despite being supported by major operating systems, there is a lack of studies investigating the characteristics of the tunneled IPv6 traffic. In this context, this paper investigates border traffic of the Czech national research and education network operator (CESNET) and attempts to answer the following question: *What are the characteristics of IPv6 transition mechanisms, in terms of their usage, popularity and impact on native IPv4 and IPv6?*

Our research is mainly motivated by an exhaustion of the IPv4 address space and an exerting pressure on network operators and content providers to deploy IPv6. The transition mechanisms are used to facilitate the IPv6 adoption. Unfortunately, they introduce extra elements in the network which add to complexity and decrease performance and security. As a result, many existing methods for measuring and monitoring large-scale networks become ineffective.

The contribution of our work is threefold: *(i)* we provide an enhanced version of our flow-based IPv6 measurement system prototype, which enables IPv6 visibility in large-scale networks, *(ii)* we analyze and show IPv6 transition mechanisms traffic characteristics including a tunneled one and *(iii)* we show how the traffic of IPv6 transition mechanisms has evolved since 2010.

The paper is organized as follows. Section II outlines related work. Section III provides an overview of Teredo and 6to4 transition mechanisms. Section IV describes the methodology and measurement setup. Section V investigates properties of IPv4 traffic carrying IPv6 payload. Section VI focuses on the characteristics of encapsulated IPv6 traffic. Section VII evaluates the use of IPv6 tunneled traffic. Finally, we draw conclusions and outline future work in Section VIII.

II. Related Work

The most widely used and discussed tunneling transition mechanisms are Teredo and 6to4. Although there are several studies focusing on performance evaluation of transition mechanisms listed below, the characteristics of the traffic generated by the tunneling transition mechanisms are not well known.

A study by Aazam et al. [1] provides a performance evaluation and a comparison of Teredo and Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) mechanisms with focus on certain parameters like throughput, end to end delay, round trip time and jitter. A study by Zander et al. [16] compares Teredo tunneling capability and performance with native IPv6 and 6to4 using measurements related to web services. Teredo increases the time needed to fetch web objects compared to IPv4 or native IPv6. The conclusion is that Teredo seems to be limited by a lack of Teredo infrastructure forcing encapsulated packets to travel long distances. Moreover, the throughput is partially limited by the performance of Teredo relay servers.

A study by Bahaman et al. [2] discusses the performance of 6to4 with focus on communication over TCP. It states that the TCP transmission ability is reduced by the use of 6to4. However, it is still suitable for early stages of the transition period.

978-1-4799-0539-3/13/$31.00 ©2013 IEEE

Other papers discuss the impact of transition tunnels on network security. Krishnan et al. [11] present security concerns with recommendations on how to minimize security exposure due to tunnels. It is pointed out that tunnels can have negative impact on deep packet inspection and that transition mechanisms such as Teredo allow inbound access from the public Internet to a device through an opening created in a network address translation (NAT) device. This increased exposure can be used by attackers to effectively attack a device hidden behind a NAT device. A generally proposed security practice is to avoid the usage of tunnels at all and deploy other transition schemes like dual-stack.

Finally, Sarrar et al. [14] provides a brief insight into tunneled traffic in a study of the world IPv6 day impact on IPv6 traffic. The Teredo and 6to4 transition mechanisms were monitored and the Teredo was discovered to carry mainly control traffic. The study also showed that IPv6 fragments were responsible for a significant portion of 6to4 traffic. The authors suspect that these fragments were caused by broken software which most likely forgot to take the IPv6 header size into account.

III. Investigated IPv6 Transition Mechanisms

The IPv6 traffic is usually divided between *native* traffic and *tunneled* traffic. The tunneled traffic is considered the one encapsulated using other protocols, e.g. UDP or IP protocol 41. This division is not necessarily accurate, since the traffic that seems to be native IPv6 can in fact originate from a client using some transition mechanism like Teredo or 6to4. To clarify this point we will differentiate between native IPv6 traffic, encapsulated tunnel traffic (IPv4 traffic containing IPv6 payload) and decapsulated tunnel traffic. The word tunnel might be omitted for the sake of brevity.

Teredo and 6to4 are the two most frequently used transition mechanisms in the CESNET network. Mechanisms like ISATAP, Anything in Anything (AYIYA) and others based on IP protocol 41 (6in4, 6over4) do not contribute to the tunneled traffic significantly and do not appear in our analysis, therefore we will not describe them in detail. We did not analyze NAT64 and DNS64 mechanisms since they should appear as a native IPv6 traffic on the outside.

Teredo [9] is designed to provide IPv6 connectivity to an endpoint behind a NAT device. It requires two network components for operation: *relays* and *servers*. Teredo servers are used for initialization of Teredo (Fig. 1, communication ①), and after that for opening a port on the user's NAT device in case of communication which is not initialized by the user. Relays are used for routing and bridging the IPv4 and IPv6 networks. Each Teredo endpoint uses a statically configured server and a relay, which can cause increased latency and low throughput in case of a distant server or relay. Teredo uses UDP for packet encapsulation making the traffic harder to identify.

6to4 [3] is only suitable for hosts with a public IPv4 address. It uses encapsulation in IP protocol 41 packets hence it is relatively easy to detect and monitor. The 6to4 relay servers

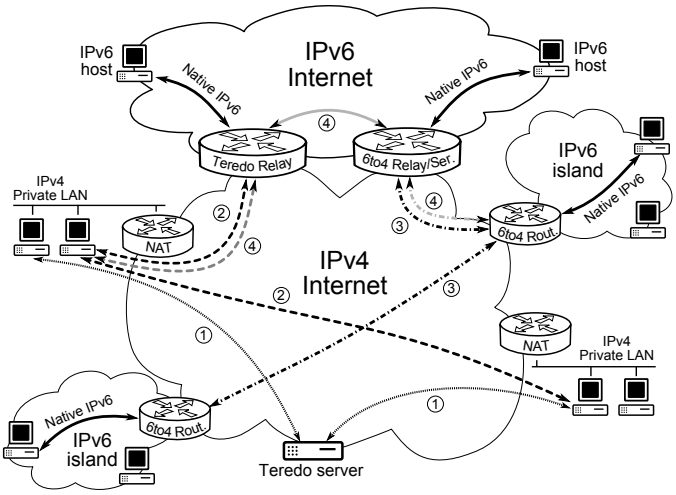


Fig. 1: Teredo and 6to4 principles: ① Teredo start setup, ② Teredo traffic transiting over IPv4 network, ③ 6to4 traffic transiting over IPv4 network, ④ communication between Teredo and 6to4 endpoint

are acting as a bridge between the IPv4 and IPv6 networks. These relays use any-cast prefix 192.88.99.0/24 therefore the optimal (nearest) relay server should automatically be used for communication.

Fig. 1 shows traffic between two endpoints (communication ④), one of which uses Teredo and the other one 6to4. The IPv6 traffic from Teredo client travels part of its path in Teredo tunnel to be later decapsulated on the edge of IPv6 Internet and shortly after that to be encapsulated again, this time by 6to4 to travel the rest of its path over IPv4 Internet to the network of its destination. Depending on where the observation point is located, the tunnel (either Teredo or 6to4) or native IPv6 traffic can be observed.

IV. Methodology and Measurement Setup

To perform a thorough inspection of tunneled traffic, we need to decapsulate packet headers of inner packets. We use the same flow-based framework as in [7] which has been further modified [8] to extract more detailed information from tunneled data. The main part of the framework is a plug-in which replaces input and processing parts of existing flow generator INVEA FlowMon Exporter [10].

Every packet is being processed to extract basic flow statistics and the processing of inner headers continues to the point when previously extracted fields indicate absence of observed encapsulation types.

Teredo protocol is detected when IPv6 header is found encapsulated in UDP packet, AYIYA is searched for in packets on TCP or UDP port 5072. Other protocols are recognized by IPv6 address format, which is protocol specific and the 6to4 protocol can be additionally identified by usage of IPv4 anycast address belonging to 6to4 relay. If encapsulation is present, its type and encapsulated IPv6 header fields are used to extend the set of extracted fields and to identify individual flows taking place inside the tunnel.

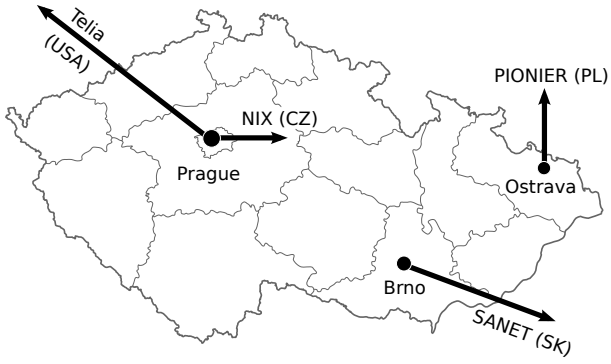


Fig. 2: CESNET monitored links

Probe	Bits/s	Packets/s	Flows/s
Telia	1.65 G	274.9 k	22.1 k
NIX	7.17 G	1072.4 k	26.7 k
PIONIER	0.51 G	75.6 k	2.8 k
SANET	1.87 G	242.3 k	5.3 k

TABLE I: Observation points IPFIX statistics

Since we need to define new elements for flow records, Internet Protocol Flow Information Export (IPFIX) protocol is used. It allows using *Enterprise Elements* which can extend flow records with additional tunnel information. The framework is able to recognize and extract information from Teredo, AYIYA and other protocols that are based on IP protocol 41 such as ISATAP, 6to4 and 6over4. We provide the source code of the measuring tool under BSD license at the project web page [8].

Resulting flow data provides us with information about the encapsulated source and destination addresses, ports and transport protocol, which is a common five-tuple used to distinguish individual flows. We respect this principle and thus have separated the flows encapsulated in the same tunnel based on the value of these elements. Apart from these key elements the framework gives information about *Time to Live* (TTL), *encapsulated HOP limit*, *TCP flags* and *ICMPv6 type* and *code*, when present. Moreover, additional information about tunnel type is provided, including Teredo header and trailer types when present. The framework also newly supports geolocation using MaxMind [12] GeoIP database for both outer and encapsulated addresses.

The data are collected from several observation points located at the borders of CESNET network by passive probes; see Fig. 2. All measured lines are at least 10 Gbit/s and together transport about 80,000 flows/s during work hours, which results in total traffic of 15.4 Gbit/s. We use IPFIXcol [15] framework to collect the extended flow data over TCP and to store them. New elements can therefore be defined and used without any further difficulties or limitations.

The IPFIX data was collected over one week in January 2013 without use of any sampling. Table I shows the average amount of traffic for all observation points. The total amount of stored data took approximately 2,485 GB of disk space. All statistics presented below are based on flow count.

V. Characteristics of IPv4 Tunnel Traffic

In this section we describe characteristics of IPv4 traffic containing IPv6 payload. The analysis is based purely on information from IPv4 headers and extended flow data are only used to accurately identify relevant flows. Three characteristics that can give us insight into tunneled traffic are addressed. Firstly, we describe TTL values of the various traffic sets, and then we look into a geolocation aspect. Lastly, the basic flow statistics are presented.

A. Frequency of TTL and HOP Values

We study the distribution of *TTL values* of the observed flows. It is known that some operating systems use specific values, as shown in [6]. Microsoft Windows has the default TTL set to 128. The value of 64 is mostly used by Mac OS X and Linux devices, including devices running Android. We expect that these operating systems form a majority and are therefore the most significant. Fig. 3 shows the most frequently used TTL values for IPv4 flows carrying IPv6 payload. The TTL values are most frequent near the values set by OS vendors and the frequency is decreasing rapidly in less than ten hops. Therefore, we assume that most of the packets reach their destination in less than 32 hops. Thus we classify the flows according to their TTL numbers into four significant groups. The Windows traffic seems to be the most frequent one taking 60.3 % of the total, while Linux machines are not present so often with only 23.8 %. Apart from the Windows and Linux ranges, there are devices that set TTL to 255 and 32. Although the 255 are usually Cisco routers, in case of tunneled traffic we observed that the 6to4 traffic from anycast addresses have TTL set to 255 as well. The portion of the 255 range is 3.8 % and most of it originates from 6to4 relays. TTL numbers 24 and 26 are dominant in the group of values from 1 to 32, which makes 12.2 % of the total number of flows. We discovered that this is caused by a 6to4 tunnel that passes two observation points. The tunnel is heavily used and causes a large portion of tunneled traffic, which also affects other 6to4 measurements.

Overall, the TTL distribution of IPv4 traffic is different as shown in Fig. 4. The Linux portion of the traffic is higher and the TTL values of 32 and 255 are not as significant. A more detailed examination of the flow records shows that this is caused mainly by a high ratio of HTTP traffic. Even though there are more clients using Windows operating system, most of the web servers are based on Linux and therefore the responses have TTL less than 64. The DNS protocol shows similar characteristics except that the DNS traffic that we observe at our metering points is mostly generated by recursive domain name servers. Since the Linux DNS servers are the most widely used, they significantly contribute to the traffic generated by Linux machines.

IPv6 uses *HOP limit* instead of TTL. Fig. 5 shows HOP limit distribution of native and decapsulated IPv6 traffic. Unlike

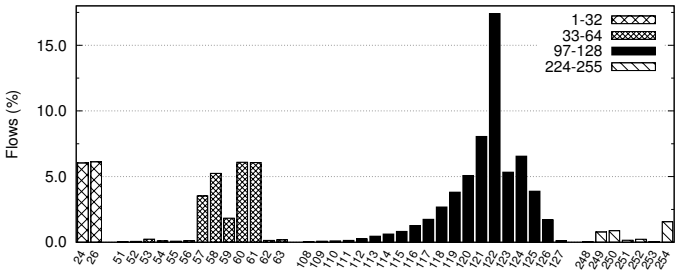


Fig. 3: TTL value distribution of IPv4 traffic containing IPv6 payload

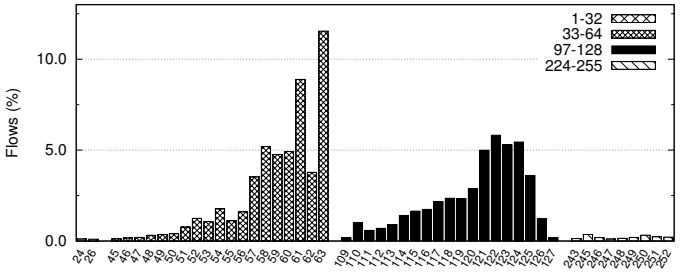


Fig. 4: TTL value distribution of total observed IPv4 traffic

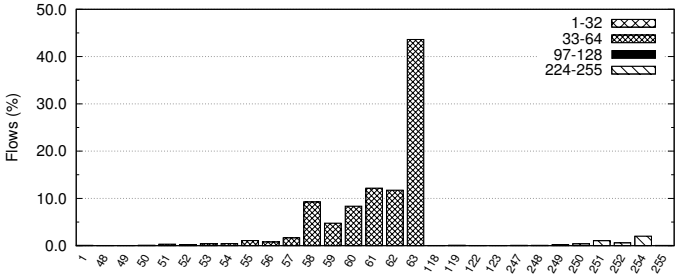
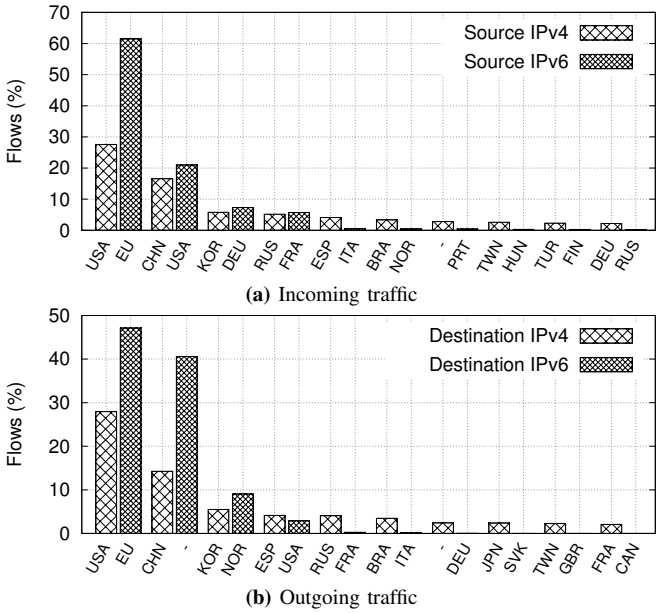


Fig. 5: HOP value distribution of IPv6 traffic

IPv4, the HOP limit of 64 is the most frequent. We assume that Linux based machines use default HOP limit 64 and Windows machines use default HOP limit 128. This setting can be overridden by Stateless Address Autoconfiguration. Therefore, clients in managed networks (e.g. universities) might have the HOP limit set to different value, regardless of their operating system. We verified this fact on several Linux and Windows based machines. Due to significant share of HTTP(S) in IPv6 traffic, large portion of Windows traffic is expected. Since the share of HOP limit 128 is negligible, we expect that common HOP limit in observed IPv6 networks is set to 64.

B. Location of IPv4 and IPv6 Endpoints

The second characteristic that we evaluate is *geolocation aspect* of the IPv4 and IPv6 traffic. We focus on data from Telia link only, which connects the CESNET network to the United States. This highlights the differences in geolocation characteristics better. The statistics are computed separately for the incoming and outgoing lines and are shown in Fig. 6. The IPv4 is more symmetric since the country statistics for both directions are similar. This is a normal behavior since most of the requests initiate a response and the routes are also



(a) Incoming traffic

(b) Outgoing traffic

Fig. 6: Top 10 country distribution for native IPv4, IPv6 and decapsulated IPv6 addresses

symmetric. The IPv6 have different properties. We discovered that the addresses that cannot be geolocated are mostly link-local addresses (fe80::/10) or local-link multicast addresses (ff02::/16). Such addresses should not be routed at all, which indicates that there are routers with erroneous IPv6 configuration. This misconfiguration also causes the asymmetry of the traffic, as such requests cannot be answered.

C. Duration and Size of Flows

The third group of characteristics is represented by flow duration, number of packets per flow and packet size (bytes per packet) statistics. For evaluation we employ *empirical complementary distribution function* (CCDF). We use the following formula to compute CCDF values:

$$\bar{F}(x) = P(X > x) = 1 - \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{x_i \leq x\} \qquad (1)$$

where $\mathbf{1}\{x_i \leq x\}$ is the indicator whether the event $\{x_i \leq x\}$ has occurred or not. The CCDF function describes how often the selected variable is above a particular level. From all traces we filtered out four subsets of traffic: TCP or UDP encapsulated traffic (TCP/UDP), all encapsulated traffic (ALL), IPv6 native or decapsulated traffic (IPv6) and IPv4 traffic (IPv4). The subsets were chosen in order to compare the tunneled traffic with other common traffic types. Further, for each of the subsets and each of the characteristics CCDF has been computed. Fig. 7, 8 and 9 show the calculated CCDFs.

The majority of the flow duration of all subsets (Fig. 7) accounts for durations shorter than 10 seconds. The flow durations longer than 10 seconds represent only 11 % or less. The TCP/UDP contains much fewer short duration flows (TCP/UDP: 32.16 % $\leq$ 0.01 sec.; ALL: 54.66 %; IPv4:

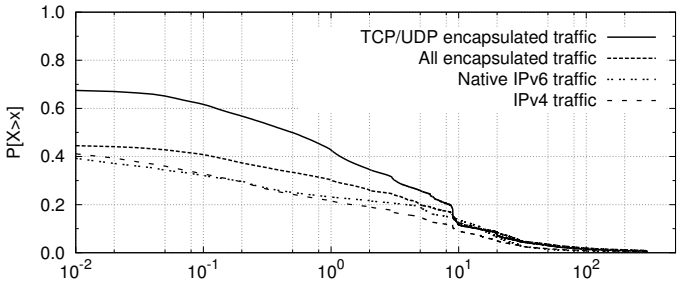


Fig. 7: CCDF of flow duration

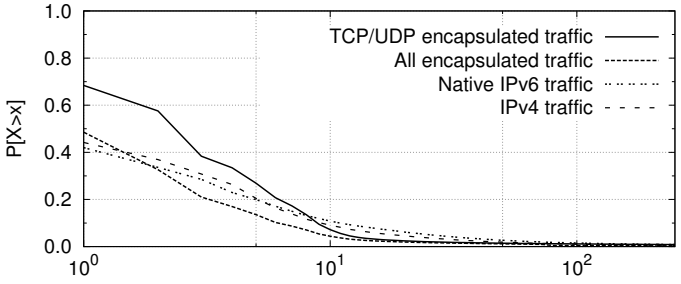


Fig. 8: CCDF of packets per flow

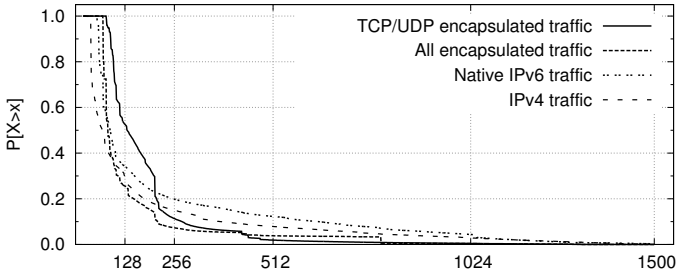


Fig. 9: CCDF of packet size

59.84 %) which is explained by the absence of connection-maintaining flows necessary for other subsets. We can observe slightly increased frequencies of the flow duration between 7 and 11 seconds especially at TCP/UDP and ALL subsets. Hence, the tunneled traffic generally contains fewer short duration flows than IPv4 or IPv6 traffic.

The packets per flow distribution (Fig. 8) suggests that single packet flows form only 31.6 % in the case of TCP/UDP, 51.42 % in the case of ALL, 57.99 % and 55.82 % in other cases. We expected tunneled traffic to behave in a similar way as IPv6 traffic. Nevertheless, we can observe a vertical shift between ALL and IPv6 CCDF. This shift is caused by single packet flows and it is caused by DNS traffic to root DNS servers, which uses IPv6 protocol. The slope of CCDF for TCP/UDP and ALL is higher than the slope for other subsets, which states higher frequencies of certain packet counts. In conclusion, the distribution of the packet counts of the tunneled traffic is slightly different from the distribution of the IPv4 and IPv6 traffic.

The last characteristic described by CCDF is packet size (Fig. 9). In the case of encapsulated traffic we consider the outer packet size including the encapsulation header. The earlier study of the packet size distribution mentions a significant difference between CCDFs of packet sizes. The authors of [4] state that the distribution of IPv4 traffic fits a heavy-tailed distribution, whereas IPv6 traffic does not. We expect the tunneled traffic to have similar characteristics as IPv6 traffic, and thus the CCDFs are expected to be similar, too. The Fig. 9 shows some discrepancies in this hypothesis. The packets larger than 400 bytes in the tunneled traffic represent only 5 %, while in the IPv6 the portion is still 15 %. Furthermore, packets smaller than 70 bytes are not present in the tunneled traffic, although they account for 26 % of IPv6. This is caused by a shift of graph to a higher packet size given by the encapsulation. The IPv4 header is usually 20 bytes long and in case of Teredo there are another 8 bytes for the UDP header. Even taking this shift into account there is still a noticeable drop at size of 200 bytes which is caused by a high share of ICMPv6 and BitTorrent control traffic.

VI. IPv6 Tunneled Traffic Analysis

In this section we focus on characteristics of encapsulated IPv6 traffic for which we use the same data set as in Section V. We show HOP limit statistics, detected Teredo servers and geolocation characteristics. We discuss the most used TCP and UDP ports inside the tunnels.

A. Distribution of HOP Limits

Fig. 10 shows HOP limit distribution for the encapsulated IPv6 traffic. The main difference from the TTL (Fig. 3) and HOP limit (Fig. 5) statistics is that the values here are distributed with much less entropy. The limits 21, 64, 128 and 255 are achieved and also the most frequent ones. This is caused by the fact that most of the traffic never traversed the IPv6 network and the HOP limit was therefore never decreased. In fact, when the values are lower, we can be reasonably certain that the packets already traversed IPv6 network and are heading towards the IPv4 destination. The value 21 is used for Teredo bubbles by Windows 7 with Service Pack 1 and earlier. The Teredo bubbles are used as a special mechanism for NAT traversal, which is consistent with the fact that most of the clients are behind a NAT. We can see that some packets reach the value of the zero HOP limit, which is a known problem when the HOP limit is set as low as to 21. The value of 255 is used for IPv6 neighbor discovery messages, so that when host receives such packet with HOP limit lower than 255, the packet is considered invalid [13].

B. Teredo Servers

There are two ways of detecting Teredo servers. Firstly, we can look at the traffic using UDP protocol on port 3544, which is a well-known Teredo port, and select the addresses that communicate most often. The shortcoming of this approach is that some other services might be using the Teredo port and therefore the results might not be accurate. Since we are able to decapsulate Teredo traffic, we can derive IPv4 addresses of Teredo servers directly from Teredo IPv6 addresses. This way

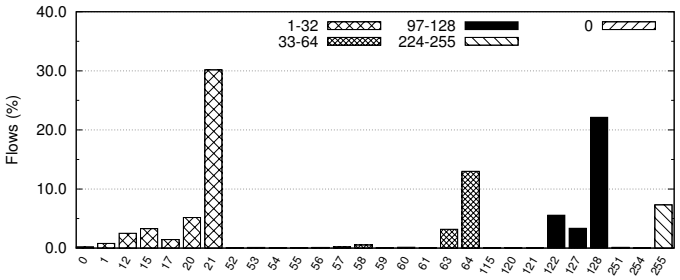


Fig. 10: HOP value distribution of encapsulated IPv6 traffic

we can even detect Teredo servers that are not communicating directly through our observation points. Table IIa shows Top 10 servers that were discovered in the encapsulated IPv6 addresses. Using the WHOIS database we confirmed that a majority of servers is operated by Microsoft, which is only to be expected since Teredo is a Microsoft technology. The most of Microsoft Teredo servers we identified are actually IP addresses of "teredo.ipv6.microsoft.com", which is the default Teredo server name configured under Windows. The address 83.170.6.76 has a hostname indicating that it serves as a Miredo server (Teredo implementation for Linux and BSD). The last address belongs to CZ.NIC (Czech top level domain operator), which is known to promote IPv6 deployment in the Czech Republic and operates local Teredo and 6to4 servers.

Table IIb shows Teredo servers discovered as the most active on Teredo port 3544. This way we detect only Teredo servers that are establishing connections through our observation points. We can see that most users use Teredo servers in the United States or Great Britain to get IPv6 connectivity. This is known to increase latency of such connections and therefore we would recommend using local servers to Czech users, such as the CZ.NIC servers.

C. Location of Tunnel Endpoints

The geolocation statistics of tunneled traffic are computed for Teredo and 6to4. We use encapsulated IPv6 addresses to determine the countries for each flow. Incoming and outgoing traffic is taken separately just as in Fig. 6. The statistics are shown in Fig. 11. The tunneled traffic shows very different geolocation characteristics compared to native and decapsulated IPv6 traffic even though both are from the same link. Most of the native and decapsulated IPv6 communication takes place inside the EU, while large portion of tunneled traffic communication is performed with the USA and Russia.

To identify applications that are using IPv6 connection provided by transition mechanisms, we created a list of the most used encapsulated TCP and UDP ports. We observed several ports that can be found both in the source and destination port Top 10. The source and destination ports Top 10 represent 32.0 % and 40.5 % of the traffic respectively. The well-known ports are - *HTTP - 80* (0.85 % of traffic as source port, 5.61 % as destination port), *HTTPS - 443* (0.58 % and 1.48 %) and *DNS - 53* (1.49 % and 1.48 %). Among the most frequent ports are ports 49001 (15.96 % and 9.91 %) and 51413 (10.12 % and 16.07 %) which are used by BitTorrent clients (namely Vuze and Transmission). We discovered that these ports are heavily used within the 6to4 tunnel as mentioned in Section V.

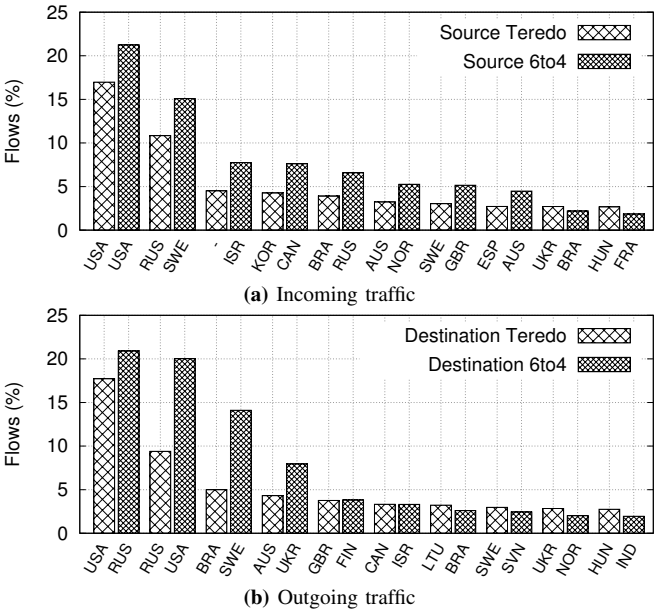
Server IP	Ratio	Owner	Country
65.55.158.118	28.33 %	Microsoft	US
94.245.121.253	27.98 %	Microsoft	GB
157.56.149.60	26.49 %	Microsoft	US
157.56.106.184	10.18 %	Microsoft	US
94.245.115.184	6.41 %	Microsoft	GB
83.170.6.76	0.04 %	B. Schmidt	DE
170.252.100.131	0.01 %	Accenture	US
94.245.127.72	0.01 %	Microsoft	GB
94.245.121.251	0.01 %	Microsoft	GB
217.31.202.10	0.01 %	CZ.NIC	CZ

(a) Based on Teredo IPv6 addresses

Server IP	Ratio	Owner	Country
94.245.121.253	43.24 %	Microsoft	GB
65.55.158.118	18.91 %	Microsoft	US
157.56.149.60	17.86 %	Microsoft	US
94.245.115.184	10.00 %	Microsoft	GB
157.56.106.184	6.50 %	Microsoft	US
94.245.121.254	0.72 %	Microsoft	GB
94.245.115.185	0.22 %	Microsoft	GB
65.55.158.119	0.18 %	Microsoft	US
83.170.6.76	0.18 %	B. Schmidt	DE
157.56.149.61	0.17 %	Microsoft	US

(b) Based on UDP port 3544

TABLE II: Top 10 discovered Teredo servers



(a) Incoming traffic

(b) Outgoing traffic

Fig. 11: Top 10 country distribution for encapsulated IPv6 addresses

VII. Evaluation of IPv6 Adoption

In this section we describe the deployment of the IPv6 protocol with respect to tunneled traffic. The overall statistics

of IPv6 and tunneled traffic are mentioned. We provide a historical comparison to our previous measurement [7].

The network activity shows the correlation with human activity. Both IPv6 and tunneled traffic are considerably smaller during the weekend than during week days. As for the IPv6 traffic, the increase of traffic volume starts at 6 AM, reaches the peak around 11 AM and holds the high level till 4 PM. Then the traffic steadily decreases and reaches the minimum at 3 AM the next day. The tunneled traffic shows slow increase that starts at 6 AM and peaks around 6 PM. The decrease begins at 10 PM and reaches the minimum at 5 AM the next day. The possible cause of this shift from the IPv6 diurnal pattern is the fact that the tunneled traffic is widely made by BitTorrent clients.

We measured the tunneled traffic back in 2010 on three CESNET border links to SANET, PIONIER and NIX. We found that the tunneled IPv6 was responsible for 1.5 % of total flows, which is the same share as we measured today. But the relative amount of bytes transferred has almost doubled from 0.66 % to 1.28 % of total bytes today. The share of the native and decapsulated IPv6 was only 0.10 % (0.21 % of bytes) compared to 3.39 % (4.42 % of bytes) today. The known services by port (HTTP, HTTPS and DNS) had a share less than 1 % of total flows. Today's share of these services is significantly higher (see Section VI). The measurements show that the overall usage of both tunneling IPv6 transition mechanisms and native IPv6 has been raising.

We distinguish between the encapsulated and decapsulated tunneled traffic, as mentioned in Section III. The decapsulated tunneled traffic is included in the measured IPv6 traffic. When we filter out the decapsulated Teredo and 6to4 traffic, they account together for 5.91 % of the measured IPv6 traffic. Teredo traffic takes part of 83.13 % of the decapsulated tunneled traffic and 6to4 16.87 %. Hence we should not consider all measured IPv6 traffic as native IPv6 traffic.

The main contributor to tunneled traffic was Teredo with an occurrence of nearly 89 % followed by 6to4 with over 11 %, therefore the relative amount of 6to4 traffic has increased. We then detected the use of 13 Teredo servers compared to 53 today. The complete list of the detected servers can be downloaded at the project web page [8].

VIII. Conclusion and Future Work

In this paper we have taken a detailed look at the IPv6 transition mechanisms. We have provided an improved version of our tool for investigating IPv6 tunneled traffic. Considerable progress has been made with regard to understanding tunneled traffic behavior, especially concerning Teredo and 6to4 traffic. The results of this paper suggest that encapsulated traffic differs from IPv4 and IPv6 in several characteristics including TTL values, geolocation aspect and flow duration. Moreover, we have provided the list of Teredo servers and described the evolution of IPv6 adoption.

This paper is the first step towards enhancing our understanding of encapsulated IPv6 traffic. We hope that our findings will be beneficial as a background to additional research into IPv6 transition mechanisms. To further our research we are planning to carry out an in-depth analysis of tunneled IPv6 traffic concerning the security matter. To be able to handle security incidents, the security threats will be identified and detection methods will be developed. Since our results are encouraging, they should be validated on other large-scale networks. On a wider level, research is also needed to evaluate the contribution of the IPv6 transition mechanisms to the IPv6 adoption.

Acknowledgement

This material is based upon work supported by Institute of Computer Science, Masaryk University, and also supported by the "CESNET Large Infrastructure" project LM2010005 and DMON100 project TA03010561.

References

[1] M. Aazam, I. Khan, M. Alam, and A. Qayyum, "Comparison of IPv6 tunneled traffic of Teredo and ISATAP over test-bed setup," in *Information and Emerging Technologies (ICIET), 2010 International Conference on*, 2010, pp. 1–4.

[2] N. Bahaman, E. Hamid, and A. Prabuwono, "Network performance evaluation of 6to4 tunneling," in *Innovation Management and Technology Research (ICIMTR), 2012 International Conference on*, 2012, pp. 263–268.

[3] B. Carpenter and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds," RFC 3056 (Proposed Standard), Internet Engineering Task Force, Feb. 2001.

[4] C. Çiflikli, A. Gerzer, and A. T. Özşahin, "Packet traffic features of IPv6 and IPv4 protocol traffic," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 20, no. 5, pp. 727–749, 2012.

[5] claffy, kc, "Tracking IPv6 Evolution: Data We Have and Data We Need," *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 3, pp. 43–48, Jul. 2011.

[6] N. Davids, "Initial TTL values," 2013. [Online]. Available: http://noahdavids.org/self_published/TTL_values.html

[7] M. Elich, M. Grégr, and P. Čeleda, "Monitoring of tunneled IPv6 traffic using packet decapsulation and IPFIX," in *Proceedings of the Third international conference on Traffic monitoring and analysis*, ser. TMA'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 64–71.

[8] M. Elich, P. Velan, and P. Čeleda, "FlowMon IPv6 Tunnel Monitoring Plugin," 2013. [Online]. Available: http://www.muni.cz/ics/920232/web/ipv6-tunnel-plugin

[9] C. Huitema, "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)," RFC 4380 (Proposed Standard), Internet Engineering Task Force, Feb. 2006, updated by RFCs 5991, 6081.

[10] INVEA-TECH a.s., "FlowMon Exporter – Community Program," 2013. [Online]. Available: http://www.invea.cz

[11] S. Krishnan, D. Thaler, and J. Hoagland, "Security Concerns with IP Tunneling," RFC 6169 (Informational), Internet Engineering Task Force, Apr. 2011.

[12] MaxMind, Inc., "MaxMind GeoIP services," 2013. [Online]. Available: http://www.maxmind.com

[13] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)," RFC 4861 (Draft Standard), Internet Engineering Task Force, Sep. 2007, updated by RFC 5942.

[14] N. Sarrar, G. Maier, B. Ager, R. Sommer, and S. Uhlig, "Investigating IPv6 traffic: what happened at the world IPv6 day?" in *Proceedings of the 13th international conference on Passive and Active Measurement*, ser. PAM'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 11–20.

[15] P. Velan and R. Krejčí, "Flow Information Storage Assessment Using IPFIXcol." in *AIMS*, ser. Lecture Notes in Computer Science, Sadre, Ramin and Novotný, Jiri and Celeda, Pavel and Waldburger, Martin and Stiller, Burkhard, Ed., vol. 7279. Springer, 2012, pp. 155–158.

[16] S. Zander, L. L. Andrew, G. Armitage, G. Huston, and G. Michaelson, "Investigating the IPv6 teredo tunnelling capability and performance of internet clients," *SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 5, pp. 13–20, Sep. 2012.