HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

Nedostaci u specifikaciji TCP protokola (TCP reset napadi)

CCERT-PUBDOC-2004-05-74

CARNet CERT u suradnji s LS&S

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem CARNet CERT kontinuirano radi.

Rezultat toga rada ovaj je dokument, koji je nastao suradnjom CARNet CERT-a i LS&S-a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

CARNet CERT, www.cert.hr - nacionalno središte za **sigurnost računalnih mreža** i sustava.

LS&S, www.lss.hr - laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

TCP je jedan od temeljnih protokola na kojem se bazira većina današnjih mreža, bez obzira radi li se o lokalnim ili javnim mrežama, odnosno Internetu. U TCP/IP stogu protokola TCP se nalazi u transportnom sloju, odnosno četvrtom sloju ISO/OSI specifikacije.
Originalna specifikacija TCP protokola dana je u IETF dokumentu RFC 793 – Transmission Control Protocol iz 1981. godine. Nakon toga dodane su mnoge specifikacije za unaprjeđenje samog protokola (http://www.faqs.org/rfcs/np.html#TCP), no originalna specifikacija se nije mijenjala.
Obzirom da je TCP temeljni protokol TCP/IP stoga protokola, velik broj proizvođača mrežne opreme i operacijskih sustava implementira ga u svojim proizvodima.
U travnju 2004. godine otkriven je sigurnosni nedostatak u originalnoj implementaciji TCP protokola. Sama ranjivost na mogućnost resetiranja TCP sjednice poznata je od ranije, no nedavno je uočeno da je vjerojatnost njenog uspješnog iskorištavanja znatno veća nego što se prije smatralo. Iskorištavanjem otkrivenog nedostatka zlonamjerni napadač može uzrokovati prekid TCP sjednice, odnosno izvesti napad uskraćivanjem usluge (eng. *Denial of Service – DoS*).
Tako uzrokovani prekid TCP sjednice utječe na aplikacijski sloj TCP/IP stoga protokola, a ozbiljnost i posljedice ovise o specifičnoj aplikaciji.

2. Ranjive aplikacije i ozbiljnost problema

Ozbiljnost nedostatka, kako je u uvodu rečeno, ovisi o specifičnoj aplikaciji. Generalno govoreći, na nedostatak su osjetljivije aplikacije koje se baziraju na uspostavi stalne TCP sjednice između dva entiteta.
BGP (eng. *Border Gateway Protocol*) je protokol koji se smatra potencijalno najranjivijim. BGP se temelji na uspostavi stalne TCP sjednice između dva BGP entiteta. Resetiranje sjednice može uzrokovati produženim vremenom potrebnim za uspostavu normalnog funkcioniranja pošto se tablice usmjerivanja (eng. *routing tables*) moraju ponovno formirati i osvježiti. Također, u slučaju kontinuiranog napada, zbog samog načina funkcioniranja BGP protokola, može doći do prestanka oglašavanja pojedinih ruta (eng. *route dampening, route suppression*). Iako ovo može biti ozbiljan problem, zbog prilično male vjerojatnosti uspješnog napada u ovom slučaju, nedostatak se ipak ne smatra kritičnim.
Također, postoji potencijalna mogućnost napada na druge protokole aplikacijskog sloja kao što je npr. DNS. Moguće bi bilo prekinuti DNS transfere zona, no obzirom da je tipično trajanje takvih sjednica relativno kratko, a sjednica može biti resetirana bez ozbiljnih ili trajnijih posljedica, ranjivost se također ne smatra ozbiljnom. Sličan je slučaj i sa SSL protokolom, gdje postoji potencijalna mogućnost prekida financijskih transakcija.
Obzirom da je nedostatak inherentno svojstvo same specifikacije TCP protokola, svi proizvodi koji koriste TCP/IP stog protokola smatraju se ranjivim. Velik broj proizvođača (Check Point, Cisco, HP, NetBSD, Sun, itd.) potvrdio je ranjivost svojih proizvoda i objavio sigurnosne zakrpe koje ispravljaju uočeni nedostatak.

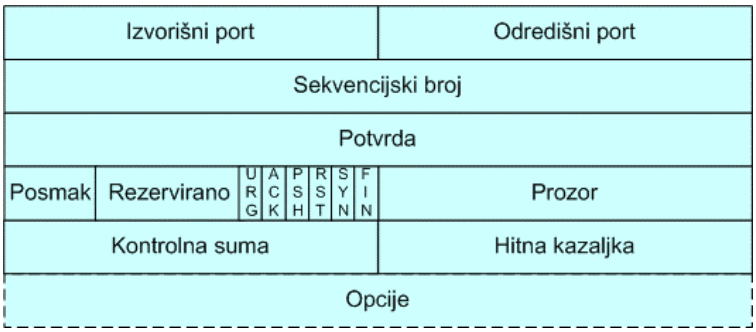
3. TCP protokol

TCP (eng. *Transmission Control Protocol*) je protokol transportnog sloja (OSI razina 4) koji osigurava usluge kao što su:

- garantirana isporuka paketa,
- potvrda paketa,
- segmentacija,
- kontrola toka podataka,
- detekcija pogrešaka i
- identifikacija aplikacija.

TCP protokol obično se koristi kod aplikacija koje generiraju relativno velike količine podataka koji moraju doći na odredište točno u onom redoslijedu u kojem su odaslani. Većina aplikacija koja se temelji na klijent-poslužitelj modelu, kao što su WWW, FTP, elektronička pošta, koristi upravo TCP za slanje podataka.

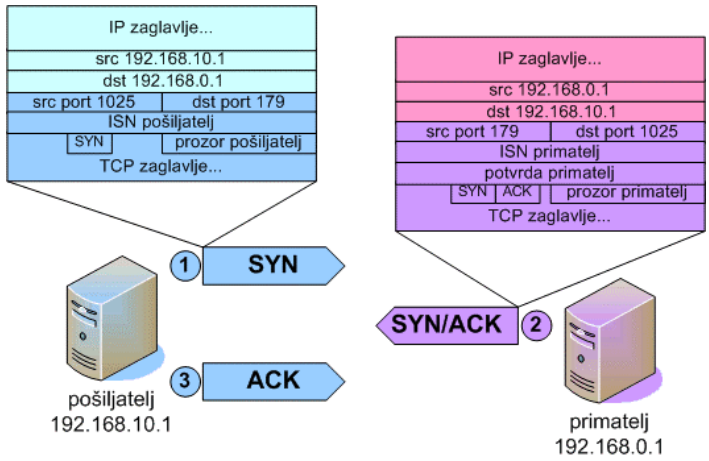
Obzirom na količinu podataka, često nije moguć prijenos u jednom paketu, pa je nužno da pošiljatelj podijeli podatke u blokove te zatim svaki blok šalje primatelju u posebnom paketu, formirajući na taj način tok podataka (eng. *TCP data stream*) koji se sastoji od zasebnih segmenata. Kada se na odredištu zaprime svi paketi koji čine jednu sekvencu, slažu se i isporučuju odgovarajućoj aplikaciji.
Kako je na početku rečeno, TCP osigurava potvrdu prijema segmenata na strani pošiljatelja, detektira pogreške i signalizira potrebne informacije pošiljatelju.
Slika 1 prikazuje TCP zaglavlje s pripadajućim poljima, a u nastavku su opisane funkcije svakog od polja.



Slika 1: TCP zaglavlje

Izvorišni port (eng. *source port*) – 2 okteta – identificira aplikaciju koja je generirala podatke koji se prenose.
Odredišni port (eng. *destination port*) – 2 okteta – identificira aplikaciju koja je primatelj informacija koje se prenose.
Sekvencijski broj (eng. *sequence number*) – 4 okteta – identificira relativnu poziciju podataka tekućeg segmenta u odnosnu na cijelu sekvencu.
Potvrda (eng. *acknowledgement number*) – 4 okteta – specificira sekvencijski broj sljedećeg segmenta koji odredište očekuje. Koristi se u ACK porukama.
Posmak podataka (eng. *data offset*) – 4 bita – specificira broj 4-oktetnih riječi TCP zaglavlja
Rezervirano (eng. *reserved*) – 6 bita – ne koristi se.
Kontrolni bitovi (eng. *control bits*) – 6 bita – sadrži šest zastavica (URG, ACK, PSH, RST, SYN i FIN) koje identificiraju funkcionalnost TCP paketa.
Prozor (eng. *window*) – 2 okteta – osigurava kontrolu toka specificiranjem koliko okteta je primatelj spreman prihvatiti.
Kontrolna suma (eng. *checksum*) – 2 okteta – sadrži kontrolnu sumu (CRC vrijednost) izračunatu na strani pošiljatelja koju primatelj koristi za detekciju pogrešaka u prijenosu.
Hitna kazaljka (eng. *urgent pointer*) – 2 okteta – pokazuje na dio podataka u segmentu koje primatelj treba tretirati kao hitne. Koristi se samo u kombinaciji s URG zastavicom.
Opcije (eng. *options*) – varijabilne duljine – mogu sadržati opcionalne parametre TCP spoja.
Podaci (eng. *data*) – segment podataka koji je dio sekvence generirane od strane aplikacije.
Zastavice (eng. *flag bits*) specificiraju namjenu svakog TCP paketa. Paket s postavljenom SYN (eng. *synchronize*) zastavicom koristi se pri inicijaciji spoja između pošiljatelja i primatelja, a generira ga pošiljatelj. Paket s postavljenom ACK (eng. *acknowledge*) zastavicom služi za potvrdu primljenih podataka i šalje ga primatelj. Paket koji ima postavljenu FIN zastavicu (eng. *finish*) koristi se za terminiranje spoja između pošiljatelja i primatelja, dok paket koji ima postavljenu RST (eng. *reset*) zastavicu resetira uspostavljenu sjednicu. Konačno, paket s postavljenom URG zastavicom označava hitni paket koji se mora obraditi prije ostalih, dok paket s PSH zastavicom označava da se podaci pošiljatelja moraju poslati primatelju,.
Uspostava TCP spoja zahtijeva razmjenu tri paketa između pošiljatelja i primatelja, a taj postupak se naziva trostruko rukovanje (eng. *three-way handshake*). *Slika 2* prikazuje uobičajeni postupak uspostave TCP spoja.
Prilikom uspostave TCP spoja primatelj i pošiljatelj, između ostalog, razmjenjuju i inicijalni sekvencijski broj, te inicijalnu veličinu prozora. Ti pojmovi će biti pojašnjeni u nastavku dokumenta, a vrlo su važni za izvođenje TCP reset napada.

Pošiljatelj inicira spoj s primateljem slanjem SYN paketa. U tom paketu specificirani su brojevi portova na klijentskoj (obično neki visoki port – >1023) i poslužiteljskoj strani (npr. port 80 ukoliko se radi o Web poslužitelju). Osim toga, na razini mrežnog sloja (OSI sloj 3), odnosno u IP paketima, moraju biti definirane IP adrese klijenta i poslužitelja.



Slika 2: Uspostava TCP spoja

Kada primatelj zaprimi SYN paket na otvorenom TCP portu on odgovara slanjem SYN/ACK paketa. Razlog slanja paketa s postavljenim SYN/ACK zastavicama leži u tome što, unatoč činjenici da su TCP spojevi dvosmjerni, svaki smjer mora biti iniciran i upravljan neovisno. Da bi se izbjeglo slanje dva različita TCP paketa; jedan za potvrdu primljenog paketa, a drugi za otvaranje TCP spoja u drugom smjeru, generira se paket s postavljenim SYN i ACK zastavicama. Brojevi portova i odgovarajuće IP adrese primatelja i pošiljatelja su zamijenjene u odnosu na inicijalno poslani SYN paket. Zaprimanjem takvog paketa pošiljatelju je potvrđeno da postoji (virtualni) put između njega i primatelja, te da primatelj prihvaća uspostavu spoja. U slučaju da primatelj ne može prihvatiti spoj, klijentu odgovara slanjem RST/ACK paketa (engl. *reset*) ili ICMP *port unreachable* paketom.

Prilikom uspostave TCP spoja vrlo su važna još dva polja koja određuju daljnju komunikaciju. U polju *sekvencijski broj* (eng. *sequence number*) prilikom svakog slanja SYN paketa postavlja se inicijalna vrijednost (eng. *ISN – initial sequence number*). ISN vrijednost je 32-bitni pseudoslučajni broj generiran od strane pošiljatelja SYN (također i SYN/ACK) paketa. Algoritam kojim se generiraju ISN vrijednosti mora osigurati da vjerojatnost da dvije aplikacije istovremeno koriste sekvencijske brojeve iz istog skupa bude svedena na minimalnu. Prilikom razmjene podataka ovo polje se inkrementira ovisno o količini zaprimljenih podataka (okteta).

Osim toga, primatelj u polju *prozor* (eng. *window*) može definirati maksimalnu duljinu segmenta (u oktetima) koju je spreman prihvatiti. Obzirom da je to polje duljine 2 okteta, maksimalna duljina segmenta može biti 65 535 okteta. Vrijednost ovog polja može se mijenjati i dinamički tijekom razmjene podataka ukoliko primatelj ima potrebu smanjiti ili povećati brzinu prijenosa podataka. Na strani primatelja, paketi se obrađuju samo ukoliko njihov sekvencijski broj ne izlazi iz definiranog prozora.

Slika 3 shematski prikazuje prozor u kojem se primaju sljedeći podaci na strani primatelja.

Slika 3: Prozor za primanje

3.1. RST zastavica

Obzirom da se ovaj dokument bavi mogućnostima resetiranja TCP sjednice, potrebno je pojasniti i značenje RST zastavice. Općenito se RST zastavica, odnosno RST paket, šalje ukoliko primatelj zaprimi paket koji nije dio uspostavljene sjednice. Osim toga, u svim stanjima TCP sjednice, osim u SYN SENT stanju (ACK paket nije zaprimljen), RST paket mora sadržati i odgovarajuće polje *sekvencijski broj*. Reset je valjan samo kada je njegov sekvencijski broj unutar prozora, osim ukoliko je RST zaprimljen kao odgovor na inicijalni SYN zahtjev pošiljatelja.
Ono što je vrlo važno istaknuti kod RST paketa je to da ukoliko primatelj zaprimi takav, ispravan paket on odmah terminira sjednicu. Isto kao i URG paketi, RST paketi se moraju obraditi čak i ukoliko je primatelj postavio prozor na veličinu 0 (trenutno ne može prihvaćati pakete od primatelja).

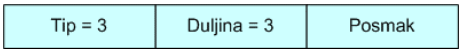
4. Napad resetiranjem sjednice

4.1. Tradicionalni pristup

Poznata mogućnost napada uskraćivanjem usluge, slanjem posebno i zlonamjerno oblikovanih paketa, (eng. *DoS – Denial of Service*) inherentna je slabost protokola TCP/IP stoga.
Osim toga, činjenica da TCP sjednica može biti resetirana slanjem odgovarajućih RST ili SYN paketa je implementacijska karakteristika TCP protokola i dio je njegove specifikacije. Protokol sam po sebi nema pogrešku, odnosno sigurnosni nedostatak, nego se sigurnosni nedostatak pojavljuje zbog toga što IP adresa i port pošiljatelja mogu lako biti lažirani (eng. *spoofed*).
Na taj način zlonamjerni napadač može pokušati slanjem posebno oblikovanog RST ili SYN paketa pokušati resetirati uspostavljenu TCP sjednicu. Obzirom da primatelj prilikom primanja takvog paketa provjerava polje *sekvencijski broj* koje je 32-bitno, smatralo se da pogađanje sekvencijskog broja, odnosno uspješno izvođenje takvih napada, nije moguće iz razloga što je vjerojatnost uspješnog napada $1:2^{32}$. Tako mala vjerojatnost ovakvu vrstu napada čini praktično neizvedivim.

4.2. Nova razmatranja

Suprotno pretpostavkama u prethodnom poglavlju, Paul A. Watson je u svom radu "Slipping In The Window: TCP Reset Attacks", objavljenom na CanSecWest 2004 konferenciji, pokazao da je vjerojatnost napada resetiranjem sjednice mnogo veća nego što se to pretpostavljalo.
Naime, u originalnoj specifikaciji TCP protokola definirano je da primatelj mora zaprimiti sve pakete pošiljatelja koji se nalaze unutar definiranog prozora. Obzirom da je najveća moguća veličina prozora prema originalnoj specifikaciji 2^{16}, odnosno 65 535, vjerojatnost uspješnog izvođenja TCP reset napada dramatično se povećava. Umjesto vjerojatnosti $1:2^{32}$, odnosno 1: 4 294 967 295, ta vjerojatnost se povećava na $1:2^{16}$, odnosno 1: 65 535.
Problem je još veći ako se uzme u obzir da se u IETF RFC 1323 - TCP Extensions for High Performance iz 1992. godine definira mogućnost skaliranja prozora. Obzirom na performanse današnjih mreža, ograničenje veličine prozora na 2^{16} pokazalo se neefikasnim, a taj problem se riješio definiranjem *Windows scale* TCP opcije duljine tri okteta (*Slika 4*) u kojoj posljednji oktet koji predstavlja broj x definira bit-posmak veličine prozora definirane u *prozor* polju TCP zaglavlja u lijevo, odnosno multiplikaciju veličine prozora za 2^x.



Slika 4: Windows scale TCP opcija

Obzirom na neke specifičnosti TCP protokola, najveći mogući posmak je 2^{14}, što daje maksimalnu veličinu prozora od 2^{30}, čime se vjerojatnost uspješnog napada resetiranjem TCP sjednice još više povećava.

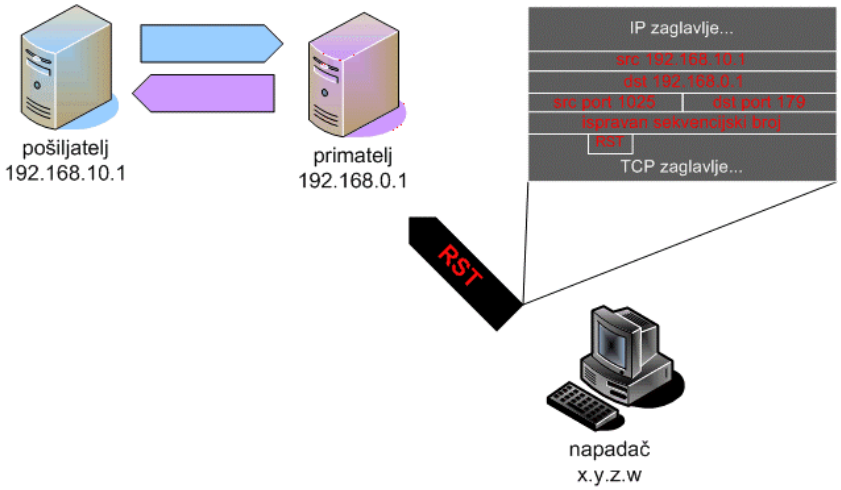
4.3. Realne mogućnosti

Da bi napad resetiranjem TCP sjednice bio uspješan, napadač mora poznavati neke stvari. Uz pretpostavku da su mu IP adresa i port ciljane žrtve poznati, te da zna s koje odredišne IP adrese žrtva prima pakete (što je realna pretpostavka), napadač mora poznavati (ili pogoditi) sljedeće:

– TCP port pošiljatelja i
– trenutni prozor.

U sljedećim poglavljima je opisano na koji način napadač može predvidjeti neke od tih podataka, obzirom na implementacije TCP/IP stoga u različitim operacijskim sustavima.

Slika 5 prikazuje shematski TCP reset napad. Da bi uspješno izveo napad, napadač mora oblikovati RST paket s ispravnim IP adresama primatelja i pošiljateljima, portovima, te sekvencijskim brojem koji odgovara trenutnom prozoru.



Slika 5: Shematski prikaz TCP reset napada

4.3.1. Određivanje TCP porta

Prva stvar koju napadač mora pogoditi jest TCP port pošiljatelja poruke (pod uvjetom da su mu poznate IP adrese primatelja i pošiljatelja, te port na kojem primatelj osluškuje). Bez toga je razmatranje svih ostalih mogućnosti bespredmetno.

Teoretski, obzirom na 16-bitno polje rezervirano u TCP zaglavlju za port pošiljatelja, napadač ima 65 536 različitih kombinacija koje valja iskušati. U praksi, taj broj se smanjuje. Naime, portovi 0 – 1023 su rezervirani za privilegirane servise, dok portovi 49152 – 65535 predstavljaju privatne portove. Štoviše, neki operacijski sustavi koriste portove 5001 – 65535 kao privatne, ostavljajući tako svega 3977 portova za dinamičku dodjelu procesima.

Osim toga, način odabira portova za dinamičku dodjelu procesima od strane operacijskog sustava kod većine operacijskih sustava vrlo je jednostavan te na taj način olakšava predviđanje porta koji će se koristiti u komunikaciji. *Tablica 1* [4] prikazuje inicijalni port koji operacijski sustav dodjeljuje procesima, te metodu kojom dodjeljuje sljedeće portove. Rezultati nisu dio nikakve specifikacije, nego su dobiveni promatranjem, što ostavlja mogućnost pogreške.

Operacijski sustav	Inicijalni port	Metoda za dodjelu sljedećeg porta
Cisco 12.2	11000	inkrementiranje za 1
Cisco 12.1	48642	inkrementiranje za 512
Cisco 12.0	11778	inkrementiranje za 512
Windows 2000 SP4	1038 / 1060	inkrementiranje za 1
Windows 2000 SP3	1060	inkrementiranje za 1
Windows XP Home SP1	1050	inkrementiranje za 1
Linux 2.4.18	32770	inkrementiranje za 1
Nokia IPSO 3.6-FCS6	1038	inkrementiranje za 1

Tablica 1: Način dinamičke dodjele portova kod pojedinih operacijskih sustava

Može se uočiti da predviđanje portova koji se dinamički dodjeljuju procesima ne predstavlja nepremostiv problem za napadača, štoviše uz određene uvjete to može biti i vrlo jednostavno.

4.3.2. Predviđanje trenutnog prozora

Da bi napadač mogao što uspješnije predvidjeti trenutni prozor za primanje paketa (*Slika 3*), potrebno je predvidjeti ISN, te veličinu prozora.
IETF RFC 1948 dokument – Defending Against Sequence Number Attacks opisuje metode kojima se mogu izbjeći mogućnosti napada predviđanjem ISN brojeva, no preporuke iz tog dokumenta nisu implementirane u većini operacijskih sustava. Pokazuje se da se kod većine sustava može prilično efikasno predvidjeti ISN, te na temelju toga povećati mogućnost uspješnog provođenja napada [5].
Veličina prozora je jedna od ključnih komponenti za uspješno izvođenje TCP reset napada. Različiti proizvođači inicijalnu veličinu prozora postavljaju na različitu vrijednost, što napadaču ostavlja veću ili manju mogućnost za uspješno izvođenje napada. Obično se inicijalna veličina prozora povećava da bi se poboljšale performanse i propusnost, no na taj način se efektivno povećava i mogućnost za uspješno izvođenje TCP reset napada. *Tablica 2* [4] prikazuje inicijalne veličine prozora kod nekih operacijskih sustava.

Operacijski sustav	Inicijalna veličina prozora	Potreban broj paketa
Windows 2000 SP4	64512	66,576
Windows XP Home	64240	66,858
HP-UX 11	32768	131,071
Nokia IPSO 3.6-FCS6	16384	262,143
Cisco 12.2(8)	16384	262,143
Cisco 12.1(5)	16384	262,143
Cisco 12.0(8)	16384	262,143
Windows 2000 SP3	16384	262,143
Linux 2.4.18	5840	735,439

Tablica 2: Inicijalna veličina prozora kod nekih operacijskih sustava

Može se uočiti kako povećanjem inicijalne veličine prozora potreban broj paketa (izračunat po formuli 2^{32}/inicijalna veličina prozora) za uspješno izvođenje napada pada. Također, može se uočiti da novije inačice Windows operacijskih sustava povećavaju veličinu tog prozora.
Prema tradicionalnom razmatranju, vrijeme potrebno za uspješno provođenje ovog napada (uz pretpostavku da je napad uspješan nakon 50% iskušanih mogućnosti) bilo bi:

$$2^{32}/\text{brzina generiranja paketa}/2$$

Uz pretpostavku da je brzina generiranja paketa 100 000 paketa u sekundi dolazi se do sljedećih rezultata:

$$2^{32}/100\ 000/2 \approx 21\ 474\text{s} \approx 358\text{min} \approx 6\text{h}$$

Ovaj rezultat implicira da bi napadaču, uz mogućnost generiranja 100 000 paketa u sekundi, za uspješno pogađanje prozora bilo potrebno oko 6 sati. Naravno, toliki promet u tako dugom vremenskom razdoblju teško bi prošao nezapaženo.

Uzme li se međutim u obzir veličina prozora, gore dobiveni rezultati znatno se mijenjaju. U tom slučaju vrijeme potrebno za uspješno izvođenje napada računa se po sljedećoj formuli:

2^{32}/brzina generiranja paketa/veličina prozora/2

Uzme li se npr. inicijalna veličina prozora kod Cisco IOS-a, ili starije inačice Windows 2000 operacijskih sustava, gdje je inicijalna veličina prozora 16 384 dolazi se do sljedećih rezultata :

2^{32}/100 000/16 384/2 ≈ 1,3s

Lako je uočiti da napadač ustvari vrlo brzo može pogoditi trenutni prozor, uz veliku vjerojatnost da u tako kratkom vremenskom razdoblju (par sekundi) njegova aktivnost neće biti primijećena.

5. Preporuke

Korisnicima je prvenstveno preporuča instalacija odgovarajućih zakrpi od strane proizvođača, ukoliko takve postoje. Osim toga, kod većeg broja operacijskih sustava inicijalna veličina prozora može se podešavati kroz odgovarajuće parametre operacijskog sustava. Smanjivanjem prozora povećava se vrijeme potrebno za uspješno izvođenje TCP reset napada, no isto tako to može negativno utjecati na sustav smanjujući njegove performanse.
Općenito govoreći, poželjno bi bilo u TCP implementacije uključiti preporuke iz IETF RFC 1948 dokumenta, a za BGP, koji je potencijalno najosjetljiviji na TCP reset napade, poželjno bi bilo implementirati TCP MD5 potpise (IETF RFC 2385 – Protection of BGP Sessions via the TCP MD5 Signature Option).
Također, uvijek dobra preporuka je implementacija filtara na usmjerivačima i/ili vatrozidima koji eliminiraju napade lažiranjem sjednica, no time se smanjuje mogućnost napada samo izvan pojedinih mreža ili mrežnih segmenata.

6. Reference

1. Transmission Control Protocol, RFC 793, http://www.ietf.org/rfc/rfc793.txt
2. TCP Extensions for High Performance, RFC 1323, http://www.ietf.org/rfc/rfc1323.txt
3. NISCC Vulnerability Advisory 236929: Vulnerability Issues in TCP, http://www.uniras.gov.uk/vuls/2004/236929/index.htm
4. Slipping in the Window: TCP Reset attacks, Paul A. Watson, http://www.packetstormsecurity.org/papers/protocols/SlippingInTheWindow_v1.0.doc
5. Strange Attractors and TCP/IP Sequence Number Analysis, Michal Zalewski, http://lcamtuf.coredump.cx/newtcp