Seznamte se – DoS a DDoS útoky

Vložil/a cm3l1k1 [1], 7 Březen, 2013 - 13:08

- Hacking method [2]
- Security [3]

DoS útoky jsou v době psaní tohoto článku znovu objeveným kolem pro mnoho zpravodajských serverů, na které byl veden útok typu SYN flood (Novinky.cz, Seznam.cz, iHned.cz, E15.cz, ...). Horším zjištěním je však to, že útoku podlehly i české banky (Česká spořitelna, ČSOB, Komerční banka, ...), kde by člověk očekával velkou míru bezpečnosti. Zdání klame.
Ačkoliv je tento útok znám přes deset let, ukázal nám, jak nepřipravené jsou společnosti, jejichž business je na webu závislý, i když by měla být dostupnost a bezpečnost jednou z jejich priorit. Nebudu zde spekulovat o zdroji, nebo důvodu útoku, zaměříme se zde na podstatu těchto útoků. V sérii článků popíšu většinu DoS a DDoS útoků, vysvětlím jejich princip fungování, metody a možnost ochrany před nimi (především pak před SYN flood).

Během čtení tohoto článku na chvilku, prosím, zapomeňte, co o DoS útocích víte, protože především na poli českých medií jsou tyto informace často zkreslené, nebo nepravdivé.

Ani nevím, proč jsem o DoS útocích nepsal už dřív. Možná je to způsobeno tím, že spousta lidí si možnost tohoto útoku vůči vlastní infrastruktuře nepřipouští, stejně jako i jiné útoky a berou to jako sci-fi, kterým straší pouze paranoidní lidé. Jde o neinformovanost těchto lidí. Jejich počítač může být součástí botnetu, jejich data mohou být odesílána na servery někde v Číně, jejich bankovní příkazy přepisuje ve formulářích banker skrytě spuštěný v jejich systému a prostě dokud se jich něco hmatatelně nedotkne, tak si riziko nepřipouštějí. Kdyby alespoň pár z nich sledovalo, co se skutečně děje na Internetu, tak jim spadne brada. Naše malá zemička se dlouhá léta vyhýbala hledáčku těchto zločineckých skupin a malwaru, ale to se postupem času mění.

Pro začátek rozdíl mezi **DoS (Denial of Service)** a **DDoS (Distributed Denial of Service)** je pouze v tom z kolika strojů je útok iniciován. Při DoS útoku je zapojen jen jeden stroj/jedinec, při DDoS dva a více (statisíce). Ve článku nebudu rozlišovat mezi DoS a DDoS, takže pokud budu psát o “DoS útoku” mohu tím myslet i DDoS.

Denial of Service znamená znepřístupnění služeb a to jakýmkoliv způsobem. Od vytržení kabelu ze serveru po využití slabiny v aplikaci, kdy po odeslaní specifického řetězce server vytuhne a přestane komunikovat. Může se jednat o využití chyby, dosažení limitu systému, síťové karty, aplikace, nebo systémových prostředků jako je CPU, paměť, místo na disku, či IO operace. Pokaždé když je výsledkem útoku nedostupnost služby daného serverů, tak se jedná o DoS.

Dalo by se říct, že útoky vedené na nižší síťové vrstvy jsou mnohem jednodušší než útoky na vrstu nejvyšší (aplikační). Přičemž úspěšné útoky nižších vrstev mnohem více afektují danou společnost (např. kompletní odstavení hraničního routeru) a její služby.

DoS útoky lze dělit podle několika parametrů či skupin. Já jsem si vybral rozdělení podle síťových/IP vrstev, protože při identifikaci rizik a postupů pro jejich zmírnění/eliminaci se budeme zaměřovat na všechny ty vrstvy, které dané zařízení ovlivňuje. Tj. u L2 switche nemá cenu identifikovat/mitigovat DoS útoky vrstev vyšších.

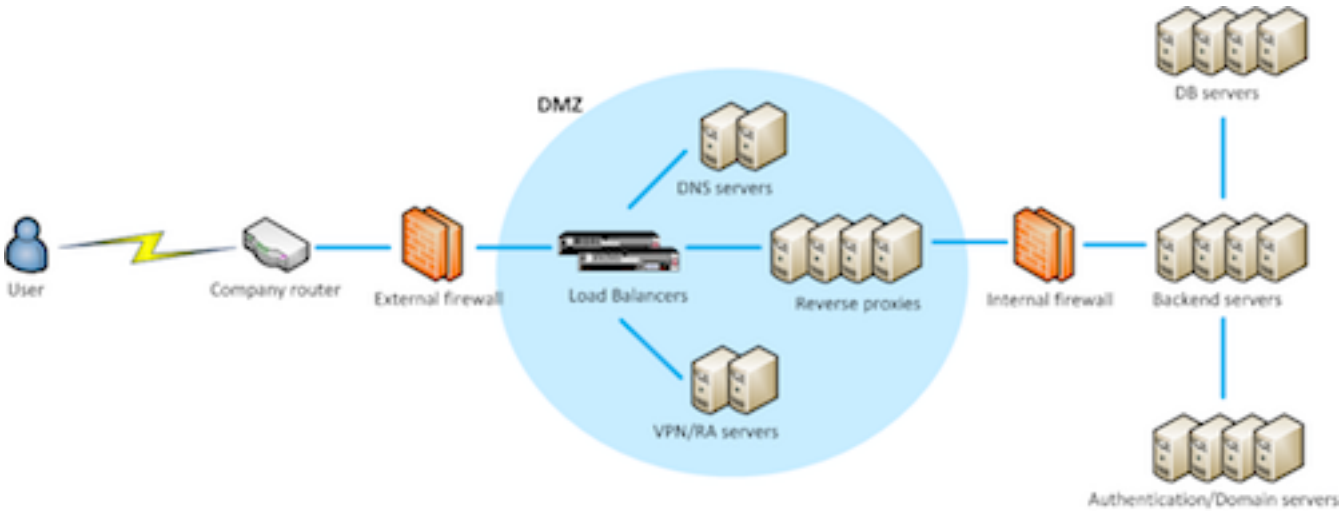
[4]

Na tomto obrázku jsem se snažil o kompletaci většiny známých DoS/DDoS útoků a jak vidíte, není jich zrovna málo. Dnes tak populární útoky (SYN flood) jsou pouhým jedním bodem v této mapě. Některé rozřazení může být sporné, ale držel jsem se schémata podle pozice v IP paketu. Podle toho jakou vrstvu (pozici v paketu) zasahuje, tam patří daný útok. Některé z nich se mohou překrývat, nebo být podmnožinou jiných, ale přišlo mi důležité je uvést. Dané útoky si vysvětlíme v průběhu série článků. Pokud máte dojem, že tam některý chybí, tak mě prosím informujte.

Většinu útoků budete schopni omezit změnou nastavení systému, nebo aktivního zařízení, jakým je firewall, load balancer, nebo IPS. Není pravdou, že se na některé typy nelze připravit, nebo že je nutné koupit zařízení za stovky tisíc eur ročně. Navíc se často jedná jen o marketingový trik a dané zařízení vás dokáže ochránit jen před pouhým zlomkem útoků (nevěřte přehnaně obchodníkům s krásnými grafy). Pokud vám některý dodavatel nabízí "super-řešení" odolávající všem útokům, tak si raději vezměte k ruce naši mapku a zjistěte, co umí a co neumí eliminovat. Tento úkol není vhodný pro manažery vyzbrojené Excel tabulkou. Musí to provést někdo technicky zdatný.
Většina vendorů vašich zařízení bude mít knowledge base věnovanou mitigaci DoS a jiných útoků, případně i guide pro optimalizaci performance. Prostě donuťte zodpovědné osoby číst dokumentaci a používat hlavu.

// Směr útoku

Při analýze rizik spojených s DoS útoky je potřeba brát v potaz všechna zařízení hrající roli při poskytování služeb vašim zákazníkům.



[5]

Nejde jen o servery a databáze. Musíme začít u hraničních zařízení, kterými jsou obvykle routery, firewally, load balancery, až po reverzní proxy a mail gatewaye, dále pak backend servery, databáze, autentizační servery, certifikační autoritu, LDAP, centrální log servery, switche apod. Oscanujte si svůj externí a interní rozsah IP adres, ať víte, o čem mluvím.

U všech musíte identifikovat potenciální hrozby (od té nejnižší až po tu nejvyšší vrstvu) a navrhnout postup jak tato rizika zmírnit.
Tuto analýzu musí dělat někdo, kdo velmi dobře rozumí systémům a infrastruktuře organizace.

Pár příkladů
Například jedno z opomíjených rizik je logování. Pokud vaše zařízení loguje každý přístup/akci na lokální disk, buďte si jisti, že při DoS útoku dané místo velmi rychle zmizí a může dojít k zahlcení. Pokud používáte centrální syslog server, tak tím spíš bude pod obrovským loadem logů od zařízení. Představte si, že jakoukoliv část systému, nebo služby začne používat X-násobek uživatelů. Zvládne to? Jak to detekovat? Jak tomu zamezit v případě útoku?
Dále například OCSP server, kterého se dotazuje uživatel, když navazuje https spojení s vaším serverem. Pokud někdo provede DoS útok na tento server (velkou mírou dotazů), tak web server jako takový bude sice fungovat, ale uživatelům se obsah https stránky nenačte, protože prohlížeč neobdržel odpověď. Toto se dá řešit například OCSP staplingem [6], kdy se během SSL handshaku klientovi rovnou pošle OCSP response, kterou si váš server jednou za čas vyžádá, předtím než mu vyprší platnost tohoto podpisu. Je to velice efektivní.
Jedná se o pouhé příklady. Není v možnostech tohoto článku popsat všechna rizika.

Po incidentu určitě uvítáte data některého z monitorovacích zařízení, jakým je například Zabbix [7] (článek o instalaci a optimalizaci Zabbixu zveřejním v příštích týdnech), Nagios, Cacti, Solarwinds, atp. Pokud tedy monitorované zařízení odpoví na snmp dotaz během DoS útoku. Tyto monitorovací nástroje vám umožní zjistit, jaký druh prostředku (CPU, Mem, IO, ...) byl během útoku vyčerpán jako první, což se vám bude hodit pro budoucí zprávu o incidentu a budete schopni navrhnout protiopatření. Navíc vás mohou včas upozornit na přicházející útok, takže fungují i jako proaktivní ochrana.

// Motivace útoků

1) Zisk a konkurenční válka

DoS útok je možné použít pro likvidaci/poškození konkurenční společnosti, zničení její aktuální marketingové reklamy/strategie a za určitých okolností může být dopadem DoS útoku (nedostupností služeb) i pokles akcií.

2) Seberealizace, společenský kredit

Často může být hnacím motorem DoS útoků potřeba někomu z komunity ukázat že na to daný jedinec má a zvýšit si tak společenský kredit, případně sám si dokázat, že je toho schopen.

3) Odplata

K tomu není třeba nic dodávat.

4) Forma demonstrace

Do této kategorie spadají politicky motivované útoky, které jsou běžně směrované na vládní a stranické systémy, případně pak proti společnostem a skupinám ohrožujícím svobodu slova, nebo pohyb na Internetu.

5) Kyber terorismus

Útoky směrované na systémy kritické pro fungování státu a jeho infrastruktury.

6) Obchod s anti-DoS produkty

Nechci ukazovat na žádnou konkretní společnost, ale často se DoS útok objeví ruku v ruce s nabídkou produktu na ochranu proti DoS útokům. Neuváděl bych to, pokud bych neměl tuto zkušenost od kolegů z několika dalších firem.

7) Skrytí sekundárního útoku

Pan Jaroslav Pinkava z Crypto-Worldu [8] správně poznamenal, že DoS útok je možné použít i k zamaskování jiného útoku na infrastrukturu. Pokud útočník ví o chybě v systému, jejíž zneužití může vzbudit pozornost správců, tak podnikne DoS útok na jinou část sítě. Tím pádem bude mít nejen dostatek času na její využití bez povšimnutí, ale také může trvat několik dalších dnů/týdnů než správci projdou všechny logy a průnik odhalí (pokud vůbec).

// Faktory napomáhající útokům

1) Zranitelné systémy

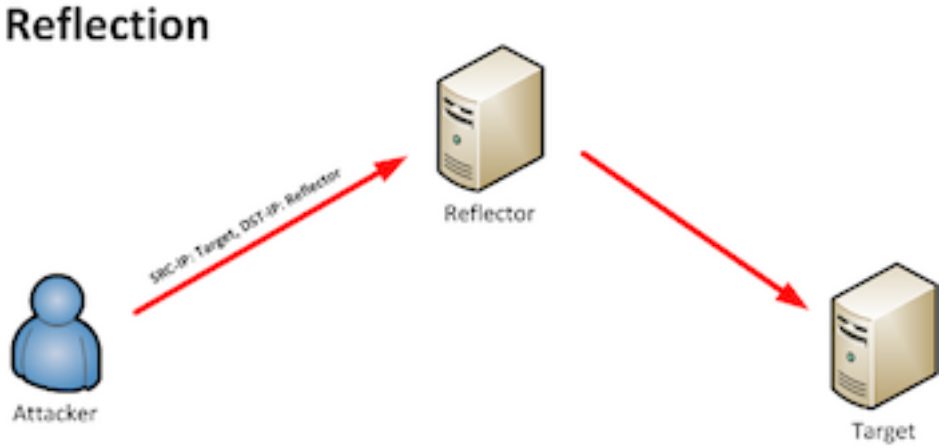
Sem spadají jak klientské stanice Windows, tak servery. Útočník využije zranitelnosti/chyby daného neopatchovaného systému a použije ho jako klienta k útoku v rámci botnetu ovládaného C&C servery. Webhostingy/VPS a freehostingy nekontrolující odchozí spojení poslouží také velice dobře. Především díky lince do Internetu, která se nedá srovnat s rychlostí uploadu ADSL uživatelů. Je to velice jednoduché, protože existuje velká spousta PHP botů, Python botů atd, takže je stačí jen nakonfigurovat, uploadnout na server a je hotovo.

2) Spoofing

Nebo-li podvržení IP adresy, jako je tomu často v případě SYN floodu. V tomto případě je výhoda tohoto faktoru jasná – anonymita. Nevíte odkud je veden útok a není ani efektivní blokovat IP adresy, případně celé subnety států. Během aktuálních DDoS útoků tak třeba Seznam prostě odřízl všechny subnety, které nejsou v České republice... ale spoofovat se dají i ty české samozřejmě. Tomuto útoku je možné na Internetu úplně zamezit, ale k tomu se vrátíme později. Spoofnuté IP se dají s relativně velkou přesností detekovat, ale to je také nad rámec tohoto článku.

3) Existence reflektorů a amplifikátorů

Reflektor
Použití reflektoru se může podobat spoofingu IP adresy, ale klient v tomto případě podvrhne zdrojovou IP za IP serveru na který chce útočit a jako destination IP uvede adresu serveru na Internetu, který za něj útok provede. Na server, na který se útočí, dorazí SYN/ACK od reflektora. Ten o tomto spojeni neví, takže paket zahodí. Po definovaném čase reflektor nabude vědomí, že se SYN/ACK paket ztratil a pošle ho znovu... a to se opakuje několikrát :] (RFC 793) To znamená, že jeden spoofnutý dotaz útočníka může vyprodukovat několik paketů z reflektora na server.



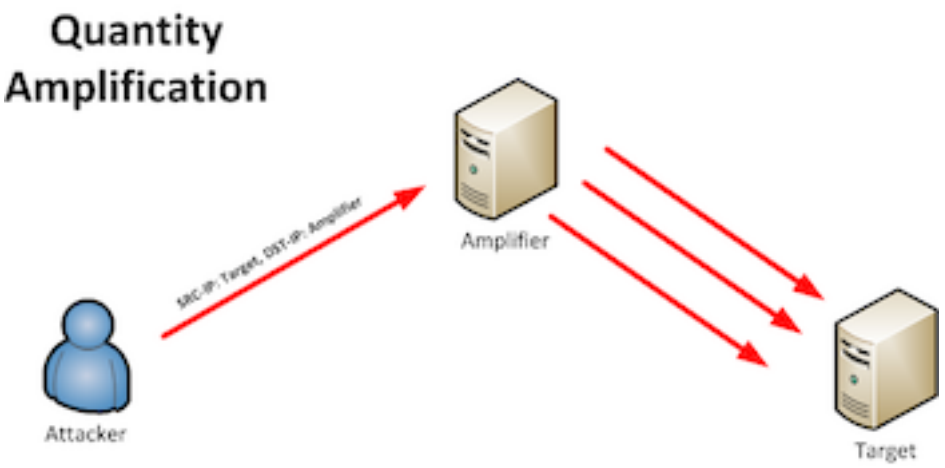
[9]

Tímto se můžete dostat do zajímavé situace, kdy na jeden z vašich serverů útočí jiný z vašich serverů. Prostě pošlete paket se zdrojovou IP např. smtp.seznam.cz a destination server bude IP adresa serveru www.seznam.cz [10] (nic proti Sezname, jen příklad). Na serveru fungujícím jako reflektor může dojít k úspěšnému SYN floodu a on sám bude na další server ve své síti útočit. Pokud spolu potřebují komunikovat, tak se může jednat o špatně řešitelný problém, protože ho prostě podle IP zablokovat nemůžete. Řešil bych to správným nastavením anti-spoof filtrů na hraničním routeru. To však řeší jen tento konkrétní příklad, ne všeobecně problém reflektorů.
Někdy v šestnácti letech mi vyšel v Computerworldu článek [11], který tomuto útoku dokáže efektivně čelit (jen náhoda, nijak zvlášť se o DoS nezajímám). Prostě pošlete RST paket, nebo ICMP unreachable a stroj již neposílá další SYN/ACK. Uvádím příklad pomocí iptables:

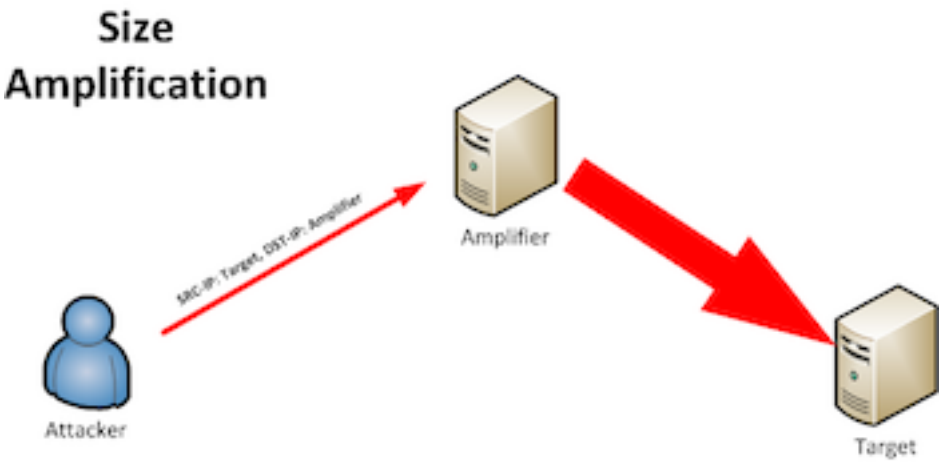
```
iptables -A INPUT -p tcp --tcp-flags SYN,ACK SYN,ACK -m conntrack --ctstate NEW -j REJECT --reject-with tcp-reset
```

Tím neznemožníme útok, ale vyhneme se amplifikaci znásobením. Tomuto útoku se říká DRDoS [12].

Amplifikátor



[13]



[14]

Amplifikátor je stroj, který dokáže znásobit náš útok. Většinou se jedná o špatně nastavený DNS server, kdy pomocí malého, spoofnutého UDP DNS dotazu server vyprodukuje mnohonásobně větší odpověď, kterou zašle na server na který útočíme. Takže nám stačí malý traffic k vyprodukování mnohem většího. V případě DNSSEC je tato odpověd ještě o řád větší. CZ.NIC o tom už před nějakým časem vydal zprávu [15]. Ověřte si, prosím, že není možné zneužít vaše servery k tomuto útoku.

4) Problematická identifikace útoku

Stejně tak jako se hraje na kočku a myš v antivirovém odvětví, tak i v případě identifikace DoS útoku je mnohdy problém rozlišit klienta od útočníka. Pokud útočník naprosto randomizuje data zasílaná na server (POST/GET data, včetně User-Agent hlavičky atd.), tak není jednoduché určit specifický pattern, kterým bychom mohli útočný traffic odfiltrovat. Vždy je dobré pro případy útoku připravit SPAN port, na který necháte mirrorovat traffic pro analýzu na ne-afektovaném zařízení.
Zde se již hodí použít Intrusion Prevention Systémy pro analýzu chování, nebo Load Balancery schopné reagovat při náhlém zvýšení aktivity o předem definovaný počet procent a změnit tak například nastavení timeoutů, nebo parametry forwardování trafficu. V některých případech je i vhodné na úrovni WAF (Web Application Firewall) vložit do vašich stránek malý javascript, který dokáže rozlišit stroj od člověka (pohyb myší, scrolování, matematický výpočet javascriptem, ...). To se většinou užívá při obraně před spammery, ale může se to hodit i ve vašem případě při detekci útočníků.

5) Špatně napsané aplikace

Každá aplikace vlastně přináší nový, specifický typ útoku. Často se útočníci zaměří na funkci aplikace, která spotřebuje hodně systémových prostředků (např. vyhledávání na fóru) a tato funkce je volána tisíckrát za vteřinu. Tím zahltíme DB servery. Dále může jít např. o proces přihlášení do aplikace, kdy se spousta “odborníků” domnívá že místo jednoho MD5/SHA hashe je bezpečnější udělat hash 100x (hash hashe)... po několika set pokusech o přihlášení se zase dostaneme na limity CPU serveru. Do stejné kategorie spadá i využití útoku typu ReDoS, nebo XML bomb, o kterých budeme mluvit v jednom z dalších článků. Určitě vás napadne stovka dalších...

6) Nedostatečný bandwidth

Jedná se o jednoduchou matematiku. Pokud mám 1Gb/s připojení do Internetu a útočí na mě bandwidthem 20Gb/s, tak mám, tak říkajíc, smůlu. Vzhledem k tomu že zatím nejsilnější (mě známý) DDoS měl kadenci 120Gb/s, tak prostě takovou linku si koupit nemůžete. Ochranu před takovýmto bandwidthem musíte řešit na úrovni ISP. Buď nabídne ochranu proti DoS jako managed službu, nebo pomoc při blokování zdroje útoku (to si raději zaneste do SLA). V některých zemích je dokonce státem přikázáno, že alespoň základní ochranu před DoS/DDoS musí ISP poskytnout automaticky a zdarma k připojení do Internetu. O tom si můžeme nechat ještě zdát.

7) Neschopnost a špatné plánování

Tuto kategorii jsem vzal spíše obecně. Nejde jen o neschopnost administrátoru, ale také o celý proces plánování, identifikace rizik, přijetí protiopatření a absenci testování. Je velký rozdíl mezi tím jestli si myslíte, že jste ochráněni proti těmto útokům, nebo jestli opravdu ochráněni jste. Nejedna arogantní firma je nejenom neochráněna, ale ještě ke všemu jsou jejich servery použity k amplification útokům.

// Zahrňte ochranu před DoS do security procesů

- penetrační testování k nalezení slabých míst, vulnerabilit, reflection a amplification systémů
- aplikační testing (source code analysis, fuzz testing, load & stress testing)
- fyzická bezpečnost, tj. nikdo neoprávněný se nedostane k serverům a systémům
- testování politik a procesů zahrnující i sociální inženýrství, abyste vědeli že nastavená pravidla se opravdu dodržují a že je všichni zaměstnanci chápou
- load a stress testing pro identifikaci maximálního throughputu skrze různá zařízení na cestě (server, firewall, IPS, ...) pro odhalení bottlenecku na cestě

Nicméně je to vždy unikátní pro potřeby dané společnosti a věřím že zodpovědné osoby budou schopny dohledat si podrobnější specifikaci/doporučení mezi ISO a dalšími standardy. Pojďme si ušpinit ruce konkrétními útoky.

// Physical link DoS útoky

Vezmeme to od té nejnižší vrstvy, protože některé útoky z aplikační vrstvy si zaslouží vlastní článek.

Fyzická manipulace se zařízením

Pokud nebude dostatečně zajištěn oprávněný přístup do serverovny, tak je možné, že se někdo dostane až k serverům, které může vypnout, odpojit od sítě, odpojit od elektřiny, nebo mechanicky poškodit. Ať tak nebo tak, server bude nedostupný. Nesmějte se, i toto je DoS. Vaše uklízečka nemá přístup do serverovny a do jiných bezpečnostních zón (zamčené kanceláře)? :]

// Data Link – ARP spoofing

Nebudeme brát v potaz vnitřní síť společnosti, ale zaměříme se na servery a routery. Pokud se útočník dostane na jedno z těchto zařízení, tak se může pokusit o spoofnutí ARP záznamu a tím ze sebe udělat např. defaultní gateway. Toto se běžně děje při Man-in-the-Middle útocích, kdy ze sebe útočník udělá bod, přes který začnou všechny počítače v daném segmentu komunikovat a on tak může jednoduše odchytávat traffic. Zkuste si to (ettercap), stanice s Windows jsou v tomto ohledu velice vstřícné :] My zde však řešíme DoS. Tj. situace je stejná, ale traffic se nebude přeposílat k cíli a bude se zahazovat. Tím se nikdo z vnitřní sítě nebude schopen nikam připojit (DoS). Pokud tohle někdo udělá v DMZ nebo na firewallu připojeném k Internet routeru, tak tím odřízne celou síť.

Ochrana:
- Existují aplikace které běží na serverech a detekují změnu MAC adresy gatewaye. Nepoužívejte je.
- Další možností je mít statické záznamy v ARP tabulce na všech serverech. To nepoužívejte už vůbec.

Jedna z efektivních cest proti ARP spoofingu (na Cisco zařízeních) je kombinace **DHCP Snoopingu s IP Source Guard**.
Tyto technologie kombinuji další návazné mechanizmy (Dynamic ARP Inspection, Port Access Control List), ale o těch nemá cenu zde mluvit.
Jak to funguje? Počítač se připojí k portu. V tu chvíli je na portu povolen jen DHCP request (nic víc), což je obvykle první paket zaslaný počítačem po připojení kabelu. V momentu jak dostane počítač IP,

tak si ji switch/router zapíše do tabulky a z daného portu nebude akceptovat žádnou jinou adresu. Nemůžete pak poslat podvržený ARP záznam, nebo paket s jinou zdrojovou adresou, protože ho switch zablokuje. DHCP Snooping navíc ještě definuje kde je povolen DHCP server (port, VLAN), což řeší problém i s podvrhnutým DHCP serverem, nebo špatně nastaveným virtuálem.
Více informací zde:
http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/white_p... [16]
http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SY/con... [17]

// Internet/IP DoS útoky

Smurf – v dřívějších dobách šlo o jeden z nejběžnějších útoků, dokud správci sítí a ISP masivně nezměnily konfiguraci zařízení a tím tak tomuto útoku ve velkém zamezily. Jedná se o amplification útok kdy jste poslal spoofnutý paket se zdrojovou IP oběti na broadcast adresu sítě. Dejme tomu že jste v síti 192.168.0.0 s maskou 255.255.255.0, tj. ve vašem subnetu může být 254 aktivních adres (více info o rozdělení sítí v článku [18]). Pokud by u vás Smurf útok fungoval, tak při pingu na adresu 192.168.0.255 dostanete 254 odpovědí. To ještě jde, ale představte si že víte o síti kde je takových adres v síti například 65.534. To už je slušný botnet. Naštěstí v dnešní době není (nebo by neměl být) tento útok použitelný. Pár sítí je tímto útokem náchylných, ale je jich zlomek promile [19].
Ochrana:
Na Cisco zařízení

```
Router(config-if)# no ip directed-broadcast
```

ICMP/Ping flood – jak název napovídá jedná se o obrovské množství pingů (ICMP) zaslaných z počítače. Tato metoda je efektivní jen pokud má útočník vyšší kapacitu připojení a oběť žádnou ochranu.
Ochrana:
Většinou se používá termín IP sweep protection, což je vlastně jen limit kolik může jedna IP adresa (případně globálně) poslat na zařízení, nebo skrz firewall. Vše ostatní je zahozeno.

Ping of death – starý typ útoku, v dnešní době snad už i nepoužitelný. Jedná se o ping, který svou velikostí překračuje RFC (65536 a více bajtů). Některá zařízení, která přijala tento pajet prostě zamrzla.
Ochrana:
Update systémů a vlastní test

Fraggle attack – představte si to samé co Smurf, ale jedná se o UDP a útok využívá služby chargen, echo, daytime a qotd.
Ochrana:
Nepoužívejte tyto služby. Jsou velice staré a v dnešní době nemají využití.

Smurf, Fraggle, POD, stejně jako další podobné útoky dokáže detekovat každý schopnější firewall.

// SYN flood

Žhavé téma, přitom jeden z nejjednodušších útoku ze všech. Útoky co jsme si do teď popsali využívali amplifikátorů, překračovali RFC, atp. SYN flood je obdoba Ping of Death. Prostě tisíce a tisíce spoofnutých SYN paketů zaslaných na server, kterému povětšinou dojde pamět, či backlog na příjem všech těch žádostí o nové spojení a přestane obsluhovat nová spojení legitimních klientů.

Útok vedený na zpravodajské servery vygeneroval podle grafů na zive.cz traffic někde kolem 30Mbit/s. To není mnoho. Je však nutné přiznat, že SYN flood generuje malý traffic. To co zabíjí servery je režie spojená s vedením a vytvářením nových spojení.
Podle našich údajů šlo v peaku o 40-50.000 nových spojení za vteřinu a to už je sakra dost. Vemte si, že se útočilo hned na několik serverů zároveň a tento údaj platí pro jeden z nich. Malý útok to rozhodně nebyl.

Jak jsem již zmiňoval, šel by ještě znásobit u firem, které nemají nastavené anti-spoof filtry na routerech a to tím, že by source i destination IP byli ze stejné organizace. Tím by na první server probíhal SYN flood útok a tento první server by sám útočil pomocí SYN/ACK DRDoS útoku na druhý, navíc ještě znásobený o retransmise.

Ochrana:

SYN cookies

To je to první co byste měli zapnout/nastavit. Ve zkratce když přijde na server SYN paket, tak server odpoví SYN/ACK a čeká na finální ACK. K tomu v případě SYN flood nedochází a proto máte najednou v SYN frontě deseti tisíce spojení kterým už jste rezervovaly prostředky systému (paměť, buffer). Tato fronta se samozřejmě rychle zaplní (čeká se na ACK, nebo až vyprší RTO) a nové požadavky o spojení jsou zahozené.
SYN cookies (ač jsou naprosto v souladu s RFC) se chovají jinak. Server přijme SYN, odpoví SYN/ACK, ale smaže ihned SYN záznam z fronty. Pokud časem přijde platný ACK paket server je schopen rekonstruovat SYN záznam z údajů v sekvenčním čísle ACK paketu. Tím pádem všechny SYN flood nedokončené spojení "nijak" nevytěžují prostředky serveru.

Systémové parametry

Potřebujeme ještě upravit nastavení backlogu a systémových bufferů, aby byl server schopen zpracovat všechny příchozích spojení.

Příklad z /etc/sysctl.conf

```
# Decrease the time default value for tcp_fin_timeout connection
net.ipv4.tcp_fin_timeout = 15

# Decrease the time default value for tcp_keepalive_time connection
net.ipv4.tcp_keepalive_time = 1800

# Enable tcp_window_scaling
net.ipv4.tcp_window_scaling = 1

# Turn off the tcp_sack
net.ipv4.tcp_sack = 0

# Turn off the tcp_timestamps
net.ipv4.tcp_timestamps = 0

# This removes an odd behavior in the 2.6 kernels, whereby the kernel stores
# the slow start threshold for a client between TCP sessions.
net.ipv4.tcp_no_metrics_save = 1

# Prevent SYN attack
net.ipv4.tcp_syncookies = 1
net.ipv4.tcp_max_syn_backlog = 4096
net.ipv4.tcp_syn_retries = 5
net.ipv4.tcp_synack_retries = 2
# Buffer size autotuning - buffer size (and tcp window size) is dynamically updated
for each connection.
# This option is not present in kernels older then 2.4.27 or 2.6.7 - update your
kernel
# In that case tuning options net.ipv4.tcp_wmem and net.ipv4.tcp_rmem isnt
recommended
net.ipv4.tcp_moderate_rcvbuf = 1

# Increase the tcp-time-wait buckets pool size
```

```
net.ipv4.tcp_max_tw_buckets = 1440000

# Increase allowed local port range
net.ipv4.ip_local_port_range = 1024 64000
```

Konfigurace byla použita z naší připravované distribuce Securix GNU/Linux *[20]*
Celou konfiguraci můžete nalézt zde: http://config.securix.org/config/securix-conf/etc/sysctl.conf [21]

Zde máte uveden výňatek sysctl konfigurace na Linuxu, který zapne SYN cookies a mnohonásobně (bez rizika) navýší hodnoty bufferu a jiných systémových parametrů, které jsou by default (RHEL, Debian, ...) příliš nízké. Především s hodnotou tcp_max_syn_backlog bych doporučil operovat, abyste nalezli optimální hodnotu.
Mám zde zapnutou auto-tuning hodnotu tcp_moderate_rcvbuf který by měl hodnoty upravovat automaticky podle aktuálního vytížení, ale někde zas píšou, že i tak má vlastní limity příliš nízko. Pokud se o tom někdo z vás chce pídit, tak vás odkážu sem:
http://fasterdata.es.net/host-tuning/linux/ [22]

Nastavení bufferu síťové karty

Vypište si na vašich strojích příkaz “netstat -i” a zaměřte se na sloupec RX-DRP

```
linux ~ $ netstat -i
Kernel  Interface  table
Iface    MTU        Met    RX-OK       RX-ERR  RX-DRP   RX-OVR  TX-OK       TX-ERR
 TX-DRP  TX-OVR  Flg
eth0     1500       0      1742981407  0       2607898  0       1176287425  0       0
      0       BMRU
lo       16436      0      126382      0       0        0       126382      0 0
0       LRU
```

Pokud tam uvidíte vysoké číslo (u mě je 2607898), které každý den narůstá, tak byste měli zvýšit RX a TX buffer na síťové kartě. Ten nárust je důležitý. Ono se může občas stát že dojde k RX-DRP, když je třeba systém chvilkově vytížen, ale nemělo by to příliš narůstat každý den. Dalším signálem problému může být velká utilizace soft IRQ. Zadejte příkaz “top” a poté zadejte číslo “1”. Uvidíte vytížení per CPU a v jednom ze sloupců (předposlední) hodnotu X.Y%si což je právě soft-irq. Pokud se zvyšuje hodnota RX-DRP a soft-irg běží nad 30-40% tak bude problém v TX/RX bufferu. V tom nám pomůže nástroj ethtool.

```
Linux ~ $ ethtool -i eth0
driver: 8139cp
version: 1.3
firmware-version:
bus-info: 0000:00:03.0
supports-statistics: yes
supports-test: no
supports-eeprom-access: yes
supports-register-dump: yes

Linux ~ $ ethtool -g eth0
Pre-set maximums:
RX: 4096
TX: 4096

Current hardware settings:
RX: 256
TX: 256
```

Jak vidíte maximum je 4096 a my máme nastaveno pouze 256.

Změníme konfiguraci TX i RX pomocí příkazu:

```
ethtool -G eth0 rx 4096 tx 4096
```

upozornění:
- tento příkaz provede ifup/ifdown, takže některá z aktuálních spojení se mohou ukončit (ne SSH)
- pokud je stroj v clusteru, tak může dojít k failoveru, takže to raději udělejte nejdříve na standby nodu a pak udělejte failover
- tato hodnota zmizí po rebootu, přidejte si proto command do rc.conf (například)

Tímto způsobem ochráníte systém jako takový. Prostudujte si dokumentaci k vaším firewallům, load balancerům a IPS sondám, protože většina z nich nabízí další úroveň ochrany proti SYN flood. Asi nemá cenu zde popisovat konkretní metody pro Cisco, BigIP, Check Point, Juniper atd.

Kdyby všichni ISP zapnuli anti-spoofing filtry na svých routerech, tak tyto útoky nemohou existovat. Prostě já jakožto uživatel konkretní sítě v ČR nemám co posílat paket se zdrojovou IP adresou např. z Ruska. Dokonce myslím že v jedné z desítky tisíc zpráv EU je to všem ISP doporučováno. Bohužel pro nás ne nařízeno.

SYN floodem tento díl ukončím, protože jsem se nějak rozepsal a většina lidí dle mého názoru ani tak daleko nedošla :]

Jsme na začátku jedné kapitoly jménem DoS v obrovské knize jménem bezpečnost. V dalších článcích (podle času) rozeberu ty zajímavější techniky.

Je to teď ve vašich rukou. Opravte si vaše systémy.

SP je na webhostingu, my moc možností nemáme :]

URL článku: https://security-portal.cz/clanky/seznamte-se-%E2%80%93-dos-ddos-%C3%BAtoky

Odkazy:
[1] https://security-portal.cz/users/cm3l1k1
[2] https://security-portal.cz/category/tagy/hacking-method
[3] https://security-portal.cz/category/tagy/security
[4] https://security-portal.cz/sites/default/files/DoS-attacks.png
[5] https://security-portal.cz/sites/default/files/network-path.png
[6] http://en.wikipedia.org/wiki/OCSP_stapling
[7] http://www.zabbix.com/
[8] http://crypto-world.info/
[9] https://security-portal.cz/sites/default/files/reflector.png
[10] http://www.seznam.cz
[11] https://www.security-portal.cz/clanky/obrana-p%C5%99ed-%C3%BAtokem-drdos
[12] http://homes.cs.washington.edu/~arvind/cs425/doc/drdos.pdf
[13] https://security-portal.cz/sites/default/files/amplifier.png
[14] https://security-portal.cz/sites/default/files/amplifier-size.png
[15] http://www.nic.cz/page/384/varovani-pred-otevrenymi-rekurzivnimi-nameservery/
[16] http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/white_paper_c11_603839.html
[17] http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SY/configuration/guide/ip_source_guard.html
[18] https://www.security-portal.cz/clanky/rozd%C4%9Blov%C3%A1n%C3%AD-ip-s%C3%ADt%C3%AD
[19] http://smurf.powertech.no/
[20] https://www.securix.org/
[21] http://config.securix.org/config/securix-conf/etc/sysctl.conf
[22] http://fasterdata.es.net/host-tuning/linux/