CSIRT.SK
www.csirt.gov.sk

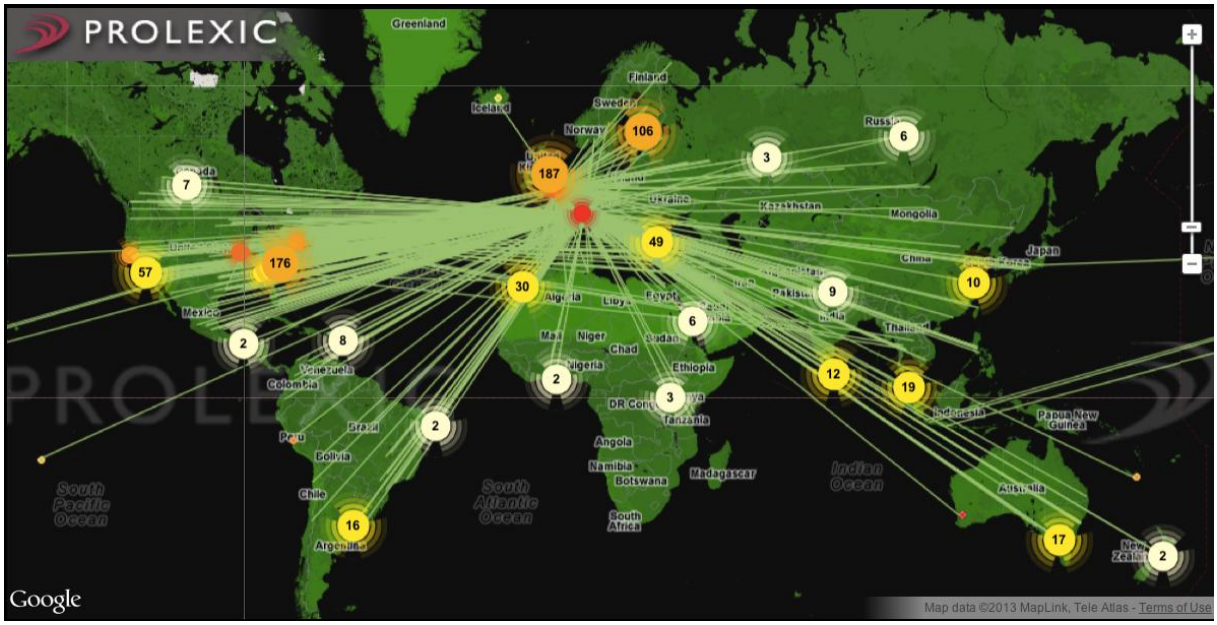
Ochrana pred útokmi DDoS

Príručka administrátora

Obsah

Čo je útok DDoS

V dnešnom svete sú kybernetické útoky chronickou záležitosťou internetu. Každý útok, ktorého cieľom je akýmkoľvek spôsobom narušiť plynulý priebeh služby, je útokom typu odoprenia služby – Denial of Service (DoS). Pokiaľ sa na takomto útoku podieľa viacero strojov útočiacich súčasne (obvykle desaťtisíce), útok sa nazýva distribuovaný, v skratke DDoS.



Zdroj obrázku: Prolexic

Klasickým spôsobom ako zaútočiť z mnohých cieľov súčasne je útočiť prostredníctvom botnetu. Botnet je sieť počítačov napadnutých vírusom. Vírus sa do počítača dostane klasickým spôsobom ako akýkoľvek iný škodlivý kód, avšak po infikovaní systému umožní autorovi vírusu prevziať kontrolu. Takýto napadnutý systém za volá „zombie“, alebo „bot“. Botnet je názov pre sieť botov. Botnet je možné nadobudnúť buď vytvorením vlastného škodlivého kódu, alebo je možné ho prenajať - odhadom 100 000 strojov je možné na deň prenajať relatívne nízku sumu. To znamená, že aj technicky nevzdelaní ľudia za pomerne malý finančný obnos môžu disponovať serióznou „palebnou silou“, ktorá dokáže zahltiť dátové linky, prípadne znefunkčniť konkrétne entity v sieti.

Dôvody a ciele útoku

Cieľom DDoS útoku je konkrétne zariadenie, služba alebo aj infraštruktúra inštitúcie. Úspešným útokom je narušená bežná prevádzka služby alebo siete a vzniká vlastníkovi škoda, ktorá je vyčísliteľná podľa povahy systému. Za predpokladu, že sa jedná napríklad o internetový obchod, ušlý zisk, poškodená reputácia a náprava škôd sa môže vyšplhať až do státisícov eur za deň. V prípade portálu verejnej správy môže byť narušené poskytovanie služby verejnosti.

Motivácia útočníkov môže byť akákoľvek, avšak vo všeobecnosti sa dá zaradiť do všeobecnejších celkov:

Peniaze alebo konkurenčný boj

Útokom na spoločnosť je možné ovplyvniť ich pozíciu v marketingu, hodnotu akcií, prípadne spôsobenie značných finančných škôd, ktoré môže využiť jednotlivec pre svoj osobný profit, prípadne

v rámci obchodnej stratégie na zvýhodnenie pozície vlastnej organizácie. Klasický cieľ predstavujú firemné siete.

Osobný kredit

DDoS útok môže byť vnímaný ako demonštrácia sily od jednotlivca, prípadne skupiny. Akýkoľvek cieľ je vhodný, čím väčší, tým útočník dosiahne väčší kredit.

Odplata

Motivácia je v tomto prípade čisto osobná, preto nie je možné označiť cieľ.

Forma demonštrácie

Klasickým príkladom tejto motivácie býva hackerský aktivizmus (tzv. hacktivizmus), ktorý sa snaží medializovať určitú agendu. Častým cieľom sú vládne, prípadne korporátne systémy.

Kybernetický terorizmus

Na rozdiel od demonštrácie je kybernetický terorizmus určený na vytvorenie paniky medzi ľuďmi, a preto je mierený na najcitlivejšie body vládnych a firemných systémov, kde takýto útok spôsobí maximálne škody. Klasický cieľ predstavujú kritické elementy štátu: banky, policajné systémy, energetika, telekomunikácie.

Odpútanie pozornosti

DDoS útok je možné použiť na odpútanie pozornosti od iného, dôležitejšieho útoku. Takýto útok môže predstavovať krádež cenných informácií, keďže v čase krádeže budú administrátori donútení venovaniu sa inej časti infraštruktúry.

Metódy DDoS

Z dôvodu, že cieľ útoku je jednotný (znefunkčniť jeden alebo viacero systémov obete), útočník si vyberá spôsob, ktorým tento stav dosiahne čo najjednoduchšie. Na výber má viacero útočných vektorov (spôsoby prevedenia útoku), ktoré sa odlišujú podľa podmienok, ktorým je útočník vystavený.

V sieti obete skoro vždy existuje mnoho zraniteľných bodov, ktoré (občas až extrémne) znižujú náročnosť vykonania DDoS útoku. Forma útoku je určená na základe analýzy siete obete a zvolenia najefektívnejšieho prístupu: zneužitie zraniteľností použitých protokolov a výber podporných prostriedkov.

Protokoly a ich zraniteľnosti

V internete je použitá rada protokolov sady TCP/IP. Delí sa na päť vrstiev, pričom existujú útoky na každú z nich. Čím nižšie číslo vrstvy, tým sú primitívnejšie.

1. Fyzické spojenia a dátové linky

Útoky na týchto vrstvách predstavujú útoky z vnútra (od zamestnancov). Útoky na fyzické spojenia predstavujú najprimitívnejšie útoky vôbec: pretrhnutie káblu, vypnutie elektrického prúdu a tak ďalej. Bezpečnosť na tejto úrovni je väčšinou riešená vnútornými smernicami o fyzickej bezpečnosti.

Dátové linky predstavujú lokálnu dátovú komunikáciu a na vykonanie útoku je nutná prítomnosť na lokálnej sieti (LAN), čo je možné dosiahnuť buď z počítača zamestnanca alebo cez zavírený počítač vo

vnútri siete, na ktorý má útočník prístup. Jeden z útokov na tejto vrstve je **ARP Spoof**, vďaka ktorému útočník vie presmerovať komunikáciu zariadení a spôsobiť ich nedostupnosť. Pokiaľ toto vykoná v časti siete ako je DMZ, odreže tým celý blok siete.

2. Internetová vrstva

Útoky na tejto úrovni môžu pochádzať z celého internetu, nie iba z lokálnej siete, preto do horných troch úrovní TCP/IP spadá gro všetkých dostupných útokov. Opäť platí, čim nižšia vrstva, tým primitívnejší útok.

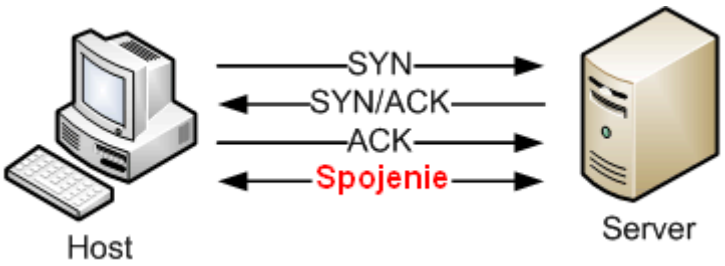
Útoky na tejto vrstve sú obmedzené na záplavu paketov, ktoré vyčerpajú kapacitu pripojenia obete do internetu.

Najčastejšie sú útoky zasielajúce paket ICMP Echo (PING) v rozličných formách. Populárna variácia tohto útoku, kde útočník zašle ICMP Echo so sfalšovanou zdrojovou adresou na broadcast adresu cudzej siete sa volá **Smurf** útok (prípadne **Fraggle**, ak sa použije UDP Echo). Tento jeden paket na neošetrených systémoch vyvolá odpoveď od každého systému na danej sieti.

O podporných mechanizmoch na znásobenie dátového toku je písané viac v sekcií Podporné faktory.

3. Transportná vrstva

Útoky na transportnej vrstve zneužívajú vlastnosti stavového protokolu TCP. Nadviazanie spojenia sa deje trojcestnou výmenou (three-way handshake) paketov, pričom server drží každé vytvorené spojenie. Pri útoku je možné použiť viaceré stavy TCP protokolu ako sú SYN, RST, PSH a veľkosť okna.



Tieto útoky vyplývajú zo správania protokolu TCP. Nadväzuje spojenie cez trojcestnú výmenu (3-way handshake) a jeho činnosť je možné ovplyvniť na základe spomenutých informácií tak, aby spotrebovalo veľké množstvo zdrojov, prípadne zablokovalo dodanie obsahu legitímnym používateľom.

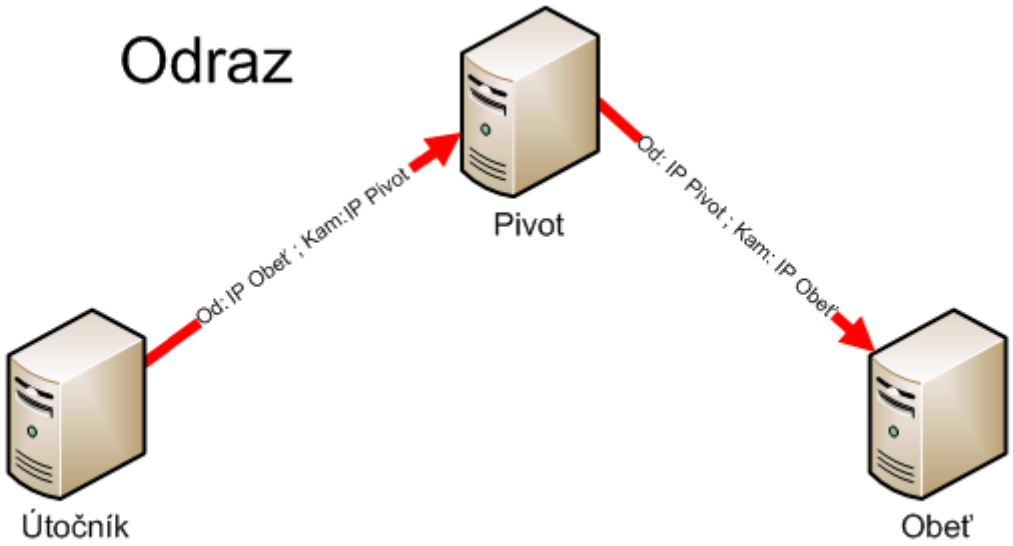
4. Aplikačná vrstva

Útoky na aplikácie zneužívajú zlú konfiguráciu aplikácií, neefektívne narábanie so systémovými zdrojmi (zahltenie pamäte, zvýšenie záťaže procesora, zahltenie diskového radiča), prípadne napádajú zraniteľnosti v aplikáciách, ktoré spôsobia ich pád. (aplikácia po prijatí špeciálne upraveného paketu zamrzne a prestane reagovať, prípadne sa vypne po chybe).

Každá aplikácia ktorá je spustená vyžaduje osobitý prístup. Klasické príklady DDoS útokov na aplikácie využívajú exploity, SQL, HTTP (**slowloris**, **pyloris**) a tak ďalej. Zraniteľností na aplikačnej vrstve je najviac.

Podporné faktory

Podporné faktory sú spôsoby, ako je možné použitím jednoduchých metód niekoľkonásobne zvýšiť silu DDoS útoku. Je potrebné týmto metódam porozumieť a znemožniť ich využitie.



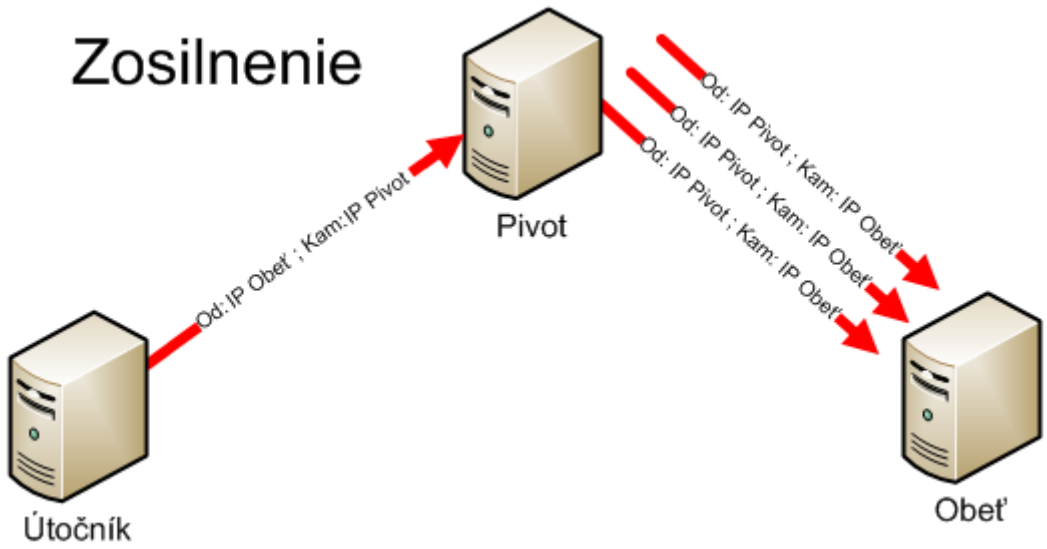
Klasická metóda odrážania paketov od cudzieho stroja spôsobí, že na pivot (záchytný bod) dorazí prvotný paket požadujúci reakciu so zdrojovou adresou obete. Následne je vygenerovaná odpoveď a zaslaná obeti. Takouto metódou obeť komunikuje s pivotom a nie s útočníkom priamo.

Útok s odrazom je možné využiť pri takzvanom **Reflected DoS**, v skratke RDoS (DRDoS). Útok spočíva v zasielaní TCP SYN paketov na obeť so sfalšovanou adresou iného stroja v sieti (ideálne v sieti obete).

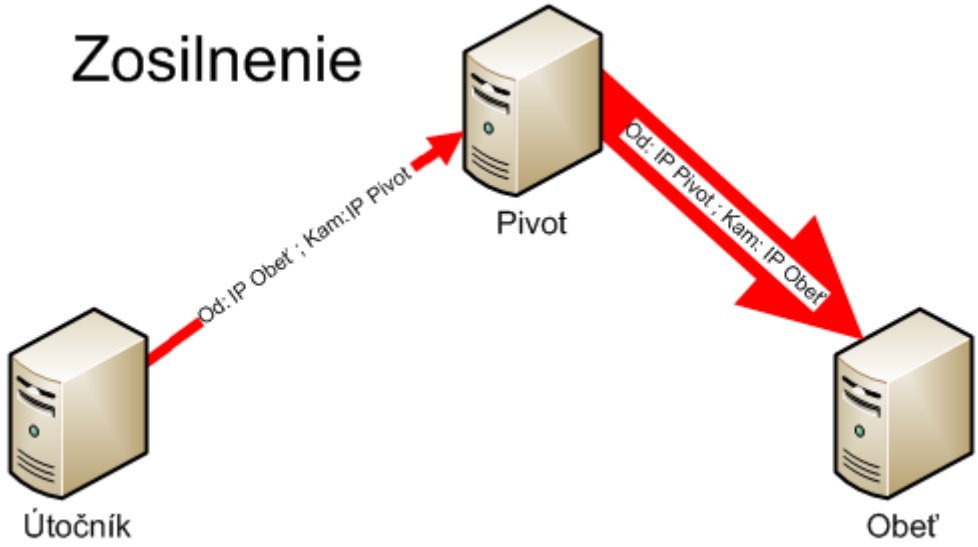
Prijatie paketu TCP SYN spôsobí odpoveď TCP SYN+ACK na zdrojovú IP adresu. Keďže tento stroj nezačal spojenie, zruší spojenie cez zaslanie TCP RST. Tým pádom server obete minie zdroje na vygenerovanie paketu (TCP SYN+ACK), navyše sa ešte raz zahltí linka prenosom paketu TCP RST.

Čiastočná obrana voči tomuto útoku je nastaviť firewall na každom serveri v sieti tak, aby blokoval všetky pakety z vnútornej siete mimo portov, ktoré sú nevyhnutné na prevádzku.

Zosilnenie má dve formy: zosilnenie počtu paketov a zosilnenie veľkosti paketov.



Takýto typ zosilnenia spôsobí, že jeden paket na vstupe vytvorí reakciu viacerých paketov. Takýto stav je možné dosiahnuť buď cez zaslanie požiadavky na viacero serverov v pozadí, prípadne na opakované generovanie odpovede smerom ďalej. Príkladom takéhoto útoku je **Smurf**, prípadne **Fraggle** útok.



Príkladom je zosilnenie útoku za použitia DNS serverov. DNS server na 70 bajtovú požiadavku odpovie zvyčajne 160 bajtmi, čo spôsobí približne 2.5násobné zosilnenie. Avšak pri zabezpečených záznamoch DNSSEC sú uvádzané digitálne podpisy pod záznamami, čo značne nafukuje objem dát, kde sa objem môže vyšplhať aj na 2500 bajtov. Takéto 30násobné nafúknutie pre obeť znamená, že útočník dokáže akumulovať jednoduchým spôsobom mnohonásobne väčšiu palebnú silu.

Na takýto útok je možné využiť všetky DNS servery, ktoré odpovedajú na rekurzívne požiadavky z internetu, takzvané otvorené prekladače (Openresolver). Ako obrana je nevyhnutné, aby každý DNS server mal umožnené prijímať rekurzívne požiadavky iba od dôveryhodných entít.

Útoky

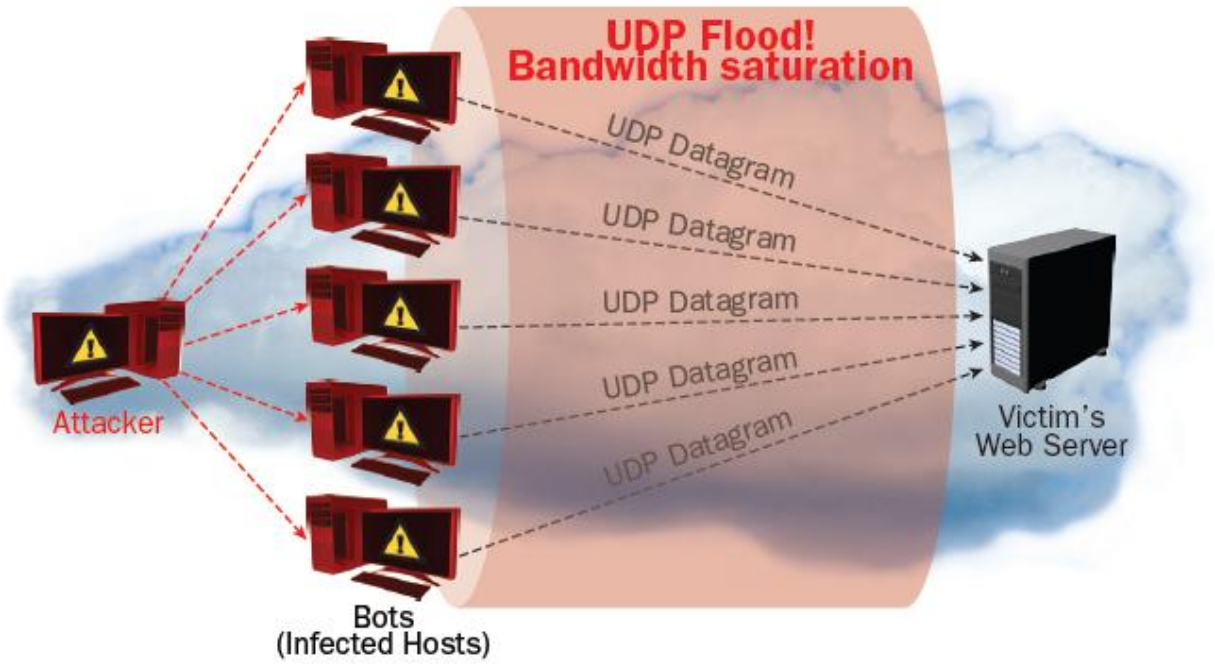
Existuje veľké množstvo útokov na rôzne platformy. Pre ilustráciu budú uvedené najzásadnejšie.

Útoky na sieťovú + transportnú vrstvu

Útoky zamerané na tieto vrstvy sa rôznia. na tejto úrovni sú možné primitívne záplavy paketov, ako aj metódy na prerušovanie spojení medzi používateľmi.

Flood (UDP, ICMP, IGMP)

Záplava paketov (packet flood) zahlcuje sieť obrovským množstvom paketov, a teda sa snažia „upchať" prístupovú cestu. Klasický prípad sú UDP, ICMP a IGMP pakety.



Zdroj obrázku: Radware

Tento útok nezneužíva žiadne zraniteľnosti, pracuje iba na základe hrubej sily. Pokiaľ útočník dokáže zaslať na obeť dostatočne veľké množstvo paketov, obeť sa stane nedostupnou ako dôsledok zahltenia. Obvyklou súčasťou útoku je botnet, ktorý generuje pakety so sfalšovanou zdrojovou IP adresou.

Pri tomto útoku sa obvykle využívajú zosilňujúce podporné faktory, ktoré dokážu zväčšiť objem zasielaných dát.

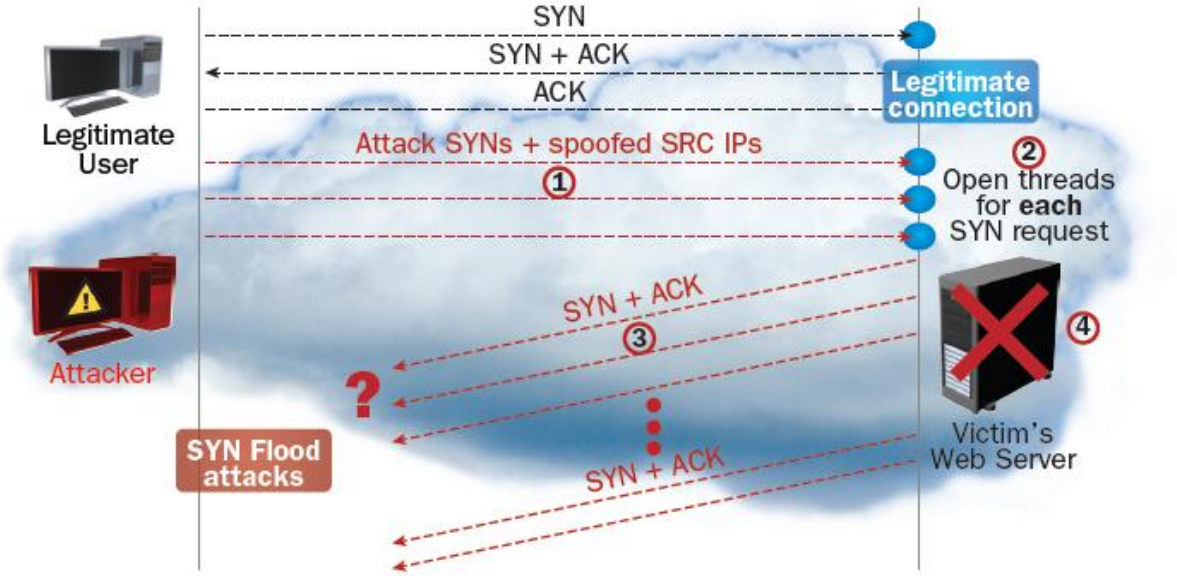
TCP SYN Flood

TCP SYN Flood zneužíva vlastnosť TCP protokolu, kde server po prijatí TCP SYN musí odoslať odpoveď TCP s príznakmi SYN+ACK, a držať toto spojenie v pamäti až do uplynutia časovača.

Jedná sa o mimoriadne jednoduchý a rozšírený útok, kde útočník zasiela obrovské množstvo TCP SYN požiadaviek zo sfalšovanej zdrojovej IP adresy. Napadnutý server musí otvoriť spojenie na každú požiadavku a **uložiť do pamäte túto reláciu.** Útok je úspešne ukončený po tom, ako sa minú systémové zdroje na vytváranie nových relácií a celý systém sa zrúti, čím spôsobí neprístupnosť na dlhšiu dobu.

Je možné vykonávať takzvaný Reflected DoS, ktorý bol popísaný v sekcií o podporných faktoroch.

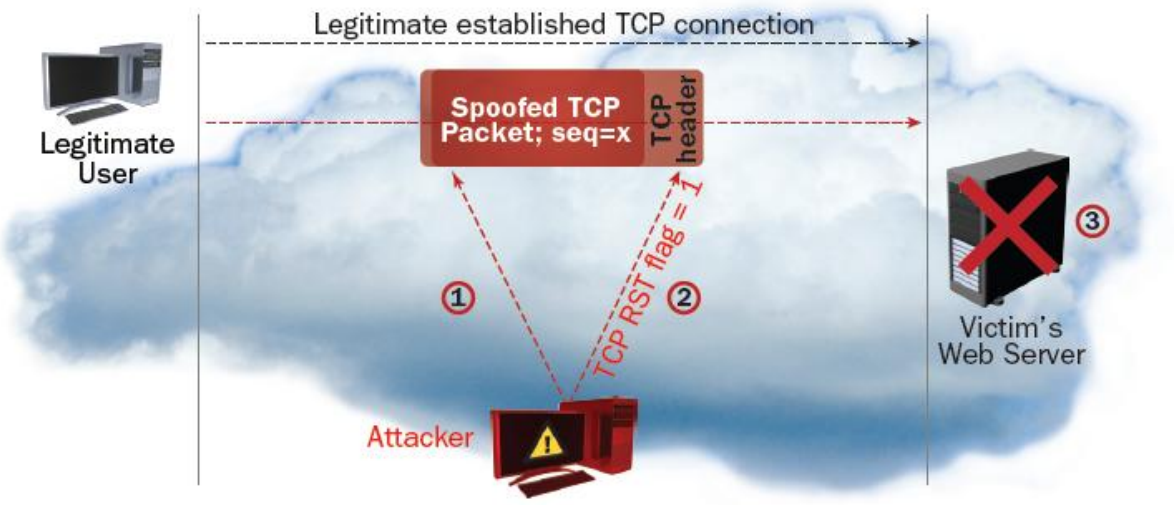
Tento útok je možné vykonať širokým portfóliom nástrojov, najtypickejšími sú Low Orbit Ion Cannon (**LOIC**), High Orbit Ion Cannon (**HOIC**) a **hping**, ktoré sú všetky hojne využívané skupinami Anonymous.



Zdroj obrázku: Radware

Jedna z možných ochrán voči tomuto útoku je tzv. TCP Cookie. Systém po prijatí TCP SYN odošle TCP SYN+ACK a nevytvorí v pamäti žiadnu reláciu. Po prijatí legitímnej odpovede TCP ACK sa spätne dopočíta TCP sekvencia paketov a vytvorí sa relácia. Takýmto spôsobom sa nezaťažujú systémové zdroje, okrem záťaže spôsobenej prijímaním enormného množstva paketov.

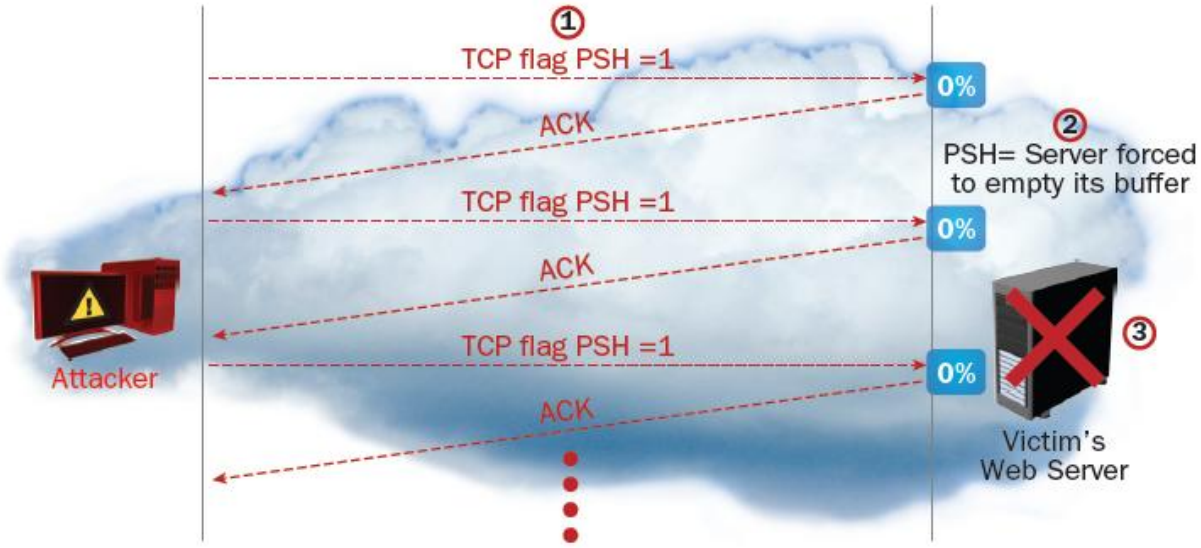
TCP RST útok



Zdroj obrázku: Radware

TCP RST útok je útok na legitímne spojenia používateľov serveru, kde útočník háda sekvenčné čísla TCP spojení a v prípade že sa trafí, spojenie je ukončené. Za predpokladu, že útočník má k dispozícii botnet, je viac než pravdepodobné že sa takýto útok podarí uskutočniť. K tomuto útoku je potrebné vedieť zdrojovú IP adresu klienta, čo znemožňuje použiť tento útok vo veľa prípadoch Avšak pri komunikácií kde sú vopred dohodnuté koncové uzly, ako napríklad elektronická aukcia, je možné použiť tento útok.

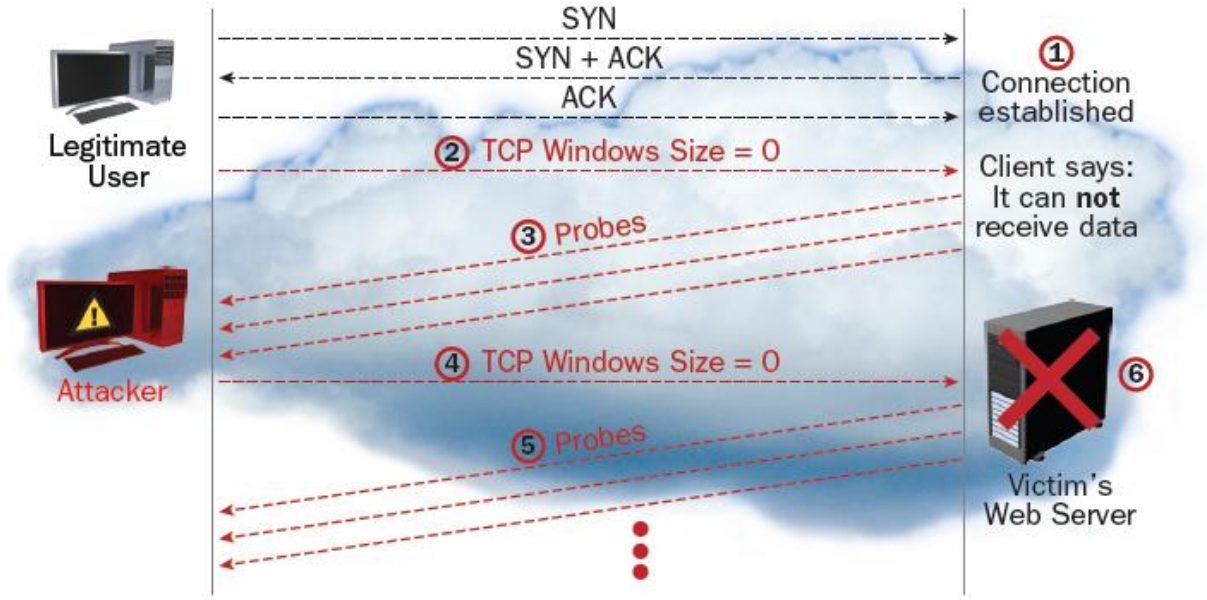
TCP PSH+ACK útok



Zdroj obrázku: Radware

Tento útok využíva príznak „PUSH“ v TCP. Server po prijatí takejto správy okamžite odošle obsah bufferu (vyrovnávacia pamäť) naspäť ku klientovi a túto akciu potvrdí odoslaním správy s príznakom ACK. Ak dokáže útočník vygenerovať dostatočne veľké množstvo paketov (napríklad použitím botnetu), je možné zahltiť server mnohými požiadavkami, až do stavu kde server nie je schopný spracúvať ďalšie požiadavky (čo spôsobí odoprenie služby pre všetkých používateľov).

Útok na veľkosť TCP okna



Zdroj obrázku: Radware

Útok na veľkosť okna v TCP nie je klasický útok ako predošlé. Nevyužíva záplavu rámcov, ale zraniteľnosť v protokole TCP. Veľkosť okna v TCP je mechanizmus, ktorým si uzly dohadujú objem dát, ktorý môže byť poslaný do siete bez toho, aby bolo potvrdené prijatie. Na základe tejto výmeny sa prispôsobuje prenos podmienkam stratovosti dát na sieti. Veľkosť okna je priamo prepojená s veľkosťou bufferu na strane servera.

Útočník môže nastaviť veľkosť okna na 0, čím oznámi serveru že nie je možné posielať žiadne dáta. Server periodicky posiela kontrolné pakety (tzv. „probes" – sondy) aby overil, kedy je používateľ dostupný. V prípade útoku sa táto dostupnosť nikdy neobjaví a teda server musí držať otvorené spojenie donekonečna.

Po prijatí mnohých spojení tohto typu server vyčerpá miesto v tabuľke spojení, čím odoprie prístup všetkým ďalším používateľom. Ak útočník nenastaví veľkosť okna na 0, ale veľmi malú, donúti server odpovedať v obrovskom kvante malých blokov dát, čím je možné spôsobiť zahltenie pamäte servera, čím sa opäť vytvoria podmienky ktoré odoprú prístup používateľom.

Jedným z nástrojov, ktorým je možné vytvoriť takýto útok sa nazýva **Sockstress**.

Útoky na relačnú vrstvu

Relačná vrstva obhospodaruje komunikačné protokoly ako je SSL a DNS. Najtypickejšie útoky na tejto vrstve míňajú sieťovú kapacitu a procesorový čas.

Útoky na SSL

SSL je vynikajúci útočný vektor z dôvodu, že je náročný na výpočty. Jeden útočník bez botnetu je schopný odstaviť server, ktorý využíva SSL. To znamená **každý server ktorý používa HTTPS**, čo je absolútna väčšina.

Záplava SSL

Záplava SSL spojení prebieha tak, že útočník v každom novom spojení vyvolá SSL handshake (nadviazanie SSL šifrovania v spojení). Napadnutý server musí vložiť záznam do tabuľky a prepočítať

určité hodnoty. Tento útok spotrebuje aj miesto v pamäti a aj výpočtový výkon. Pri veľkom množstve spojení server prestane reagovať.

Útok na SSL renegotiation

Tento útok spočíva v legitímnom nadviazaní TCP spojenia so serverom a následnom vyžiadaní SSL spojenia. Po prijatí certifikátov potrebných na vytvorenie spojenia sa okamžite zašle požiadavka na opakovanie negociácie. Takáto požiadavka spotrebuje 15x viac výpočtového výkonu na strane servera ako na strane klienta. Útočník neustále pridáva ďalšie spojenia až do momentu, kedy je server zahltený.

Riešenie tohto problému nespočíva vo vypnutí SSL (ktoré je mimoriadne dôležité na ochranu používateľských dát), ale v blokovaní požiadaviek na renegotiation. Prípadná ďalšia obrana spočíva v použití SSL offloadingu – požitie špecializovaného hardvérového zariadenia na šifrovanie, dešifrovanie a renegociáciu pri použití SSL.

Nástroj ktorý vykonáva tento útok je napríklad **THC-SSL-DOS**

Útoky na aplikačnú vrstvu

Aplikačná vrstva využíva zdroje ako je výpočtový výkon, pamäť a prístup na disky. Je ich možné vyčerpať a znemožniť tým prístup pre legitímnych používateľov. Otvára sa tu obrovský priestor bezpečnostných zraniteľností samotných aplikácií (Apache web server, Asterix VoIP PBX,...), ktoré sú náchylné na útok. Navyše, protokoly ako HTTP(S), DNS, SMTP, FTP, VOIP a SSL môžu vo svojich implementáciách obsahovať zraniteľnosti, vďaka ktorým bude mať útočník triviálnu úlohu.
V súčasnosti je evidované veľké množstvo útokov, ktoré sú často analyzované na bezpečnostných konferenciách.

Príkladom takýchto zraniteľností protokolov je pomalý útok posielania HTTP hlavičky, vykonávaný nástrojom **Slowloris**. Voči útokom na webové aplikácie a služby je potrebné postupovať metodickým prístupom na odhaľovanie zraniteľností, ako je napríklad OWASP. Útoky na aplikácie a služby presahujú rozsah tohto dokumentu.

PDoS

PDoS je skratka permanentného DoS útoku. Spočíva v takej modifikácií systému, kde jediné možné riešenie je výmena hardvéru, prípadne reinštalácia systému. Príkladom môže byť inštalovanie aplikácie na telefón, ktorá poškodí operačný systém a tým sa telefón stane nepoužiteľný. Podobné situácie môžu nastať s určitými špecializovanými zariadeniami.

Chyby vo firmvéri

Zneužitie chyby vo firmvéri môže byť napríklad volanie funkcie radiča, ktorá spôsobí pád systému.

Použitie hrubej sily na aplikáciu/funkciu

Klasický príklad je pokus o lámanie hesiel s použitím veľkého výpočtového výkonu v botnete. Pokiaľ v systémoch sú neošetrené vstupy, kde sa útočník môže prihlásiť, s pomocou hrubej sily je možné prelomiť slabšiu ochranu.

Exploity/Zraniteľnosti

Zverejnené bezpečnostné zraniteľnosti tvoria riziko pre každý systém. Bez častých aktualizácií je možné zmeniť sofistikovaný útok na triviálny. Príkladom je, že sa objaví špeciálny reťazec znakov, ktorý keď je zaslaný na server, tak daný server prestane reagovať.

R-U-Dead-Yet? (RUDY)

Rudy je nástroj, ktorý generuje útok typu „low and slow“. Útok zneužíva zraniteľnosť v HTTP protokole. Generuje malé množstvo dát, preto je možné že prejde „pod radarom“ množstva ochranných prvkov. Rudy útočí na formuláre HTTP POST, kde zasiela extrémne pomalým spôsobom po 1 bajte dáta. Toto núti server aby držal spojenie dlhodobo otvorené. Veľké množstvo spojení vyminie systémové zdroje a server je nedostupný.

Slowloris

Slowloris pracuje na podobnom princípe ako RUDY, avšak namiesto HTTP POST zneužíva zraniteľnosť HTTP protokolu pri posielaní požiadaviek. Tieto požiadavky posiela extrémne pomalým spôsobom, čo ako pri RUDY spôsobí neprimerane dlhé trvanie relácií, nahromadením ktorých je možné zahltiť server.

Billion Laughs (XML Bomb)

Tento útok je namierený na softvér, ktorý spracúva XML. Tento útok spočíva v násobení reťazcov, ako je uvedené v príklade: reťazec 1 je definovaný ako krátky reťazec. Reťazec 2 je definovaný ako desať reťazcov 1, reťazec 3 je definovaný ako desať reťazcov 2 a tak ďalej. Nafukovaním sa posledný reťazec nafúkne až na jednu miliardu opakovaní prvého reťazca. Takýto útok veľmi rýchlo vyčerpá pamäť obete (minimálne sa tak dialo v čase zverejnenia zraniteľnosti).

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
 <!ENTITY lol "lol">
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
 <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
 <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
 <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
]>
<lolz>&lol9;</lolz>
```

Fork bomb

Fork bomb je zneužitie zraniteľnosti vykonania kódu na strane obete. Fork je činnosť, kedy sa bežiaci proces rozdelí na dva simultánne bežiace procesy. Fork „bomba“ zneužíva túto činnosť tým, že nekontrolovateľne delí procesy až do zahltenia pamäte.

SQL Wildcard

Tento útok sa snaží zahltiť server cez vyťaženie procesora cez výpočtovo náročné požiadavky do databázy použitím viacero wildcard znakov ako napríklad „[]“ alebo „[^]“.

Vyčerpanie DHCP

Vyčerpanie DHCP znamená, že útočník donúti server aby pridelil všetky dostupné voľné IP adresy útočníkovi. Legitímny používateľ nebude schopný dostať IP adresu a teda všetkým novým používateľom bude odoprený prístup do siete.

Útoky na DNS

DNS servery sú mimoriadne dôležitý bod sieťovej infraštruktúry, z dôvodu že poskytujú preklad textových adries na IP adresy. V prípade že je tento server odstavený, žiadny jeho používateľ nemôže zadávať URL adresy (www.example.com). DNS servery sú preto jeden z prvých cieľov útoku.

Záplava DNS

Útoky s cieľom zaplaviť DNS server je zväčša ťažké detegovať, keďže aj rozličné legitímne služby dokážu generovať obrovské množstvo požiadaviek na DNS. Útočník sa snaží dosiahnuť až tak veľké množstvo požiadaviek, že ich server nedokáže zvládať a prestane reagovať.

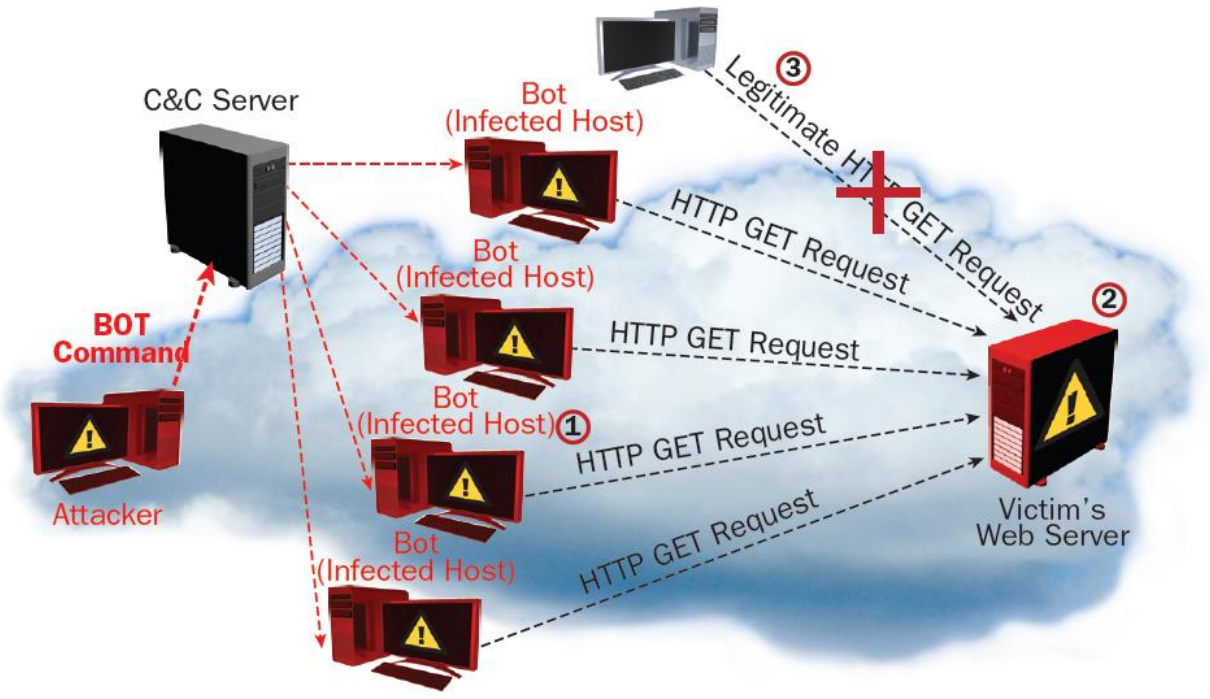
Útoky na HTTP

Útoky na HTTP sú najčastejšie sa opakujúce útoky. Pohybujú sa od triviálnych až po sofistikované, kde najtriviálnejší útok tvorí opakované načítavanie multimediálneho obsahu (zahltenie linky) cez automatizovaný nástroj.

Záplava HTTP

Záplava dát je najjednoduchším útočným vektorom, kde útočník vygeneruje enormné množstvo http požiadaviek a zahltí server, čím znemožní činnosť legitímnym používateľom. Útok prebieha zasielaním požiadaviek GET alebo POST, najčastejšie cez botnet (kvôli objemu dát).

Ako každá iná záplava, takýto útok počíta čisto s kvantitou požiadaviek, ktorá zahltí daný server. Pokiaľ je požadovaný multimediálny obsah, takáto komunikácia môže spotrebovať veľké množstvo dát, čo môže mať katastrofálne dôsledky, pokiaľ napadnutá organizácia platí za prenesené dáta.

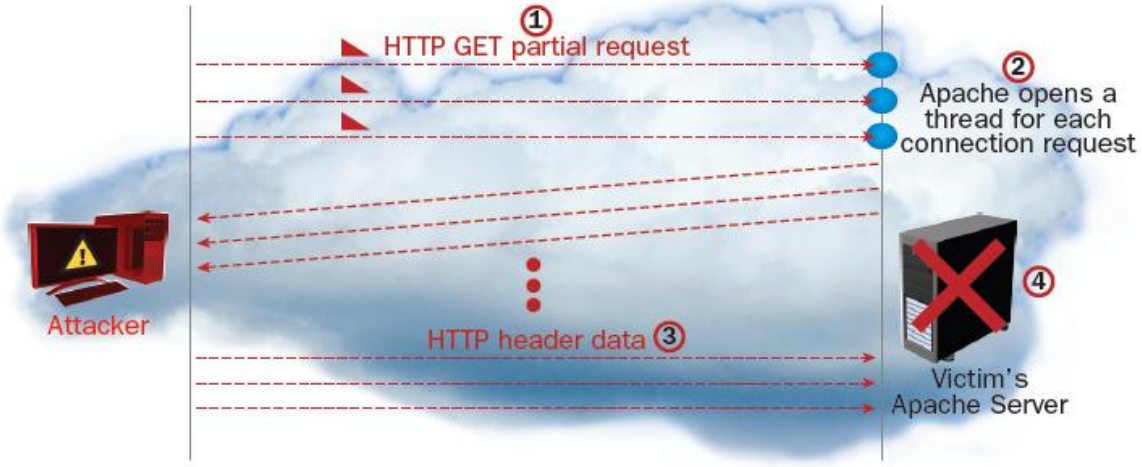


Zdroj obrázku: Radware

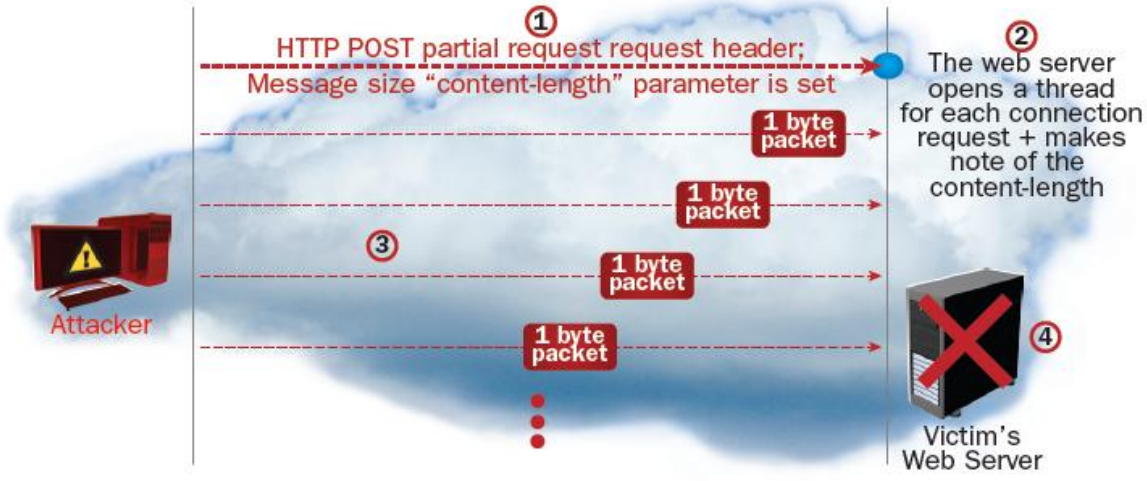
Záplava HTTPS

Záplava HTTPS paketov spočíva v rovnakom princípe ako útok na HTTP, avšak s pridanou záťažou šifrovania/dešifrovania SSL spojenia.

Pomalý HTTP GET a POST



Zdroj obrázku: Radware



Zdroj obrázku: Radware

Podstata tohto útoku je v zasielaní požiadavky HTTP GET (prípadne POST) po čo najpomalšie, teda v najmenších fragmentoch v čo najväčších odstupoch. Týmto sa spôsobí, že server musí neúmerne dlho držať otvorené spojenie, čím sa zapĺňa tabuľka spojení. Mnoho paralelných spojení spôsobí zahltenie.

Prítomnosť pomalého prijímanie dát je prirodzená súčasť web servera, aby mohol prijímať pripojenia aj od používateľov s pomalým pripojením.

Obrana pre obidva prípady je nastavenie maximálneho časového intervalu pre vypršanie spojenia.

Hash DoS

Hash DoS je útok na webový server, ktorý využíva slabé hashovacie funkcie. Útočník sa snaží vytvoriť na serveri kolízie týchto funkcií na základe špeciálne upravených vstupov. Server po vzniku kolízie (už použitého hashu) musí daný prepočítať a určiť novú hodnotu. Útok spočíva v snahe vytvoriť čo najdlhšiu možnú reťaz kolízií, čo spôsobuje nadmernú výpočtovú záťaž. Po veľkom množstve požiadaviek na výpočty server prestane reagovať.

Regex DoS (ReDoS)

Tento útok zneužíva zraniteľnosť v knižnici spracúvajúcej regulárne výrazy. Útočník zašle špeciálny regulárny výraz (zbežne nazývaným „evil regex“), ktorý spôsobí výraznú výpočtovú záťaž na systéme obete. Zasielaním veľkého množstva takýchto požiadaviek je server zahltený.

Útok na databázy

Útoky na databázy predstavujú útok, ktorý sa zameriava na úzke hrdlo medzi databázou a webovou aplikáciou. Generovaním veľkého množstva požiadaviek na čítanie dát z databázy je možné spôsobiť zahltenie komunikačného kanálu, ktorým sa tieto dáta prenášajú. Požiadavky je väčšinou možné generovať vo veľkom množstve, pokiaľ to umožňuje štruktúra stránky (?page=151651)

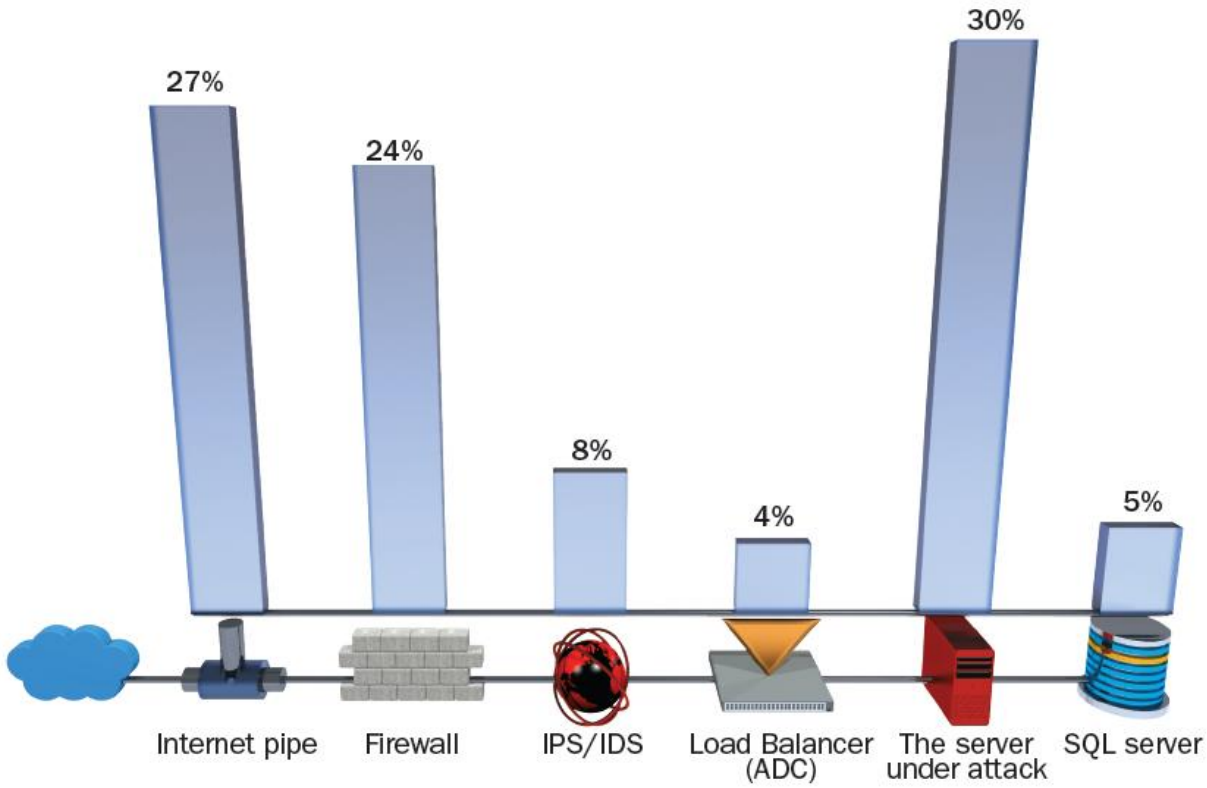
Útoky na SIP (VoIP)

Voice over IP je protokol na prenos hlasovej komunikácie (telefón). Čoraz častejšie sa nasádza vo firemných priestoroch, preto je potrebné dbať aj na tento aspekt. Najbežnejší protokol na VoIP je SIP.

Záplava INVITE

INVITE je SIP správa určená na vytvorenie telefónneho hovoru. Každý nový hovor je uložený do tabuľky hovorov a následne sú odoslané správy pre volaného. Za predpokladu, že týchto správ je enormné množstvo, telefónna ústredňa (server) prestane reagovať a odoprie prístup pre používateľov. Možnosti ochrany na konkrétnych zariadeniach

Voči týmto útokom je možné sa efektívne brániť správnou konfiguráciou zariadení, prípade prítomnosťou špecializovaných zariadení v sieti. DDoS je primárne útok zameraný na vyplytvanie zdrojov, pričom kľúčové sú úzke hrdlá. Podstatná informácia o problémoch úzkych hrdiel je, že problémy s nimi súvisiace sa nedajú odstrániť, iba oddialiť.



Zdroj obrázku: Radware

Chybná konfigurácia je taká, ktorá umožňuje útočníkom vytvoriť neprimeranú záťaž na zariadeniach. V nasledujúcej sekcií sú popísané odporúčania pre konfiguráciu niektorých vybraných systémov a zariadení.

Cisco ASA

Smurf útok je v základnej konfigurácií blokovaný, avšak pri starších verziách IOS je možné, že nie. Aplikovať konfiguráciu *Router(config-if)# no ip directed-broadcast*

ASA dokáže automaticky rozpoznávať daktoré útoky, napr. bez ďalšej konfigurácie filtruje pakety PSH+ACK, ktoré nepatria do žiadnej komunikácie. Ďalej filtruje SSL renegociačný útok

TCP SYN Flood

Príklad ochrany voči TCP SYN Flood. Táto konfigurácia nastaví maximálny limit pripojení na port 80 na 100 spojení a max. 200 polo-otvorených (neukončený handshake, embryonic) spojení. ASA po prekročení limitov automaticky aplikuje na spojenia kontrolu leigitímnosti cez SYN Cookies.

```
ciscoasa(config)#class-map tcp_syn
ciscoasa(config-cmap)#match port tcp eq 80
ciscoasa(config-cmap)#exit
ciscoasa(config)#policy-map tcpmap
ciscoasa(config-pmap)#class tcp_syn
ciscoasa(config-pmap-c)#set connection conn-max 100
ciscoasa(config-pmap-c)#set connection embryonic-conn-max 200
```

```
ciscoasa(config-pmap-c)#set connection per-client-embryonic-max 10
ciscoasa(config-pmap-c)#set connection per-client-max 5
ciscoasa(config-pmap-c)#set connection random-sequence-number enable
ciscoasa(config-pmap-c)#set connection timeout embryonic 0:0:45
ciscoasa(config-pmap-c)#set connection timeout half-closed 0:25:0
ciscoasa(config-pmap-c)#set connection timeout tcp 2:0:0
ciscoasa(config-pmap-c)#exit
ciscoasa(config-pmap)#exit
ciscoasa(config)#service-policy tcpmap global
```

Reverse path forwarding

Reverse path je zaslanie paketu, ktorý má zdrojovú adresu falošne nastavenú na takú, aby bola rovnaká ako adresa siete, kam sa vysiela paket. Je dôležité nastaviť tento filter na rozhraniach smerujúcich do internetu.

```
hostname(config)#ip verify reverse-path interface interface_name
```

Falošný rozsah IP

Ako falošné môžeme automaticky pokladať všetky IP adresy, ktoré prichádzajú z **internetového** rozhrania a majú privátne adresy. Je mimoriadne dôležité, aby sa overilo či sa firewall nenachádza v NAT, aby neboli filtrované legitímne pakety.

Vytvoríme rozšírený access list s číslom napríklad 110.

```
access-list 110 deny ip host 0.0.0.0 any
access-list 110 deny ip host 10.0.0.0 255.0.0.0 any
access-list 110 deny ip 127.0.0.0 0.255.255.255 any
access-list 110 deny ip host 172.16.0.0 255.240.0.0 any
access-list 110 deny ip 192.168.0.0 0.0.255.255 any

následne aplikovať Access-list na rozhranie smerujúce do internetu v smere
IN.
```

Detailné odporúčanie na mitigáciu útokov je možné nájsť tu:
http://www.cisco.com/en/US/docs/solutions/Enterprise/Security/SAFE_RG/safemediumentnetworks.html

Linux

Operačný systém Linux je populárna voľba systému na servery. Konfiguráciou balíka TCP/IP je možné zvýšiť ochranu sieťovej a transportnej vrstvy pre všetky služby, ktoré sú na danom stroji spustené.

TCP SYN Flood

Príklad ochrany voči TCP SYN Flood, plus ochrana cez SYN Cookies.

```
# 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
```

Apache HTTP FLOOD

Ochrana web serveru Apache cez službu aplikačného firewallu **mod_evasive**

```
<IfModule mod_evasive20.c>
# size of hash table
DOSHashTableSize 4096
# requests for the _same_ page per interval and client
DOSPageCount 20
# requests for any object by same client
DOSSiteCount 300
# threshold in second intervals
DOSPageInterval 1
DOSSiteInterval 1
DOSBlockingPeriod 30
#DOSCloseSocket On
#DOSSystemCommand "/sbin/iptables -I INPUT -s %s -j DROP"
DOSWhitelist 127.0.0.1
DOSEmailNotify your@email.com
DOSLogDir /var/log/httpd/evasive.log
</IfModule>
```

Ochrana pred pomalými útokmi

Na určitých operačných systémoch je možné nainštalovať mod_qos. Je to služba, ktorá slúži ako Web Application Firewall a určuje parametre spojení.

Je to modul, ktorý je v súčasnosti nutné stiahnuť a skompilovať na systéme.

Príklad pre Debian (Lenny):

```
cd /tmp/
wget http://downloads.sourceforge.net/sourceforge/mod-qos/mod_qos-8.13-
src.tar.gz?use_mirror=freefr
tar xvfz mod_qos-8.13-src.tar.gz

apt-get install apache2-threaded-dev gcc

cd mod_qos-8.13/apache2/
apxs2 -i -c mod_qos.c
```

V zložke /etc/apache2/mods-available vytvorte súbory qos.load a qos.conf .
Do súboru qos.load vložte riadok:

```
LoadModule qos_module /usr/lib/apache2/modules/mod_qos.so
```

Do súboru qos.conf vložte nasledujúce riadky:

```
## QoS Settings
<IfModule mod_qos.c>
    # handles connections from up to 100000 different IPs
    QS_ClientEntries 100000
    # will allow only 50 connections per IP
    QS_SrvMaxConnPerIP 50
    # maximum number of active TCP connections is limited to 256
    MaxClients              256
    # disables keep-alive when 70% of the TCP connections are occupied:
    QS_SrvMaxConnClose      180
    # minimum request/response speed (deny slow clients blocking the
server, ie. slowloris keeping connections open without requesting
anything):
    QS_SrvMinDataRate       150 1200
    # and limit request header and body (carefull, that limits uploads and
post requests too):
    # LimitRequestFields      30
    # QS_LimitRequestBody     102400
</IfModule>
```

DNS openresolver

Je možné, že vo vašej inštitúcií je DNS server chybne nakonfigurovaný a umožňuje jeho využitie útočníkom ako Open resolver. Nasledujúca konfigurácia odstraňuje tento problém:

BIND

```
  # príklad, nahradiť 192.0.2.0/24 za list vaších IP
  acl "trusted" {
      192.0.2.0/24;
  };

  options {
      recursion no;
      additional-from-cache no;
      allow-query { none; };
  };

  view "trusted" in {
      match-clients { trusted; };
      allow-query { trusted; };
      recursion yes;
      additional-from-cache yes;
  };
```

Windows DNS

Riešenie je možné nájsť tu: http://technet.microsoft.com/en-us/library/cc787602.aspx

Záplava DNS

Príklad pre BIND:

```
rate-limit {
responses-per-second 5;
window 5;
};
```

Príklad pre Knot:

```
system {
rate-limit 200; # Each flow is allowed to 200 resp. per second
rate-limit-slip 2; # Every other response is slipped (default)
}
```

F5 BigIP ASM

BigIP ASM je zariadenie určené na ochranu voči DDoS útokom na sieť. Sama rieši veľké množstvo útokov: TCP SYN flood, ICMP flood, UDP flood, UDP fragment attack, ping of death, Land attack a teardrop. Pochopiteľne, útoky typu záplava sú riešené filtrovaním paketov, čiže nie je možné ochrániť internetové spojenie pred zaplnením kapacity.

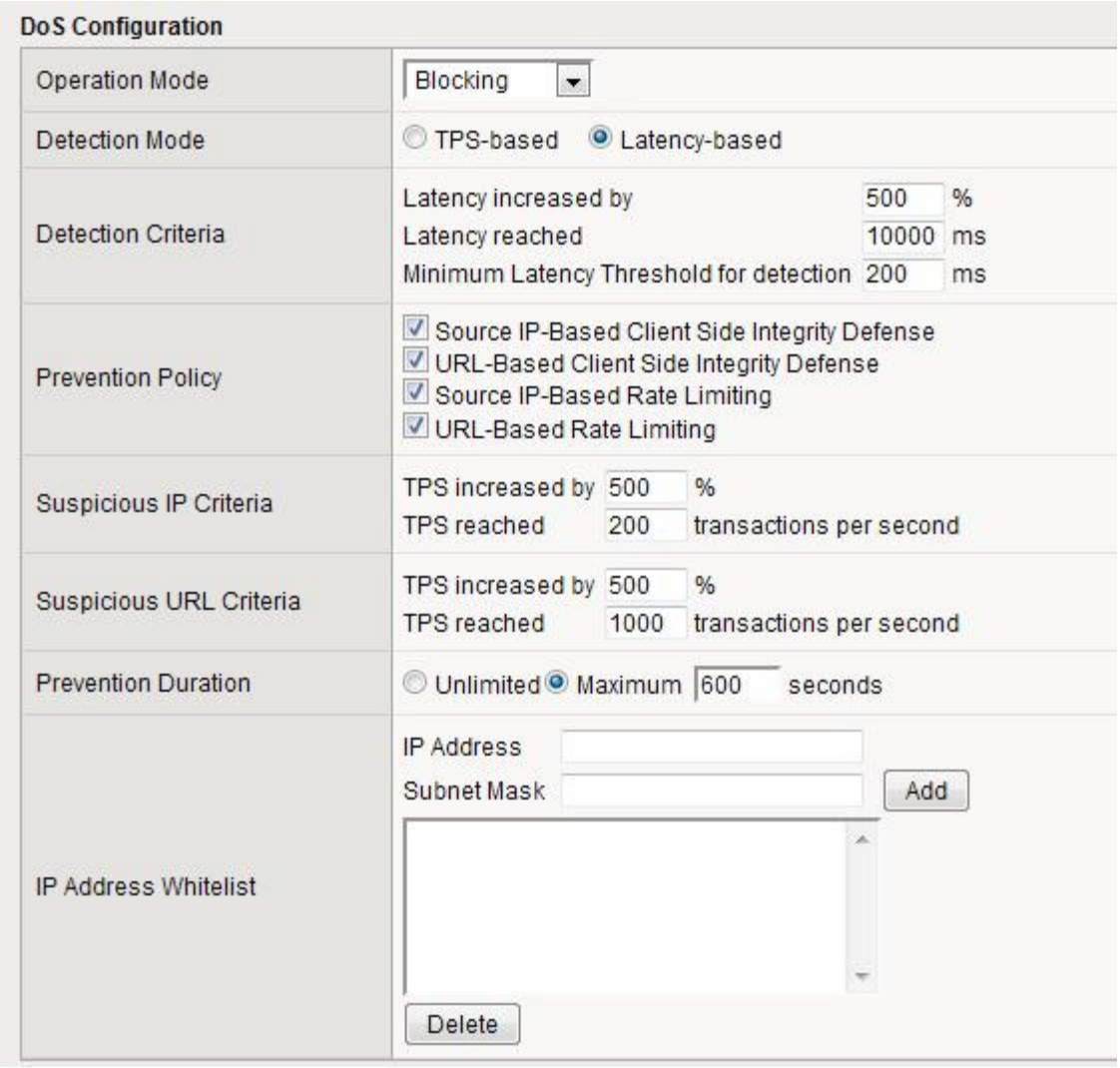
Je potrebné nastaviť rozumné limity na spojenia. Celkový limit pripojení sa určí ako

Connection Limit = Množstvo RAM v KB * 0.8

To znamená, že ak máme k dispozícii 256MB RAM, tak limit je **256,000 * 0.8 = 20480.**

Tento limit aplikujeme v hlavnej časti navigačného panelu (Main), a zmeníme nastavenia **virtuálnych serverov** (Local traffic > Virtual Servers).

Všeobecnú konfiguráciu ochrany pred DoS je možné vidieť na screenshote:



Zdroj obrázku: security-session.cz

SSL renegotiation

Obrana voči tomuto útoku je tvorená na základe pravidla iRule.

```
when RULE_INIT {
       set static::maxquery 5
       set static::mseconds 60000
}
when CLIENT_ACCEPTED {
       set ssl_hs_reqs 0
}
when CLIENTSSL_HANDSHAKE {
       incr ssl_hs_reqs
       after $static::mseconds { if {$ssl_hs_reqs > 0}
       {incr ssl_hs_reqs -1}
if { $ssl_hs_reqs > $static::maxquery } {
       after 5000
       log "Handshake attack detected, dropping
       [IP::client_addr]:[TCP::client_port]"
       drop
}
}
```

Ďalšie informácie je možné nájsť tu: http://support.f5.com/kb/en-us/products/big-ip_ltm/manuals/product/ltm_implementation/sol_dos.html#1064682

Zdroje

Security Session 2013, Brno
Konferencia

Prolexic
http://www.prolexic.com/

Cisco
http://www.cisco.com/

Radware
http://www.radware.com/

F5 Networks
http://www.f5.com/