České vysoké učení technické v Praze

Fakulta elektrotechnická

Siťová bezpečnost a firewall

Oleksandr Petronyuk

Obsah

1.1 Úvod

V této semestrální praci si kladu za cíl seznámit všechny začínajíci a pokročile siťové správce a zájemce o siťovou problematikou s zabezpečením sítě pomoci firewallů a upozornit na bezpečnostní úskalí a rizika napadení sítě.

1.2 Prvky ideálního firewallů

Ideální firewall by mněl fungovát na těchto třech principech:
Filtrování paketů
Překládání siťových adres(NAT)
Služby proxy

Filtrování paketů

Filtrování paketů je jedná z klíčových funkcí dnešních firewalů.Filtry pracují na principu porovnávaní síťových protokolu(**IP** a další) a paketu transportní vrsty(**TCP**, **UDP**) s databází pravidel a propouštějí pouze ty pakety, které vyhovují kriteriím uvedeným v databázi pravidel. Tato implementace firewallů může být implementována buď ve směrovačích nebo v implementaci **TCP/IP** protokolů na serverech.

Překlad siťových adres(NAT)

Tento překlad řeší problém skrývání interních hostitelů.NAT je v podstatě proxy na síťové vrstvě, kdy požadavky jménem všech interních hostitelů provádí jediný hostitelský počítač, takže totožnost interních hostitelů je před veřejnou sítí skryta.Funkce NAT skrývá interní IP adresy tak, že všechny adresy interních hostitelů zkonvertuje na adresu firewallu.Tento Firewall pomocí čísla portu **TCP** přepošle datovou část z interního hostitelského počítače z jeho vlastní adresy.NAT skrývá veškteré informace o interních počítačích na úrovni **TCP/IP** před Internetem.

Služby Proxy

Proxy se používa před únosem **TCP** spojení před kterým nás neochrání firewall ani **NAT** směrování.Služba proxy pracuje na aplikační úrovnim kdy umožňuje prostřednictvím firewallu úplně přerušit tok protokolů na síťové úrovní a omezit provoz pouze na protokoly vyšší úrovně jako je **HTTP**,**FTP**,**SMTP**.Tato služba proxy na aplikační úrovní je kombinaci serveru a webového klienta.Z hlediska klienta se proxy chová jako server, z hlediska cílového serveru se chová jako klient.

1.3 Co firewall nevyřeší

Na svetě existuje tolik různých způsobu napadení sítě, že úplně bezpečná není žádna metoda.Dalším problémem je ochrana proti protokolům, které se rozhodnete propouštět.Častý útok je na port 80 TCP a zneužití webovýho serveru.Protože HTTP je služba, bez které se neobejde žádna počítačová síť.Dalším nebezpečním můžou být skryté hraniční přechody.

1.4 Služby a jejích slabiny

V tomto přehledu chci uvést míry hackovatelnosti bežných internetových služeb.Tyto služby na aplikační vrstvě nejsou samy o sobě vrstvené a nemají jednotnou autentizační službu,jednotnou službu šifrování ani nejsou závislé na jiných protokolech.

Bežné internetové služby jsou

DNS(53 UDP)
FTP(20 a 21 TCP)
HTTP(80 TCP)
IMAP(143 TCP)
NTP(123 UDP)
POP(110 TCP)
SMTP(25 TCP)

DNS(Domain Name System)

Protokol DNS je pro útočníky v síti prvotřídním cílem,ale pokud ho deaktivujete, deaktivujete i všechny své klientské počítače připojené k síti.Nejčastěji hackeři na tento protokol utočí **DNS spoofingem** a **DNS cache poisoningem.**

DNS spoofing

Tento typ útoku spočívá v podvržení informací v DNS, čímž nějaké doménové jméno nasměrujeme na jinou IP adresu, popř. pozměníme jiné záznamy, třeba MX záznamy pro doručování pošty. Efekt úspěšného útoku je umocněn v případě, že se nám podaří podvrhnout informaci cachovacímu DNS serveru, který je využívám dalšími uživateli a mezi ně se podvržená informace rozšíří.
Nejjednodušší útok, používající podvržení informace v DNS komunikaci, je založen na odposlouchávání komunikace na síti a generování falešných odpovědí. Spoléhá se na to, že podvržená odpověď dorazí klientovi dříve než skutečná odpověď od dotazovaného DNS serveru.Proti tomuto útoku není odolný žádný DNS software. Jedninou možnou ochranou proti těmto útokům je kombinace šifrování a kryptografického podepisování požadavků a odpovědí, což se však zatím téměř vůbec nepoužívá.

DNS cache poisoning

Využivá nedokonalost některých implementaci DNS serverů, která spočívá v neprovádění kontrol obsahu tzv **additional section** v DNS odpovědi.Do této sekce DNS servery vkládají další informace související s odpovědí na dotaz. Například ptáme-li se na MX záznam pro doménu example.com, nachází se v odpovědi název příslušného mailserveru pro doručování pošty pro doménu example.com a v **additional section** autoritativní DNS server rovnou posílá také IP adresu tohoto mailserveru (pokud je pro takovou informaci taktéž autoritativní, což většinou bývá).
Pokud však oběť (v roli cachovacího DNS serveru, kterému chceme nějakou informaci podvrhnout) dostatečně obsah **additional section** nekontroluje a automaticky si bere její obsah a ukládá do paměti, můžou hackeři pro podvržení libovolných dat použít následující postup:

Na autoritativním DNS serveru naší domény firma.com utočník připraví podvrženou **additional section** s falešnou IP adresou cílové domény www.example.comOběti se zeptá na naši doménu www.firma.comOběť se zeptá na doménu www.utocik.com našeho autoritativního DNS serveru, do odpovědi do **additional section** se přidá A záznam s falešnou IP adresou pro doménu www.example.com Oběť si do paměti zaznamená falešnou IP adresu pro uvedenou doménu a bude ji na požádání posílat svým klientům.Pro zlepšení efektu utočník zvoli vysokou hodnotu **TTL**, aby si oběť tuto informaci pamatovala co nejdéle.

Jako možnost opatření případa v úvahu to, že servery DNS bychom měli chránit například tím, že zablokujeme pakety s přenosem zón nebo použitim služby proxy DNS.

FTP(File Transfer Protocol)

FTP je protokol s příkazovou řádkou pro přenášení souborů přes Internet.Na portu 20 dochází k vlastnímu přenosu dat.Port 21 slouží k řízení.

Nejčastějším útokem na FTP protokol je **FTP bounce-scanning ,** který využívá zneužitelné implementace FTP protokolu.

Útok na dva FTP servery Další možností útoku je nechat dva FTP servery si vyměňovat data. Toho se dá docílit kombinací pasivního a aktivního režimu. Tento útok může sloužit jako efektivní **DoS** útok, jako šikovný trik, jak kopírovat soubory mezi FTP servery bez toho, aby to pocítila vaše linka (tj. aniž byste byl prostředník).Celý útok spočívá v tom, že na jednom serveru aktivujete pasivní režim. On vám dá IP adresu a port, na kterém bude čekat spojení. Na druhém serveru aktivujete aktivní režim a IP adresu s portem zvolíte podle toho, co vám dal první. Nyní na prvním zvolíte příkazem **stor** a jménem souboru, že si přejete uložit data. Na druhém serveru příkazem **retr** a uvedením souboru zahájíte přenos dat. Nyní se druhý server připojí k prvnímu a začne mu posílat data. To je celý princip. Výhodou tohoto útoku je také to, že stačí, aby byl k němu náchylný jenom jeden server, jelikož pasivní režim není nijak omezen.

Útok na lokální síť pomocí FTP

Díky aktivnímu režimu můžeme posílat data i dovnitř takové sítě, která není z venku přístupná. Stačí jen mít přístup k **FTP** serveru uvnitř sítě a možnost ukládat na něj soubory. Trik spočívá v nahrání souboru, který obsahuje data, která chceme určitému portu (službě) předat. Pak už jen stačí nastavit aktivní režim na daný počítač a vybrat port. Vše odstartujeme tím, že zvolíme stáhnutí souboru, který jsme na tento **FTP** server nahráli. Server se totiž připojí k danému počítači na zadaném portu a veškerá tato data mu pošle.

Vetšínu výše popisovaných utoku se provádělo na starších verzich ftp v dnešní době je aktivní režim ošetřen a nedá se již zneužit.Je naivni si myslet , že s těmito útoky se jíž nesetkáme. Na Internetu je ještě spousta serverů, které jsou k této chybě náchylné a jsou připojeny linkami o slušných kapacitách.Pokud děláte začínajicího správce v menší firmě je dobrý o těcho technikách útoku vědět.

Při nastavování **FTP** zabezpečení bychom měli znát, že **FTP** ustavuje s klientem komunikaci, při níž si může klient prohlížet soubory na FTP serveru a ztahovát tyto soubory.Autentizace **FTP** se provádí pomocí uživatelských jmen a hesel operačního systému.Když uživatel získá přístup k serveru **FTP** jako anonymní uživatel může se dostat k životně důležitým souborům operačního systému, pokud nemáte vyhovujíci zabezpečení souborů a adresářů.

Když povolíte na server **FTP** přístup z umístění mino siť, nepoužívejte na serveru **FTP** stejná jména účtů a hesla, která používáte pro přihlašování k LAN.

HTTP(Hypertext Transfer Protocol)

Velikým problémem toho protokolu je, že je to standartní služba, která je na Firewallů povolená a je oblíbená a často zneužívaná služba hackerami.

Prostřednictvím **HTTP** protokolu se přenáší text, video , zvuk a dokonce programy.V současné době jsou webové servery složité programy s mnoha možnostmi konfigurace.Hackeři mohou pomocí **HTTP** zneužít webové stránky a to obsahem nebezpečných prvků jako jsou ovládací prvky **ActiveX** nebo **aplety** jazyka **Java**. Rešením toho problému je buď uplně blokovat přes Firewall **ActiveX** a **aplety Javy** nebo požadovat, aby se mohli stahovat pouze prvky **ActiveX** digitálně podepsané organizacemi, kterým důvěřujete. Pro Javu existuje lepší zabezpečení a to, že zkontrolujeme a nastavíme všechny počitače v siťi tak, aby apletům jazyka Java nepovolovaly přístup k hardwarovým zdrojům.

Pro lepší bezpečnost sitě spravujte uživatelská jména a hesla webových stránek odděleně od uživatelských jmen a hesel operačního systému. Dále mužete zaznamenávat přístup k síti a vyhledávat neobvyklé sekvence(např příliš mnoho chyb 404).Nejčastější a nejobliběnější zneužití HTTP v dnešní době je XSS útok.

XSS útok

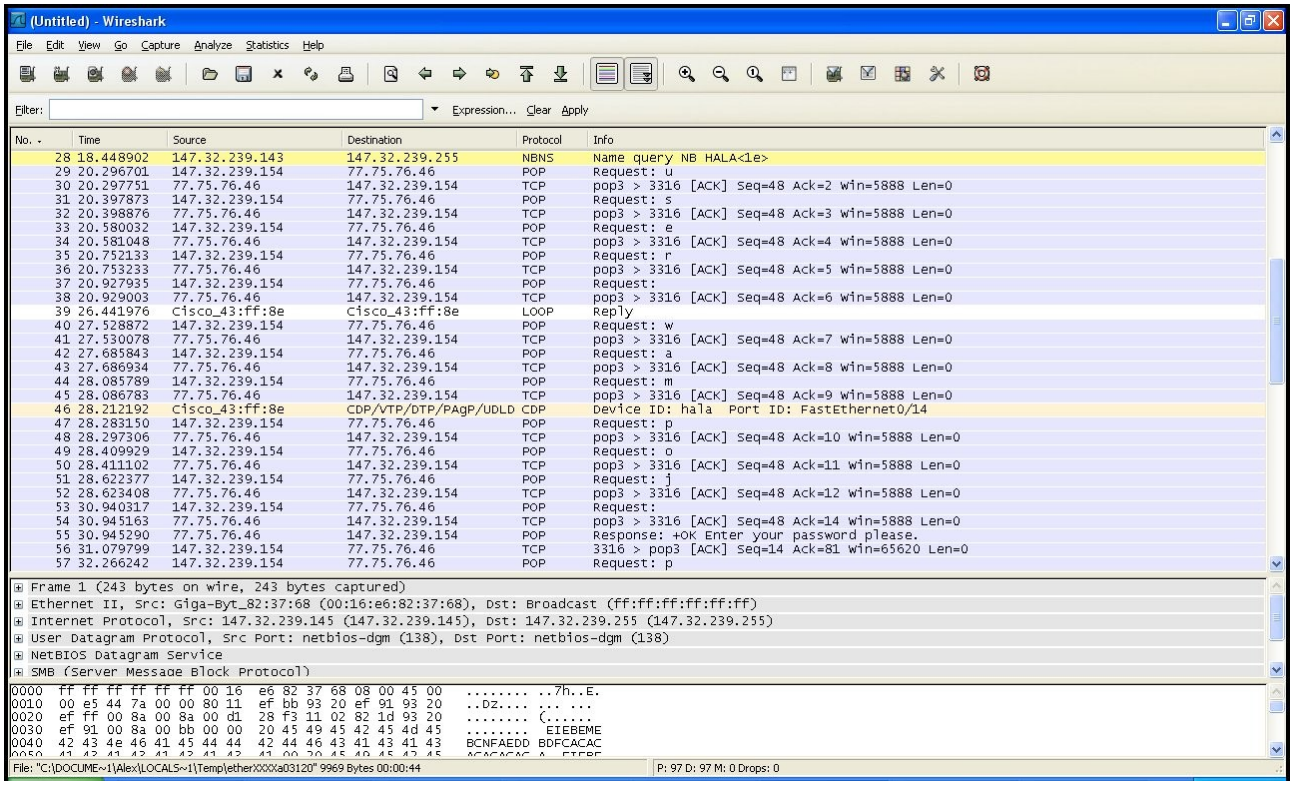
Neboli **Cross Site Scripting** je jeden ze způsobů jak lze hacknout webové stránky.Tento útok nastává, když webová aplikace shromažďuje neošetřená data od uživatelů.Útočník obykle skript zakóduje pomoci hexadecimálních znaků, takže se data stávají méně podezřelá. A tento kód vloží na populární fóra.Muže se stát, že se oběť příhlási na toto fórum, který obsahuje závadný script, který může získát **cookies** uživatele a tím i přístupové heslo. Proti tomuto typu útoku nás žádný Firewall neochrání.Zde je namístě školení zaměstnanců a pokud tyto webové stránky vyvíjimé my například v PHP tak lze ošetřít výstupy předávané metodou **get** či **post** pomoci funkce htmlspecialchars('výstup', ENT_QUOTES).

NTP(Network Time Protocol)

Timto protokolem probíha synchronizace vnitřních hodin počítačů po paketové síti.Tento protokol zajišťuje, aby všechny počítače v síti měly stejný a přesný čas.Je to jeden z nejstarších dosud používaných **TCP/IP** protokolů.Ukázalo se, že mnoho implementaci **NTP** je zranitelných vůči útoku prostřednictvním přetečení vyrovnávací paměti.Jako spravce sitě je vhodný používat nenovější verzi toho protokolů.

POP(Post Office Protocol)

Pomocí poštovního protokolu si klientské počítače kontrolují elektronickou poštu.Velkou nevýhodu **POP** protokolu je, že nešifruje uživatelská jména ani hesla takže je zranitelný proti Sniffingu kdy hacker může odposlechnout citlivé údaje.Doporučením je používát protokol Imap.



Obr 1: *Ukázka odchytnuti citlivých údaju programem Wireshark*

IMAP(Internet Message Access Protocol)

V dnešní době tento protokol je nejčastějí používaným protokolem pro stahování pošty ze serverů. Je bezpečnější než **POP** kdy hesla se neposílají v nezašifrované podoběTento protokol není tak zneužitelný jako předchozí protokoly ale přesto na tento protokol může Hacker utočít slovnikovým útokem za cílem získat přihlašovácí údaje.Tomu to typu utoku zamezime složitějším autentizačním schématem.

SMTP(Simple Mail Transfer Protocol)

Tento protocol akceptuje poštu prostřednictvím jednoduché komunikace ale nekontrojule pověření nebo dokonce totožnost odesilatele.I když je protocol sám o sobě jednoduchý, software , který poštu zpracovává často jednoduchý není.A v tom je kámen urazu.Problémem je, že software zpracovávajicí poštu ma hodně možností konfigurace, kde se muže z hlediska administrace snadno udělat chyba.Dalé chybna konfigurace může ovlivňovat výkon sítě nebo dokonce způsobit selhání poštovního serveru.V tomto protokolů neni autorizace odesilatele, takže je dále náchylný vůči spamu.Řešením je kvalitní software a dobrá konfiurace.

1.5 Profily utoku

Ping of Death

Tento útok je předchůdcem všech útoků typu odepření služby.Tento typ útoků zneužíva toho, že implementace protokolu TCP/IP se často spoléhají na správný format paketů ICMP a neprovadějí chybovou kontrolu.Útok začína vytvořením chybně formátovaného paketu zprávy typu Echo u protokolu ICMP, ve kterém velikost paketu přesahuje maximální možnou velikost.Výsledkem je ,že pakety ICMP které udavají svou velikost větší než 65 500 bajtů způsobi chyby protokolu TCP /IP kdy dojde k havárii implementace TCP/IP kvůli chybě alokace paměti.
V dnešní době většina Firewallů tento útok implicitně kontroluje .Pokud tak tomu není tak lze v konfiguraci firewallů blokovat ICMP protokol.

Teardrop

Tento typ útoku využíva potenciální slabiny v procesu zpětného seskupení fragmentů kdy napadá implementace TCP/IP, které spolehají na informace v hlavičkách paketů fragmentů IP.Některé implementace TCP/IP havarují, pokud dostanou falšované fragmenty, který obsahují překrývající offsety. Obranou proti tomuto útoku je aktualizace operačního systému.Dobrou zprávou je, že většína implementací protokolu TCP/IP ve firewalech by mněla být proti tomuto útoku odolná.

UDP floods

Tento útok je velice jednoduchý pro hackera kde hacker zfalšuje připojení UDP ke službě **Chargen** spuštěné v hostitelském počitači, které má jako zpáteční adresu hostitele uvedenou adresu počítače se spuštěnou službou **Echo**.Hacker vytvoří nesmyslný proud dat procházejících mezi oběma hostiteli kdy dojde k vyčerpání šířky pásma a odepření služby.Abychom zabránili tomuto útoku, tak je vhodné nakonfigurovát směrovače tak, aby blokovaly pořadavky UDP na služby **Chargen** a **Echo** příchazející z Internetu.Lepší obranou proti tomuto útoku je ,že nebudeme spouštět nepotřebné služby, které tyto útoky zneužívají.

SYN floods

Tento útok využíva mechanizmy navazování spojení protokolem TCP.Tento útok používa falešnou zprávu SYN odesílané na servery.Výsledkem toho útoku je, zahlceni serveru SYN požadavky kdy dojde k přetečení vyrovnávací paměti a odepření služby.Dobrou obranou proti tomu útoku je stavový Firewall, který dokáže kontrolovat stav spojení.A jen SYN spojení zahazovát.

Smurf

Tento typ útoku je mimořádně účinným útokem odepření služby, který je založen na funkci příme broadcast adresace protokolu IP kdy tato funkce umožňuje hostiteli vysílat data všem hostitelům v jeho podsítích.Hacker zaplaví hostitele pakety zpráv **Echo** protokolu ICMP, které mají zpáteční adresu nastavenu na broadcast adresu cílové sétě.Všichni hostitelé na zprávy Echo protokolu ICMP odpoví a tím ještě zvýší provoz kdy dojde k zahlcení přenosového pásma.
Učinnou obranou proti **Smurf** útoku je vypnutí funkci broadcast adresace svého externího směrovače nebo nastavit firewall tak, aby odstraňoval zprávy ping protokolu ICMP.

Fraggle

Tento útok je jednoduchou úpravou útoku Smurf , kdy používá zprávy typu Echo protokolu UDP místo ICMP protokolu.Díky tomu může útok překonat firewally, které filtruj pouze ICMP protokol. Obranou je na firewallů aktivovat filtrování zpráv typu Echo protokolu UDP.

Chybně formatované zprávy

Mnohé služby v různých operačních systémech havarují po příjmu chybně formátovaných zpráv, protože neprovádějí dostatečnou kontrolu chyb ve zprávách před jejich zpracováním. Všechny operační systémy mají slabá místa, která lze zneužít různými chybně formátovanými zprávami. Například přetečením vyrovnávací paměti e-mailů nebo výřazením RPC služeb.Jako správní administrátoři bychom mněli sledovat nejnovější informace o slabých místech daného operačního systému a používaného software.

1.6 Detekce pruníku do sitě

Podstata detekce pruníku vychází z předpokladu, že úkony narušitele budou na základě přímých či nepřímých indicií odlišitelné od běžné činnosti uživatelů.Toto odlišení lze provést různými metodami.Jako zákadní prostředek dekekce vniknuti pro administrátora sitě je zavést záznamy o chodu systému a sledování provozu v síti.

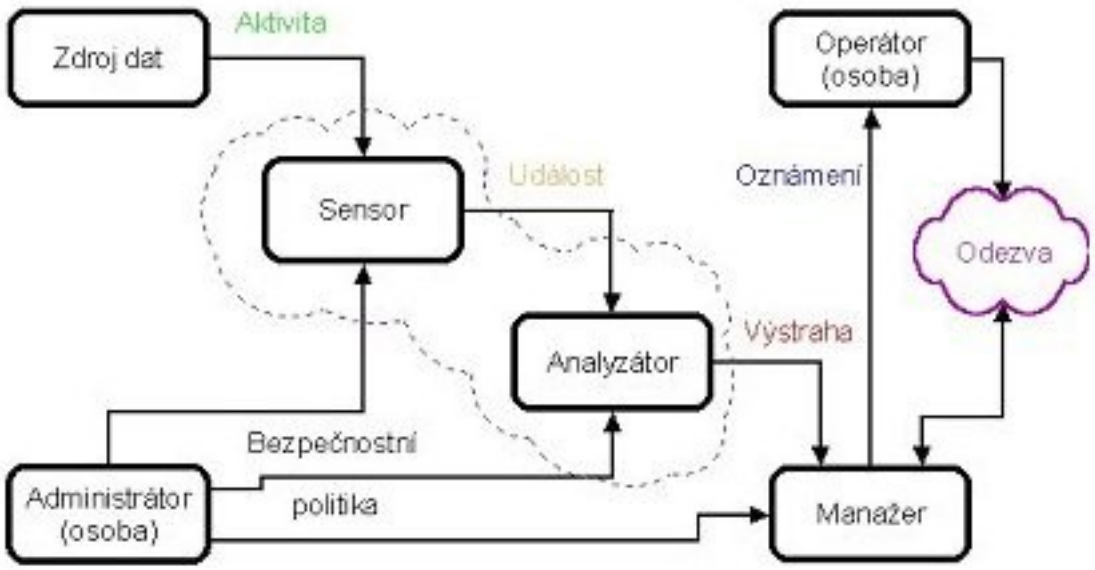
Obecná struktura detekce průniku
Tuto strukturu navrhla **Dorothy Denningová** v létech 1986 a 1987 kdy tvrdí, že je užitečné pracovat s detekčním modelem, jehož hlavní části jsou

-generátor událostí

-analytický modul

-ukládací mechanismus

-modul protiopatření



Obr 2: *Tento obrázek znázorňuje základní komponenty systému pro detekci průniku*

Tato stuktura systému detekce nezáleží na implementaci.Jednotlivé moduly mohou být implementovány samostatně nebo mohou být součastmi složitějších celku.

Generátor událostí

Poskytuje informaci o vzniku dané události v systému.Událostí se rozumí jak interakce mezi aplikačními programy, tak aktivita na nižších vrstvách síťové architektury. Není zapotřebí, aby se jednalo o útok. Generátor událostí představuje čidlo systému detekce průniku. Bez něj by neměl žádné informace, ze kterých by mohl učinit závěry.

Analytický modul

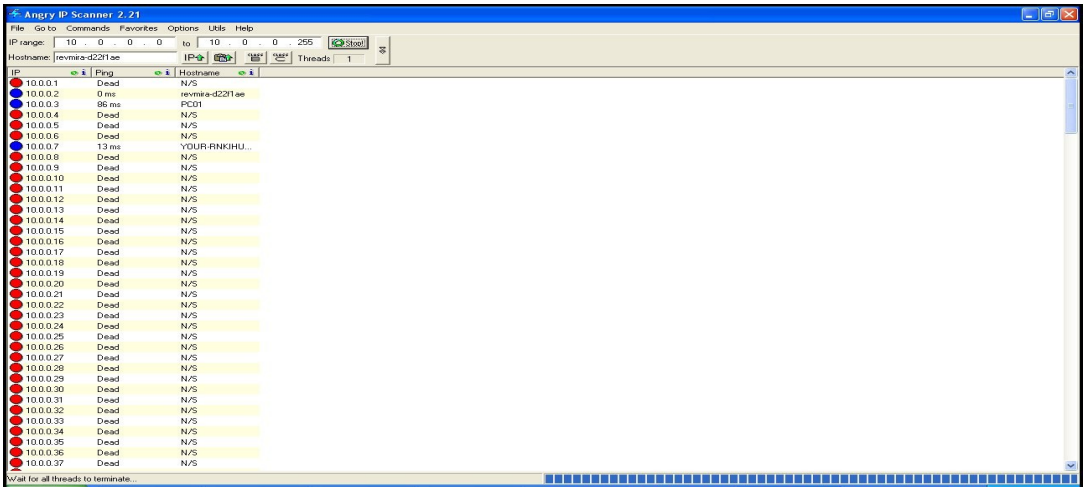
Zpracovává udaje přicházející od generátoru událostí.A snaži se je analýzovat.Tento modul uzce spolupracuje s **modulem protiopatření** kdy pokud je modul protiopatření kvalitně impementovan, tak muže bez administrátora učinit bezpečnostní kroky k chodu sitě a várovat administrátora.

Ukládací mechanismus zaznamenáva informace o provozu síti.Kdy tuto informace, mohou být kdykoliv dostupné a sloužici k anlýze problému.

1.7 Typický postup hackera pří vniknutí do sitě

1.Skenování ip adres

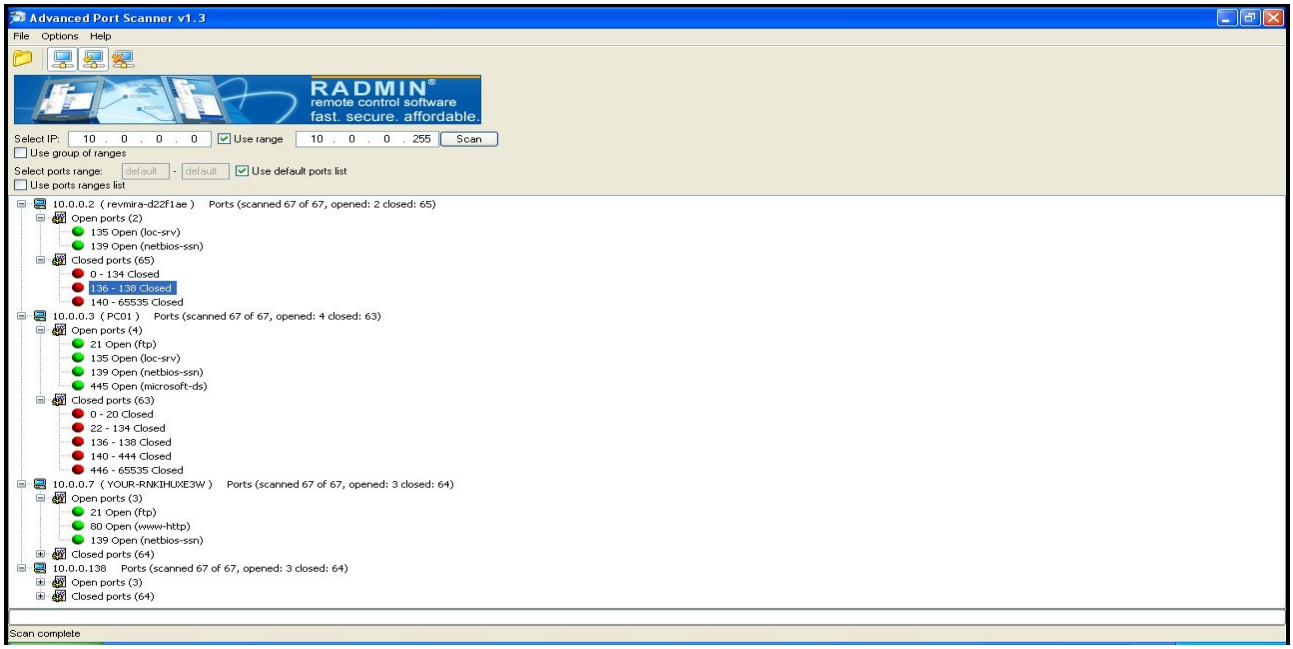
Toto skenování v rámci rozsahu síťových adres.Hackeři obvykle skenují v celém rozsahu adres IP v okolí hostitele a mohou pomocí reverzního vyhledávání názvů DNS zjišťovat, zda jsou další hostitelé registrováni pro vaši společnost.



Obr 3: *skenování adres pomoci Angry Ip Scanner*

Na *obr 3* mužeme vidět tři hosty, na které utočník může útočit.Dále hacker postupuje scannováním portu.

2.Skénovaní portu



Obr 4: *Scanování portů pomoci programu Advanced Port Scanner*

3. Průzkum služeb
Umožňuje zjistit operační systém jednotlivých hostitelů.Hacker pomoci scanovani portů zjisti jaký operační systém hostitlel pouziva podle bežicích služeb. Například bežicí služby jako **NetBios** a **RPC Locator** charakterizují operační systém Windows.Služby **SSH** nebo **TFTP** charakterizují Unix. Protože většina aplikací je svázána s určitým operačním systémem, je určení operačního systému velice snadné.

4. Výběr cíle a průzkumy slabých mist
Hackeři se zaměřují na hostitele, který má spuštěno nejvice služeb, protože předpokladají, že výchozí konfigurace hostitele nebyla téměř nebo vůbec zabezpečena.Velice oblíbenou službou pro útok je **NetBios** , protože zneužití této služby může zajistit úplné řízení příslušného počítače.Dalším popularním útokem jsou služby pro vzdálené řízení jako je **Terminal Services**, **VNC** nebo **pcAnywhere**

5. Automatizované útoky na hesla
Nejvic se tyto útoky používají proti službám jako je FTP, HTTP, NetBios , VNC nebo k jiným službám, které umožňují přístup k systému souborů, nebo vzdálené konzole.Používají jednoduchý slovnikový útok.Pokud hacker získá přístup do počítače pomocí konzoly, muže v hostiteli spustit vysokorychlostní nástroj pro automatické zjištění hesla(Crack nebo NT Crack), aby mohl zneužit další účty.

6. Útoky na specifické služby
Tyto útoky zahrnují neobvyklé, zvláštní nebo obtížné útoky, ke kerým se hackeři mohou uchýlit. K těmto útokům patří útoky typu přetečení vyrovnávací paměti, útoky využívající přímé směrování, pokusy o zcizení spojení, siťový odposlech hesel nebo klamavé e-maily určené k instalaci trojských koní.Dalším problémem může být pokud spávujete vzdáleně firewall.Pokud hacker získa heslo pro správu toho firewallů tak muže napadnout překonfiguraci firewallů vaší síť.

1.8 Jak ochránít středně malý podník

Stanovení nároku na zabezpečení
U středně malých podníku není třeba při zabezpečování sitě jit do extrému.Analýza nákladu a výnosu prozradí, že postačí méně přisné zabezpečení.Většinou tyto firmy nemají tolik důležitých informaci, které by zajímali náhodného hackera nebo konkurenci.

Nejčastější hrozby
Mezi nejčastější hrozby těcho síti patří počitačové víry a červy a spam.

Možn řešení
Centralní ochrana počítačové sítě
Každou firmu by měl chránit siťový firewall.Tento firewall by mněl být umístěn na vstupu do firemní sítě.Kdy oděluje firemní PC od internetu.Hlavní a základní obranou sitě je centrální ochrana počítačové síťě pomoci firewallu.Základem je kvalitní síťový firewall pomoci kterého mužeme chránit síť jako celek, včetně ochrány klíčových serverů, na kterých běží aplikace mail serveru, file serveru , databázových serverů a dalších.

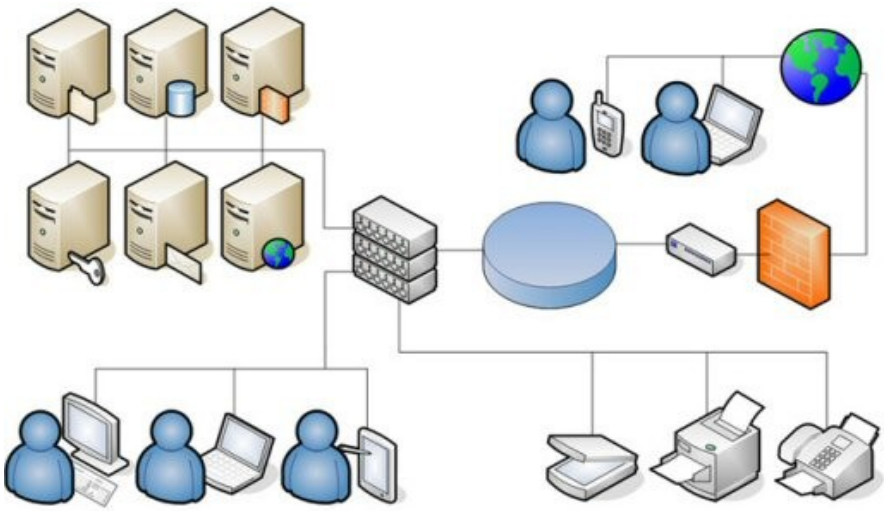
Síťové firewally kontrolují, povolují nebo zamítají jednotlivé pokusy o přístupy z nebo do podnikové sítě. Firewall na základě stanovených pravidel propouští pouze některé pakety síťové komunikace a jiné odmítá, nebo rovnou odstraňuje. Například síťový firewall může automaticky povolit přijetí určitých dat jen tehdy, pokud si tyto data předtím některý z PC v síti vyžádal.

Nejlacinějším řešením v segmentu malých a středních firem bývají často malé hardwarové firewally, které v sobě integrují jednoduchý firewall a router a mohou být doplněny o další funkce, jako je vestavěný hub nebo switch, případně umožňují připojení přenosných počítačů pomocí bezdrátové technologie WiFi.

Ochrana osobního počítače
Tato ochrana je proti hrozbě z lokální podníkové sitě.Nebezpečí, která můžeme očekávat, jsou v drtivém procentu nechtěná a zaviněná uživateli.Typickým příkladem hrozby je napadení viru z vnitřku sítě z nezabezpečeného PC nebo notebooku.Dalším problémem jsou přenosová media jako cd nebo flash disk, která mohou obsahovat viry a před kterým nás firewall neochrání.Možným řešením je personální firewall.Tento firewall nabízi kontrolu výměny informací výměny informací mezi počítači v rámci lokální sítě dale tento firewall může blokovát vyskakující pop-up okna, filtrovát reklamní bannery, zákazat cookies a spyware, které umožňují sledovat jaké stránky uživatel navštěvuje.Toto zabežpečení doplněto o kvalitní antivirový prográm vytváří solidné zabezpečení stanic.

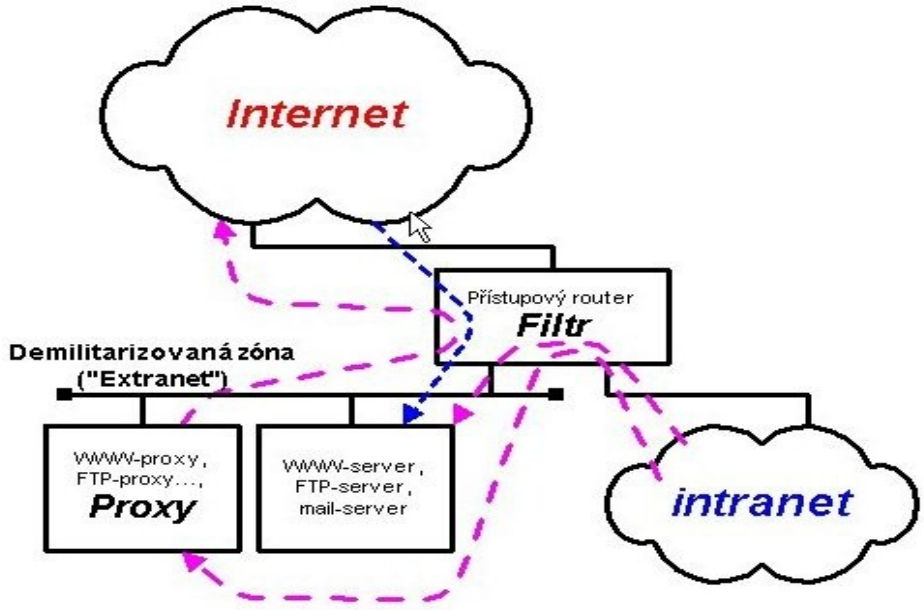
Ochrana serverových stanic
Tuto ochránu lze zajistit specialním firewallem pro serverové stanice.Tento firewall chrání před útoky zevnitř sítě a brání zneužití bezpečtnostích chyb v operačním systému a aplikacích, pro které dosud neexistuji záplaty.Ale tato ochrana je u středně malých firem zbytečna, protože představuje dálši finanční náklady na zabezpečení.

Siťová architektura středně malých podniku
Obr 5 ukazuje optimální siťovou architekturu středně malých podniku.Kdy jsou na třech portech switchem od sebe odděleny klienti, tiskárny a serverové stanice.Dále tato architektura umožnuje vytvořit delimitarizovanou zonu, kdy mužeme všechen provoz směrovat přes server, kdy pokud utočník se dostané na server podniku tak není schopen se dostát ke klientům.



Obr 5: *Typická siťová architektura malých podniků*

Obr 6 ukazuje delimitarizovánou zónu.Je to další z oblíbených řešení siťové bezpečnosti.



Obr 6: *Demilitarizovaná zóna*

1.9 Malá ukázka konfigurace Linuxovýho firewall pomoci iptables

Zde uvádím tuto konfiguraci Linuxovýho firewallu iptables jako málou ilustraci. Tuto konfiguraci jsem přebral ze stránek http://jk.myserver.cz/hack/iptables/ Josefa Kufnera.

```
#!/bin/bash
#------------------------------------------------------------------------#
#
#       iptables initialization
#       by Josef Kufner <jk(a)myserver.cz>
#
#
#       special thanks to:
#               http://www.petricek.cz/mpfw/mpfw.sh.txt
#
#------------------------------------------------------------------------#

## config
IPTABLES=/sbin/iptables
LO_IFACE=lo
INET_IFACE=eth0
LAN_IFACE=eth1

## vyber rozhrani na ktere to bude pustene
## (v pripade ze je spousteno z /etc/network/if-pre-up.d/ -- Debian)
[ "$IFACE" != "" ] && [ "$IFACE" != "$INET_IFACE" ] && exit
```

```
#------------------------------------------------------------------------#
#               Moduly & inicializace
#------------------------------------------------------------------------#

echo
echo -n "Loading iptables settings"

## Zavedeme moduly pro nestandardni cile
/sbin/modprobe ipt_REJECT
/sbin/modprobe ipt_MASQUERADE

## Moduly pro FTP prenosy
/sbin/modprobe ip_conntrack_ftp
/sbin/modprobe ip_nat_ftp

## clear all
$IPTABLES -F -t mangle
$IPTABLES -F -t nat
$IPTABLES -F
$IPTABLES -Z
$IPTABLES -X

echo -n "."

#------------------------------------------------------------------------#
#               Default
#------------------------------------------------------------------------#

$IPTABLES -P INPUT      DROP
$IPTABLES -P OUTPUT     ACCEPT
$IPTABLES -P FORWARD    DROP

echo -n "."

#------------------------------------------------------------------------#
#               retezce
#------------------------------------------------------------------------#

## Retezec pro stanoveni limitu prichozich SYN konexi (ochrana pred SYN floods)
## propusti pouze 4 SYN segmenty/sec
$IPTABLES -N syn-flood
$IPTABLES -A syn-flood -m limit --limit 20/s --limit-burst 5 -j RETURN
$IPTABLES -A syn-flood -j DROP

echo -n "."

#------------------------------------------------------------------------#
#               INPUT
#------------------------------------------------------------------------#

## Pakety od navazanych spojeni jsou v poradku
$IPTABLES -A INPUT -i $INET_IFACE -m state --state ESTABLISHED,RELATED -j ACCEPT

## loopback bez omezeni
$IPTABLES -A INPUT -i $LO_IFACE -j ACCEPT

## server sit bez omezeni
$IPTABLES -A INPUT -i $SERVER_IFACE -j ACCEPT
```

```
## lan bez omezeni -- jedine neco jako transparentni proxy, ale to az casem
$IPTABLES -A INPUT -i $LAN_IFACE -j ACCEPT

## ping -- max 5 za sec
$IPTABLES -A INPUT -i $INET_IFACE -p icmp --icmp-type 'echo-request' \
        -m limit --limit 1/s --limit-burst 10 -j ACCEPT

## Odfiltrovat pokusy o syn-flooding
$IPTABLES -A INPUT -i $INET_IFACE -p tcp --syn -j syn-flood

## Odfiltrovat pokusy o zahlceni icmp
$IPTABLES -A INPUT -i $INET_IFACE -p icmp -j syn-flood

## auth nerado DROP
$IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport auth -j REJECT --reject-with tcp-reset

## povolene sluzby
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 21   -j ACCEPT      #  FTP server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 22   -j ACCEPT      #  SSH server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 25   -j ACCEPT      #  SMTP server
# $IPTABLES -A INPUT -i $INET_IFACE -p UDP --dport 53   -j ACCEPT      #  DNS server UDP
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 53   -j ACCEPT      #  DNS server TCP
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 80   -j ACCEPT      #  WWW server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 110  -j ACCEPT      #  POP3 server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 143  -j ACCEPT      #  IMAP server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 443  -j ACCEPT      #  HTTPS server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 873  -j ACCEPT      #  rsync server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 995  -j ACCEPT      #  POP3s server
# $IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 10000 -j ACCEPT     #  webmin server

## DC++
# $IPTABLES -A INPUT -p tcp --dport 9176 -j ACCEPT
# $IPTABLES -A INPUT -p udp --dport 9176 -j ACCEPT

## otevreni firewallu pro LAN
$IPTABLES -A INPUT   -i $LAN_IFACE -j ACCEPT

echo -n "."

#------------------------------------------------------------------------------#
#               OUTPUT
#------------------------------------------------------------------------------#

## vse je defaultne povoleno

## TOS flagy slouzi k optimalizaci datovych cest. Pro ssh, ftp a telnet
## pozadujeme minimalni zpozdeni. Pro ftp-data zase maximalni propostnost
$IPTABLES -t mangle -A OUTPUT -o $INET_IFACE -p tcp     \
       --sport ssh -j TOS --set-tos Minimize-Delay
$IPTABLES -t mangle -A OUTPUT -o $INET_IFACE -p tcp     \
       --dport ssh     -j TOS --set-tos Minimize-Delay
$IPTABLES -t mangle -A OUTPUT -o $INET_IFACE -p tcp     \
       --sport ftp     -j TOS --set-tos Minimize-Delay
$IPTABLES -t mangle -A OUTPUT -o $INET_IFACE -p tcp     \
       --dport ftp     -j TOS --set-tos Minimize-Delay
$IPTABLES -t mangle -A OUTPUT -o $INET_IFACE -p tcp     \
       --dport telnet -j TOS --set-tos Minimize-Delay
$IPTABLES -t mangle -A OUTPUT -o $INET_IFACE -p tcp     \
       --sport ftp-data -j TOS --set-tos Maximize-Throughput
```

```
echo -n "."

#-----------------------------------------------------------------------#
#               FORWARD
#-----------------------------------------------------------------------#

# MTU je na ppp0 a eth0 ruzny:
#    This is called MSS-clamping and influences the amount
#    of data per TCP packet.
$IPTABLES -I FORWARD -p tcp --tcp-flags SYN,RST SYN \
                -j TCPMSS --clamp-mss-to-pmtu

## NAT - maskarada
echo "1" > /proc/sys/net/ipv4/ip_forward
$IPTABLES -t nat -A POSTROUTING -o $INET_IFACE -j MASQUERADE

## jedna zlobiva IP
$IPTABLES -A FORWARD -i $LAN_IFACE -s 192.168.1.99 -j REJECT

## povoleni provozu
$IPTABLES -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPTABLES -A FORWARD -i $LAN_IFACE -p all -m state --state NEW -j ACCEPT


echo -n "."

#-----------------------------------------------------------------------#
#               Konec
#-----------------------------------------------------------------------#

echo done.
exit
#-----------------------------------------------------------------------#
```

1.10 Závěr

Jak útoky hackerů, tak i viry, červi a spam jsou velkým problémem dnešního internetu. Žádnou síť, která je připojena k internetu, bohužel nelze zabezpečit stoprocentně. Avšak vhodným výběrem řešení a kombinací různých druhů ochrany můžeme tato rizika minimalizovat.

1.11 Seznam použité literatury

Matthew Strebe, Charles Perkins **Firewally a proxy-servery Praktický průvodce**

http://www.pweb.cz
http://jirivanek.eu/hackersky-utok-na-ftp
http://cs.wikipedia.org/wiki/File_Transfer_Protocol
http://www.systemonline.cz/clanky/bezpecnost-it-v-prostredi-malych-firem.htm
http://icons.cz/48-xss-utoky-a-zabezpeceni-stranek.html
http://cs.wikipedia.org/wiki/Network_Time_Protocol
http://www.lupa.cz/clanky/systemy-detekce-pruniku/
http://jk.myserver.cz/hack/iptables/
http://www.cpress.cz/knihy/tcp-ip-bezp/Cd-II/CD-uvod/um.htm