Proxy firewall

Úvod

Počítačová bezpečnost patří k ožehavým tématům dneška. V době stále se rozšiřujícího Internetu číhá nebezpečí na každém rohu. Profesionálové o něm dobře vědí a málokdo z nich bere tento problém na lehkou váhu, avšak spousta běžných uživatelů není dostatečně informována, a ti, kteří jsou, si myslí, že se jich nebezpečí napadení cizím útočníkem netýká. To je jeden z největších omylů v této oblasti. Všichni máme na pevném disku informace, které bychom neradi zveřejnili nebo ukázali někomu cizímu. Na jedné straně se oháníme zákony na ochranu osobnosti, na straně druhé necháváme často až příliš nápadně „otevřené dveře“ do našeho systému. Problém počítačové kriminality je čím dál hrozivější. Počítače dnes využívá kdekdo, staly se součástí našich životů a svou práci si bez nich nedokážeme představit.

Největší hrozbou je jistě připojení k počítačové síti. A platí přímá úměra mezi její velikostí a nebezpečím, které se v ní ukrývá. Z toho plyne, že největší síť znamená také největší riziko. Touto je bezesporu celosvětová síť Internet. Díky své rozlehlosti, anonymitě jednotlivých uživatelů, kterých jsou miliony a nedůvěryhodné povaze znamená pro naše data a soukromé informace obrovské riziko, jemuž je potřeba rázně čelit. Vznikly tedy síťové programy a zařízení nazývané firewally, jejichž úlohou je oddělit sítě s různou úrovní důvěryhodnosti a kontrolovat datový tok mezi těmito sítěmi. Kontrola dat probíhá na základě pravidel, která určují podmínky a akce. Podmínky se stanovují pro údaje, které je možné získat z datového toku. Úloha firewallu spočívá ve vyhodnocení podmínky a provedení akce, pokud je podmínka splněna. Základní akcí je povolit či blokovat datový paket. Existují i takové akce, které slouží pouze na zaznamenání nebo změnu hlaviček paketu. Pro snadnější integraci do sítě dnešní firewally podporují i směrování.

Mezi základní způsoby ochrany sítě firewallem patří filtrování paketů na základě zdrojové či cílové IP adresy nebo podle TCP/UDP portů, dále stavová inspekce, která umí přiřadit pakety k příslušnému spojení (díky této vlastnosti firewall rozpozná, že se jedná o pakety vracející se do sítě v rámci spojení, které bylo navázáno zevnitř) a proxy firewall/aplikační brána pracující na aplikační úrovni a vyhodnocující průběh komunikace na této nejvyšší vrstvě. Proxy jednoduše řečeno ukončují spojení směrem od klienta a směrem k serveru navazují spojení nové. Míra bezpečnosti je proto pro konkrétní aplikace velmi vysoká. Spolu s firewally, chránícími síť jako celek, jsou často zmiňovány i tzv. proxy servery, umožňující oddělit klienty (hostitelské stanice) od přímého styku s počítači (resp. servery) „na druhé straně“ informačního řetězce a poskytnout tak anonymitu a bezpečnost uživateli.

Tato práce se bude z velké části zaobírat především tematikou principu činnosti proxy firewallů, ovšem budou zde zmíněny i ostatní typy firewallů, spolu se vzájemným porovnáním jejich vlastností, výhod a nevýhod. Další dvě velké kapitoly budou tvořit překlady síťových adres (NAT) a proxy servery, kterým bude poskytnuto více prostoru. Praktickou část práce bude představovat návrh řešení proxy firewallu postavené na platformě GNU/Linux. Této části spolu s popisem principu činnosti proxy firewallů bude věnována největší pozornost. V samotné příloze na závěr přiložím komentované konfigurační soubory a skripty, které budou na linuxovém serveru otestovány a funkčně využity.

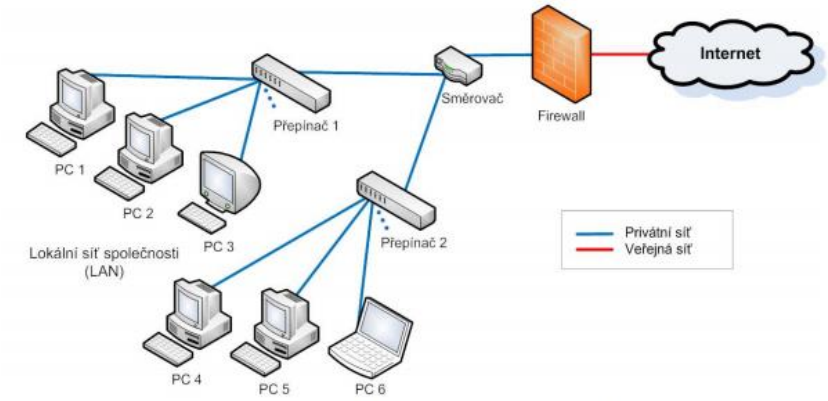
Nezanedbatelná část internetových serverů dnes běží na strojích, které jsou spravovány některým z operačních systémů patřících do rodiny Unix. Proto veškeré, později jmenované programy či softwarové aplikace, budou přizpůsobeny operačním systémům založeným na svobodném jádru Linux a spadající pod licenci GNU General Public License.

Firewally

Technologie firewallů

Firewall (česky by se dalo říct „bezpečnostní brána“ či „ochranný počítač“, pojem se však nepřekládá) je jednoduše řečeno brána oddělující sítě. Může být realizován jako program spuštěný na počítači nebo jako samostatné zařízení. Primárním účelem firewallu je zabránit nechtěné komunikaci z jedné sítě (zóny) do jiné sítě (zóny) a jeden firewall může oddělovat i více než dvě různé sítě (zóny). Rozlišení, která komunikace bude povolena či zakázána, je řízeno bezpečnostní politikou. Tato politika je vložena do konfigurace firewallu a pro každý požadavek na průchod firewallem jsou

aplikována pravidla bezpečnostní politiky, podle nichž firewall rozhodne, zda komunikaci povolit či zakázat. Tato činnost je často nazývána filtrování komunikace a to je příčina, proč jsou firewally někdy označovány jako síťové filtry.



Obr. 1: Postavení firewallu v síťovém komunikačním řetězci

Přehled typů firewallů

Rozdělení firewallů je možné podle úrovně, na které firewall filtruje komunikaci. Buď může fungovat na vrstvě síťové (nejrychlejší a zpravidla také nejméně nákladná varianta, ale filtruje velmi povrchně - rozhoduje se pouze dle informací dostupných na této vrstvě), tyto firewally jsou zpravidla označovány jako paketové. Pro jejich rychlost je vhodné je umístit na místa s hustým provozem, např. vstupní brána do sítě apod.

Firewall může být i stavový, když dokáže rozlišit již navázaná spojení a k nim příbuznou komunikaci (například pro FTP protokol). V tomto případě již ale musí pracovat na vrstvě transportní.

Na aplikační vrstvě pracující firewall je často označován jako proxy firewall. Výhodou těchto firewallů je, že zpravidla plně „rozumí“ fungování aplikací a protokolů a jsou schopny detekovat např. případy jejich nestandardního chování apod., tedy filtrují velmi důkladně. Je tedy nejvhodnější je umístit až na hostitelské stanice či v jejich bezprostředních blízkostech (předřazený jednoúčelový server - firewall).

Poslední dvě možnosti rozdělení tvoří analyzátory paketů a NAT (Network Address Translation)

SW/HW firewally

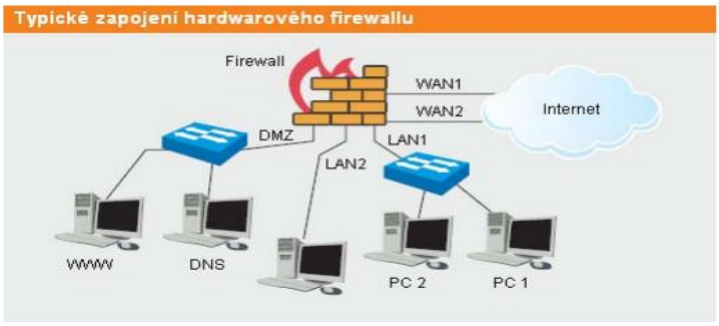
Hardwarové firewally

HW firewall je aktivní síťové zařízení, které slouží zejména ke kontrole a filtrování datového toku. Obvykle má alespoň jedno vstupní a jedno výstupní síťové rozhraní. Firewall kontroluje procházející data podle definovaných pravidel. Pravidla se aplikují buď na jednotlivé pakety (tzv. paketový filtr) na základě směru toku (vstupní/výstupní), nebo IP adresy a portu (zdroje/cíle). Pokud se zaznamenává stav pro rychlejší zpracování paketů patřících do jednoho toku dat, jedná se o stavový filtr. V případě zpracování aplikačních protokolů jde o tzv. aplikační brány.

D-Link DFL-800 [31] Cisco PIX 501 firewall [25]

Obr. 2: Ukázka hardwarových firewallů

Hardwarové firewally jsou specializovaná síťová zařízení, která na první pohled připomínají menší přepínače. Umějí však mnohem více. Kromě přepínání a filtrace paketů provádějí např. analýzu dat na aplikační úrovni (obsah WWW stránek, e-mailů, atp.). Kombinují funkce stavového i paketového filtru a přidávají kontrolu obsahu dat pomocí vyhledávání řetězců. Oproti softwarovému řešení jsou rychlejší, poměrně jednodušeji se konfigurují a spravují (velmi relativní). Poskytují komplexní ochranu a monitorování bezpečnosti sítě. Konfigurovat je můžeme pomocí grafického rozhraní (přes WWW) nebo textového CLI (Command Line Interface)



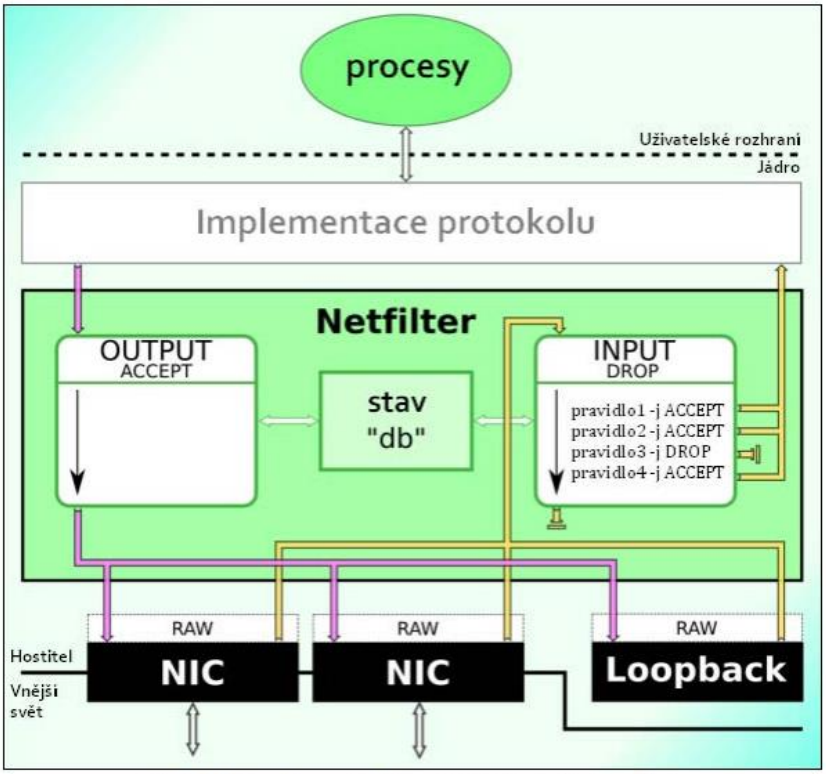
Obr. 3: Obvyklé zapojení zařízení – dvě interní LAN sítě s omezeným přístupem do Internetu (WWW, ping, ssh), demilitarizovaná zóna (DMZ) s kontrolovaným přístupem z Internetu a blokováním přístupu do vnitřní sítě LAN a Internet (WAN síť) [35]

Z technického pohledu je hardwarový firewall vlastně specializovaný počítač, který obsahuje procesor, paměť a síťová rozhraní.

Součástí mohou být i hardwarové akcelerátory postavené na technologiích ASIC nebo FPGA pro rychlý výpočet šifrovacích algoritmů u VPN či vyhledávání řetězců u systémů IDS (Intrusion Detection System). Např. u firewallu ZyWall 35 umožňuje přídavná PCMCIA karta prohledávání obsahu dat a antivirovou kontrolu. Operačním systémem těchto zařízení může být (kromě proprietárních řešení) například Linux.

Softwarové firewally

Dobře známé jsou softwarové firewally, které mohou běžet na osobním počítači, který může být pro tento účel vyhrazen nebo také ne. V prvním jmenovaném případě bychom se bavili o serverech s více síťovými kartami, které kromě filtrování provádějí i překlad adres NAT. Obvykle je na serveru s firewallem také služba DHCP pro dynamické přidělování IP adres, případně DNS služba pro překlad doménových jmen na IP adresy. Druhý případ značil (koncové) počítače uživatelů. SW řešení je levné a oblíbené – můžeme využít volně dostupný software a přizpůsobit si ho podle svých požadavků. Nastavení však vyžaduje zkušeného administrátora, a také analýza přenosů nebývá příliš jednoduchá. Pod OS Linux je k dispozici již vestavěný firewall v jádře operačního systému s názvem Netfilter.



Obr. 4: Ukázka softwarového firewallu Netfilter (nástroj Iptables)

Kde vytvořit firewall

Podle chráněného subjektu můžeme firewally rozdělit na dva druhy - firewally na ochranu (jednoho) počítače (tzv. „host-based“ firewally) a firewally na ochranu celé sítě (tzv. „network-based“ firewally).

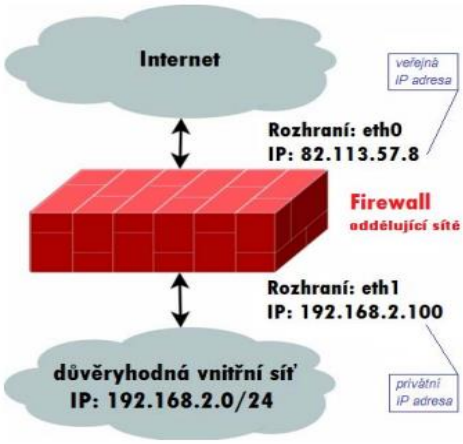
Firewally na ochranu (jednoho) počítače

Firewally určené k ochraně jediného počítače jsou relativně „jednoduché“. Jejich úlohou je chránit jeden jediný počítač (typicky s jednou síťovou kartou), nepotřebují se zaobírat přesměrováním paketů ani překladem adres. Jsou nakonfigurované na počítači, který mají chránit, a tak mohou být pravidla těchto firewallů velmi specifická a umožňovat jen ten síťový provoz, který je potřebný pro zabezpečení služby (služeb) či aplikací na daném počítači.

Takovéto firewally se často používají na doplňkovou ochranu serveru, který je umístěný za nějakým již existujícím („front-line“) firewallem, například (ovšem ne výlučně) ve webhostingovém centru a dále k ochraně jednotlivých osobních počítačů (pracovních stanic) připojených k síti. Proto jsou také často nazývány osobními firewally.

Firewally na ochranu sítě

Ve většině případů je úlohou firewallu oddělit dvě či vícero sítí s různými přístupovými právy nebo obsahem. Dále se budu držet příkladu, ve kterém firewall odděluje Internet (vnější síť) a Intranet (vnitřní síť). Z předešlého logicky vyplívá, že přes firewall musí procházet všechny pakety pohybující se mezi sítěmi. V opačném případě je firewall není schopen kontrolovat. Firewall funguje pro sítě jako bezpečnostní brána (ve skutečnosti vykonává i směrování paketů, takže tato analogie není od věci).



Obr. 5: Firewall, který má chránit síť, musí být umístěný mezi sítěmi, mezi kterými má kontrolovat a analyzovat síťový provoz

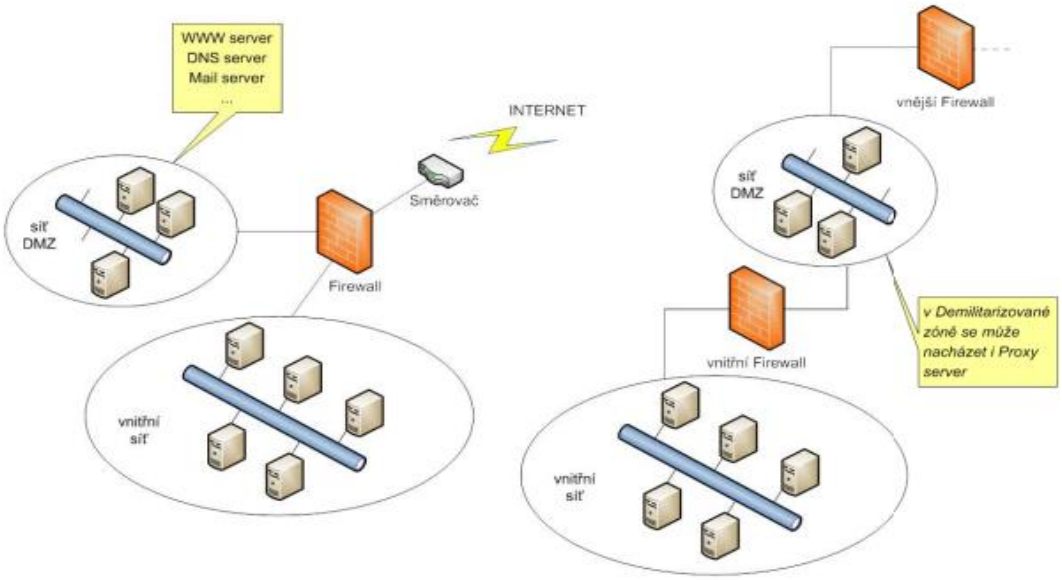
Politika firewallu

V kapitole o bezpečnosti serveru v síti budu hovořit o tom, jak důležité je stanovit si dobrou bezpečnostní politiku. V případě firewallu je důležité zejména rozhodnutí, jak firewall naloží s pakety, jejichž osud explicitně neurčíme. Ze zkušeností odborníků na počítačovou bezpečnost se obecně uplatňuje postup, při kterém „co není výslovně povoleno, je zakázáno“. Tato zvolená strategie se z hlediska zabezpečení jeví jako ucelenější.

Firewall & server versus samostatný firewall

Existují v podstatě dvě řešení, jak implementovat firewall do existující sítě. První řešení předpokládá, že firewall bude na serveru, který bude poskytovat určité služby pro některou ze sítí (nebo obě), které vzájemně propojuje. Bude tedy bránou, firewallem a serverem současně. Toto řešení je méně nákladné, protože nepotřebujeme samostatný počítač na firewall, ale (relativně) není příliš bezpečné. V případě, že na firewallu běží nějaké služby, je možno je za určitých okolností zneužít (ať už jsou chráněné jakýmkoliv způsobem), získat práva administrátora a upravit/smazat pravidla firewallu. Naproti tomu, když se firewall umístí na úplně samostatný počítač, na kterém neběží žádné služby, je řešení mnohem bezpečnější. V takovém případě je vhodné, jestliže se na firewall přihlašuje administrátor pouze z konzole, aby nemusely na počítači běžet ani služby jako SSH. Všechny služby, které mají být dostupné z Internetu, běží fyzicky na jiném počítači (serveru) a firewall zabezpečí správné směrování paketů pomocí routovací tabulky či překladu adres NAT.

V případě, že síť bude poskytovat služby do Internetu, nejbezpečnějším řešením se stává takové, při kterém firewall propojuje ne dvě, ale tři sítě (Internet, Intranet a DMZ – demilitarizovanou zónu). DMZ je síť, která se používá pro servery s veřejnými službami přístupnými z Internetu. Firewall umožňuje chránit Intranet (do něho proniknou maximálně odpovědi na spojení již iniciované z Intranetu), veřejně dostupné služby jsou v odděleném segmentu sítě a firewall nad všemi drží „stráž“.



Obr. 6: Ukázky demilitarizované zóny (DMZ)

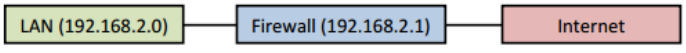
Paketové filtr

Paketový filtr umožňuje sledovat síťový provoz na třetí vrstvě ISO/OSI modelu (tj. logické IP adresy). Je mnohem rychlejší, ale jeho správa je komplikovanější a nemá tolik možností jako aplikační proxy, protože se například nedostane ke jménu uživatele, kterému patří zachycené pakety. Nevýhodou paketového filtru je i to, že někdy musí na serveru fungovat služby, které při komunikaci využívají náhodně vybrané porty (např. při přenosu souborů pomocí služby FTP). Paketový filter „nevidí“ (nepozná) souvislosti mezi pakety a každý analyzuje samostatně. Je tedy třeba povolit buď celý rozsah portů (příliš benevolentní) nebo jej zakázat a takovéto služby nepoužívat (někdy nerealizovatelné) .

Postup paketového filtru:

1. Zjištění zdrojové i cílové IP adresy a portů.
2. Průchod tabulkou po jednotlivých řádcích odshora dolů, dokud není nalezen řádek s pravidlem, který odpovídá danému paketu. Většinou definována i poslední výchozí „policy“.
3. Provede se akce uvedená v příslušném řádku pravidla. Zpravidla se paket propustí („allow“) nebo zahodí („deny“; „drop“), popřípadě i zaloguje, čili se zaznamená přísl. akce do souboru.

Příklad filtračních pravidel pro lokální síť s adresou 192.168.2.0:



pořadí pravidla	zdrojová IP adresa	zdrojový port	cílová IP adresa	cílový port	akce	popis
1	any	any	192.168.2.0	>1023	allow	příjem reakcí na výzvy zevnitř
2	192.168.2.1	any	any	any	deny	nelze zneužít adresu firewallu
3	any	any	192.168.2.1	any	deny	blokování spojení s firewallem
4	192.168.2.0	any	any	any	allow	povolení komunikace zevnitř
5	any	any	192.168.2.2	80	allow	přístup na www server zvenku
6	any	any	any	any	deny	vše ostatní nepustit (zakázat)

První pravidlo zajišťuje navázání TCP spojení iniciovaných z vlastní sítě. Druhé pravidlo zabraňuje útočníkovi vystupovat v síti jako firewall. Třetí pravidlo zabraňuje vnějšímu útočníkovi komunikovat s firewallem. Čtvrté pravidlo umožňuje všem prvkům sítě komunikovat s kýmkoliv z vnější sítě a použít

přitom jakýkoliv protokol. Páté pravidlo umožňuje projít všem paketům z vnější sítě, pokud nesou data protokolu HTTP. Poslední šesté pravidlo je velmi důležité - vše co nespadá pod výše uvedená pravidla nepustit do (ze) sítě. Jak je názorně vidět z tabulky, paketový filtr je založen na pravidlech stanovených pro dvojici zdrojová IP adresa a cílová IP adresa. Tyto údaje jsou často ještě upřesněny pro konkrétní TCP nebo UDP porty

Stavové firewally

Speciálním případem paketového filtru je stavový paketový filtr. Ten si dokáže uvést v souvislost procházející pakety a tak si uchovat stav spojení. Jedno navázané spojení a všechny pakety, které k němu patří, se nazývá relace (session). Stavový firewall se oproti paketovému filtru liší také tím, že není obecně univerzální, ale je využitelný pouze pro TCP/IP protokoly.

Stav spojení je možné uplatnit v pravidlech, a tak například automaticky povolit odpovědi na všechny odeslané pakety, povolit spojení, která souvisí s daným, již navázaným spojením po dobu jeho trvání, atd. Toho se s výhodou prakticky využívá např. u řešení již zmíněného problému s FTP.

Tento typ firewallu rozezná paket, který otevírá nové spojení, od paketů, které tuto komunikaci realizují, a díky tomu může precizněji filtrovat datové toky. Pokusy o spoofing, podvržení paketů, které se tváří, jako by se "vracely do sítě" v rámci fiktivního spojení, jsou blokovány.

Firewally se stavovou inspekcí využívají k filtraci stejně jako paketové filtry IP adresy i čísla portů. Navíc však využívají i informaci o stavu spojení, takže umožňují hlídat souvislosti mezi pakety.

Příklad: Klient vnitřní sítě s adresou 192.168.2.10 si chce prohlédnout webové stránky umístěné na internetové adrese 77.75.72.3 (*http://www.seznam.cz*). Klient tedy vyšle na danou adresu paket s TCP segmentem typu SYN (Synchronization) pro navázání spojení:

- **zdroj:** IP = 192.168.2.10, port = 1030 (port alokovaný prohlížečem),
- **cíl:** IP = 77.75.72.3, port = 80 (protokol HTTP).

Firewall si tento paket zapíše do své stavové tabulky a propustí jej do vnější sítě (ovšem pouze tehdy, splnil-li příslušné pravidlo pro přístup do vnější sítě - Internetu). Očekává, že potvrzující paket (od protistrany) by měl být TCP segmentem typu SYN+ACK:

- **zdroj:** IP = 77.75.72.3, port = 80,
- **cíl:** IP = 192.168.2.10, port = 1030.

Paket je propuštěn do vnitřní sítě, pouze pokud splňuje očekávané parametry. Do vnitřní sítě se tak zvenčí dostanou pouze pakety z komunikace iniciované vnitřním prvkem sítě => zvýšení bezpečnosti.

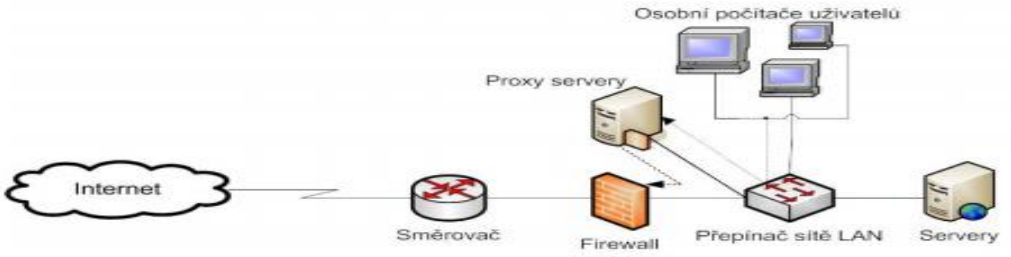
Softwarový firewall je implementovaný přímo v jádře Linuxu. Ve verzích jádra 2.4.X a 2.6.X je to bezpečnostní nástroj s názvem "iptables", který kromě správy filtrovacích pravidel umožňuje i správu NATu. Samotný firewall (framework pro manipulaci s pakety) se nazývá "netfilter".

Protože je Iptables z technologického hlediska nejdokonalejší (nejpokročilejší), a jelikož je doporučeno používat aktuální stabilní verze jádra Linuxu, budu se v dalších částech textu zaobírat pouze tímto mocným firewallovým nástrojem, který umožňuje linuxovému systému plně pracovat se síťovou komunikací.

Proxy firewally

Firewally se stavovou inspekcí paketů jsou v podstatě rozšířenou verzí běžného paketového filtru. Zde popisovaná zařízení jdou ještě dále a provádějí analýzu paketů na aplikační vrstvě (ochrana na úrovni aplikací). Tuto úroveň ochrany implementuje několik různých technologií, které se označují různými názvy; každá z nich pracuje trochu jinak, ale jejich cíl je stejný - posílit bezpečnost sítě.

Firewally na aplikační úrovni zajišťují nejbezpečnější typ datových spojení, protože dokáží v komunikačním procesu zkoumat úplně všechny vrstvy modelu TCP/IP. Pro zajištění této úrovně ochrany musí uvedené firewally neboli proxy (proxy servery) fakticky vstoupit do probíhající komunikace (do role „prostředníka“, což je v podstatě český význam slova „proxy“) a detailně kontrolovat každé spojení. Jestliže proxy označí dané spojení za povolené, otevře směrem k serveru druhé spojení od sebe sama, jménem původního hostitele, jak vidíme na obrázku 7.



Obr. 7: Činnost proxy serverů a jejich pozice

Při této inspekci je nutné odříznout datovou část každého procházejícího paketu, zkontrolovat jej, znovu sestavit a odeslat po druhém spojení. Uvedené funkce lze zajistit v různých typech firewallů:

- **Standardní proxy firewally**. Běžný proxy firewall neprovádí směrování paketů a pouze je přeposílá; pracuje v aplikační vrstvě modelu TCP/IP. Z funkčního hlediska přijímá nad jedním síťovým rozhraním pakety, kontroluje je podle definované množiny pravidel, a pokud se rozhodne pro jejich povolení, odešle je přes jiné rozhraní. Mezi vnějším a vnitřním počítačem nikdy neexistuje přímé spojení; z pohledu počítače ve vnitřní síti tak veškeré informace zdánlivě pocházejí od proxy firewallu.
- **Dynamické proxy firewally**. Tento typ proxy firewallů se vyvinul ze standardních proxy firewallů, oproti kterým je navíc rozšířen o filtrování paketů. Dynamický proxy firewall tak provádí úplnou inspekci paketů; po prvotním vytvoření spojení a zejména po jeho schválení již stačí ostatní pakety kontrolovat v rychlejším, i když slabším mechanismu filtrování paketů. Podtrženo a sečteno, spojení se nejprve zkontroluje na aplikační vrstvě, a poté již další kontrola probíhá jen na vrstvě síťové.

Tyto proxy firewally tak „vidí“ veškeré informace z aplikační vrstvy modelu TCP/IP. To znamená, že mohou vyhledávat přesněji definované údaje, než jakékoli jiné typy dosud probíraných technologií. Dokáží například rozlišit mezi paketem s e-mailovou zprávou a paketem s javovým appletem či nebezpečným prvkem ActiveX, jak vidíme na obrázku 8. Speciálním případem je SSL proxy, což je varianta proxy firewallu založená na protokolu SSL. S pomocí tohoto protokolu je umožněn uživatelům z vnější sítě (př. Internetu) bezpečný přístup do vnitřní sítě.

Obr. 8: Inspekce paketů v proxy serveru (HTTP)

Při vstupu do proxy serveru podle obrázku 8 se z paketu odstraní veškeré parametry hlavičky TCP/IP a samotné inspekci dále podléhají vlastní přenášená data. Informace zjištěné při této inspekci se poté předloží firewallovým pravidlům a podle výsledků bude průchod paketu povolen nebo zamítnut. Je-li paket shledán „nezávadným“, tedy povoleným, uloží si proxy firewall informace o daném spojení z hlavičky, přepíše novou hlavičku a upravený paket odešle dále. Zamítnutý paket se jednoduše odstraní.

Omezení možností proxy

Každá technologie má svá omezení nebo určité svoje nevýhody. Zde jsou některá obecná omezení proxy firewallů:

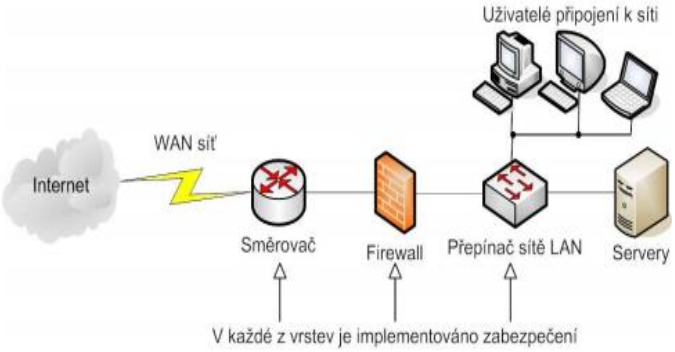
- **Pomalejší činnost.** Vzhledem k důkladnému zkoumání a pečlivému zpracování paketů jsou proxy firewally velice bezpečné, ale zároveň také dosti pomalé. Protože se na této úrovni

zabezpečení kontrolují v podstatě všechny části všech paketů, bývá činnost proxy firewallů opravdu pomalejší oproti ostatním.

- **Nejsou vždy aktuální**. S vývojem nových protokolů a aplikací je nutné odpovídajícím způsobem doplnit či rozšířit i proxy servery, které musí daný provoz označit za přípustný či nikoliv. To znamená, že je k nové aplikaci nutné také vyvinout a otestovat nové proxy servery. To ovšem nějaký čas trvá a do té doby je příslušné bezpečnostní zařízení neaktuální.

Z bezpečnostního hlediska se za nejbezpečnější firewall dá považovat standardní proxy firewall, který provádí inspekci veškerého provozu na aplikační vrstvě. V některých dnešních sítích není takové řešení vždy zrovna praktické a patřičné. Pro návrh správného a dostatečně silného zabezpečení je proto důležité se pečlivě připravit a seznámit se s charakterem síťového provozu i s požadovaným zabezpečením. Firma pro údržbu parků a zahrad bude mít například jiné potřeby zabezpečení než společnost, která vyvíjí elektronické komponenty pro armádní zbraňové systémy.

V každé navrhované síti je nutné v rámci vrstveného zabezpečení provozovat alespoň jeden ze dvou probíraných typů firewallů - stavový nebo proxy firewall. Pokud kromě některé z těchto technologií spustíme také na hraničním směrovači paketový filtr a ve firewallovém zařízení zapneme překlady adres NAT, dostaneme solidní základ vrstvené bezpečnosti sítě.



Obr. 9: Místa vrstveného zabezpečení sítě

Jelikož primárním cílem diplomové práce je detailně popsat princip činnosti proxy firewallů, věnuji této bezpečnostní problematice samostatnou kapitolu, která podrobně rozvede všechny možnosti využití proxy, objasní jak proxy fungují a nakonec vyzdvihne přednosti tohoto řešení.

Omezení firewallů

Firewall je důležitou komponentou zabezpečení každé sítě a jeho úkolem je řešit problémy spojené s integritou dat a s autentizací síťového provozu (prostřednictvím stavové inspekce paketů), a dále zajišťovat důvěrnost vnitřní sítě (pomocí překladového mechanismu NAT). Pokud bude síť přijímat veškerý provoz jen přes firewall, dostane se jí plnohodnotné ochrany. O důležitosti firewallu ve strategii zabezpečení nelze pochybovat, přesto je ale důležité si uvědomit, že i firewall se potýká s některými omezeními:

- Firewall nedokáže zabránit uživatelům a útočníkům s modemy v přímém přihlášení do vnitřní sítě (nebo naopak ze sítě ven); tím ovšem tento uživatel úplně obchází jak firewall, tak i jeho ochranu.
- Firewally nedokáží uskutečňovat přijaté zásady práce s hesly a nezabrání ani v možném zneužití hesel. Stanovit zásady pro práci s hesly je proto velice důležité, a to včetně sankcí za případné porušování.
- Naprosto neúčinné jsou firewally také proti netechnickým bezpečnostním rizikům, jako je například známé „sociální inženýrství“.
- Každý firewall tvoří úzké hrdlo síťového provozu, protože se v něm veškerá komunikace a zabezpečení koncentrují do jediného místa; tím vzniká jediné kriticky zranitelné místo sítě.

Porovnání jednotlivých typů firewallů

Výsledné stručné srovnání jednotlivých typů firewallů z hlediska jejich výhod a nevýhod. U každé vzpomenuté technologie se krátce zmíním o vlastnostech a připomenu základní princip.

Paketové filtry (Packet filter)

Nejjednodušší a nejstarší typ firewallu, kde jsou pakety řízeny pravidly, která uvádějí, z jaké adresy a portu na jakou adresu a port může být doručen procházející paket. Tyto informace jsou získány z hlaviček paketů. Paketový filtr pracuje na síťové vrstvě ISO/OSI modelu.
Toto řešení se vyznačuje jednoduchostí a transparentností, což se projevuje vysokou rychlostí. Z toho důvodu je řešení hojně využíváno v místech, kde není potřeba důkladnější analýza procházejících dat.
Další nespornou výhodou proti aplikačním proxy je přizpůsobivost na libovolný typ protokolu.
Nevýhodou je nízká úroveň kontroly procházejících spojení. To je způsobeno schopností sledovat pouze jednotlivé pakety bez možnosti hledání závislostí mezi nimi. Další nevýhody jsou absence autentizace vůči firewallu, relativně omezené možnosti logování, sledování pouze hlaviček paketů, zneužití nedokonalosti TCP/IP protokolu a konfigurace filtrů.
Typickým představitelem paketových filtrů je ipchains, který byl součástí jádra. Od jádra verze 2.4 je nahrazen nástrojem iptables (frameworkem netfilter).

Stavové paketové filtry (Stateful packet inspection)

Stavové paketové filtry poskytují to samé jako paketové filtry a navíc umožňují ukládání informací o povolených spojeních, které lze používat při rozhodování o budoucnosti paketů. Dochází ke zvýšení bezpečnosti, protože lze nastavit, která komunikující strana může otevřít spojení a firewall bude povolovat i pakety jdoucí z druhé strany jako odpovědi na požadavky (request <-> response).
Mezi výhody patří stejně jako u paketových filtrů rychlost zpracování požadavků a současně lepší možnost zabezpečení, než u paketových filtrů. Další výhodou je poměrně jednoduchá konfigurace minimalizující škody způsobené například překrýváním některých pravidel a kontrola stavu spojení.
Nevýhodou je nižší úroveň zabezpečení, než poskytují aplikační brány a o něco náročnější režie oproti paketovým filtrům (musí se udržovat informace o spojeních a kontrolovat souvislosti).
Typickým představitelem je iptables v linuxovém jádře.

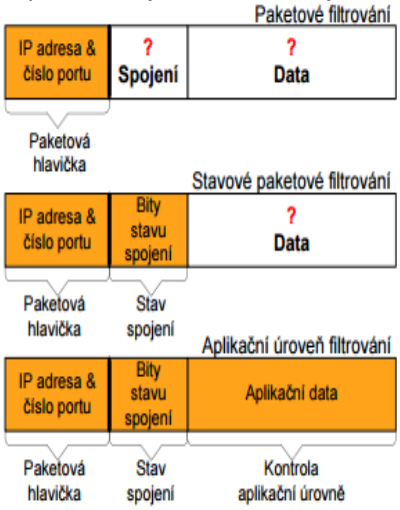
Stavové paketové filtry s kontrolou protokolů a IDS

Moderní stavové paketové filtry kromě informací o stavu spojení a schopnosti dynamicky otevírat porty pro různá řídící a datová spojení složitějších známých protokolů implementují něco, co se v marketingové terminologii různých společností nazývá nejčastěji Deep Inspection nebo Application Intelligence. Znamená to, že firewally jsou schopny kontrolovat procházející spojení až na úroveň korektnosti procházejících dat známých protokolů i aplikací.
Nejnověji se do firewallů integrují tzv. in-line IDS (Intrusion Detection Systems – systémy pro detekci útoků). Tyto systémy pracují podobně jako antiviry a pomocí databáze signatur a heuristické analýzy jsou schopny odhalit vzorce útoků i ve zdánlivě nesouvisejících pokusech o spojení, např. skenování adresního rozsahu, rozsahu portů, známé signatury útoků uvnitř povolených spojení apod.
Výhodou těchto systémů je vysoká úroveň bezpečnosti kontroly procházejících protokolů při zachování relativně snadné konfigurace, vyšší rychlost kontroly ve srovnání s aplikačními branami, nicméně je znát významné zpomalení proti stavovým paketovým filtrům.
Nevýhodou je zejména to, že z hlediska bezpečnosti designu je základním pravidlem bezpečnosti udržovat bezpečnostní systémy co nejjednodušší a nejmenší (i když aktuálním trendem je opak). Tyto typy firewallů integrují obrovské množství funkcionality a zvyšují tak pravděpodobnost, že v některé části jejich kódu bude zneužitelná chyba, která povede ke kompromitaci celého robustního systému.
Podobná funkce je k dispozici ve formě experimentálních modulů pro iptables v linuxovém jádře.

Aplikační brány (Application proxy firewall gateway)

Aplikační brána (proxy firewall, proxy server, aplikační proxy) je ochrana na aplikační vrstvě. Dochází k úplnému oddělení sítí. Spojení probíhá tak, že klient pošle proxy severu požadavek na otevření spojení s nějakou službou v jiné síti a aplikační brána toto spojení otevře. Všechna data prochází vždy přes proxy server, který rozhodne o jejich osudu. Jinými slovy, proxy server slouží jako prostředník mezi klientem v jedné síti a službou v druhé síti. Vedlejším efektem tohoto způsobu komunikace je skrytí zdrojové adresy klienta, protože jako klient vždy vystupuje aplikační brána.
Výhodou tohoto řešení je možnost kontroly obsahu přenášených paketů (např. antivirová kontrola, filtrování nevhodného obsahu, systém detekce průniků, apod.), autentizace uživatelů, možnost cachování dat, skrytí zdrojové adresy klienta (anonymizace) a četné logovací možnosti.

Proti tomuto řešení hovoří především vyšší hardwarové nároky (výkonová a paměťová náročnost) a netransparentnost (nutná úprava aplikací), kdy musí každá aplikace podporovat připojení pomocí proxy a tyto aplikace musí být správně nastaveny. Pro každou aplikaci musí být zvláštní proxy. Tj. nejsou univerzální jako paketové filtry - HTTP proxy, FTP proxy, atp.
Pro Linux existuje mnoho proxy serverů, jedněmi z nich jsou např. Squid, FWTK nebo SOCKS.



Pro úplnost dále uvádím přehled vlastností a charakter. atributů ostatních síťových technologií.

Network Address Translation (NAT)

Překlad síťových adres slouží ke změně hlaviček paketů, kde jsou přepsány zdrojové - DNAT (Destination NAT), nebo cílové adresy - SNAT (Source NAT). Toto řešení lze použít, pokud nemáme mnoho IP adres, aby všechny počítače v síti mohly mít svou vlastní veřejnou IP adresu nebo ke zvýšení zabezpečení. Za výhodu i nevýhodu lze považovat „neviditelnost“ počítačů z vnější sítě, což vede k již zmíněnému zvýšení zabezpečení a také k omezení, protože nelze používat některé služby vyžadující inicializaci spojení serverem (řeší Port Forwarding, česky přesměrování/mapování portů). Na překladu adres je založena aplikační brána.

Analyzátory paketů (Network Packet Analyzer)

Analyzátory paketů fungují podobně jako aplikační brána, ale jsou pasivní a transparentní. Umožňují neměnnou analýzu paketů a případné reakce na podněty. Příkladem může být IDS balík Snort. Mezi výhody tedy patří pasivní kontrola přenášených dat s možností definice reakce na potenciální útok (např. zablokování příjmu paketů z podezřelé IP adresy). Nevýhodou je náročnost na systémové prostředky a fakt, že nejsou univerzální.

HW versus SW firewally

Pravděpodobně by bylo vhodné uvést i přednosti obecně fyzických řešení firewallů. Hardwarové firewally mají vůči softwarovým firewallům výhodu především v rychlosti, nezávislosti na operačním systému, jednoduchosti nasazení a správy a v méně bezpečnostních „trhlinách“. Zřejmou výhodou může být i integrace firewallu s jinými zařízeními (např. směrovači). Dražší a výkonnější HW firewally mohou rovněž podporovat i externí moduly přidávající komplementární funkce antivirové kontroly či systému detekce průniku IDS. Toto zvolené řešení má ale i své nedostatky v podobě vyšší ceny (některé SW firewally jsou poskytovány zdarma) a pružnosti/adaptace konfigurace. Pouze jako poznámku doplním, že HW firewally jsou spíše firemní záležitostí [6].

Překlad síťových adres (NAT)

Když se poprvé objevilo adresování protokolu IPv4, zdálo se, že je množství dostupných adres prakticky neomezené a že musí stačit „na věčné časy“. Teoreticky je v tomto adresovém prostoru k dispozici 232 neboli 4 294 967 296 jedinečných veřejných adres; skutečný počet dostupných adres se pohybuje někde mezi 3,2 a 3,3 miliardami, a to z důvodu rozdělení adresového prostoru do tříd (A, B, C) a vyhrazení určitých adres pro vícesměrné vysílání, testování a další speciální účely (třída D). Uvedené dělení nadefinovalo sdružení IETF (Internet Engineering Task Force), ale později se od něj

upustilo, protože nebylo příliš hospodárné. Efektivním se ukázalo být tzv. podsíťování (členění na subsítě), kdy se několik prvních bitů původně určených k adresaci stanice použije na definici podsítě, zbývající bity pak adresují stanice v této podsíti. Subsíť je definována adresou třídy a síť. Maskou. Vzhledem k obrovskému rozmachu sítě Internet a neustále rostoucí potřebě přidělování IP adres v domácích sítích a v malých firmách je zřejmé, že počet dostupných adres IPv4 zkrátka nemůže stačit. Jednoduchou úvahou dojdeme k řešení, kterým je změna schématu adresování a rozšíření adresového prostoru. Toto nové adresování je skutečně ve vývoji a nasazení, a to pod označením IPv6, ale jeho implementace potrvá ještě řadu let, protože je u ní nutné změnit činnost infrastruktury celého Internetu. Proces přechodu z dosavadní verze IPv4 na novou verzi IPv6 je proto velmi pomalý a nejspíše bude i nadále postupovat zdlouhavým tempem, protože život dosavadních adres IPv4 dále prodlužuje překladový mechanismus NAT.

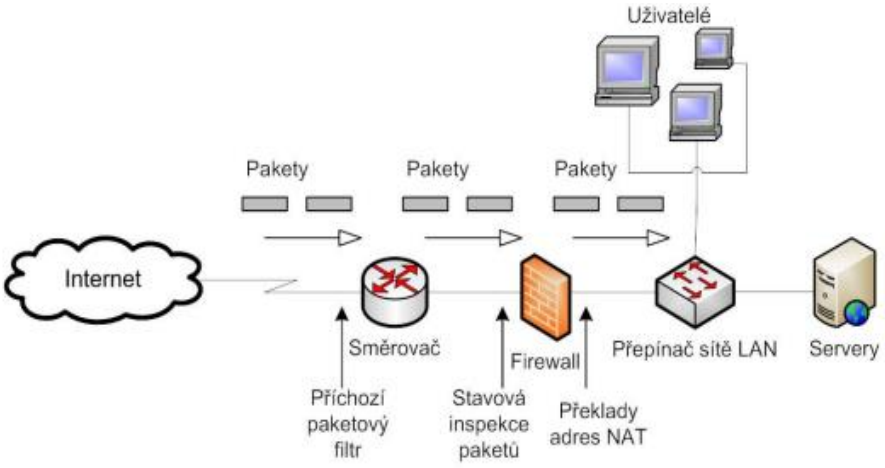
Pomocí překladů síťových adres NAT mohou organizace vyřešit nedostatek veřejných IP adres ve své síti, připojené do Internetu. Sítě, které dosud nemají veřejné IP adresy, registrované u centra NIC, si je musí vyžádat od sdružení IANA (Internet Assigned Numbers Authority) a registru ARIN (American Registry for Internet Numbers), což znamená zdlouhavý úřední postup. Mnohá pracoviště dokonce těmito nepříjemnými úředními procedurami neprojdou; pro většinu sítí je tudíž řešením právě NAT.

Sdružení IANA vyhradilo pro potřeby privátních sítí následující tři bloky adresového prostoru IP:

- 10.0.0.0 – 10.255.255.255 (prefix 10/8)
- 172.16.0.0 – 172.31.255.255 (prefix 172.16/12)
- 192.168.0.0 – 192.168.255.255 (prefix 192.168/16)

Při zapojení překladů NAT může firma používat veřejné IP adresy na vnější straně sítě (tedy u zařízení, přímo připojených k veřejnému Internetu). Jak jsem ale naznačil, těžko bude mít stejná firma dostatek veřejných IP adres, aby je přiřadila každému serveru, osobnímu počítači, síťové tiskárně, směrovači, bezdrátovému zařízení, atd. Všechna uvedená zařízení potřebují ale pro komunikaci v protokolu TCP/IP nějakou IP adresu, a proto budeme ve vnitřní síti přiřazovat privátní IP adresy. Díky tomu mohou všechna zařízení vnitřní sítě komunikovat v protokolu TCP/IP – a to je také naším cílem. Směrem do veřejného Internetu, se již ale privátní adresy dostat nesmí, a proto musíme uvést do provozu mechanismus NAT.

Překlady síťových adres NAT (Network Address Translation) zavádíme a provozujeme na vhodném zařízení (firewallu, směrovači nebo počítači), umístěném mezi vnitřní sítí s privátními IP adresami a vnějším Internetem s veřejnými IP adresami. Zmíněné zařízení pak provádí takzvané převody neboli překlady adres z privátních na veřejné. Jedna jeho strana bývá připojena k vnitřní síti, druhá pak k Internetu nebo jiné vnější síti. Umístění mechanismu NAT v rámci vrstvené obrany dokresluje obrázek 10.



Obr. 10: Pozice mechanismu NAT v síti

Mechanismus překladů adres NAT znamená také další úroveň zabezpečení sítě. Samotný NAT má několik různých podob a může pracovat ve třech základních režimech činností:

- Statický NAT – definuje jednoznačné mapování neboli zobrazení privátních IP adres na veřejné (tedy jedna k jedné). To je užitečné zejména u zařízení, která musí být dostupná z veřejného Internetu (z vnější sítě). Pokud má například webový server takovouto vnitřní IP adresu (10.0.0.1) a má být dostupný z Internetu (je to veřejný webový server), musíme definovat statický překlad NAT, který zajistí trvalý a jednoznačný převod uživatelských požadavků s veřejnou adresou webového serveru na jeho vnitřní adresu 10.0.0.1. Právě pro zařízení dostupná z vnější sítě, jako jsou webové servery apod., jsou statické překlady NAT velice běžné.
- Dynamický NAT – tento režim zajišťuje mapování privátních IP adres na veřejnou IP adresu, vybranou ze skupiny registrovaných adres. I při tomto typu překladů NAT je mezi privátními a veřejnými IP adresami jednoznačné zobrazení (jedna k jedné); má-li například náš osobní počítač privátní adresu 10.0.0.2 a kolegův počítač 10.0.0.3, dostaneme při komunikaci s veřejným Internetem přiděleny od firewallu dvě různé veřejné IP adresy (ty mohou být ovšem při každé komunikaci jiné). Dynamický NAT je bezesporu užitečný, ale na některé stanice je krátký; může se například stát, že již firewall všechny veřejné IP adresy vyčerpal a naši komunikaci tak zamítne. To může být v určitých situacích vážný problém, proto byl vyvinut takzvaný přetížený NAT.
- Přetížený NAT – jedná se o speciální typ dynamického NAT, který převádí větší skupinu privátních IP adres na jedinou veřejnou IP adresu, přičemž je rozlišuje pomocí různých portů TCP. Uvedený mechanismus se označuje také jako překlady portů PAT (Port Address Translation), případně jednoadresový NAT. Název ale není tak podstatný jako mechanismus činnosti: pro jedinou IP adresu je totiž k dispozici více než 64 000 portů (celkem 65 535) TCP, a proto PAT umožňuje efektivní přístup k Internetu velkému množství uživatelů s privátními IP adresami. Tento poslední typ překladu NAT se používá nejčastěji, protože dokáže najednou obsloužit největší množství uživatelů.

Zvýšení bezpečnosti sítě

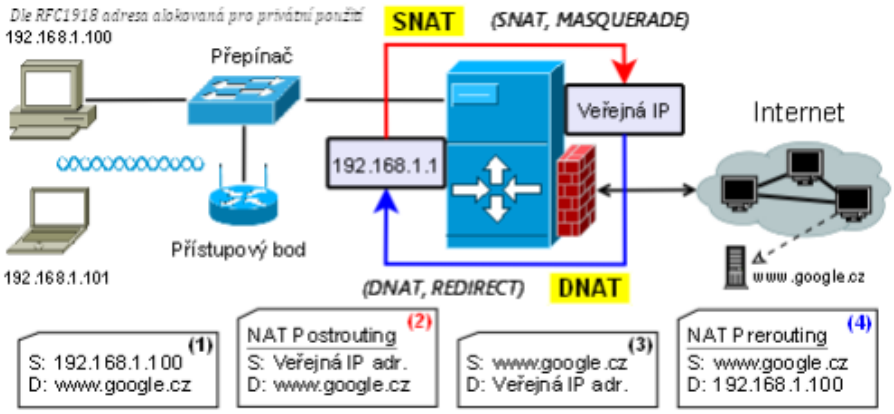
Vývoj překladového mechanismu NAT byl veden především snahou o vyřešení problému s nedostatkem veřejných adres protokolu IPv4. NAT poskytuje ale další vrstvu zabezpečení a ochrany sítě. Obecně se totiž dá říci, že NAT útočníkovi velmi výrazně ztěžuje:

- Mapování topologie cílové sítě a zjišťování informací o konektivitě
- Zjištění počtu provozovaných systémů v síti
- Zjištění typu provozovaných počítačů a jejich operačních systémů
- Vedení různých útoků s odepřením služeb (DoS), jako jsou například záplavy synchronizačních paketů SYN, prohledávání portů a vnášení paketů (injekce)

Zařízení uvnitř sítě LAN nejsou díky technologii překladu adres NAT adresovatelná z Internetu, což taktéž posiluje bezpečnost (útočník nezná strukturu sítě a nemůže se spojit přímo s konkrétním počítačem, lépe řečeno zařízením v síti, vždy se dostane „pouze“ na hraniční směrovač nebo počítač provádějící službu překladu adres). NAT ovšem firewall nenahrazuje a existují způsoby, jak počítače za NATem napadnout.

Překlad adres a OS Linux

Zde je uvedena příkladová situace. Uživatel vnitřní lokální sítě s IP 192.168.1.100 má přístup k Internetu a požaduje po internetovém prohlížeči webovou stránku http://www.google.cz. Internetový prohlížeč naváže TCP spojení pomocí HTTP protokolu s webovým serverem (na portu 80) a vrátí zpět uživateli požadovaný dokument, www stránku, soubor apod. Mezitím se událo (z pohledu překladu síťových adres) několik skutečností zachycených na obrázku 11.



Obr. 11: SNAT (přepis zdrojových adres), DNAT (přepis cílových adres)

Situace, která je symbolizována tabulkou s pořadovým číslem (1), zachycuje požadavek klienta v síti LAN na webový vyhledávací portál s doménovým názvem www.google.cz. S pořadovým číslem (2) následuje přepis zdrojové IP adresy (+ portu) SNAT tohoto požadavku, dále ve (3) kroku příchozí odpověď od serveru na požadavek klienta, a ve (4) finální fázi přepis cílové IP adresy (+ portu) DNAT.

SNAT v POSTROUTING – mění se zdrojové adresy a zdrojové porty.

DNAT v PREROUTING – mění se cílové adresy a cílové porty.

Využití je zřejmé: připojení sítí LAN k Internetu pomocí jedné veřejné IP adresy.

Pozn. Ne všechny pakety prošlé PRE/POSTROUTING projdou, musí mít pravidlo v řetězci FORWARD!

Omezení mechanismu NAT

Je zřejmé, že příchod technologie NAT do světa počítačových sítí a Internetu znamenal alespoň částečné vyřešení problémů s nedostatkem IP adres. Mnozí lidé se tudíž ptají, jestli sítě vůbec někdy přejdou na novou verzi protokolu IPv6, když překlady NAT tak výborně fungují. Otázkou ovšem fakticky není zdali, ale kdy. Asijský region dnes například v implementaci protokolu IPv6 jasně vede a mnohé sítě jej již skutečně používají.

S ohledem na stále rostoucí konektivitu a konvergenci sítí budou potřeba stále další a další IP adresy. Nakonec se tedy budeme muset k protokolu IPv6 uchýlit. Mechanismus NAT tyto nevyhnutelné změny pouze oddálil. NAT má svoje nepopiratelné výhody, ale na druhé straně má také jistá omezení:

- **Problémy s protokolem UDP** – překladový mechanismus NAT sleduje a kontroluje stav spojení. V protokolu UDP ovšem nelze stav spojení nijak určit (protokol UDP je nespojovaný – spojení se v něm vůbec nevytvářejí). NAT nedokáže tudíž žádným způsobem určit, jestli určitý paket spadá do nějaké probíhající konverzace, nebo jestli tvoří izolovaný přenos dat. Zařízení s překlady NAT tak musí odhadovat, jak dlouho může konverzace UDP trvat a jak dlouho ji tedy po posledním paketu ponechat otevřenou; hovoříme o tzv. době nečinnosti. Například ve firewallech od firmy Cisco je možné nastavením doby nečinnosti tyto případy omezit.
- **Citlivé protokoly** – některé protokoly skrývají, pozměňují nebo jinak zastiňují atributy paketů, které NAT potřebuje ke správnému překladu adres. Příkladem jsou protokoly Kerberos, X Window, vzdálený shell (rsh) nebo protokol SIP (Session Initiation Protocol), které mohou mít při průchodu zařízením s NAT jisté problémy. Příčina problémů se skrývá v aplikacích, jež vkládají IP adresu do paketů. Firewally Cisco mají pro různé protokoly opravné funkce (fixup), například Skinny pro telefonii (SCCP).
- **Vzájemné vlivy systémů šifrování a autentizace** – mnohé systémy šifrování dat se pokoušejí zajistit integritu paketů a kontrolují tak jejich neporušenost při přenosu. Překladový mechanismus NAT ale z principu pakety pozměňuje, a proto šifrovací a autentizační technologie s ním často nedokáží spolupracovat.
- **Komplikovaný záznam do systémových protokolů** – pokud zařízení odesílá přes překladové zařízení určité informace do systémových protokolů, musí cílové zařízení znát překlady prováděné mechanismem NAT. Dosažení souladu systémových protokolů s překlady NAT pak

může být velice složité a často se těžko zajišťuje, který z interních systémů vlastně dané události zaznamenal.

Závěrem ale poznamenejme, že překladový mechanismus NAT je skutečně velmi užitečný – celé firemní síti nabízí přístup k Internetu a zároveň tvoří další úroveň zabezpečení. Technika NAT umožňuje komunikaci zařízení v privátní síti se zařízeními jiných sítí, sdílet jednu veřejnou IP adresu více zařízeními privátní sítě, vyrovnávat zatížení serverů poskytujících stejnou službu, v případě výpadku zajistit adekvátní náhradu serveru záložním serverem a hlavně (z hlediska bezpečnosti) skrýt před vnějším útočníkem strukturu vnitřní sítě. Tímto disponují i proxy servery, probírané v další kapitole, jež pracují na nejvyšší aplikační vrstvě. Dokáží navíc zabránit prozrazení některých důležitých informací o hostitelské stanici, jako např. typ OS, www prohlížeč, interní IP adresa a další. Vzdálený server se tak o klientu dozví minimum informací (je odkázán pouze na informace zpřístupněné proxy serverem), což umožňuje určitý (pokročilejší) stupeň anonymizace.

Proxy servery

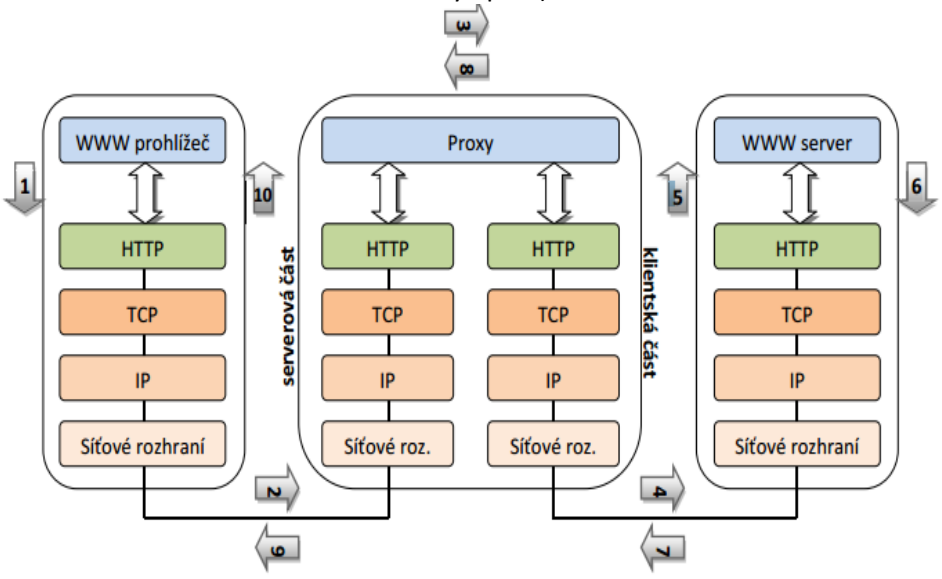
Jak proxy servery fungují

Přímá komunikace mezi klientem a WWW serverem pomocí HTTP protokolu je nejběžnější způsob získávání informací z WWW. V některých případech je však tato forma komunikace zprostředkována pomocí prostředníka – proxy serveru. Proxy server je zpravidla program (běžící na serveru organizace), který pracuje současně jako klient i server, schematické znázornění funkce je možné nalézt na obrázku 12. V obrázku je záměrně opomenuto DNS zjišťování IP adresy hostitele, které probíhá jak na straně klienta (hledání IP proxy serveru), tak na straně proxy serveru (hledání cílové IP webového serveru).

Podstatnou vlastností je, že proxy server tím, že vystupuje za klienta, poskytuje anonymitu uživatele vzhledem k www serveru.

Jak již bylo nastíněno, proxy server je server počítačové sítě, který umožňuje klientům nepřímé připojení k jinému serveru. Proxy server funguje jako prostředník/zprostředkovatel mezi klientem a cílovým serverem, překládá klientské požadavky a vůči cílovému serveru vystupuje jako klient. Přijatý požadavek následně odesílá zpět na klienta. Může se jednat jak o specializovaný hardware, tak o software běžící například na počítači klienta.

Aplikační proxy server je server speciálně určený pro určitý protokol resp. aplikaci. Za pomocí něj lze analyzovat obsah komunikace, případně ji pozměňovat (např. odstraňování reklamních bannerů z HTTP požadavků, blokování webových stránek podle obsahu, zákaz přístupu na nepovolené webové servery apod.).



Obr. 12: Fungování komunikace zprostředkované proxy serverem

Stručný popis jednotlivých kroků:

(1) Uživatel chce zobrazit určitou stránku na internetu, např.: http://www.server.cz/stranka.htm, což se předá nižším vrstvám k přenosu. Stručný popis jednotlivých kroků [18]:

(2) Nižší vrstvy navážou spojení se serverovou částí proxy serveru a předají požadavek prohlížeče (HTTP metoda GET).

(3) Proxy server rozumí struktuře HTTP protokolu a proto je schopen detailně vyhodnotit požadavek. Na základě vlastní logiky může proxy tento požadavek přepsat a de facto tak např. změnit cílový server, z kterého se bude stránka získávat. Dále může ověřovat, zda je vůbec uživatel oprávněn takový požadavek provést. Další netriviální činností, kterou může proxy provádět, je ukládání odpovědí do své cache paměti a při opakování dotazu na stejnou stránku tak zajistit rychlejší odezvu. Tato činnost však mírně ztrácí význam v souvislosti s rozvojem dynamicky generovaných stránek. Komplexnější proxy dokáže na základě spolupráce s firewallem případně antivirovým programem účinně filtrovat provoz a chránit tak uživatele. Pokud proxy nenajde ve své paměti požadovanou stránku a uživatel je oprávněn, předá se požadavek klientské části proxy.

(4) Klientská část zajistí navázání spojení s www serverem a předání metody (GET /stranka.htm).

(5) WWW server přijme požadavek a vyhodnotí ho.

(6) Výsledkem je odpověď WWW serveru, zpravidla požadovaná stránka.

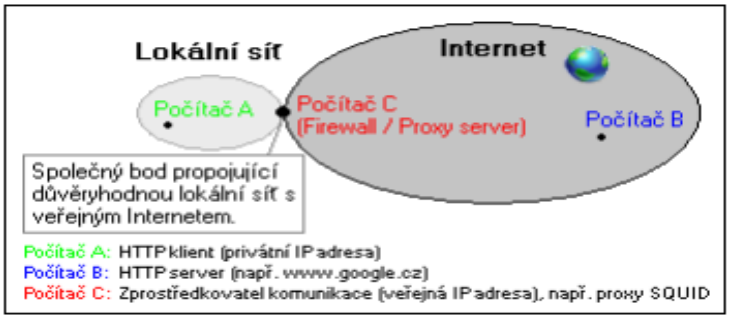
(7) Na základě nižších vrstev je odpověď předána zdroji metody GET, což je v tomto případě klientská část proxy serveru.

(8) Stejné operace jako v bodě 3), jen zpětně. Např. přepsání odpovědi do formy v jaké ji očekává uživatel, případně zápis stránky do cache paměti.

(9) Odpověď se pomocí nižších vrstev předá uživateli (původní tazatel).

(10) Prohlížeč odpověď zpracuje a stránku zobrazí.

Jak je patrné z předcházejícího textu, proxy rozumí aplikačnímu protokolu (HTTP) a je schopno samo navazovat spojení, což jsou nejmarkantnější rozdíly např. od techniky překladu síťových adres (NAT). Umístění proxy serveru v rámci sítě není pevně dáno. Nejčastějším použitím je oddělení vnitřní sítě a Internetu, takže proxy server je umístěn na pomezí těchto dvou oblastí (viz obrázek 13). K proxy serveru se však lze připojovat i za hranice vnitřní sítě (do Internetu). V takovém případě se jedná o veřejné (public) proxy servery, kterých existuje velké množství, avšak jejich důvěryhodnost je často pochybná, protože do souhrnu (ne)žádoucích činností, které provádí, nemáme možnost nahlédnout.



Obr. 13: Ukázka umístění proxy serveru v rámci sítě (LAN ⇔ Internet)

V textu je pojednáno pouze o variantě proxy pracující s HTTP protokolem (nejčastější), proxy server však zpravidla podporuje i další typy, jako např. FTP protokol atp

Oddělení sítí – bezpečnost

V lokálních sítích se často používají privátní adresy (např. 10.0.0.0/8, 192.168.0.0/16), takže počítače nemohou komunikovat (resp. přistupovat) přímo do Internetu. Používá se buď technika překladu adres na firewallu (NAT), nebo řešení pomocí proxy serveru. Klient používá proxy server na své vlastní privátní síti (má privátní IP adresu) a ten komunikuje (nejčastěji pomocí jiného síťového

připojení) s Internetem (tedy vnější sítí). Nutno poznamenat, že pro některé sítě je použití přístupu pomocí proxy serveru jedinou možností přístupu na Internet.

Výhodou použití proxy serveru z hlediska bezpečnosti je:

- Možno ho použít pro přístup k (určitým službám) Internetu i v případě, že klienti používají privátní adresy (netřeba použít NAT).
- Možno ho použít jen pro přístup k těm službám, které podporuje proxy server.
- Klient se připojuje na (lokální) proxy server a ne přímo na vzdálený server – nekomunikuje přímo s vnější sítí, čímž se dosahuje vyšší bezpečnosti, neboť je nemožné zneužít probíhající spojení na útok směřující ze serveru na klienta.

Existují nejméně tyto tři zmíněné důvody, proč nasadit proxy server v praxi. Přirozeně, řešení má i své nevýhody či nedostatky:

- Možno ho použít jen na přístup k těm síťovým službám, které podporuje samotný proxy server (toto je výhoda i nevýhoda zároveň, záleží na úhlu pohledu).
- V případě použití aplikačního proxy serveru, zpracování aplikační vrstvy modelu OSI (běžné síťové protokoly založené na TCP) zanáší do komunikace jisté zpoždění – proxy server musí analyzovat nejvyšší vrstvu modelu OSI.

Obecné doporučení:

V případě užití proxy serveru, vzdálený server nekomunikuje přímo s klientem, takže nemůže proti němu podniknout útok, může však zneužít spojení se samotným proxy serverem. Tento proxy server je třeba proto co nejlépe zabezpečit (pomocí firewallu):

- Proxy server nepotřebuje přijímat spojení z vnější sítě, pouze odpovědi na už vytvořené spojení (stavový filtr v IPTABLES: ESTABLISHED).
- Proxy server se nepotřebuje připojovat do vnitřní sítě, posílá pouze odpovědi na už vytvořené spojení (stavový filtr v IPTABLES: ESTABLISHED).
- Proxy server by měl „poslouchat“ jen na portech, které jsou nevyhnutelné pro správný provoz dané služby (služeb).
- Proxy server by v ideálním případě neměl poskytovat žádné jiné služby (WWW, FTP, MAIL, aj.); důvodů existuje několik, mezi hlavní patří bezpečnostní hledisko (možné zneužití poskytovaných služeb a následné získání vyšších práv - root) a rozprostření zátěže mezi více serverů (přičemž každý server plní svou funkci => při pádu jednoho stále fungují ostatní).

Řízení přístupu

Jestliže zabezpečíme, že všechny klientské počítače v síti budou používat proxy server, stane se jediným „úzkým hrdlem“ komunikace mezi klientem a serverem. Vznikne tak místo v síti, kde je možné pomocí stanovených pravidel regulovat přístup k Internetu.

Výhody použití proxy serveru:

- Možnost nasazení a vynucování bezpečnostní politiky z hlediska přístupu k Internetu – např. k určitým nevhodným či nebezpečným webovým stránkám.
- Jednoduché nastavení bez nutnosti konfigurovat politiky na všech klientech.

Nevýhody použití proxy serveru:

- Výkon proxy serveru má vliv na rychlost a efektivitu přenosu (čas odezvy na požadavek)
- Vzniká rizikové místo v síti („single point of failure“), v případě výpadku proxy serveru je nemožné se dostat na Internet (či obecně do vnější/WAN sítě). Toto lze řešit pomocí vícenásobného množství proxy serverů v clusteru, ale jen v případě, že je možné si to dovolit.

Druhy proxy serverů

Jak už jsem v předešlém poznamenal, proxy je síťová brána, obecně nejen pro HTTP protokol. Většinou umožňuje cachování dat, kontrolu přístupu, filtrování obsahu a některé další služby.

V zásadě rozlišujeme dva druhy proxy serverů:

1. **Aplikační proxy servery** – jsou navrhnuté jen pro vybrané síťové služby, jako např. HTTP, HTTPS, FTP, apod. Výhodou je, že nastavování a konfigurace klienta nejsou složité, stačí uvést adresu proxy serveru a jeho port. V zásadě rozlišujeme dva druhy proxy serverů:

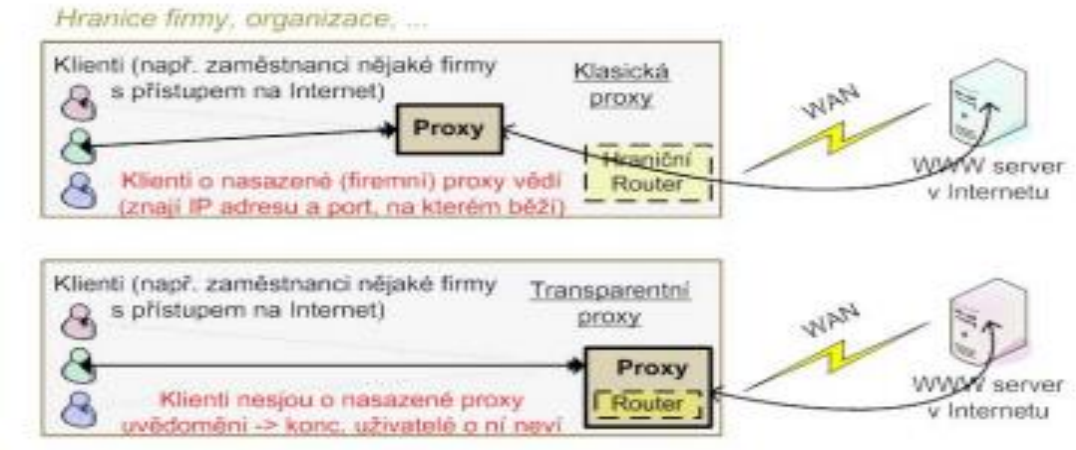
2. **SOCKS proxy servery** – fungují pro libovolné služby za předpokladu, že aplikace podporuje tento typ proxy serveru (musí být speciálně upravená).

Další eventuální rozdělení je taktéž na forward proxy a reverse proxy. Záleží, zda jde o caching či balancing – umístění před klientem nebo serverem.

Forward (dopředná) proxy – klasická síťová brána s možností cachování, umístěna blíže ke klientovi. Typická je speciální konfigurace klienta (nemusí být vždy pravda).

Vlastnosti: urychlení síťového provozu, snížení datového toku, bezpečnost (lze doplnit o antivir aj.).

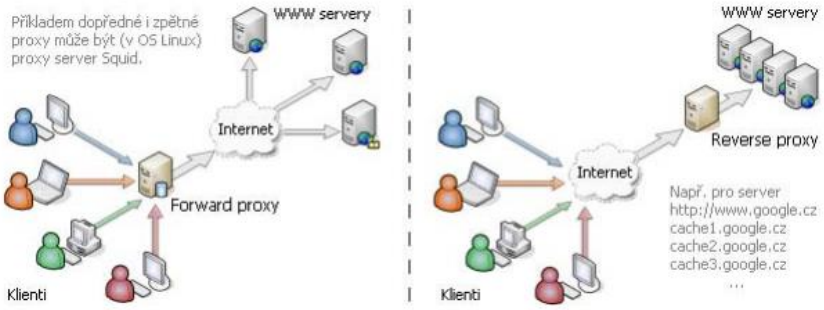
Podrobnější rozčlenění dopředné proxy může být na klasickou proxy a transparentní proxy.



Reverse (zpětná) proxy – je transparentní z pohledu klienta, umístěna před serverem. Výhodou je možnost balancingu (rozložení požadavků na více strojů). Klient nevyžaduje speciální konfiguraci.

Vlastnosti: urychlení síťového provozu (některé požadavky lze vyřídit z cache paměti, eliminace prodlevy ve spojení mezi klientem a serverem, mezi klientem a proxy může probíhat SSL spojení, vlastní server se tak může věnovat vyřizování skutečných požadavků), rozložení zátěže na více serverů, zvýšená bezpečnost (klienti se připojují k proxy, ne k samotnému serveru).

Reverzní proxy server někdy také slouží k šifrování/dešifrování spojení. Nedochází tím k degradaci potřebného výpočetního výkonu serverů. Zajištěno je šifrování směrem ke klientovi, spojení mezi serverem a proxy šifrované není, ale tato oblast se považuje za bezpečnou (součást např. DMZ).



Obr. 14: Aplikované rozdělení proxy na Forward (dopřednou) a Reverse (zpětnou)

Klasická proxy

Klasická proxy se používá zejména pro protokoly Telnet, FTP, HTTP a HTTPS. Například při použití Telnetu se uživatel nejdříve připojí na proxy, kde zadá příkaz connect, ve kterém určí název nebo IP adresu skutečného serveru. Klientská část proxy naváže spojení s cílovým serverem a začne předávat mezi klientem a tímto serverem data. Klientovi se pak jeví, jako by mezi ním a cílovým serverem žádná proxy nebyla.

Generická proxy

Někteří klienti nemohou nebo nechtějí vést úvodní dialog s proxy, aby jí sdělili cílový server. Generická proxy je program, který může spouštět a konfigurovat správce sítě podle momentálních požadavků. Dokáže pracovat s TCP i UDP.

Serverová část proxy je spuštěná na konkrétním portu, který určil správce, a je připravena přijímat požadavky od klientů ve vnitřní síti. Klientská část proxy je pevně nastavena na jeden cílový server. Jedna spuštěná generická proxy dokáže obsloužit jeden cílový server. Naráz jích pochopitelně může být spuštěných více. Generické proxy se používají hlavně pro protokoly POP3, IMAP4, SSH, LDAP, SMTP a NNTP.

Transparentní proxy

Klient naváže spojení s transparentní proxy a má pocit, že navázal spojení se skutečným serverem. Klientská část transparentní proxy hned navazuje spojení s cílovým serverem. Transparentní proxy nemusí vést s klientem žádný dialog.

Předpoklady pro práci transparentní proxy jsou:

- Klient ve vnitřní síti má možnost přeložit jméno cílového serveru nebo počítače v Internetu na IP adresu. Cílová adresa IP datagramu je adresa cílového serveru.
- IP datagram odeslaný do Internetu musí projít přes transparentní proxy.
- IP datagramy jednoho spojení musí procházet přes jednu transparentní proxy. Může existovat více transparentních proxy najednou.

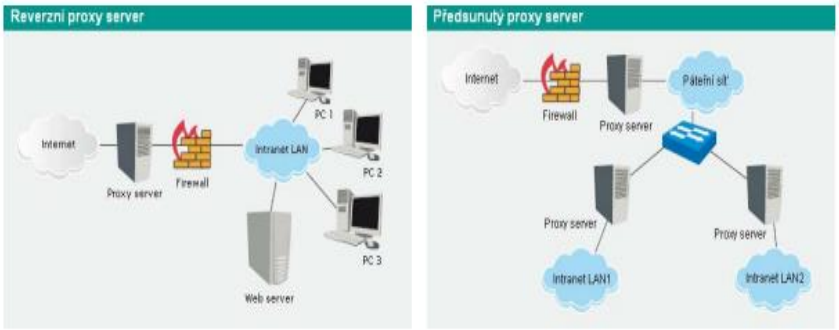
Používá se především pro protokoly HTTP, HTTPS, FTP, NNTP, IMAP, POP3 a Telnet.

Reverzní proxy

Reverzní proxy využívají jinou technologii proxy serverů. Můžeme je využít vně firewallu, kde reprezentují bezpečný obsahový server určený pro vnější klienty a zabraňují přímému, nesledovanému přístupu k datům na vnitřních serverech z vnější sítě. Reverzní proxy také urychlují činnost sítě, protože před silně vytížený server můžeme umístit i několik proxy serverů a zátěž mezi nimi vyrovnávat. Dopřednou a reverzní aplikaci může zajišťovat stejný proxy server.

Je patrné, že složitý obvod sítě s různými proxy servery, tunely a paketovými filtry, které pracují v obou směrech, se může velmi rychle zkomplikovat a znepřehlednit. V diagramech propojení sítě je proto vhodné zavést si jednoduchou konvenci a každý z typů serverů označit např. následovně:

- C Client (Klient) – systém, který vyžaduje určitou službu
- S Server – systém s požadovanou internetovou službou
- L Listener (Posluchač) – vnitřní strana proxy serveru, která očekává spojení od klientů
- I Iniciátor – vnější strana proxy serveru, která navazuje spojení s vlastním serverem



Obr. 15: Ukázka reverzního a předsunutého proxy serveru

SOCKS

SOCKS je souprava proxy nástrojů, která umožňuje zprostředkovanou proxy komunikaci aplikací bez nutnosti vytváření zvláštních proxy modulů. Hostitel na jedné straně serveru SOCKS může přistupovat k hostitelům na druhé straně serveru bez potřeby přímé konektivity IP. Server SOCKS přesměruje spojení i požadavky služeb mezi hostiteli; zajišťuje přitom autentizaci a autorizaci požadavků, zavádí zprostředkované proxy spojení a přeposílá data mezi oběma propojenými hostiteli.

Program SOCKS má dva samostatné konfigurační soubory. První definuje povolené typy přístupu a druhý směruje požadavky služeb na odpovídající proxy server. Dále je zde možné identifikovat IP adresy a uživatele pro filtrování.

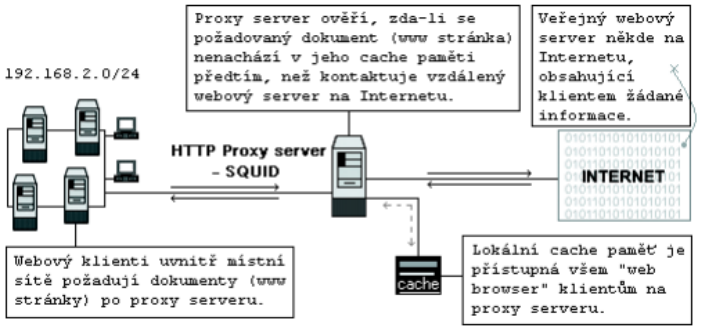
Aplikace, které mají s proxy serverem SOCKS spolupracovat, musí být zvlášť upravené. Pro každou z podporovaných služeb jsou potřeba hned dvě aplikace – například dva různé programy Telnet, z nichž

jeden zajišťuje přímou komunikaci a druhý komunikaci přes proxy server SOCKS. Instrukce pro nezbytné úpravy najdeme přímo u serveru SOCKS. Určité úpravy jsou potřebné v každé aplikaci; klientský software pak komunikuje přes server SOCKS. Celý SOCKS se skládá ze dvou součástí, a sice ze serveru SOCKS a klienta SOCKS. Server SOCKS je implementován na aplikační úrovni, zatímco klient SOCKS pracuje mezi aplikační a transportní vrstvou.

Cache pro obsah (WWW)

Některé typy aplikačních proxy serverů (např. HTTP proxy) mohou disponovat cache pamětí, ve které se určitý čas uchovávají odpovědi vzdálených serverů a obsah této paměti je k dispozici pro další požadavky. Jestliže proxy server najde požadovanou odpověď v cache paměti, může odpovědět klientovi rychleji, než kdyby musel odpověď znovu získat. To je výhodné zejména v případě, že připojení na Internet není velmi rychlé, zatímco přenosová rychlost lokálních sítí dnes běžně dosahuje 100 Mbit/s i více.

Mohlo by se zdát, že hlavní výhodou je tedy kratší čas odezvy na požadavky klienta, což je i pravda, ale pouze v případě, že se hledaná odpověď nachází v cache paměti proxy serveru. Její velikost je totiž vždy omezená (kromě fyzické paměti se používá i pevný disk). Aby se do cache mohly ukládat stále nové údaje, musí se z ní něco odstranit. Na to existuje několik algoritmů, např. odstraní se nejméně používané údaje. Nejrychlejší odezvy na klientský požadavek lze dosáhnout pomocí proxy serveru tehdy, jestliže bude v případě HTTP proxy přistupováno na nejčastěji využívané stránky.



Obr. 16: Proxy cache

Výhoda použití proxy serveru je tedy, kromě již zmíněné bezpečnosti, zmenšení času odezvy (subjektivně „rychlejší načítání“) pro nejčastější požadavky nacházející se v cache paměti. Nevýhody lze zmínit dvě. První spočívá v omezené velikosti cache paměti a druhá v limitu na minimální a maximální velikost cachovaného objektu. Může se stát, že stránky, které je potřeba uchovávat v paměti, nevyhoví zmíněným limitním požadavkům, a bude tedy nutností je vždy získat z webového serveru. Pro lepší představu, jak funguje cachující proxy server pro WWW stránky, vyzdvihnu následující orientační model. Požadavkem je URL adresa, kterou klient požaduje od HTTP proxy.

Nachází se webová stránka v cache paměti?

¬ ANO – Je odpověď stále ještě platná? (Pozn. zjišťuje se například čas poslední modifikace stránky)

- Ano – Proxy server vrátí stránku ze svojí cache paměti.
- Ne – Proxy server aktualizuje kopii stránky v cache paměti a vrátí jí klientovi.

¬ NE – Proxy server se pokusí stáhnout webovou stránku z cílového serveru. Podařilo se stránku získat?

- Ano – Vyhovuje stránka kritériím pro uložení (velikost, typ, direktivy pro proxy server, apod.)?
  - Ano – Proxy server uloží stránku do cache paměti a vrátí jí klientovi.
  - Ne – Proxy server vrátí stránku klientovi bez uložení v cache paměti.
    - Ne – Proxy server vrátí klientovi hlášení o chybě.

Proxy servery s cache pamětí se používají téměř výhradně na uchovávání obsahu WWW stránek.