1.5 Bezpečnost počítačových sítí

Zajištění bezpečnosti počítačových sítí, která byla ještě donedávna dosti opomíjena, se v dnešní době stává prakticky samozřejmostí. To do značné míry souvisí s tím, že použití informačních systémů i počítačem řízených procesů je dnes běžné a na jejich funkčnosti jsou již mnohé instituce přímo závislé. Současně je třeba tyto systémy bránit proti zneužití neoprávněnými osobami, ať už ve smyslu vyřazení z funkce nebo odcizení či modifikace citlivých dat. Toto riziko samozřejmě narůstá v prostředí, kdy prakticky veškeré organizace provozující počítačové sítě jsou alespoň nějakým způsobem připojeny k Internetu.

Přestože se to řada manažérů v IT domnívá, bezpečnost sítí nelze zajistit jednorázově a ani nákupem libovolného drahého a výkonného zařízení. Zajištění bezpečnosti sítě je totiž neustálý proces, který zahrnuje nejen implementaci technických opatření, ale také stanovení a vynucování Konkrétní politiky k používání sítě, včetně sankcí pro uživatele za její za porušení. Jelikož se však požadavky instituce na provozované aplikace a způsoby sdílení informace mezi uživateli neustále vyvíjejí, musí být i bezpečnostní politika organizace periodicky revidována.

Je třeba si uvědomit, že implementace bezpečnostní politiky je vždy pro uživatele omezující. Smyslem rozumného návrhu bezpečnostní politiky je najít vhodný kompromis mezi pohodlím uživatelů a bezpečností sítě tak, aby síť mohla nepřetržitě a efektivně pomáhat v práci svým uživatelům.

Druhým mnohdy opomíjeným faktem je, že problematika bezpečnosti nezahrnuje pouze síťovou infrastrukturu jako takovou (inspekce provozu a filtrace nežádoucích toků, obrana před útoky na síťové prvky) ale i operační systémy koncových stanic. Síťové prvky mohou účinně pomáhat proti šíření virů mezi koncovými stanicemi a také proti útokům typu Denial of Service, kdy se útočník pokouší zahlcením serveru nebo využitím známých slabin v software přivést operační systém nebo software serveru k havárii. Zamezení šíření virů je ovšem důležité i pro správce síťové infrastruktury, poněvadž infikované stanice mohou útočit i na síťovou infrastrukturu, v nejjednodušším případě pokusem o její zahlcení.

Mimo zabezpečení síťové infrastruktury a koncových stanic zahrnuje problematika bezpečnosti sítí samozřejmě i otázky zabezpečení dat přenášených mezi uživateli, které se bude věnovat následující kapitola.

1.5.1 Základní pojmy

Data přenášena sítí mezi uživateli nebo síťovými službami procházejí mnohými linkami a síťovými prvky, o jejichž provozovatelích komunikující strany nic nevědí. Proto je často potřeba data při přenosu šifrovat. Také budeme chtít zajistit, aby u dat nemohla být podvržena identifikace jejich odesílatele. Ujednoťme si nejprve termíny používané v oblasti krypografie, tedy vědy zabývající se utajením dat při jejich přenosu.

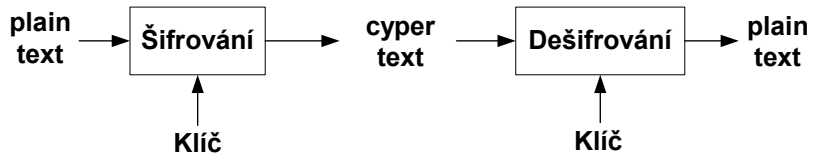
1.5.1.1 Utajení, integrita a autentizace

Pojmem **utajení** (angl. confidentality) rozumíme přenos dat takovým způsobem, že cizí posluchač naslouchající na přenosovém kanále významu přenášených dat nerozumí. **Autentizace** (angl. authentication) zdroje dat dává příjemci jistotu, že odesílatel dat je skutečně tím, za koho se vydává. Zajištěná **integrita dat** (data integrity) dává příjemci jistotu, že data nebyla na cestě nikým zmodifikována.

Pro praktické aplikace je také často užitečná **nepopiratelnost** (angl. non-repudiation), což je mechanismus zajišťující, že zdroj dat nemůže popřít, že jistá konkrétní data odeslal.

1.5.1.2 Symetrický a asymetrický kryptografický systém

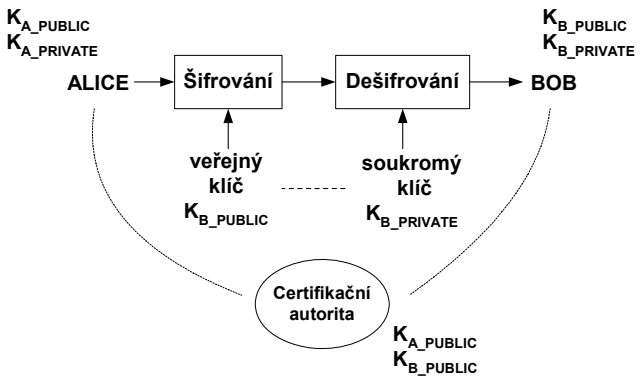
Kryptografickým systémem rozumíme systém, který na straně odesílatele dat šifruje zprávu (angl. plain text), přenáší ji zašifrovanou (angl. cyphertext) a na straně příjemce původní zprávu dešifruje (obr. 1.29). Šifrování lze principiálně realizovat dvěma způsoby. Prvním z nich je vyvinout konkrétní šifrovací algoritmus a ten přede všemi mimo příjemce utajit. V případě prozrazení je však (možná i hardwarová) implementace takovéhoto systému k ničemu. Proto se zpravidla používá druhý způsob a to použít (i veřejně známý) algoritmus, jenž však bude parametrizován pomocí klíčů, které jediné musí zůstat utajeny. Podmínkou je pouze to, aby bylo dost možných klíčů, aby útočník nebyl schopen jednoduše vyzkoušet všechny z nich.



Obr. 1.29 – Šifrování a dešifrování informací během přenosu

S ohledem na způsob zacházení s klíči rozlišujeme kryptografické systémy symetrické a asymetrické. U symetrického systému je použit sdílený klíč, totožný na vysílači a přijímači. Existuje řada rychlých, efektivních a i hardwarově dobře implementovatelných symetrických algoritmů (DES, 3DES, AES). Problémem je však distribuce klíčů, aby se klíč používaný vysílačem dostal pouze a jedině k zamýšlenému příjemci. Toto je třeba typicky zajistit osobním předáním nebo jiným dostatečně bezpečným způsobem nesouvisejícím se sítí přenášející šifrovaná data.

Naopak v asymetrickém systému se klíče generují jako doplňující se pár - veřejný (public) a soukromý (private) klíč. Jeden z těchto klíčů je vždy použit pro šifrování, druhý pro dešifrování. Je lhostejné, který z nich bude použit k čemu. Výhodou je, že generování páru si může každý uživatel uskutečnit lokálně, privátní klíč si bezpečně ukrýt a veřejný klíč zveřejnit. Na obr. 1.30 je vidět, jak uživatel Alice při přenosu zprávy uživateli BOB zašifrovala zprávu všem dostupným veřejným klíčem uživatele BOB (K_{B_PUBLIC}) a jedině uživatel BOB, vlastnící svůj soukromý klíč ($K_{B\_PRIVATE}$), je schopen tímto soukromým klíčem zprávu dešifrovat.



Obr. 1.30 – Šifrování veřejným a soukromým klíčem

V asymetrickém systému odpadá problém s utajenou distribucí klíčů. Používané algoritmy (např. RSA, El-Gammal) jsou však mnohem složitější, náročnější na výpočty a tudíž pomalejší. Asymetrický systém se tak typicky využívá pro předávání (dynamicky generovaných a periodicky měněných) klíčů pro symetrické šifrování.

I v asymetrickém systému však narážíme na problém s bezpečnou distribucí veřejných klíčů. Obsah klíče sice již nemusí být utajen, ale musíme zabránit jeho podvržení. Pokud by např. na obr. 1.30 místo veřejného klíče K_{B_PUBLIC} podvrhnul zlovolný uživatel C svůj veřejný klíč K_{C_PUBLIC} , byl by C schopen zprávu od A dešifrovat, avšak uživatel B nikoli. Uživatel A by přitom nic nepoznal.

Podvržení veřejně distribuovaných veřejných klíčů se obvykle zabraňuje s použitím tzv. certifikační autority. Certifikační autorita je instituce, které je veřejně důvěřováno a která svým privátním klíčem "podepisuje" certifikáty jednotlivých uživatelů. Certifikátem rozumíme blok dat tvořený jednak veřejným klíčem a jednak identifikací vlastníka veřejného klíče. Pravost podpisu lze ověřit pomocí veřejného klíče certifikační autority, která certifikát podepsala. Jednotlivým uživatelům tak stačí spolehlivým způsobem získat jediný veřejný klíč certifikační autority, který mu již zajistí ověření pravosti veřejných klíčů ostatních uživatelů. Veřejné klíče některých obecně používaných certifikačních autorit bývají typicky vypáleny na originálních instalačních CD a instalovány spolu s operačním systémem.

1.5.1.2.1 Autentizace v symetrickém a asymetrickém systému

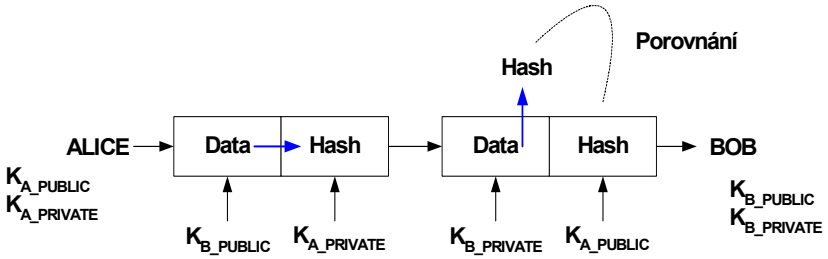
Šifrování v symetrickém i asymetrickém systému je realizováno konkrétním šifrovacím algoritmem. Podívejme se ale nyní, jak lze v obou těchto systémech prakticky zrealizovat autentizaci a na ní se vážící zabezpečení integrity dat.

V symetrickém systému založeném na sdíleném tajemství (hesle) mezi vysílačem a přijímačem můžeme uživatele autentizovat jednoduše tak, že jeho uživatelské jméno zašifrujeme klíčem a na přijímači je opět dešifrujeme a porovnáme se seznamem autorizovaných uživatelů. Pokud nemáme na přijímači správný klíč, dešifrováním vznikne nesmyslný řetězec a ted v seznamu uživatelů nenajdeme. Jestliže chceme pouze ověřit identitu odesílatele, aniž bychom k tomu potřebovali seznam uživatelských jmen na přijímači, můžeme postupovat tak, že na vysílači nejprve ke jménu

uživatele připojíme jeho otisk (hash) a celou zprávu poté zašifrujeme. Přijímač, který zprávu jako celek dešifruje, pak může smysluplnost dešifrované informace ověřit tak, že z dešifrovaného uživatelského jména stejnou hash funkcí jako předtím vysílač vypočte otisk a srovná jeho shodu s dešifrovaným otiskem
Pro výpočet otisku používáme vhodnou jednosměrnou funkci (hash), která z velkého bloku dat vypočte tzv „otisk“ – blok několika bajtů, jenž blok dat charakterizuje. Hash funkce je volena tak, aby nebylo možné najít a podvrhnout jiný text zprávy, který bude mít stejný otisk jako zpráva původní.

Společně s autentizací odesílatele se zpravidla implementuje i zajištění integrity přenášených dat. Zajištění integrity mezi uživateli sdílejícími tajný klíč se realizuje tak, že odesílatel v paměti za zprávu připojí sdílený tajný klíč a z celého bloku **[zpráva+sdílený tajný klíč]** vypočte otisk. Poté vyšle původní zprávu a otisk celé dvojice. Přijímač pak za došlou zprávu v paměti připojí sdílený klíč a stejnou hash funkcí jako vysílač vypočte otisk. Ten poté srovná s otiskem, který spolu se zprávou vyslal vysílač.

V asymetrickém systému se autentizuje poněkud odlišným způsobem, na kterém jsou mj. založeny i elektronické podpisy. Princip je vidět z obr. 1.31. Uživatel Alice chce poslat zprávu uživateli Bob. Z dat nejprve vypočte otisk a ten zašifruje svým soukromým $K_{A\_PRIVATE}$. Data zprávy zašifruje veřejným klíčem Boba, který jako jediný může zprávu dešifrovat svým soukromým klíčem $K_{B\_PRIVATE}$. Mimo to Bob prověří, zda zprávu vyslala Alice tak, si z dešifrované přijaté zprávy sám spočítá otisk a dešifrovaný otisk od Alice dešifruje všem dostupným veřejným klíčem Alice K_{A_PUBLIC}. Dešifrování proběhne úspěšně pouze v případě, že byl otisk zašifrován soukromým klíčem Alice, který vlastní jedině ona. A pouze v případě úspěšného dešifrování otisku se bude dešifrovaný otisk zprávy shodovat s otiskem vypočteným Bobem a Alice bude považována za autentizovaného odesílatele přijaté zprávy.



Obr. 1.31 – Autentizace v asymetrickém systému

1.5.2 Bezestavová a stavová filtrace provozu

Zabezpečení operačních systémů a na nich běžícího software před útoky nebo neoprávněným přístupem v prostředí počítačové sítě se nejjednodušeji řeší vhodnou filtrací nežádoucího provozu. Filtrace může probíhat buďto na úrovni inspekce jednotlivých paketů za použití tzv. paketových filtrů, výsledkem jejichž vyhodnocení je vždy propuštění nebo zahození kontrolovaného paketu. Hovoříme zde o filtraci bezestavové, jelikož jednotlivé pakety jsou kontrolovány nezávisle na sobě, aniž by se filtrující zařízení (typicky směrovač) snažil rekonstruovat datový proud nesený pakety,

které k sobě logicky patří. Bezestavová filtrace se tedy děje pouze na základě dat obsažených v paketu, nikoli podle „stavu“ logického spojení zprostředkovávaného jednotlivými pakety.

Náročnější variantou filtrace je filtrace stavová, při které si filtrující zařízení z procházejících paketů rekonstruuje obsah jednotlivých datových toků. Tímto zařízením může být buďto proxy server, nakonfigurovaný v jednotlivých komunikujících zařízeních jako prostředník-firewall, nebo může být filtrující zařízení pro komunikující účastníky zcela transparentní (typicky realizováno jako inteligentní most). Stavová filtrace dovoluje filtrovat na základě chování aplikačních protokolů na 7. vrstvě OSI RM, avšak platí za to omezenou škálovatelností, kdy pro každé jednotlivé spojení je třeba udržovat v paměti jeho stav a procesor musí nahlížet nejen do hlaviček, ale i do aplikačních dat.

1.5.2.1 Bezestavová filtrace pomocí ACL

Implementačně nejjednodušší metodou filtrace je inspekce jednotlivých procházejících paketů bez ohledu na ostatní pakety, s nimiž společně tvoří jeden datový tok. Výhodou je, že není třeba procházející pakety rozřazovat do datových toků a udržovat informaci o každém z nich, nevýhodou zase skutečnost, že můžeme kontrolovat pouze korektnost dat zcela obsažených v jednom paketu, nikoli tedy nebezpečná data rozložená do více samostatných paketů[1].
Nejčastější forma implementace bezestavové inspekce paketů jsou tzv. Access Control Lists (ACL). ACL je v podstatě filtr umístěný na některé rozhraní aktivního prvku. Zpravidla se jedná o směrovač nebo L3 přepínač, i když i některé přepínače dovolují vložit na port nebo VLAN ACL filtrující podle zdrojové nebo cílové MAC adresy, popřípadě typu protokolu 3. vrstvy. Dále se budeme zabývat jen ACL na směrovačích, které budou kontrolovat informace ze 3., popřípadě 4. vrstvy OSI RM.

ACL je sekvence položek zakazujících nebo povolujících průchod paketů vyhovující kritériím definovaným danou položkou. Při průchodu paketu rozhraním, na němž je ACL aplikován, se postupně odshora procházejí jednotlivé položky ACL, až se narazí na první položku, jejímž kritériím zkoumaný paket vyhovuje. Podle toho, zda je položka definována jako příkaz k propuštění vyhovujícího paketu nebo k jeho zahození se pak paket propustí nebo zlikviduje. Další položky ACL se pak už dále nezkoumají. Na konci ACL typicky bývá implicitní položka přikazující „zahodit vše“, vlivem čehož ACL respektují filosofii „co není explicitně povoleno, je zakázáno“.

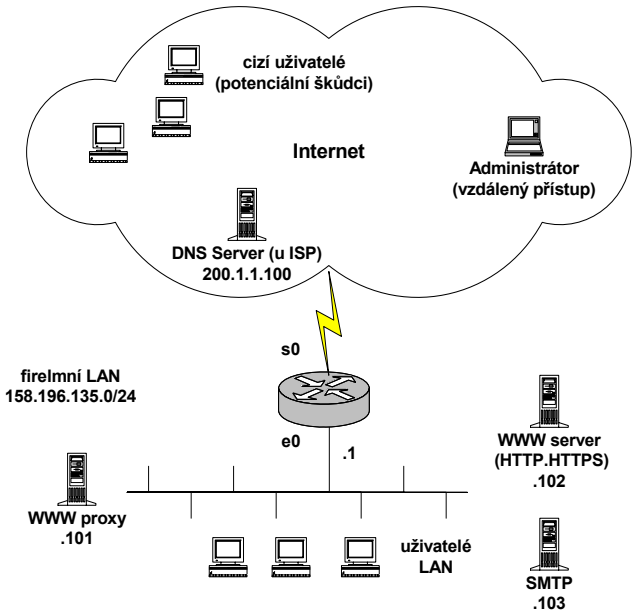
Při návrhu filtrace provozu pomocí ACL je vždy třeba stanovit

- na kterém rozhraní kterého směrovače bude ACL aplikován
- zda bude filtrovat provoz vstupující do tohoto rozhraní nebo z tohoto rozhraní vystupující
- jaká kritéria způsobující propuštění nebo zahození procházejících paketů bude ACL obsahovat

[1] Nebezpečí pro úspěšné použití bezestavové inspekce plyne i z fragmentace.V hlavičce fragmentů s výjimkou prvního totiž není záhlaví 4. vrstvy, takže podle něj nelze filtrovat. Lze to řešit např. timeoutem na propouštění fragmentů se shodnou hodnotou v poli Identification spouštěným prvním fragmentem, ale to již nese potřebu ukládat stavovou informaci. Častá implementace kontroluje 4. vrstvu jen je-li v IP hlavičce Fragment Offset=0 (tedy u prvního fragmentu), ostatní fragmenty propouští. To přináší nebezpečí, pokud první fragment úmyslně nepřenáší celou hlavičku 4. vrstvy.

Zabezpečení sítě není zpravidla možné zajistit pouze jedním ACL, ale kombinuje se funkčnost několika různých ACL umístěných na vhodná rozhraní. Na každé rozhraní lze typicky umístit vždy jeden ACL filtrující provoz vstupující do rozhraní a jeden ACL filtrující provoz vystupující.

Při aplikaci bezestavové filtrace zpravidla postupujeme tak, že nejprve analyzujeme aplikace, které v síti hodláme podporovat. Definice požadavků v konkrétním prostředí může vypadat například takto (viz obr. 1.32):



Obr. 1.32 – Příklad bezpečnostní politiky realizované pomocí ACL

- Firma provozuje svůj vlastní poštovní server (SMTP) přístupný zvenčí na adrese 158.196.135.103.
- Firma provozuje svůj vlastní WWW server (HTTP, HTTPS) přístupný zvenčí na adrese 158.196.135.102.
- Přístup lokálních klientů ke službě WWW (HTTP i HTTPS) jde výhradně přes proxy server s adresou 158.196.135.101.
- Ze stanic na LAN lze do Internetu otevírat pouze spojení SSH.
- DNS server provádějící rekurzivní vyhledávání jmen pro všechny klienty na LAN je u poskytovatele Internetu (ISP) na adrese 200.1.1.100.
- Je povolen ping z LAN do Internetu, v opačném směru však z bezpečnostních důvodů nikoli.
- Vzdálený administrátor je schopen připojit se odkudkoli z Internetu na počítač s WWW serverem pomocí služby SSH.

U každé aplikace přesně zjistíme, jaký protokol transportní vrstvy a jaké porty tohoto protokolu používá. Komplikaci přinese případ, kdy zjistíme, že aplikace si porty volí dynamicky a zvolená čísla portů předává účastníkům komunikace v aplikačním protokolu.

Následně rozhodneme, na kterém rozhraní směrovače chceme provoz filtrovat. V typickém případě používáme jeden ACL pro filtraci provozu dovnitř firemní sítě a jiný pro filtraci provozu ven Následně definujeme obsah jednotlivých ACL s tím, že nesmíme zapomínat na povolení zpětného směru provozu. Obvykle je dobré vycházet ze zásady, že implicitně je vše zakázáno a pouze provoz protokolů explicitně vyjmenovaných v ACL je povolen. Zvláštní pozornosti musíme dbát, pokud ACL aplikujeme na směrovači při jeho vzdálené správě, abychom neodfiltrovali protokol, pomocí něhož směrovač vzdáleně konfigurujeme.

U situace vyobrazené na obrázku 1.32 by analýza používaných protokolů mohla vypadat např. takto:

Služba (aplikační protokol)	**Protokol**	**Port serveru**
HTTP	TCP	80
HTTPS	TCP	443
SMTP	TCP	25
DNS	UDP	53
ping	ICMP	Zprávy Echo request a Echo reply

Žádná z použitých služeb nevyužívá dynamicky přidělované serverové porty. Naopak porty klientů jsou vždy přidělovány dynamicky operačním systémem, takže jejich hodnotu ACL nekontroluje.

Dále bychom zavedli dva ACL, označili je a přiřadili k nim rozhraní a směr toku, který budou filtrovat. V našem případě označíme číslem 101 ACL na rozhraní s0 filtrující provoz vstupující do tohoto rozhraní, tedy do vnitřní sítě. Číslem 102 označíme ACL filtrující provoz vstupující do rozhraní e0. Filtraci bychom tedy prováděli ještě před tím, než necháme směrovač vyhledávat cílové rozhraní při směrování přicházejících paketů.
Obsah ACL 101 a 102 můžeme vyjádřit bez ohledu na syntaxi používanou konkrétní implementací ACL tabulkami podobnými Tab. 1.1 a Tab. 1.2.

Tab. 1.1 Filtrace provozu přicházejícího z Internetu - ACL 101 (s0, in)

Pořadí položky	Povolení / zákaz	Protokol	Zdrojová IP adresa	Zdrojový port	Cílová IP adresa	Cílový port	poznámka
1	zakázat	IP	158.196.135.0/24		*		anti-spoofing filtr (podvržení src IP)
2	povolit	TCP	*	*	158.196.135.103	25	SMTP do LAN
3	povolit	TCP	*	*	158.196.135.102	80	HTTP do LAN
4	povolit	TCP	*	*	158.196.135.102	443	HTTPS do LAN
5	povolit	TCP	*	*	158.196.135.102	22	SSH do LAN na stroj WWW serveru
6	povolit	UDP	200.1.1.100	53	158.196.135.0/24	*	DNS odpovědi
7	povolit	ICMP	*		158.196.135.0/24		odpovědi ping (zpráva Echo reply)
8	povolit	TCP	*	80	158.196.135.101	*	odpovědi pro HTTP proxy
9	povolit	TCP	*	443	158.196.135.101	*	odpovědi pro HTTPS proxy
10	povolit	TCP	*	22	158.196.135.0/24	*	odpovědi SSH klientům
11	zakázat	IP	*		*		zákaz ostatního provozu

Tab. 1.2 – Filtrace provozu odcházejícího z LAN - ACL 102 (e0, in)

Pořadí položky	Povolení / zákaz	Protokol	Zdrojová IP adresa	Zdrojový port	Cílová IP adresa	Cílový port	poznámka
1	povolit	TCP	158.196.135.101	*	*	80	HTTP z WWW proxy do Internetu
2	povolit	TCP	158.196.135.101	*	*	443	HTTPS z WWW proxy do Internetu
3	povolit	TCP	158.196.135.0/24	*	*	22	SSH do Internetu
4	povolit	UDP	158.196.135.0/24	*	200.1.1.100	53	DNS dotazy
5	Povolit	ICMP	158.196.135.0/24		*		dotazy ping (Echo request)
6	povolit	TCP	158.196.135.103	25	*	*	odpovědi cizím SMTP klientům
7	povolit	TCP	158.196.135.102	80	*	*	odpovědi cizím klientům HTTP
8	povolit	TCP	158.196.135.102	443	*	*	odpovědi cizím klientům HTTPS
9	povolit	TCP	158.196.135.102	22	*	*	odpovědi administračního SSH
10	zákaz	IP	*		*		zákaz ostatního provozu

Implementaci výše uvedeného příkladu ve formě ACL pro operační systém Cisco IOS lze najít v [6].

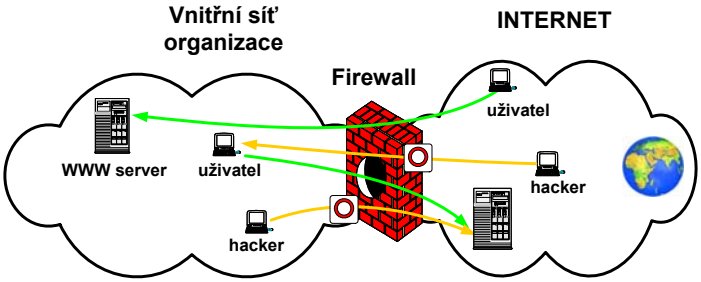
Při návrhu ACL je třeba mít neustále na zřeteli, že nestačí povolit datový tok povoleného aplikačního protokolu pouze ve směru k serveru, ale i „odpovědi“ ve směru opačném. Při tom si je

třeba vždy ujasnit, že čísla zdrojového a cílového portu budou pro zpětný směr přehozená. Nerespektování tohoto faktu je nejčastější chybou při konfiguraci ACL.

Implementace ACL mohou být nejrůznějším způsobem vylepšovány. Časté a užitečné rozšíření je například uvedení časového intervalu, ve kterém má konkrétní položka ACL platit. Tím je např. možné v pracovních hodinách zakázat přístup na WWW servery nesouvisející s pracovní činností a v mimo ně přístup umožnit.

1.5.2.2 Stavová filtrace pomocí firewallu

Stavovou filtraci na základě nedovolených aktivit detekovaných v datových tocích aplikačních protokolů realizuje zpravidla firewall (obr. 1.33). Ten může být pro komunikující strany transparentní nebo ve funkci proxy serveru.



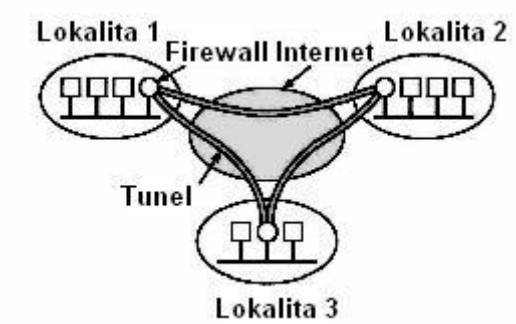
Obr.1.33 – Funkce firewallu

Firewall oddělují důvěryhodnou a nedůvěryhodnou část sítě, zkoumá datové toky mezi nimi a podle nakonfigurovaných pravidel je dovoluje nebo naopak zakazuje. Implementace firewallu může být buďto „hardwarová“ (specializované zařízení) nebo „softwarová“ s použitím vhodného operačního systému (Linux, NetBSD, …)
Firewally se často implementují ne pouze se dvěma rozhraními jako na obr. 1.33, ale s dalším třetím rozhraním, ke kterému je připojena tzv „demilitarizovaná zóna“ (DMZ). V DMZ se typicky vyskytují servery, které mají být přístupny jak z vnitřní sítě, tak z Internetu. Tyto servery (tzv. „bastillon hosts“) mají řádně zabezpečený operační systém, aby nemohlo dojít k jejich napadení. Na servery v DMZ smějí přistupovat jak uživatelé z vnitřní sítě, tak uživatelé z Internetu. Přímý přístup uživatelů z Internetu do vnitřní sítě je však zakázán.

1.5.3 Virtuální privátní sítě

Virtuální privátní sítě (VPN) jsou mechanismem, umožňujícím organizacím budovat „privátní“ sítě s použitím sdílené infrastruktury, např. veřejného Internetu. Při tom je ale dosaženo stejné úrovně Flexibility a bezpečnosti, jako při použití vlastní infrastruktury. Princip VPN spočívá v tunelování šifrovaných dat přes sdílenou infrastrukturu za použití autentizace mezi jednotlivými konci tunelů (obr. 1.34). Tunelem zde rozumíme virtuální dvoubodové spojení přes sdílenou infrastrukturu, které

nese pakety jednoho protokolu zabalené v jiném (nebo i tomtéž) protokolu. Obalujícím protokolem je v Internetu vždy IP, protokolem tunelovaným může být také IP, lze ale přenášet i jiné síťové protokoly (např. Novell IPX) nebo dokonce přímo rámce 2. vrstvy OSI RM.

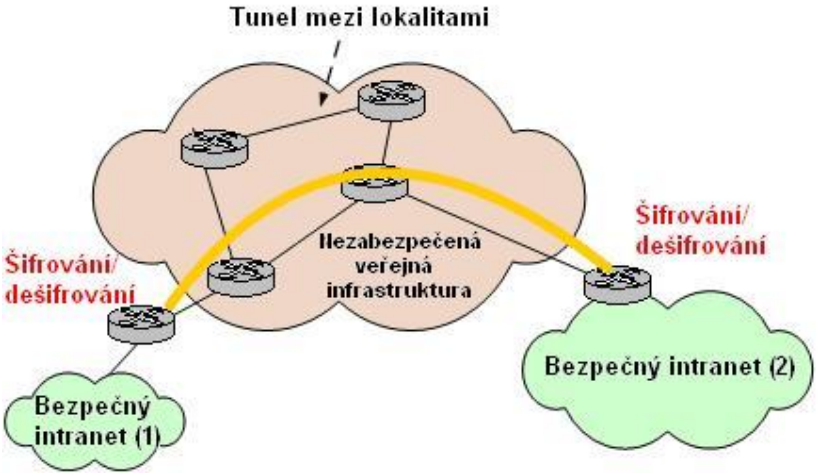


Obr. 1.34: Implementace VPN s použitím firewallů a veřejné infrastruktury

Výhodou použití VPN oproti udržování vlastní privátní infrastruktury je nižší cena, flexibilita (virtuální) topologie dána pouze konfigurací firewallů/směrovačů na koncích VPN tunelů. O realizované virtuální topologii nemusí poskytovatel sdílené infrastruktury nic vědět. Také odpadá potřeba dozoru nad vlastními WAN linkami, který je přenechán poskytovateli veřejné infrastruktury.

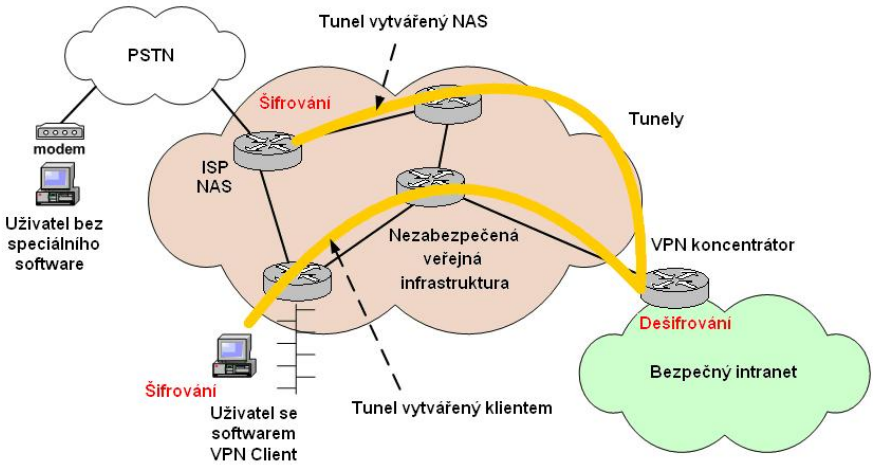
1.5.3.1 Použití VPN

VPN bývají nejčastěji aplikovány ve dvou základních modelech – virtuální propojení sítí a připojování vzdálených uživatelů.
Model s tunelem mezi směrovači (firewally) lokalit propojených s použitím VPN přes veřejný Internet je vidět na obr. 1.35. Obecně lze vytvářet i složitější topologie tunelů mezi více lokalitami.



Obr. 1.35 - VPN propojující dvě odlehlé lokality

Obrázek 1.36 ukazuje připojení vzdálených uživatelů do domovské sítě prostřednictvím směrovače/firewallu ukončujícího tunely od jednotlivých uživatelů, tzv. VPN koncentrátoru. Konec tunelu na straně uživatele je vytvářen pomocí speciálního software, který má uživatel nainstalován – obvykle hovoříme o tzv. VPN klientu.



Obr. 1.36 Použití VPN klienta pro bezpečný přístup do intranetu

Při připojení uživatele přes operátora telekomunikační sítě připadá v úvahu i varianta, že tunel z intranetu organizace nebude ukončen až u uživatele, ale na přístupovém serveru (Network Access Server, NAS) realizujícím přemostění spojení od uživatelů sítě operátora do Internetu. Této varianty se však v našich podmínkách zatím příliš nevyužívá.

1.5.3.2 Technologie VPN na 3. a 4. vrstvě modelu OSI RM

Z praktického hlediska lze šifrování a autentizaci využívající principů uvedených v kap. 1.5.1 realizovat na všech vrstvách OSI modelu a to třeba i na více vrstvách současně. Můžeme se např. setkat s šifrováním na 2. vrstvě, to však není pro komunikaci v Internetu příliš efektivní, protože zpráva musí být vždy při každém přeskoku mezi směrovači znovu a znovu šifrována a dešifrována. Naopak šifrování na 7. vrstvě bude sice efektivní v tom, že každá aplikace může zvolit optimální algoritmus s ohledem na charakter přenášených utajovaných dat, avšak nevýhodou bude, že jednotivé aplikace budou muset bezpečnostní mechanismy implementovat opakovaně jako součást svého vlastního kódu. V praxi se také setkáváme s šifrováním na principu vkládání bezpečnostní vrstvy SSL (Secure Sockets Layer) nad protokol TCP, to však není použitelné pro aplikace využívající na transportní vrstvě datagramově orientovaný protokol UDP. Proto se z praktického hlediska jeví výhodné řešit bezpečnost na vrstvě síťové, čímž zajistíme nezávislost na konkrétní síťové technologii a současně i na aplikaci. Standard, který řeší šifrování, autentizaci a zabezpečení

integrity dat u protokolu IP se nazývá IPSec a je dnes chápán jako vcelku široce implementované rozšíření protokolu IPv4.

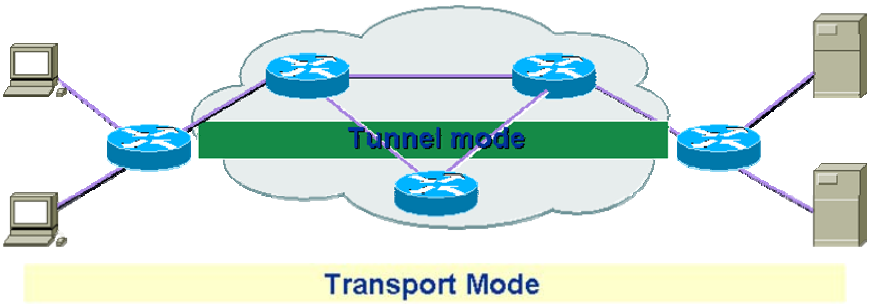
1.5.3.2.1 IPSec

IPSec (RFC2401) je architektura pro zajištění autentizace, integrity dat a šifrování, technicky realizovanou pomocí dynamicky navazovaných zabezpečených tunelů na síťové vrstvě OSI RM. Není závislá na konkrétních algoritmech, jedná se spíše o obecný framework, v rámci něhož se mezi konci tunelů konkrétní šifrovací a autentizační algoritmy dynamicky dohadují.
Soubor parametrů dohodnutých pro šifrování a autentizaci mezi konci tunelu se nazývá bezpečnostní asociace (Security Association, SA). SA obsahuje informaci o dohodnutých šifrovacích a autentizačních algoritmech a příslušné klíče. Pro každý směr provozu je dohodnuto zvláštní SA a každá SA má omezenou životnost, takže se bezpečnostní parametry periodicky mění, což ztěžuje potenciálním útočníkům odposlech.
Technicky se dynamická dohoda SA mezi konci tunelu realizuje pomocí protokolu IKE (Internet Key Exchange), což je konkrétní implementace protokolu pro bezpečnou dohodu klíčů vyhovující obecnému rámci pro protokoly tohoto určení nazývanému ISAKMP (Internet Security Association and Key Management Protocol). Parametry pro autentizaci a šifrování jednotlivých paketů se přenášejí přímo v těchto paketech v pomocných hlavičkách, nazývaných Authentication Header (AH) a Encapsulating Security Payload (ESP). AH zabezpečuje autentizaci odesílatele a chrání integritu IP hlavičky i přenášených dat a také obsahuje ochranu před útoky zachycením provozu a jeho zopakovaným přeposíláním (tzv. replay útoky). Novější ESP hlavička může zajistit všechny tyto funkce a navíc šifrování.

Je samozřejmé, že IP hlavička, kterou potřebují směrovače sítě pro přenos paketu k cíli, nemůže být šifrována. Také při použití autentizaci IP hlavičky si musíme uvědomit, že nemůže být použita spolu s mechanismy, které hlavičku modifikují, například s překladem adres (NAT).

IPSec může pracovat v jednom ze dvou režimů – transportním a tunelovém (obr. 1.37). Transportní režim zajišťuje bezpečnost mezi koncovými zařízeními a vyžaduje podporu operačních systémů koncových stanic. Hlavičky AH, resp. ESP jsou zde vkládány mezi hlavičky 3. a 4. vrstvy, takže data od transportní vrstvy výše jsou šifrována. Originální IP hlavička samozřejmě šifrována není, ale je chráněna mechanismem autentizace a zabezpečení integrity dat.

Častěji se můžeme setkat s režimem tunelovým, kdy je zřízen tunel mezi směrovači (tzv. IPSec branami) připojujícími LAN považované na bezpečné na nebezpečnou sdílenou infrastrukturu. IPSec brány tunelují a detunelují pakety z koncových zařízení, která nemusí o IPSec vůbec vědět. Tunelují se celé pakety z lokálních sítí včetně hlaviček, takže není ani možné odposlouchat, které konkrétní stanice z tunelem propojených sítí spolu komunikují.



Obr. 1.37 – Tunelový a transportní režim IPSec

Nevýhodou IPSec je, že jej lze použít pouze pro protokol IP a nepodporuje ani multicast provoz. To však lze v některých případech technicky obejí tak, že provoz jiných protokolů nebo provoz multicastový před předáním do IPSec jednoduše zabalíme do protokolu IP.

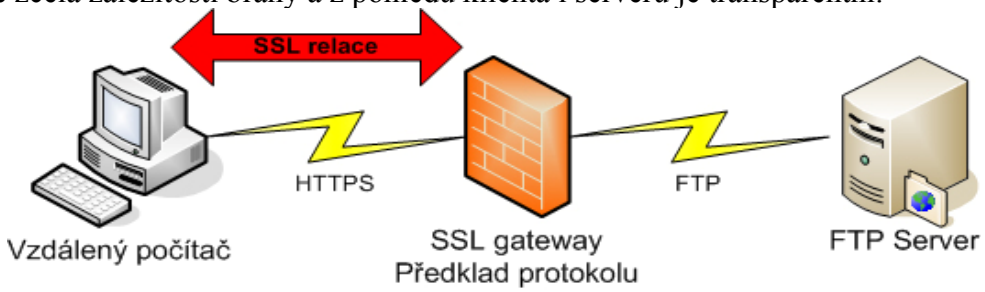
Druhou nevýhodou je, že provoz IPSec nesmí omezovat operátor sdílené infrastruktury. Vložení hlavíček AH a ESP je v originální IP hlavičce indikováno číslem protokolu vyšší vrstvy 51, resp. 50 a původní číslo protokolu transportní vrstvy je uvedeno až v přídavné hlavičce. Také provoz protokolu IKE vyžaduje průchodnost datagramů UDP na portu 500. Operátor podporující IPSec proto musí implementovat případné paketové filtry tak, aby provoz těchto protokolů neomezil.

1.5.3.2.2 SSL VPN

Nejnovějším typem VPN je technologie založená na kombinaci symetrické a asymetrické kryptografie, nejčastěji nazývaná SSL VPN (Secure Socket Layer Virtual Private Network) [7].
Termínem SSL VPN je dnes označována řada často vzájemně nekompatibilních technologií, nicméně všechny jsou postaveny na stejné základní myšlence. Je jí využití asymetrické kryptografie a knihoven SSL (Secure Socket Layer) pro šifrovanou komunikaci. Protože SSL VPN pracují na transportní vrstvě, je jejich použití pro síťovou infrastrukturu (a tedy i poskytovatele Internetu) transparentní.
Prakticky se SSL VPN technologie realizuje nejčastěji jako SSL brána (obr. 1.38), což úplně neodpovídá definici VPN. SSL brána je zařízení mezi vnitřní síti obsahující standardní služby a vzdáleným klientem realizujícím šifrování svého připojení pomocí protokolu SSL. K využití SSL VPN často není třeba žádný specializovaný software na straně klienta, protože se vzdálenou SSL bránou je možné komunikovat i standardním HTTPS protokolem a tedy přes běžný WWW prohlížeč. SSL brána emuluje obě protistrany jak klientovi tak i serveru. Provádí dešifrování příchozím požadavků a šifrování odpovědí serveru předtím, než je odešle zpět klientovi. SSL relace existuje tedy mezi SSL bránou a vzdáleným počítačem.

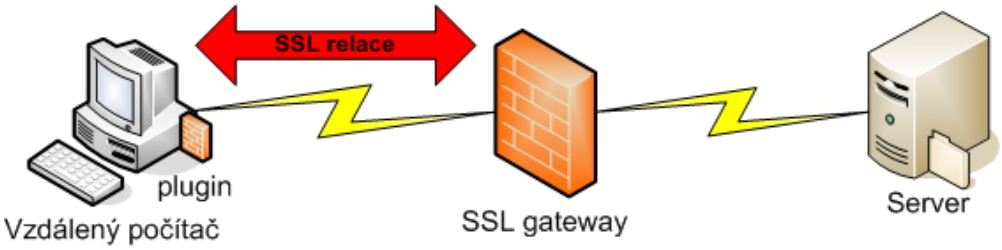
Obr. 1.38 – SSL VPN se SSL bránou

Konkrétní funkce SSL brány se může pro jednotlivé scénáře použití i dosti odlišovat. Pro aplikační protokoly, které je možné adaptovat na protokol HTTP, je někdy výhodné použít SSL brány provádějící překlad aplikačního protokolu (obr. 1.39). Klient této služby je standardní webový prohlížeč s podporou HTTPS a také server může zůstat zcela bez modifikací. Překlad aplikačního protokolu je zcela záležitostí brány a z pohledu klienta i serveru je transparentní.



Obr. 1.39 - Překlad aplikačního protokolu

Další variantou implementace SSL brány je použití předávání portů (port forwarding), viz obr. 1.40. Vzdálený klient zde ze SSL brány získá komponentu (Java, ActiveX), která bude ve vzdáleném stroji vystupovat jako VPN SSL Proxy. Klientská aplikace (prohlížeč) bude komunikovat přímo s touto komponentou, jako by se jednalo o lokální proxy server. Lokální proxy server (komponenta) provádí port forwarding. Mění tedy cílovou adresu IP paketu (případně cílový TCP port) v závislosti na původním cílovém TCP portu v paketu zaslaném klientem. SSL relace je tedy navázána přímo mezi komponentou vzdáleného klienta a SSL bránou. SSL brána žádným způsobem nemanipuluje s obsahem zasílaných dat, pouze spravuje mapování portů příchozích SSL relací na cílové adresy a porty serverů a jejich aplikací. Problémem při této koncepci jsou služby, které nepoužívají jeden fixní port, ale rozsah dynamicky přidělovaných portů (např. služba FTP).



Obr. 1.40 – Předávání portu

1.5.4 Útoky na počítačové sítě

Útoky na počítačové sítě mohou mít velmi různorodý charakter. Rozlišujeme pokusy o průnik, kdy se útočník dostává k zařízením, ke kterým měl být provoz od něj filtrován a útoky, kdy se útočník pokouší úplně zničit, případně ve svůj prospěch modifikovat chování sítě nebo serveru poskytujícího jistou službu.

1.5.4.1 Útoky typu Denial of Service

V dnešní době se nejčastěji setkáváme s útoky, jejichž cílem je zhroucení infrastruktury oběti a zamezení funkčnosti jí poskytované služby. Proto hovoříme o útocích typu „Denial of Service“ (DoS). Cílem je zpravidla diskreditovat provozovatele služby před svými uživateli (zákazníky), což oběti může přinést nemalé finanční ztráty.

Cílem útočníka provádějícího DoS útok je vyčerpání systémových prostředků síťového prvku nebo serveru a jeho zhroucení nebo změna požadovaného chování ve prospěch útočníka (např. zhroucení systému obrany proti útočníkovým dalším aktivitám). Systémovými prostředky, o jejichž vyčerpání může být usilováno, jsou typicky paměť, procesorový čas nebo šířka pásma jistého segmentu sítě.

DoS útoky zpravidla přicházejí z podvržených zdrojových adres, aby útočníci obešli případné paketové filtry. Nejnebezpečnější jsou DoS útoky v distribuované variantě (Distributed Denial of Service Attacks, DDoS), kdy na útoku spolupracuje velké množství útočníků rozmístěných na různých místech Internetu. Nebezpečí DDoS útoků spočívá hlavně v tom, že charakter útočného provozu se mění rychleji, než stačí správce reagovat a škodlivý provoz analyzovat a odfiltrovávat.

1.5.4.2 Detekce útoků a obrana proti průnikům

Pro detekci útoků a průniků dnes existuje celá řada systému, obecně nazývaná Intrusion Detection Systems (IDS). Tyto systémy neustále sledují provoz na sítí a na základě podezřelých vzorů chování vyhodnocují stavy, které mohou s jistou pravděpodobností indikovat nějaký typ útoku. Vyhledávání provozu indikujícího útok je věcí dosti komplikovanou a může zahrnovat jak různé heuristiky, tak i prvky umělé inteligence. Způsob reakce na oznámené riziko je v IDS již ponechán na administrátorovi. U systému typu IPS (Intrusion Prevention Systems) je automatizace dotažena dále – při vyhodnocení rizika je automaticky provedeno opatření, aby byl další provoz od útočníka odfiltrován. Typicky se jedná o automatickou definici a aplikaci vhodného ACL.
Systémy pro detekci a prevenci útoků jsou nezbytné zejména v souvislosti s distribuovanými DoS útoky, u kterých již není v lidských silách dostatečně rychle reagovat na měnící se charakter strojově generovaného útočného provozu a aplikovat příslušná protiopatření.

1.5.5 Bezpečnostní mechanismy přepínaných a směrovaných sítí

Zabezpečit síť před útoky je úkol velmi komplexní. Potenciální rizika jsou samozřejmě závislá na použitých technologiích a protokolech a není možné zabývat se jimi v tomto textu v plné šíři. Proto si dále ukážeme jen nejzákladnější možnosti zabezpečení v lokálních sítích založených na přepínačích a ve směrovaných sítích s protokolem IP.

Pro bezpečnost sítě je samozřejmě nejdůležitější zabezpečení managementu síťových prvků, aby neoprávněná osoba nemohla změnit jejich konfiguraci. Je třeba používat vhodné přístupové heslo a vyhnout se manažovacím protokolům, které heslo přenášejí v nešifrované podobě (Telnet, HTTP, SNMP). Je vhodné pomocí ACL specifikovat zdrojové adresy, ze kterých může být management uskutečňován, také bývá užitečné pro management vyhradit samostatnou VLAN. Často opomíjeno

také zůstává zabránění neoprávněnému fyzického přístupu k zařízení, bez něhož jsou ostatní způsoby zajištění bezpečnosti managementu zpravidla k ničemu.

1.5.5.1 Možnosti zabezpečení v přepínaných sítích LAN

V přepínaných lokálních sítích patří mezi základní zabezpečení omezení zařízení vpouštěných do sítě pouze na zařízení autorizovaná. Chceme zpravidla zajistit, že na jistý port přepínače může být připojeno jen určité konkrétní zařízení nebo naopak že jisté zařízení se smí připojit jen na určitou množinu portů. Na mnohých přepínačích lze u portu explicitně vyjmenovat zdrojové MAC adresy, zařízení, které se smí přes daný port připojovat. Často také máme možnost omezit počet MAC adres na portu, což zabraňuje útokům typu source-spoof DoS, u kterých útočník generováním velkého množství rámců s různými podvrženými zdrojovými MAC adresami dosáhne přeplnění přepínací tabulky a následného rozesílání rámců na všechny porty, kde mohou být odposlouchávány. U některých přepínačů také můžeme využít dynamického zařazování zařízení do VLAN na základě zdrojové MAC adresy, popsaného v kapitole 1.1.2.2.

Dalším typickým způsobem zabezpečení je omezení provozu mezi porty přepínače. Na port může být u některých přepínačů aplikován ACL, který filtruje podle zdrojové či cílové MAC adresy, někdy i podle informací ze síťové a transportní vrstvy. ACL lze také někdy aplikovat na veškerý provoz procházející jistou VLAN jako celkem.

Užitečná je také možnost zákazu vzájemné komunikace mezi klientskými porty, kdy je klientům dovolen přístup pouze na porty se servery nebo vedoucí do páteřní sítě. Tím se zabrání nežádoucím aplikacím typu peer-to-peer často nelegálně provozovanými mezi uživateli (hry, chat, DirectConnect apod.).

Vpouštění do sítě na základě identifikace konkrétního hardwarového zařízení je nevýhodné v tom, že nezabrání vpuštění útočníka, který oprávněné zařízení zcizil. Jedno zařízení také může být legálně sdíleno několika uživateli. Proto je často výhodné vázat oprávnění k přístupu ne na konkrétní zařízení, ale na uživatele s tímto zařízením pracujícího. Uživatel se v takovém případě před připojením do sítě musí autentizovat různými formami hesel nebo certifikátů, které jsou přístupovému přepínači předány pomocí k tomu určeného protokolu IEEE 802.1x. Přístupový přepínač autentizační parametry uživatele ověří u autentizačního serveru (typicky RADIUS-Remote Access Dial-In User Server) a pouze v případě pozitivního výsledku autentizace provoz z portu, kterým je uživatel připojen, do sítě vpustí.

Dosti užitečné je také chránit v přepínaných sítích protokol Spanning Tree. Pro útočníka totiž často může být výhodné přitáhnout k sobě kořen stromu, protože přes ten bude procházet velká část provozu. Na portech, kde mají být připojeny pouze klientské stanice, je tedy vhodné filtrovat vstupující BPDU, aby tyto nemohly chování algoritmu Spanning Tree nežádoucím způsobem ovlivnit.

1.5.5.2 Základní zabezpečení v intranetech a WAN s protokolem IP

V sítích s protokolem IPv4 přináší velké riziko možnost zneužití protokolu ARP. Jelikož MAC adresy strojů, jimž má být zaslán paket s určitou IP adresou jsou hledány dynamicky, existuje zde riziko falešných odpovědí na ARP dotaz, jimiž může stanice na sebe stáhnout provoz určený pro stanici jinou. Zabránit tomu lze pouze tak, že protokol ARP zcela vypustíme a na celém segmentu vložíme do ARP cache záznamy pro všechny komunikující stanice staticky. Jelikož toto není příliš praktické a v případě jakékoli změny v připojených stanicích je třeba ARP cache všech strojů rekonfigurovat, setkáváme se se statickým vkládáním záznamů do ARP obvykle jen u směrovačů – výchozích bran segmentů LAN.

V sítích, ve kterých je použit dynamický směrovací protokol, je dobré se chránit také proti falešné směrovací informaci generované útočníkem. V některých směrovacích protokolech (RIPv2, OSPF, EIGRP, BGP) mohou být zdroje směrovací informace (sousedé) autentizováni pomocí hesla nebo ještě lépe pomocí otisku zprávy, na směrovači lze také aplikovat ACL definující adresy, ze kterých jsou směrovací informace akceptovány. Cesty propagované sousedy lze před dalším zpracováním také filtrovat.

Samostatnou kapitolou je bezpečnost systému DNS. Jelikož v současné době ještě příliš nejsou implementována rozšíření dovolující autentizovat zdroje informace, existuje zde reálné riziko podvržení informací z DNS a tím i falešného mapování doménových jmen na IP adresy, které může klienty zavést na podvržené servery. Podobně lze podvrhnout například falešné MX záznamy, čímž může být odchozí pošta pro určité domény směřována na podvržené poštovní servery. Je tedy vhodné povolit zasílání odpovědí z DNS pouze z portu, kde je připojen skutečný DNS server.