7. Bezpečnost na Internetu

7.1 Základní problémy bezpečnosti přenosu dat po Internetu

Jakmile připojíte svůj počítač k Internetu, musíte počítat i s tím, že ho připojujete k obrovské počítačové síti, která nepatří právě mezi nejbezpečnější.
Tím prvním a nejvíce viditelným, co ohrožuje váš počítač denně (a nejen ten připojený k Internetu), jsou **viry**. Virus počítačový, stejně jako ten biologický, není schopen samotného života. Parazituje na hostitelské buňce (souboru), množí se a škodí. Klasický virus se připojoval ke spustitelným souborům a do zaváděcí oblasti disků a disket.
Situaci změnily **makroviry**. Ve složitějších programech využívají speciálního programovacího jazyka pro tvorbu maker, prográmků usnadňujících práci s dokumenty. Trvalo poměrně dlouho, než se začaly šířit viry psané v makrojazycích. Infekce však byla o to rychlejší. Kolik pošlete při práci spustitelných souborů? A kolik dokumentů? Dnes jsou stále populárnější skrýší soubory s nápovědou, a u nových virů je seznam typů nakazitelných souborů delší než popis viru. Makrovir je schopen zničit systém stejně vydatně jako běžný vir a může být o to nepříjemnější, že mu nečiní potíže vkládat do textu třeba neslušné výrazy. Podotýkám, že viry ničí data, ne hardware.
Trojští koně žijí samostatné a tváří se jako užitečné programy. Kromě aktivity, která je natolik zajímavá, že se o ni budete chtít podělit, nabízí i škodlivou funkci. Vzhledem k tomu, že se nemusí skrývat, nabízí složité chování a může ve vašem počítači zajišťovat např. vzdálený přístup do počítače, odposlech klávesnice nebo obrazovky a prozradit všechna vaše data včetně hesel. Chování může být natolik složité, že kůň bude odesílat e-maily vašim známým. Bude se snažit vzbudit co nejmenší podezření, posílat sám sebe známým z vašeho adresáře, těm, kteří vám právě poslali poštu, nebo těm, kterým zrovna píšete. Množí se sám prostřednictvím sítě a říká se mu **červ**. Červ není nic jiného než spustitelný soubor, většinou něco.exe nebo něco.doc, neboť červi mohou být také tvořeni makry; mezi makročervy patří např. Melissa.
Proti virům pomůže jedině dobrý antivir. Ten se skládá z několika částí. Pravidelně prohledává soubory na pevných discích, přičemž by neměl zapomenout na prohledávání komprimovaných souborů. Nezapomeňte si aktivovat kontrolu přístupových cest, jako například *Příchozí pošta,* aby se nic nebezpečného do počítače nedostalo nepoznáno. Asi nejdůležitější je prohledávání všech souborů při jakékoli manipulaci, abyste nic zavirovaného nespustili a neodeslali. Antiviry vyhledávají podle řetězců v těle souboru známé viry. Existují sice viry matoucí změnou těla, přesto však platí, že nalezení známého viru zpravidla nebývá problém. Pro hledání nových virů je dobrá heuristická analýza. Bohužel je velmi obtížné - až nemožné - vyladit ji tak, aby našla vše nebezpečné a přitom nehlásila plané poplachy. Antivirové firmy neustále analyzují nové a nové viry a vydávají nové a nové popisy virů. Dobré antiviry si takové aktualizace najdou na Internetu a samy sebe aktualizují. Nemělo by to být méně často než jednou za týden.
Internet znamená hlavně prohlížení stránek. Klasické stránky byly psány v jazyce HTML, který vlastně jen popisuje stránku a je naprosto neškodný. Pak začaly vznikat aktivní prvky, Java a ActiveX. Java a ActiveX mohou být i v e-mailu.
Programy psané v Javě by měly být spustitelné na libovolném počítači s libovolným prohlížečem podporujícím tento jazyk. Prohlížeč se stará o bezpečnost. Kontroluje běh programu a nepovolí mu žádné potenciálně nebezpečné akce typu zápisu na disk. Program vás může požádat o rozšíření pravomocí - nečiňte tak, pokud není podepsán spolehlivou firmou.
Komponenty ActiveX jsou běžné programy spustitelné pouze v prostředích Windows, bez zvláštních bezpečnostních prvků. Při prohlížení internetových stránek se dá bez podpory tohoto prvku bez velké újmy obejít, získáme za to menší riziko napadení našeho počítače.

Pořád se nacházejí chybičky v javascriptu, které umožňují přístup k některým vašim datům. Záleží na konkrétním webovém prohlížeči a zpravidla jsou okamžitě po zjištění odstraňovány.
Stále více programů, které komunikují po Internetu, představuje potenciální bránu do pekel. Za vrata jako hrom je považováno ICQ. Tento populární program umí několik perliček. Uživatelé spolu komunikují nešifrovaně, takže není nejmenší problém je odposlechnout. Horší je, že nešifrovaně je zasíláno i heslo pro přihlášení, a proto není problém je zachytit a ukrást vám vaši identitu. Děje se to a spolehlivá obrana není. Chyby v ICQ umožňující průnik do systému zatím známy nejsou. Přes ICQ však můžete přijímat soubory a tím i viry, nebezpečná je možnost rovnou spustit příslušný soubor. Na spustitelný soubor si snad dáte pozor, horší je to s obrázkem, ve kterém byste žádné nebezpečí neočekávali. Ale přesto: ICQ zobrazuje při přijímání souboru jen určitý počet znaků, takže není problém vytvořit spustitelný soubor nazvaný obrazek.jpg.exe, kde při vhodné volbě délky názvu nebude přípona .exe zobrazena. Vy potom v domnění, že prohlížíte obrázek, spustíte trojského koně. Obrana je jasná: Obrázek rovnou neotvírejte, ale uložte a podívejte se na něj z disku.
Ale i bez virů lze počítačovou síť vcelku účinně napadnout. Největší servery se hroutily jako domečky z karet pod DDoSem, což je zkratka spojení Distributed Denial of Service. Tisíce nakažených počítačů (tedy distribuovaná síla) začaly na povel bombardovat servery nesmyslnými dotazy, čímž je zahltily a znemožnily jejich normální fungování. Nebylo nic, co by tomuto útoku mohlo zabránit. Normální uživatel má proti DDoSu obranu jako proti atomové pumě v koupelně, naštěstí jsou obě varianty podobně pravděpodobné. Důležitější je uvědomit si, že bezpečnost už není jen vaším problémem, protože aktivní trojský kůň na vašem počítači připojeném do Internetu se může podílet na nejhorších zvěrstvech v celosvětovém měřítku.
Starším bratříčkem je DoS, neboli Denial of Service. Jedná se o útok jednoho počítače na druhý, většinou využívající chyby v systému. Klasickým zástupcem je kdysi populární program WinNuke sestřelující spolehlivé Windows 95. Instalace záplaty do systému je spolehlivou obranou.
Mezi útočné programy patří i mailbombery odesílající na danou adresu tisíce e-mailů a zahlcující tak schránku. Při opakovaných útocích je dobré požádat o pomoc administrátora sítě, který může zamezit příjmu e-mailů z vybraných adres a omezit maximální velikost příchozí zprávy. Filtrování příchozích zpráv podle adresy nebo domény odesílatele však může mít drsný dopad na nevinné. Představme si například, že mailbomber odešle tisíce zpráv z domény nasefirma.cz. Správce pošty u příjemce se naštve, zakáže příjem zpráv z této domény a vzájemná e-mailová komunikace skončila.
Nikdo by neměl znát číslo vaší platební karty. Nikdo by neměl znát vaše přístupová hesla. Nikdo by neměl vědět o vašem poměru se sekretářkou. Přístupová hesla volte tak, abyste si je zapamatovali a nikdo je neuhádl. Kombinujte písmena a čísla, velké a malé znaky. A hlavně si je nezapisujte.
Data se na Internetu přenáší přes mnoho počítačů a s trochou šikovnosti je může kdokoli odposlechnout. Takže platí dobrá zásada: šifrujte. Prohlížeče jsou schopny bezpečně šifrovat data; zda probíhá komunikace šifrovaně se dozvíte třeba klepnutím pravým tlačítkem na stránku a vybráním jejích vlastností. V Netscapu je šifrování vyhrazena ikona v nástrojové liště. V Internet Exploreru o přístupu na zabezpečené stránky informuje ikona v dolní liště a prohlížeč vás upozorní při vstupu do zašifrovaných stránek i při jejich opouštění.
Občas můžete zvolit mezi šifrovanou a nešifrovanou stránkou, není ale důvod nevolit bezpečnost. Komunikace mezi serverem a prohlížečem je šifrována symetrickou šifrou. Šifry mohou být různě dobré a není radno důvěřovat kratšímu klíči než 80 bitů. Volena je vždy nejsilnější šifra zvládaná oběma systémy (serverem a klientem). Nejnovější prohlížeče mohou po uvolnění exportu šifer z USA použít šifrovací klíče s délkou 128 bitů (dříve jen 40 a později někde také 56) a toto uvolnění **se** bude postupně dostávat do praxe (využívá jej například MSIE 6.0). V současné době je však bezpečná komunikace ještě výjimkou a týká se hlavně

komunikace mezi americkými prohlížeči a webovými servery. S příchodem nových verzí prohlížečů by se měla situace napravit i v Evropě.
Podobným způsobem můžete šifrovat e-maily. Je to opravdu snadné, oblíbená je freewarová verze PGP. Ta se umí zasunout do Outlooku či Eudory a dopisy pak zabezpečujete jedním klepnutím myší. Kromě šifry je možné e-mail i podepsat, čímž je zaručeno, že dopis odchází skutečně od vás a nebyl od napsání pozměněn. Podepisování všech dopisů může být mj. skvělou obranou před červy (automaticky podepisovat odesílané přílohy se vám však nepodaří např. v Outlook Expressu). Pokud ovšem trojského koně (byť nechtěně) nepřidal do zprávy sám odesílatel.

OBRANA

1. Nikdy nespouštějte nic, o čem nevíte, co to je.
2. Pokud vám od někoho neznámého dojde spustitelný soubor, patří do koše.
3. Pokud vám dojde od někoho blízkého spustitelný soubor, zeptejte se, o co jde.
4. Pokud neví, že vám něco posílal, měl by nechat vyšetřit sebe nebo svůj počítač.

Internet Explorer verze 5.0 CZ používá ještě 40bitovou šifru. Na webu Microsoftu si lze pro MSIE od verzí 4.0 po 5.01 pro různé jazyky i operační systémy stáhnout High Encryption Pack umožňující používat 128bitové šifrování. Stačí, když si zobrazíte položku O *aplikaci Internet Explorer* z menu *Nápověda* a v dialogu se dozvíte, jak silnou šifru váš prohlížeč používá. Pak stačí klepnout na zvýrazněný text a dostanete se na stránku s nabídkou bezpečnostních aktualizací.

Stažení oprav MSIE:
http://www.microsoft.com/windows/ie_intl/cs/download/default.asp

PROXY-SERVER

Základní funkcí aplikací typu *proxy-server,* je realizace připojení lokální sítě (LAN) přes jedinou IP adresu (účet u poskytovatele připojeni) do Internetu. V celé lokální síti se nachází pouze jediný počítač, který je prostřednictvím modemu a běžné telefonní linky připojen přímo do Internetu. Na tomto počítači je nainstalována a správně nakonfigurována některá s mnoha existujících aplikací typu proxy-server (např. aplikace *WinProxy).* Ostatní počítače v lokální síti do Internetu přistupují nepřímo, právě přes tento zmíněný počítač a aplikaci p*roxy-server.* Toto řešení je výhodné, především z ekonomického hlediska, právě pro lokální sítě malého a středního rozsahu, jejichž výstavbu v rámci této knihy probíráme. Aplikace *proxy-server* ovšem umí pracovat i s jakoukoliv jinou běžnou variantou připojení lokální sítě k Internetu (ISDN linka, bezdrátového připojení, pevná metalická linka). *Proxy-servery* rovněž umožňují zaznamenávání veškerého provozu do textového soboru nebo do některé databáze. Lze zaznamenat který z uživatelů se připojil do Internetu, kdy se tak stalo, které internetové stránky navštívil, jakou použil službu či protokol, kdy se od Internetu odpojil, a mnoho dalších informací.
Dalším závažným důvodem instalace aplikace *proxy-server* může být potřeba ochrany vaší lokální sítě, připojené k Internetu, před pirátským útokem neoprávněných osob zvenčí. Počítač s nainstalovanou aplikací např. *WinProxy* v tomto případě nefunguje jen jako *proxy-server,* ale rovněž jako *firewall.*

Aplikace *WinProxy* a vůbec v podstatě všechny existující aplikace typu *proxy-server* dokážou vykonávat funkci tzv. *Cache serverů.* Anglické slovo „cache" je ve světě počítačů označením

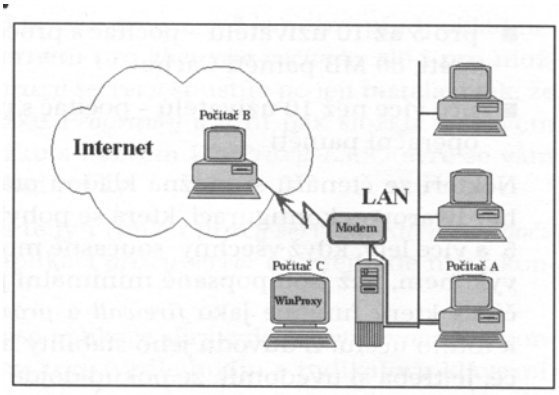
pro rychlou vyrovnávací paměť. *Cache server* funguje tak, že na pevném disku počítače alokuje určitý diskový prostor pro ukládání kopií internetových stránek navštívených uživateli v poslední době. Pokud se později některý z uživatelů pokusí připojit k internetové stránce, jejíž kopie je z její předešlé návštěvy, klidně i jiným uživatelem, uložena v paměti *cache,* nedojde k načtení této stránky v prohlížeči z jejího originálního uložení, ale dojde k načtení její kopie právě z paměti *cache.* Výsledným efektem pak může být lepší využívání síťových zdrojů spolu se snížením zatížení linky do Internetu a rychlejší odezvou na požadavky zobrazení stránek.

WinProxy je v současnosti jednou z nejrozšířenější aplikací české provenience realizující služby *firewallu* a *proxy-serveru.* Kromě výše zmíněných služeb je součástí této aplikace i *server elektronické pošty* (Mail server).
WinProxy podporuje většinu populárních internetových protokolů a služeb. Z nejpoužívanějších jmenujme alespoň protokol HTTP vyžívaný pro prohlížení internetových stránek (služba WWW), protokol FTP využívaný pro rychlý přenos souborů prostřednictvím Internetu, službu Telnet pro emulaci terminálového přístupu ke vzdáleným počítačům, protokoly POP3 a SMTP nezbytné pro odesílání a příjem zpráv elektronické pošty (E-mail) a mnohé další protokoly či služby.

V současné, poslední uvolněné verzi aplikace *WinProxy* l.5 jsou integrovány tři způsoby zpracování elektronické pošty. Konkrétně se jedná o *poštovní bránu* (Mail Gateway), *přesměrování pošty* (Mail Forwarding) a nejpoužívanější službu *serveru elektronické pošty* (Mail serveru), obstarávající komplexní práci s e-mailovými zprávami, včetně třídění došlé pošty do lokálních poštovních schránek, příjmu a odesílání v určitý čas.

Princip funkce firewallu a proxy-serveru

Princip funkce aplikace *WinProxy* je poměrně jednoduchý a lze jej lehce pochopit z uvedeného obrázku.



Představme si situaci, kdy máme nějaký počítač A, který se nachází v naší lokální síti. V této lokální síti se rovněž nachází počítač C, k němuž je připojen modem a je na něm nainstalována aplikace *WinProxy.* Počítač C tedy funguje jako *firewall* a *proxy-server.* Někde ve světě existuje počítač B, který je připojen do Internetu a s nímž se potřebuje počítač A spojit. Jak to provede?
Následujícím způsobem: Jestliže se chce počítač A spojit s počítačem B, musí se nejprve spojit s počítačem C. Počítač A musí počítači C zaslat žádost o spojení s počítačem B. Počítač C naváže spojení s počítačem B a nyní může začít výměna dat mezi počítači A a B. Formát žádosti o spojení přicházející na počítač C může být různý a je typem používané služby a protokolu. Také úloha počítače C při výměně dat mezi počítači A a B je různá, závislá od

používané služby. V jednodušších případech počítač C do procházejících dat vůbec nezasahuje, ve složitějších případech může provádět transformaci protokolů. Počítač C může také provést ověření žádosti o spojení a podle určitých kritérií rozhodnout, zda bude požadavek vyřízen nebo zamítnut. To umožňuje řídit přístup uživatelů lokální sítě k vybraným službám na Internetu.

Aby nebylo možné použít opačného přístupu, tj. aby počítače v Internetu nemohly přistoupit přes počítač C na lokální síť, umožňuje *WinProxy* definovat tzv. bezpečná síťová rozhraní nebo rozsah IP adres, ze kterých je povoleno využívat služeb *WinProxy*. Spojení na počítač C je pak možné jedině z lokální sítě.

OSOBNÍ FIREWALL

Osobní firewall je užitečná věc, která bude potřeba stále víc. Přibývají totiž domácí uživatelé připojení permanentně k Internetu, například přes síť kabelové televize, ale také ti, kteří i u obyčejného vytáčeného připojení tráví celé večery. Určitě si nikdo z nich nepřeje, aby se jim útočník pod rukama díval na soukromá data nebo dokonce páchal škodu na jejich počítačích.

Firewall (doslova by se dalo říci „protipožární zeď") je softwarové nebo hardwarové zařízení, které zabraňuje neautorizovanému přístupu do privátní sítě, intranetu. Osobní firewall je pak zpravidla software, který podobně chrání samotný počítač. Osobní firewally nabízejí i některé antiviry. Firewally izolují interní domácí (podnikovou) síť spolu s jejími počítači od Internetu; uživatelům Internetu však přesto umožňují přístup k jisté omezené části dat a služeb. Firewally můžeme zhruba rozdělit do tří širokých kategorií:

- Bezstavové filtry neboli filtry na úrovni paketů
- Filtry okruhů
- Filtry na úrovni aplikací

Bezstavové filtry neboli filtry na úrovni paketů

Filtry na úrovni paketů zasílají nebo blokují příchozí pakety výhradně na základě povahy paketu; neuvažují tedy jeho historii ani stav. Filtrování paketů se neprovádí podle zdrojové ani cílové adresy či portu. Takovéto filtry mohou provádět také analýzu hlavičky protokolu síťové vrstvy. Uvedený typ firewallů se nejsnáze implementuje a jeho funkce nemají prakticky žádný znatelný dopad na propustnost šitě. Úroveň bezpečnosti takovéhoto firewallů však bohužel není příliš vysoká. Pravidla filtrováni se dají snadno obejít například takzvaným tunelováním. Jinými slovy, jestliže pravidla nepřipouštějí doručováni paketů protokolu HTTP, ale dovolují protokol Telnet, dá se protokol HTTP tunelovat v relaci Telnet.

Filtry okruhů

Filtr okruhu se jako typ firewallů nachází zhruba mezi bezstavovým filtrem a filtrem na úrovni aplikace. Filtr okruhu si „pamatuje" určitý omezený úsek historie paketu a rozhodováni o směrování paketu provádí nejen podle obsahu paketu, ale také podle jeho historie. Tato metoda filtrování je zcela zřejmě složitější než bezstavové filtrováni, protože se při ní musí aktualizovat stav paketu; navíc se provádí analýza obsahu paketu. Filtrování zohledňuje zdrojovou a cílovou adresu a typ služby. Filtry okruhů jsou ve skutečnosti implementovány jako brána: klient komunikuje s filtrem, který pak jeho jménem kontaktuje požadovaný server. Typ filtry se používají častěji pro odchozí pakety než pro příchozí pakety (například pro Telnet).

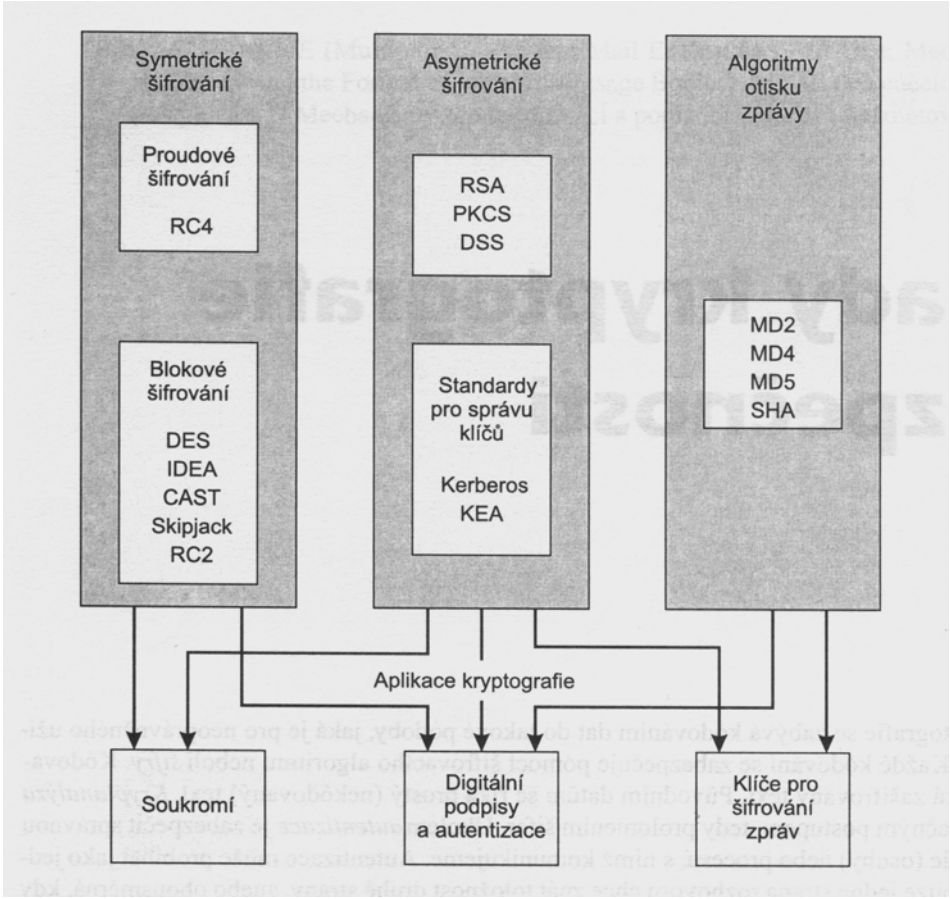
Filtry na úrovni aplikací

Filtry na úrovni aplikací zajišťují vysoký stupeň bezpečnosti, avšak za cenu nižší rychlosti a větší složitosti celého systému. Tyto filtry pracují na firewallovém serveru. Vlastní aplikační server se nachází uvnitř, na privátní síti za firewallovým serverem. Klient se připojí k firewallovému serveru, který se také chová jako aplikační server. Klient ve skutečnosti vůbec nezjistí, že komunikuje s nějakým firewallovým serverem (tomu se v tomto případě říká také *proxy aplikační server)*. Firewall se poté prohlásí za klienta a přijatý klientský požadavek odešle na skutečný aplikační server. Před tímto odesláním však firewall vykoná určitý podmínkový blok, ve kterém rozhodne, jestli je daný požadavek platný a jestli je klient k provedení takovéto operace oprávněn. Z toho vyplývá, že firewall musí mít detailní znalosti příslušné aplikace a protokolu. Potenciální nevýhodou takovéhoto firewallů neboli proxy aplikačního serveru je, že se může snadno stát úzkým místem celé sítě.

7.2 Základy kryptografie a bezpečnosti

Věda zvaná kryptografie se zabývá kódováním dat do takové podoby, jaká je pro neoprávněného uživatele nečitelná. Každé kódováni se zabezpečuje pomocí šifrovacího algoritmu neboli *šifry*. Kódovaným datům se říká zašifrovaný text. Původním datům se říká prostý (nekódovaný) text. *Kryptoanalýza* se pak zabývá opačným postupem, tedy prolomením šifer. Úkolem *autentizace* je zabezpečit správnou totožnost uživatele (osoby) nebo procesu, s nímž komunikujeme. Autentizace může probíhat jako jednosměrná, kdy pouze jedna strana rozhovoru chce znát totožnost druhé strany, anebo obousměrná, kdy si obě strany sděluji totožnost vzájemně.

Základní schéma použití kryptografie v počítačové praxi



Metody šifrování
Obecně existují dva základní typy šifrovacích metod, a sice *metody symetrického šifrování* (kterému se také říká konvenční šifrování) a *asymetrického šifrování* (neboli šifrování veřejným klíčem).

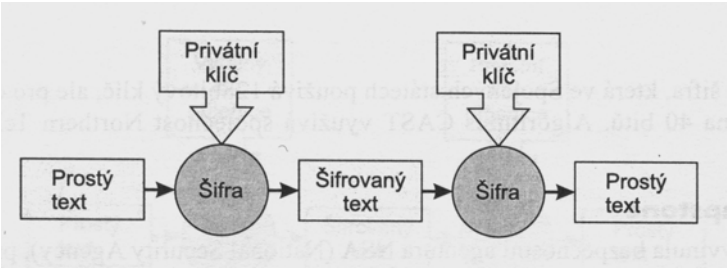
Symetrické šifrování
Symetrické šifrování si můžeme představit jako matematický zápis funkce

Šifrovaná data = Funkce (Data, Klíč)

ke které existuje inverzní funkce vyjádřená vztahem

Data v podobě prostého textu = Inverzní funkce (Šifrovaná data, Klíč)

Tok dat při symetrickém šifrování



Podstatný trik zde spočívá v tom, že šifrovací funkce i funkce k ní inverzní jsou veřejně známé, obnovit prostý text bez znalosti klíče (šifrovacího klíče) je však nemožné.
Symetrickému šifrování se také říká šifrování s tajným klíčem, protože odesílatel a příjemce dat musí znát jistý tajný klíč. V následujících odstavcích textu budeme hovořit o některých významných symetrických šifrovacích algoritmech neboli symetrických šifrách. Obecně existují dva typy šifer. Při *blokovém šifrování se* ze vstupu převezme vždy blok dat o pevné délce a vygeneruje se z něj blok zašifrovaných dat o jiné pevné délce. *Proudové šifrování převádí* prostý text na šifrovaný vždy jeden bit po druhém.

Standard DES
Standard šifrování dat DES (Data Encryption Standard) je blokový šifrovací algoritmus, který pomocí 56bitového klíče zpracovává bloky dat dlouhé 64 bitů. Algoritmus DES byl důkladně analyzován a testován a v současné době se považuje za velice bezpečný systém. DES pracuje ve dvou různých módech, a sice v módu ECB (Electronic Code Book) a CBC (Cipher Block Chaining). V módu ECB zpracovává algoritmus DES vždy ucelený blok 64 bitů dat a používá stejný 56bitový klíč. Každá množina dat o velikosti 64 bitů se tudíž zašifruje nezávisle na zbytku dat. V módu CBC se každý 64bitový blok před vlastním šifrováním spojí logickou operací XOR s předchozím blokem dat. Výsledkem šifrování jednoho stejného 64bitového bloku dat jsou potom různé hodnoty závislé na přesném kontextu (tedy na místě, kde se blok uvnitř odesílané zprávy nachází).
Algoritmus DES je mimořádně rychlý a dobře se hodí pro hardwarovou implementaci. Vývoz produktů, které DES používají, však vláda Spojených států omezuje. DĚS je schválen jako standard ANSI.
V některých situacích se DES považuje za málo bezpečný; pro takové případy slouží určitá obměna algoritmu DES se jménem Triple-DES. Varianta Triple-DES se používá několika různými způsoby. Základní variantou je šifrování dat pomocí klíče DES. Výsledná data se

zašifrují podle druhého klíče a data z tohoto druhého šifrováni se dále zašifrují podle třetího klíče. Všechny tři klíče jsou vzájemně nezávislé.

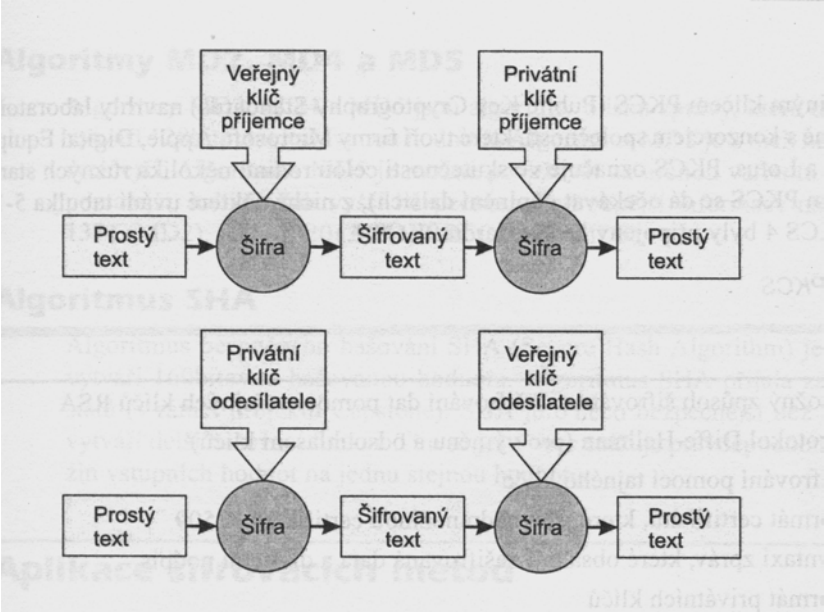
Algoritmus IDEA

Algoritmus IDEA (Intemational Data Encryption Algorithm) představuje další šifru blokového typu. Používá klíč o délce 128 bitů. Algoritmus IDEA je evropským standardem (vytvořila jej společnost ETH z Curychu) a byl přijat v roce 1990. Algoritmus IDEA je považován za bezpečný a srovnání jeho rychlosti implementace i bezpečnosti šifrování proti kryptanalýze vyznívá vůči algoritmu DES velice příznivě.

Asymetrické šifrování

Asymetrické šifrování bylo vyvinuto v sedmdesátých letech. V algoritmech asymetrického šifrování vystupují dva různé klíče. První klíč je veřejně k dispozici (takzvaný veřejný klíč). Pomocí tohoto klíče se šifrují veškerá data, která někdo potřebuje odeslat v zabezpečené podobě vlastníkovi klíče. Druhý klíč je privátní a zná jej pouze vlastník klíče. Asymetrické šifrování je zajímavé zejména tím, že oba klíče bývají často vzájemně zaměnitelné. To znamená**, že** privátní klíč se dá použít k zašifrování dat, která lze dešifrovat pouze pomocí veřejného klíče (a naopak). Uvedená vlastnost asymetrického šifrování vede přímo k myšlence digitálních podpisů, o nichž hovoříme v následujících částech textu. Principem asymetrického šifrování je předpoklad, že jestliže určité dostatečně velké číslo vznikne jako součin dvou prvočísel, pak je mimořádně obtížné najít tato dvě čísla. Toto tvrzení se neopírá o žádný matematický důkaz, jeho správnost prověřil spíše praktický život.

Princip asymetrického šifrování.



V horní polovině obrázku vidíme šifrování dat z podoby prostého textu pomocí veřejného klíče příjemce zprávy. Příjemce dešifruje přijatý zašifrovaný text pomocí svého privátního klíče. Tímto způsobem může zprávu dešifrovat skutečně pouze oprávněný příjemce (který je pravděpodobně jedinou osobou vlastnící správný privátní klíč příjemce).
Spodní část obrázku ukazuje jinou metodu asymetrického šifrování. Zde šifrujeme text pomocí privátního klíče odesílatele a příjemce jej dešifruje pomocí veřejného klíče

odesílatele. Pokud je navíc výchozí prostý text dopředu známý, dostáváme tak základ techniky pro digitální podpis. Zde totiž bezpečně víme, že daná zpráva mohla přijít pouze od konkrétního odesílatele, protože jen tato osoba vlastní (a zná) správný privátní klíč odesílatele.

Algoritmus RSA

Algoritmus RSA je pojmenován podle svých objevitelů: byli to Ron Rivest, Adi Shamir a Leonard Adieman, kteří byli zakládajícími členy výboru RSA Data Security. Algoritmus RSA je možná nejen nejslavnější šifrovací metodou pro asymetrické šifrování, ale také nejslavnější šifrovací metodou vůbec. Algoritmus RSA je založen na matematickém předpokladu, podle kterého je mimořádně obtížné najít dvě prvočísla, jejichž součinem vzniklo určité dostatečně velké číslo. Z daného veřejného klíče se dá tedy jen poměrně dosti obtížně zjistit odpovídající privátní klíč. Algoritmus RSA byl důkladně analyzován a obecně se považuje za bezpečný, jestliže pracuje s dostatečně dlouhým klíčem. Pro účely bezpečnosti se zpravidla 512 bitů pokládá za nedostačující, l 024 bitů však již dává dobré výsledky. V nedávné době se objevily námitky, že díky rostoucímu výpočetnímu výkonu počítačů (rychlostí procesorů) není principiálně žádný problém nabourat šifru RSA útokem typu hrubá síla. Díky stejnému nárůstu ve výkonu procesorů je ale možné použít pro šifrování delší klíče, nebo dokonce provést dvojité šifrování; obě metody vedou k vyšší bezpečnosti.

Standard DSS

Standard pro digitální podpisy DSS (Digital Signatuře Standard) je šifra schválená vládou Spojených států. Délka klíče se může pohybovat mezi 512 a l 024 bity. Standard DSS je určen pro vytváření digitálních podpisů (o nich budeme hovořit dále v této kapitole), nikoli pro zabezpečení soukromí dat. Ve standardu DSS byly objeveny určité bezpečnostní díry a nyní se již příliš nepoužívá.

Algoritmy otisku zprávy

Algoritmy otisku (haše) zpráv představují spolu s asymetrickým šifrováním základní metodu pro vytváření digitálních podpisů (message digest algorithms, doslova jsou to algoritmy, které zprávu určitým způsobem „sežvýkají" - digest - a vytvoří z ní jistou kódovanou reprezentaci). Nejprve uvažujme hašovací funkci, která na vstupu přebírá proměnný počet bitů a vrací řetězec o pevné délce - basovanou hodnotu. Pokud je nalezení inverzní funkce k hašovací funkci velice obtížné, říká sejí funkce otisku zprávy (message digest). Algoritmy otisku zpráv prakticky zajišťují jedinečnost basované hodnoty.

Algoritmy MD2, MD4 a MD5

Algoritmy MD2, MD4 a MD5 jsou algoritmy otisku zprávy, které objevil Ron Rivest. Každý z nich vytváří 128bitovou basovanou hodnotu. Algoritmus MD2je z nich nejpomalejší, zatímco MD4je nejrychlejší. Algoritmus MD5 je možné považovat za určitou variantu algoritmu MD4, který je o něco pomalejší, nabízí však vyšší bezpečnost.

Algoritmus SHA

Algoritmus bezpečného basování SHA (Secure Hash Algorithm) je algoritmus otisku zprávy, který vytváří 160bitovou basovanou hodnotu. Algoritmus SHA přijala za svůj standard vláda Spojených států (v rámci projektu Capstone). SHA je o něco bezpečnější než algoritmy MD4 a MD5, protože vytváří delší hašovací funkci. Tím se ještě více snižuje pravděpodobnost basování dvou různých množin vstupních hodnot na jednu stejnou hodnotu.

Aplikace šifrovacích metod

Na základě výše popsaných technik byla vyvinuta celá řada zajímavých aplikací. Některé z nich si popíšeme v následujících odstavcích.

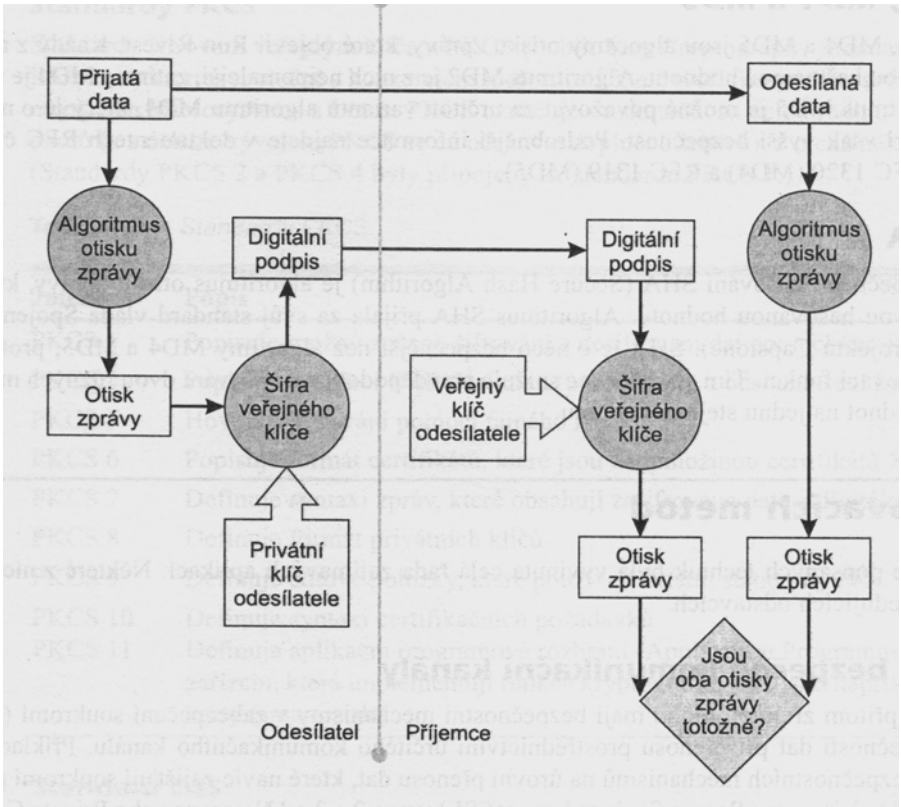
Soukromí dat a bezpečné komunikační kanály

Nejdůležitější a přitom zřejmou úlohu mají bezpečnostní mechanismy v zabezpečeni soukromí (privátnosti) a bezpečnosti dat při přenosu prostřednictvím určitého komunikačního kanálu. Příkladem implementace bezpečnostních mechanismů na úrovni přenosu dat, které navíc zajišťují soukromí přenášených dat, může být vrstva Secure Sockets Layer (SSL) verze 2 a 3 od Netscape nebo Private Com-munications Technology od Microsoftu. Ani Secure Sockets Layer, ani Private Communications Technology však zdaleka nejsou jen kryptografickými aplikacemi. Jsou zároveň aplikačními programovými rozhraními (API) pro programátory, kteří vytvářejí rozhraní na úrovni přenosu dat. O SSL i PCT budeme ještě v této kapitole hovořit podrobněji.

Digitální podpisy

Digitální podpisy představují mechanismus pro ověření (verifikaci) obsahu zprávy a totožnosti odesilatele. Implementují se pomocí asymetrických šifrovacích algoritmů a hašovací funkce. Digitální podpisy vycházejí ze dvou předpokladů: prvním je reverzibilita neboli vratnost asymetrického šifrování, druhým je pak velice těsné svázaní původní zprávy, podpisu a dvojice klíčů, díky němuž algoritmus ověřování při změně libovolného (byť i jednoho) z těchto údajů selže.

Funkce digitálního podpisu.



Odesílatel si připraví otisky zprávy (které se nacházejí v levém horním rohu schématu). Tento otisk zprávy pak zašifruje pomocí svého privátního klíče (privátního klíče odesílatele). Příjemce obdrží takto zašifrovanou zprávu a dešifruje otisk zprávy pomocí veřejného klíče odesílatele. Příjemce si také vypočte otisk zprávy, který musí odpovídat dešifrovanému otisku zprávy. Jestliže se obě zprávy nerovnají, znamená to, že buďto během přenosu dat došlo k narušení obsahu zprávy, nebo že má zpráva padělaný podpis.

Služby časových razítek

Určitá důvěryhodná autorita, která potvrdí platnost digitálního podpisu, může k podpisu doplnit časové razítko (time stamp). Tím se dostáváme k myšlence služeb časových razítek. Tato důvěryhodná autorita (nazývaná také „síťový notář") může tím pádem potvrdit, že daný subjekt podepsal dokument v určitý datum a čas.

Autentizace

Proces autentizace stvrzuje, že náš protějšek v komunikaci je skutečně tou osobou, za kterou se prohlašuje. Autentizace se zabezpečuje několika různými prostředky. Může být založena na určité jedinečné vlastnosti, kterou daná osoba nebo proces vlastní. Příkladem takovéto identifikace mohou být otisky prstů. Jiná možnost je vycházet při autentizaci z určité informace, kterou zná výhradně daná osoba nebo proces. Takovouto znalost představuje například digitální podpis; jedinečnou informaci zde tvoří privátní klič odesilatele. Autentizační služby poskytují například mechanismy SSL a PCT (ty kromě toho zajišťuji také soukromí dat).

Jinou obecnou metodu autentizace představuje protokol Microsoft NTLM Challenge/Response. Jeho původ sahá až do první verze LAN Manageru, který se dodával s operačním systémem OS/2. Tento autentizační mechanismus je založen na určité entitě, kterou vlastní daná osoba nebo proces; v tomto případě je touto entitou tajné heslo. Klient se připojí k serveru a server odešle klientovi zpět výzvu (k přihlášení). Klient na tuto výzvu vygeneruje odpověď, kterou tvoří basované uživatelské heslo. Tato odpověď se poté odešle na server. Server systému Microsoft Windows NT zná heslo daného uživatele, takže může nezávislým způsobem vypočítat správnou odpověď na výzvu a může tím pádem ověřit, jestli je odpověď přijatá od klienta správná. (Přesněji řečeno, heslo uživatele může ověřovat server, který se fyzicky nachází v jiné doméně; v takovém případě první server validuje odpověď klienta pomoci něčeho, čemu se *fiká pověřená validace* - pass through validation.) Systém obsahuje kromě toho dobře dokumentované funkce rozhraní API, pomoci kterých může každý programátor protokol Microsoft NTLM Challenge/Response snadno implementovat; to znamená, že se programátor na straně serveru nebo na straně klienta nemusí zabývat všemi detaily kryptografického algoritmu. Samotné heslo uživatele se zde nikdy neposílá po síti; tato myšlenka je z hlediska bezpečnosti zcela zásadní. Podrobnější informace najdete v části věnované rozhraní Windows NT S SPI dále v této kapitole.

Authenticode

Authenticodeje nová technologie firmy Microsoft, která se na trhu poprvé objevila s Microsoft Internet Explorerem verze 3. Celé Authenticode se ve skutečnosti skládá ze dvou částí: jednak je to technologie, pomocí níž si klient může ověřit povahu načítaného kódu, jednak je to také rozhraní API, pomoci kterého mohou vývojáři vytvářet digitální podpisy programového kódu.

Nad kódem načteným ze sítě Internet provádí tato technologie dvě kontroly: jednak ověřuje totožnost odesílatele a jednak zjišťuje, jestli se kód po cestě nezměnil. To je přímá aplikace

asymetrického šifrování. Jako certifikační autorita vystupuje v Authenticode služba Verisign, kterou Microsoft smluvně zajistil. O certifikaci mohou žádat klienti a vývojáři aplikací. Internet Explorer pak umožní klientovi zvolit si takovou úroveň bezpečnosti, která vyhovuje jeho veškerým datům přenášeným po Internetu. Klient může mechanismem asymetrického šifrování ověřit autenticitu libovolného načteného kódu. K volbě konkrétní úrovně bezpečnosti slouží klientům speciální modul se jménem Windows Trust Verification Service. Klient tak může ze svého vlastního rozhodnutí považovat za bezpečný nebo nebezpečný například veškerý kód načtený od určité společnosti.

Klíče pro šifrování zpráv

Kterou metodu šifrování - symetrickou nebo asymetrickou - by tedy měla aplikace používat? V praxi se obě metody používají současně! Jedním důvodem je, že asymetrické šifrování je mnohem pomalejší než symetrické, takže k zašifrováni dat se používá náhodně vygenerovaný privátní klíč. Tento privátní klíč se pak odešle po síti spolu se šifrovanými daty; klíč se ale zašifruje pomocí veřejného klíče příjemce. Uvedený náhodně vygenerovaný privátní klíč se nazývá *klíč pro šifrování zpráv.*

Jiná situace, kdy je vhodné použít jak symetrické, tak i asymetrické šifrování, nastává při rozesílání zprávy více příjemcům. Pokud bychom zde totiž chtěli uplatnit výhradně asymetrické šifrování, museli bychom zprávu opakovaně zašifrovat pomocí veřejného klíče každého jednotlivého příjemce. Takovýto postup by byl pochopitelně dosti časově náročný. Zprávu tedy namísto toho zašifrujeme pouze jednou, a to pomocí náhodně vygenerovaného privátního klíče. Nyní již pomocí veřejného klíče každého jednotlivého příjemce opakovaně zašifrujeme pouze tento klíč pro šifrování zprávy.

Protokol SET

Protokol pro bezpečné elektronické transakce SET (Secure Electronic Transaction) je jedním z protokolů, jejichž snahou je rozšířit a zdokonalit obchod na Internetu. Specifikaci protokolu SET vyvíjí společnosti Visa a MasterCard. Na vývoji se dále podílí společnosti Microsoft, Netscape, IBM a GTE. Poslední specifikace protokolu SET se skládá ze tří hlavních částí: je to obchodní popis, vlastní popis protokolu a návod pro programátory. Technologie protokolu SET slouží na Internetu k bezpečnému přenosu informací o kreditních kartách; současně definuje rozhraní API, pomocí něhož ji mohou využit programátoři při vývoji komerčních aplikaci orientovaných na Internet. Pro šifrování dat používá protokol SET algoritmu DES; šifrování symetrického šifrovacího klíče a čísla kreditní karty probíhá algoritmem RSA.

Certifikáty

Z asymetrického šifrování vyplývá, že privátní klíč je opravdu nutné považovat za privátní a jako takový jej uchovat v tajnosti. Dále zde vyvstává požadavek na důvěryhodnou metodu spojení určitého veřejného klíče s osobou, procesem nebo jinou entitou. Jak si ověříme, že určitý veřejný klíč náleží opravdu Liboru Neužilovi? (Pokud by nyní svůj klíč zveřejnila Jana Neužilová, ale označila by jej za veřejný klíč Libora Neužila, mohla by číst data určená Liborovi právě ona, nikoli pouze Libor.) Uvedené potřeby vedou k myšlence *certifikátů.* Certifikát je objekt, který bezpečným způsobem definuje vazbu uživatele a jeho klíče. Certifikáty obsahují také určité doplňující údaje, jako například datum platnosti. Certifikáty vydává certifikační autorita (certification authority, CA) neboli síťový notář; zavedením takovéhoto notáře jsme ale celý problém certifikací a celou práci s nimi spojenou pouze přesunuli z jedné entity na jinou.

Certifikační servery

Certifikáty se mohou ukládat do objektů adresářové služby nebo na servery, které jsou pro certifikace vyhrazené. Jako příklad certifikačního serveru si můžeme uvést servery od firem Microsoft a Netscape. Certifikáty se čas od času musí také zrušit (odvolat). V takovém případě vydá síťový notář seznam odvolaných certifikátů (certificate revocation list, CRL). (Uvedený seznam musíme opět načítat bezpečným způsobem; tento požadavek vede k dalším problémům, které jsou již mimo rámec této knihy.)

X.509

X.509je standard, který definuje formát a syntaxi certifikátů. Přesněji řečeno, standard X.509 se zabývá také specifikací autentizačních služeb (těmto službám však již neurčuje kryptografický algoritmus); více se však věnuje syntaxi certifikátů. Certifikáty standardu X.509 využívají různé standardy, které mají co do činění s důvěrností a autentizací dat - například SSL, Secure-HTTP a PEM (Privacy Enhanced Mail). Certifikace podle X.509 mají sloužit zejména oblasti elektronické komerce. První verze standardu X.509 spatřila světlo světa v roce 1988 a v současné době již existuje třetí verze.
Základním smyslem certifikátů je svázat jednoznačným způsobem uživatele a jeho veřejný klíč. Standard X.509 obsahuje proto mimo jiné následující pole. (Seznam polí zdaleka není vyčerpávající.)

- Číslo verze **X.509**
- Identifikátor algoritmu certifikační služby
- Jméno autority, která certifikát vydala
- Období platnosti certifikátu
- Informace o veřejném klíči.