Szakdolgozat

Peleskey Miklós Pál

Debrecen

2009.

Debreceni Egyetem

Informatikai Kar

Hálózati csomópontok védelme hardver és szoftver eszközökkel

Témavezető:
Gál Zoltán
DE TEK ITK igazgatója

Készítette:
Peleskey Miklós Pál
informatikatanári szakvizsga

Debrecen

2009.

Tartalomjegyzék

Bevezetés

Az informatikai rendszerekben tárolt, feldolgozott és az informatikai hálózatokon továbbított adat/információ egyre több, újabb és összetettebb fenyegetéseknek, veszélyeztetéseknek van kitéve. Naponta újabb és újabb vírusok bukkannak fel, kéretlen levelek tömege árasztja el a felhasználókat, adatokat lopnak el, változtatnak meg, veszélyeztetve a normális működést, óriási kieséseket, károkat, anyagi veszteségeket okoznak cégeknek, vállalatoknak és magánszemélyeknek egyaránt. Az Interneten való adattovábbítás módja alapvetően nyílt abban az értelemben, hogy bárki hozzáférhet, megszerezheti, lehallgathatja, sőt meg is változtathatja, meg is hamisíthatja az ott továbbított információt. De nem garantált a számítógépes adattárolás biztonsága sem, hiszen adott a hálózatra kapcsolt számítógépekre, informatikai rendszerekbe történő belépés, betörés lehetősége is. Ráadásul a betörő, a tolvaj, a cracker mindezt szinte észrevétlenül meg tudja tenni, álcázni képes magát, s mire rájönnek, észreveszik, addigra a baj már nagy lehet. Ily módon a tárolt vagy továbbított információ gyakran érzékeny abban az értelemben, hogy annak illetéktelen megismerése, esetleges megsemmisítése vagy rosszindulatú módosítása anyagi, erkölcsi kár okozására, jogosulatlan előnyszerzésre ad módot. Ezért a biztonság és az adatvédelem az információs és kommunikációs technológiák alkalmazhatóságának alapvető kritériumává vált. Eddig nem látott mértékű védelmi és biztonsági beruházásokra kényszerülnek a számítógépek, informatikai rendszerek tulajdonosai, üzemeltetői.

Az elektronikus adatvédelem és informatikai biztonság feladata, hogy védelmet nyújtson az informatikai és kapcsolódó elektronikus rendszerekben tárolt, megjelenített, feldolgozott és továbbított információk véletlen vagy szándékosan okozott bizalmasságának, sértetlenségének és rendelkezésre állásának csökkenése ellen. Az információs és kommunikációs rendszerek biztonságának növelését, az adatok védelmének minél nagyobb fokú biztosítását csakis komplex, átfogó, integrált megoldások alkalmazásával érhetjük el. Az integrált védelem széleskörű, hatékony, költségtakarékos megoldások együttesét jelenti. A központi, egységes telepítés, üzemeltetés és karbantartás, a frissítési tevékenység nem csupán jelentős mértékben növeli a védelem hatékonyságát, gyors reagálást biztosít a változásokra, hanem komoly pénz- és időmegtakarítást is eredményez. E területek a hardver eszközök, az

azokra telepített és megfelelően konfigurált szoftveres megoldások, továbbá a fizikai védelmi rendszerek, illetve rendszabályi eljárások, technikák alkalmazásának összehangolását jelentik.

Amikor az informatikai számítógépes rendszerek – és ezen belül a hálózati csomópontok – biztonságáról és védelméről beszélünk, célszerű a rendszerszemléletű megközelítési módot választani. Világosan kell látni, hogy mik a rendszer összetevői, alrendszerei, elemei, milyen a rendszer fizikai és logikai struktúrája, hierarchikus felépítése, működése, a közvetlen és tágabb környezettel való kapcsolata. A hálózati csomópontok védelmét illetően az egyes biztonsági területek és védelmi megoldások szempontjából nem hanyagolható el az egész rendszer működtetésére, biztonságos üzemeltetésére, védelmére, karbantartására és fejlesztésére vonatkozó elvárások és ezek szabályozási rendszere. Így a számítógép, mint hálózati csomópont veszélyforrás lehet saját maga, és a vele közvetlen, hálózati vagy közvetett kapcsolatban álló gépek, eszközök számára; és ugyanakkor védelemre is szorul mindezekkel szemben. A határt, a kapcsolatot a környezettel a hálózati interface jelenti. Hasonló a helyzet a számítógépek csoportját alkotó hálózati szegmensek, a LAN-ok, illetve VLAN-ok esetén is, itt a kapcsolat a rendszer azonos hierarchiájú szintjén a routerek, illetve switchek (hub-ok) révén valósul meg.

Az alábbiakban a széleskörű integrált védelem összetevőit, szükségességét, hatékonyságát és feladatait a hálózati csomópontokra összpontosítva vizsgáljuk. Áttekintjük a hálózati csomópontokon megjelenő veszélyforrások fajtáit, az elhárítási lehetőségeket, az adatvédelem és biztonság csomópontokat érintő megvalósítási lehetőségeit, alkalmazási módszereit. Nem kétséges, hogy erre a területre koncentrált figyelmet kell fordítani, hiszen az adatok többnyire itt kerülnek elektronikusan feldolgozásra, tárolásra és itt kell megfelelő védelemmel ellátva előkészíteni a továbbításra. A feldolgozás és továbbítás során a gyakran legnagyobb veszélyt jelentő felhasználók az üzemeltetés és alkalmazások révén itt kerülnek közvetlen kapcsolatba a védendő adatokkal és számítógépes rendszerekkel. A fenyegetettség tehát nem csupán kívülről jelentkezik, hanem gyakran az adott helyi hálózat gépeiben, illetve gépei felől. A helyi, belső hálózat külső és belső védelme mellett figyelmet kell fordítani a csomópontok, számítógépek, hálózati eszközök saját, külön védelmére is. A védelem ugyanakkor folyamatos figyelést is jelent, felkészülést a várható és az előre nem látható támadások kivédésére.

A mellékletekben a védelmi technológiákhoz kapcsolódó kiegészítések, informatikai és matematikai eljárások vázlatos ismertetése, gyakorlati alkalmazása, illetve szómagyarázat található. A dolgozatban tárgyalt védelmi, biztonsági (security) informatikai megoldások nagyon új jellege miatt a magyar nyelvű terminológia még nincs kialakulva. Emiatt inkább az angol megnevezéseket használjuk, az 1. számú mellékletben felsorolva a témakörhöz közvetlenül kötődő, gyakrabban használt elnevezéseket, fogalmakat, meghatározásokat. A 8. számú mellékletben pedig az előforduló angol rövidítések, mozaikszavak listája olvasható.

A dolgozat elkészítése során nyújtott szakmai segítségnyújtásáért, hasznos tanácsaiért, útmutatásaiért köszönetemet fejezem ki mentoromnak, Gál Zoltán egyetemi informatikai igazgató úrnak.

1. Támadási fajták, veszélyek

1.1. Támadási fajták

A rendszert fenyegető támadások mind céljukat, mind pedig a módszereiket és a támadási felületeket tekintve igen sokrétűek, széleskörűek. A leggyakoribb, legismertebb célok és támadásfajták az alábbiak:

- Információgyűjtés a rendszerről, portszkennelés, védelmi és hitelesítési eljárások feltérképezése, adatátviteli csatornák figyelése, szolgáltatások szkennelése (ftp, web, mail, irc, icmp stb.)
- Synflood, UDP flood, DoS (Denial of Service), Buffer túlcsordulás
- IP Spoofing (Ip hamisítás), Teardrop, Ping of death
- Vírusok, trójai falovak, férgek stb.
- Lehallgatások, spywarek
- Belső veszélyforrások

A támadási irány szempontjából külső- és belső behatolási kísérletekről beszélhetünk.

1.2. Külső behatolási kísérletek

1.2.1. Hálózati kommunikációt veszélyeztető támadások

A használt szolgáltatások egy része nem biztonságos, mivel a kommunikáció ez esetben titkosítatlanul folyik. Ráadásul ezeket a szolgáltatásokat gyakran a rendszeradminisztrációhoz használják, ezáltal még nagyobb a várható veszély és károkozás mértéke. Ilyen szolgáltatások pl. a telnet, az ftp, rsh, rcp stb. A titkosítatlan, védelem nélküli kommunikáció így lehetőséget kínál a támadó számára hasznos információk (például belépési nevek, jelszavak) megszerzésére. A támadás módja az adott hálózati forgalom figyelése révén az adatok megszerzése. Ez ellen legegyszerűbb védekezési mód a biztonságos

protokollok és kapcsolatok használata (mint például az SSH (Secure Shell), az SRP (Secure Remote Password), VPN (Virtual Private Network)).

1.2.2. Snifferek, lehallgatás

Lényege a gépek közötti forgalom figyelése, ami által a megfigyelő értékes információkhoz juthat a titkosítatlan kapcsolatokból. Ehhez szükséges, hogy a hálózaton valahol legyen egy root accountja. Ha ez megvan, több program is rendelkezésére áll a tcpdump-tól az intelligensebb programokig (pl. sniffit vagy tcpflow).

A snifferek működése a hálózati interface promiscuous (lehallgatási) módba kapcsolásán alapszik, melynek lényege, hogy az ethernet interface nem csak a neki címzett csomagokat fogadja el, hanem minden csomagot. Ezen alapul a detektálásuk is, amit a promiscuous módban működő ethernet interface-ket kereső programokkal lehet végezni, illetve maguk a sniffer programok állítják be az interface ezen üzemmódját. Ezáltal általában meghatározható, milyen címről fut a sniffer (feltéve, hogy az elkövető nem olyan operációs rendszert használ, mely a detektálást megakadályozza). A detektorok működési elve általában a lokális hálózat pingelésén alapszik, így megkapja minden interface hardver címét (MAC address). Ezután egyenként minden IP-hez tartozó hardver címet véletlenszerűen megváltoztat az ARP cache-ben és újra pingeli. Amelyik gép erre válaszol, ott az interface már promiscuous módban van. (Persze ez még nem jelenti, hogy sniffert futtat, mert néhány más program is átkapcsolja az interface-t ebbe a módba.) Ezáltal lehetőség van a hálózati szegmensen áthaladó csomagok figyelésére. A csomagokból pedig kiszűrhetők az adatok, többek között a felhasználói nevek és jelszavak lehetnek érdekesek.

Az ilyen típusú támadásokkal különösen az üzenetszóró (pl. kapcsolt Ethernet) csomópontok és hálózatok, sebezhetők. A vezeték nélküli hálózatok gépei még nagyobb veszélyben vannak, hiszen esetükben közvetlen fizikai kapcsolatra sincs szükség. A kapcsolt Ethernet környezetben – bár fennáll, de jóval kisebb a veszély.

A lehallgatás elleni védekezés elsősorban az üzenetszóró szolgáltatások kerülésével, illetve az adatok titkosításával történhet.

1.2.3. Portscan-ek

A portscan a portok egymás utáni letapogatása az adott portra történő kapcsolat kezdeményezése révén. Ezzel a támadó feltérképezi a gépen futó szolgáltatásokat, tulajdonképpen egy felmérés, tájékozódás, előkészület a betörési kísérlet előtt. Ezt a módszert

használják a rendszerellenőrző programok. A portscannelés célja tehát a felderítés, hogy majd melyik szolgáltatást kihasználva lehet betörni a célgépre.

Védekezésként log-olhatjuk a TCP/UDP/ICMP scan-t. A rendszeresen visszatérő próbálkozók gépének adminisztrátorait érdemes tájékoztatni ezekről az esetekről. Ugyanakkor célszerű a helyi hálózaton, illetve közvetlenül a csomóponti gépen tűzfal program alkalmazásával a támadók címeit kiszűrni.

1.2.4. Address spoofing

Másképpen IP címhamisítás, amely egyidejűleg több más támadási eljárás alapjául is szolgál, többek között alkalmas a tűzfallal védett rendszerek megkerülésére. Működési elvének lényege, hogy a támadó hamis címmel ellátott mesterséges adatcsomagok küldésével azt a látszatot kelti, mintha azok egy belső csomópont csomagjai lennének. Így könnyen becsapható az olyan tűzfal, amely csak a kimeneti porton kiküldendő csomagokat szűri, ugyanis nem megállapítható, hogy az adott csomag egy belső vagy egy hamisított külső csomag-e. A tűzfal saját belső tartományából származó csomagként küldi tovább a hamis csomagot az áldozat címére. Ezzel a címhamisítással a tűzfalrendszeren keresztüljutva megvalósulhat a belső hálózat gépeinek szekvenciaszámos támadása.

Az IP spoofing egyik esete az ICMP redirect használata, amikoris a támadó képes figyelni a hálózati forgalmat. Legyen adott az A és B gép, illetve C a támadó, aki át akarja venni B szerepét. Ebben az esetben C tehát képes figyelni B szegmensét (mert pl. már betört valamelyik lokális hálózaton lévő gépre), A routerének küld egy ICMP redirect csomagot B routere nevében. Ezért A routere már úgy tudja, hogy B - akivel kommunikálni akar - a C routerének irányában van. E támadás ellen védekezést jelent a belső és külső hálózat között elhelyezett tűzfalon a kívülről érkező ICMP forgalom tiltása.

A másik eset, amikor a C támadó nem tudja figyelni a hálózat forgalmát. Ebben az esetben az A routerét kell becsapnia, hogy az úgy tudja majd, hogy B routere a C irányában van, tehát az A gép a C felé fogja a kapcsolatot kezdeményezni azáltal, hogy a C gép a B gép IP címét veszi fel. Ez az eljárás általában a routing táblák átírását jelenti. Lehetséges úgy, hogy a támadó az A routerébe tör be először és ott írja át a táblát. Hibásan beállított router esetén esetleg még be sem kell törni a routerre annak átkonfigurálásához. Egy jól beállított router az ilyen jellegű támadások nagy részét megfogja.

A harmadik esetben B és A ugyanabban a szegmensben található. Ekkor a támadó C gép először kiiktatja B-t (azaz valamilyen formában eléri, hogy B leszakadjon a hálózatról), majd átveszi B hardware címét is. Ezután az A gép a B helyett C-vel fog kommunikálni. Védekezésül belső tűzfal használata jöhet szóba, és a belső gépeket megfelelően kell konfigurálni, karbantartani. Így a C-nek még abban az esetben is nehéz kiiktatni B-t, ha egy szegmensen van vele. A szegmensek tűzfalas védelme mellett nagyon fontos tehát az egyes csomópontok megfelelő egyedi védelme! Emellett érdemes még arpwatch-ot is használni a hálózati forgalom figyelésére. Az általa észlelt IP és hardver címek kiírásával észrevehető az esetleges MAC cím változás.

1.2.5. DNS spoofing

Ebben az esetben a támadó a DNS –t használja az IP cím cseréjére, hamisítására. Az előző példához hasonlóan tegyük fel, hogy A és B egymással akarnak kommunikálni, a C támadó pedig át akarja venni B helyét. Ehhez C megkeresi B domain name server-ének címét és betör rá (ha ez nem sikerül, akkor itt a vége), átállítja B IP címét a sajátjára, majd újraindítja a name server szolgáltatást. Ezután A, ha B hostnevét használja a kapcsolat felvételéhez és nem ellenőrzi annak valódiságát (azaz nem kéri a reverse name server-től a B nevéhez tartozó IP címet, ami most már a C-é), akkor B helyett C-vel fog kommunikálni.

Elég jó védelmet nyújt ellene a tcp-wrapper, illetve a megfelelő kliens és szerver programok használata, a DNS szerver megfelelő védelme.

1.2.6.TCP/UDP spoofing

Hibásan tervezett vagy implementált protokoll stack esetén vagy a protokollok hiányosságaiból fakadó okok miatt lehetséges a már felépült vagy kezdeményezett kapcsolatokba való belépés, a kapcsolatok "elrablása" vagy a kapcsolatba idegen csomagok "becsempészése" harmadik személy által.

Védelmet jelent ellene a felfedezett hibák gyors kijavítása és egy jól konfigurált tűzfal.

1.2.7. Mail spoofing

A feladó címének hamisítása az SMTP egyszerűségét használja ki. Mivel az SMTP a teljes átviteli folyamatot titkosítatlan szöveggel bonyolítja, nincs lehetőség a feladóra

vonatkozó adatok vizsgálatára. Ezáltal mind a feladó neve, mind pedig az üzenet szövege hamisítható.

A megbízható azonosítás és az üzenetek valódisága az elektronikus aláírással (PGP), illetve titkosítási eljárások (PKI, RSA) használatával valósítható meg.

1.2.8. Man-in-the-middle attack

Arra az esetre vonatkozik a támadás, amikor a kommunikációs csatorna végpontjai, a csomóponti gépek megfelelően védettek, támadásuk sikertelen. Ekkor a csatorna csomópontok közti megcsapolása lehet a támadás kiinduló lépése. Tegyük fel, hogy A és B kommunikációjába C be szeretne avatkozni. Ehhez a C támadó rákapcsolja a saját terminálját a csatornára és az ott folyó kommunikációt lehallgatja. Amikor lehetősége van, a csatornát észrevétlenül átvágja, majd mindkét végét a saját termináljára kapcsolja, így C egyfajta átjáróként szolgál A és B között, megadva ezzel C-nek a beavatkozás lehetőségét. Tehát C küldhet A-nak és B-nek is a másik nevében csomagokat, a választ pedig elfogja. A számára érdektelen kommunikációba nem avatkozik be.

Védekezni csak a teljes kapcsolat kódolásával lehetséges, de még ez sem nyújt teljes az olyan esetekben, amikor C a kapcsolat kezdetétől fogva képes beavatkozni a kommunikációba. Az utóbbi eset ellen bizonyos protokollok (pl. SSL) implementációja megfelelő védelmet biztosít a tanúsítványok használatával.

1.2.9. Denial of Service (DoS)

Ez a támadások egy speciális fajtája, amikor a támadó nem betörni akar, hanem egy adott szolgáltatás működését szeretné megbénítani. Az adott kiszolgáló program vagy az operációs rendszer hibája segítségével a program rendeltetésszerű működését megakadályozza. A hiba lehet például egy távolról kihasználható buffer overflow (buffer túlcsordulás), amelynek következtében a program illegális műveletet hajtana végre vagy számára nem hozzáférhető memóriaterületre írna, így az operációs rendszer a futását megszakítja. Elképzelhető olyan hiba is, amit felhasználva a program egy része vagy egésze lefagyasztható vagy hibás működésre bírható. Lehetséges olyan támadás is, amely az egyszerre kiszolgálható kapcsolatok számának a határérték fölé növelésével teszi használhatatlanná az adott szolgáltatást.

Védekezni a szerver programok karbantartásával és helyes beállításával lehet.

1.2.10. Resource starvation

A támadó a host vagy a hálózat erőforrásait (memória, processzor, háttértár stb.) igyekszik olyan mértékben kihasználni, hogy ezzel más felhasználó számára a rendszert használhatatlanná tegye.

Lehetséges lokálisan - azaz a gépre bejelentkezve - (pl. fork bomba), illetve kívülről végrehajtani a támadást (pl. a syslogd üzeneteit felhasználva megtölthető a partíció vagy a régebbi Apache httpd-hez intézett kérésben megfelelő számú hivatkozást elhelyezve a httpd által okozott processzorterhelés ugrásszerűen megnövelhető).

Gyakori technika a kiszolgáló programok flood-olása. Ezzel általában az a támadó célja, hogy az egyszerre megnyitható kapcsolatok számának a határérték fölé növelésével használhatatlanná tegye az adott szolgáltatást, illetve a szerver terhelését növelve lassítsa a valós kérések kiszolgálását, esetleg a merevlemezt töltse meg log bejegyzések generálásával vagy levélbombázással.

Ide sorolható a hálózati sávszélesség kihasználása is. A támadó a hálózati protokollok hiányosságait vagy egy rosszul beállított eszközt felhasználva olyan mértékben megnövelheti a hálózati forgalmat, hogy a felhasználók által kezdeményezett kapcsolatok timeout-tal szakadjanak meg. Erre jó példa a smurf attack, amikor a támadó a célhálózat címére állított forráscímű csomagokkal bombázza a hálózat broadcast címét. Az eredmény elképzelhető.

E támadások elleni védekezés legjobb módja a megelőzés. Ebben komoly szerepe lehet a hostok megfelelő erőforrásvédelmének, a programhibák csökkentésének, a jól beállított hálózati eszközöknek (router, switch), illetve a firewall-nak. A felesleges kiszolgáló programok letiltásával is csökkenthető a kockázat.

1.2.11. Web browser attack

A web böngészők méretének és szolgáltatásainak növekedésével egyre több kritikus hiba kerül napvilágra. A kliens oldali Javascriptekkel kapcsolatos hibákat sem szabad elhanyagolni. A Java biztonsági modellje elég jó, az előforduló hibák általában a hibás vagy

hiányos implementációból adódnak. Bizonyos hibák kihasználásával a gépünkön tárolt és számunkra olvasható file-ok hozzáférhetővé válnak a látogatott oldal készítője számára.

Védekezésül ajánlott letiltani a Java support-ot a böngészőben, ha nem megbízható site-okat látogatunk.

1.3. Gyakori támadási felületek

1.3.1. Csomóponti alkalmazások, programok veszélyei

A csomópontokon használt alkalmazások, illetve az ezekbe beágyazott különböző plug-in-ek, vezérlők, lejátszók és scriptek (ActiveX vezérlők, XSS (Cross Site Scripting), SQL használat, JAVA, JavaScript, Flash lejátszók) biztonsági hiányosságaik, a nem megfelelő beállítások vagy a megfelelő csomóponti védelem hiánya miatt beláthatatlan lehetőségeket nyitottak a támadók számára.

Elsősorban a számítógépvírusok, a trójai programok és kémprogramok (spyware-k) jelentik e téren a legnagyobb veszélyt.

1.3.1.1. Számítógépes vírus

A számítógépes vírus automatikusan terjedő, önmagát reprodukálni képes, kár okozására alkalmas program. Megfertőzött szoftverekkel, adathordozókkal (USB drive, floppy stb.), illetve hálózaton terjednek. A számítógépes vírus olyan program, amely saját másolatait helyezi el más, végrehajtható programokban vagy dokumentumokban. Többnyire rosszindulatú, más állományokat használhatatlanná, sőt teljesen tönkre is tehet. A számítógépes vírusok működése hasonlít az élővilágban megfigyelhető vírus viselkedéséhez, mely az élő sejtekbe hatol be, hogy önmaga másolatait előállíthassa. Ha egy számítógépes vírus kerül egy másik programba, akkor ezt *fertőződés*nek nevezzük. A vírus csupán egyike a rosszindulatú szoftverek *(malware)* számos típusának. Ez megtéveszthető lehet a számítógép-felhasználók számára, mivel mára lecsökkent a szűkebb értelemben vett számítógépes vírusok gyakorisága, az egyéb rosszindulatú szoftverekhez, mint például a férgekhez képest.

A számítógépes vírusok lehetnek kártékonyak (például adatokat semmisítenek meg, megbénítják a számítógép működését stb.), a vírusok bizonyos fajtái azonban csupán

zavaróak. Némely vírus késleltetve fejti csak ki hatását, például csak egy bizonyos számú gazdaprogram megfertőzése után. A vírusok domináns kártékony hatása az ellenőrizetlen reprodukciójuk, mely túlterhelheti a számítógépes erőforrásokat.

A gazdaprogramok megfertőzése és az önsokszorosító viselkedés valamennyi vírusra jellemző. Ezen kívül gyakran rendelkeznek a következő tulajdonságokkal:

- nagyon kis méret;
- legtöbbjük a Microsoft Windows operációs rendszereken okoz gondokat;
- futtatható állományokat képesek megfertőzni;
- általában ártó szándékkal készítették őket;
- gyakran akár válogatva, időzítve tönkretesznek más fájlokat;
- rejtetten működnek, esetleg akkor fedik fel magukat, ha feladatukat elvégezték;
- egyre fejlettebb intelligenciával rendelkeznek, például változtathatják saját kódjukat és aktivitásukat

Alaptípusaik:

- EXE-COM vírusok
- BOOT-vírusok
- MR-vírusok
- Makró vírusok
- New Exe vírusok
- Multi platform vírusok
- Önátíró (polimorf) vírusok
- E-mail vírusok
- zenefájl vírusok

1.3.1.2. Számítógépes féreg

A számítógépes féreg (*worm*) a számítógépes vírushoz hasonló önsokszorosító számítógépes program. Míg azonban a vírusok más végrehajtható programokhoz vagy dokumentumokhoz kapcsolódnak hozzá illetve válnak részeivé, addig a férgeknek nincs

szükségük gazdaprogramra, önállóan fejtik ki működésüket. Az operációs rendszer biztonsági réseit és a hálózati kapcsolatokat felhasználva terjednek.

Az önsokszorosításon kívül a féreg sokféle dologra beprogramozható, például a fájlok törlésére a gazdarendszeren, vagy önmaga elküldésére e-mailben. Az újabban megfigyelt férgek több végrehajtható állományt is visznek magukkal. Még valódi ártó szándékú kód nélkül is súlyos fennakadásokat okozhatnak, csupán azzal, hogy sokszorozódásuk kiugróan magas hálózati forgalmat generálhat. Például a Mydoom féreg terjedése csúcsán világszerte észrevehetően lelassította az internetet.

1.3.1.3. Trójai programok

A trójai programok: számítógépes értelemben a **trójai faló** (röviden *trójai*) egy olyan program, ami mást tesz a háttérben, mint amit a felhasználónak mutat. A trójai nem feltétlenül tartalmaz rosszindulatú programkódot, azonban a többségük tartalmazza az úgynevezett hátsó kapu telepítését, ami a fertőzés után biztosítja a hozzáférést a célszámítógéphez. A vírusokkal ellentétben általában nem többszörözi önmagát, terjedése főként egyedi támadásoknak és az emberi hiszékenységnek köszönhető.

Az egyszerűbb trójai programok csak kívülről tűnnek hasznos programnak, míg fejlettebb változataik a kémkedés mellett valóban képesek az ígért funkciók elvégzésére is – így csökkentik a lebukás veszélyét. Trójaival való megfertőződésnek forrása lehet egy e-mail üzenet csatolmánya vagy azonnali üzenetküldő program, de megkaphatjuk CD-n vagy egyéb adattárolón is. A leggyakoribb fertőzési módszert azonban a letöltések és a veszélyes honlapok jelentik.

Gyakoribb támadási felületei, módjai:

- FTP szervert telepít a gépen, lehetővé teszi fájlok le- és feltöltését a megtámadott gépen
- billentyűzetleütés figyelő (keylogger) - felhasználói nevek, jelszavak ellopására
- ál dialógusablak – szabályosnak tűnő űrlap kitöltetésével jut információhoz, s küldi el a támadónak
- telnet – a támadó távolról irányítja a megtámadott gépet

- automata trójai, e-mail küldő trójai – a megszerzett információkat (név, jelszó, IP cím stb.) weben (az ellopott adatokkal általa kitöltött form a támadónak való elküldéssel), vagy e-mail útján küldi a támadónak

1.3.1.4. Kémprogramok

Kémprogramnak (spyware) nevezzük az olyan, főleg az interneten terjedő számítógépes programok összességét, amelyek célja, hogy törvénytelen úton megszerezzék a megfertőzött számítógép felhasználójának személyes adatait. Feltelepülése általában észrevétlenül történik, a felhasználó figyelmetlenségének és a számítógép böngészőprogramja biztonsági hiányosságainak kiaknázásával. Léteznek azonban magukat álcázó - trójaiakhoz hasonló - programok is, amelyek a felhasználó közreműködésével települnek egy rosszindulatú honlapon. A megszerzett információkat általában bűncselekmények (hitelkártya számok, online szolgáltatások jelszavainak megszerzése) elkövetésére vagy enyhébb esetben böngészési szokásaink, érdeklődésünk, ízlésünk megfigyelésére használják fel. A kémprogramokat a számítógépes kártevők (malware) kategóriájába sorolhatjuk.

A vírusok, trójai programok és spyware-k elleni védekezés az összetett víruskereső programok (amelyek állandóan figyelik a rendszert (letöltött és indított programokat, e-maileket)), intelligens tűzfalprogramok, illetve spyware-k ellen kifejlesztett specifikus programok segítségével történik. Nem elegendő az eltávolításuk, nagyon fontos a megtalált biztonsági rések befoltozása is. A megelőzésben nem elhanyagolható a felhasználó ébersége sem.

1.3.2. Universal Plug and Play használatának veszélyei

Az Universal Plug and Play vagy UPnP számítógépes hálózati protokollok egy csoportja, amelynek célja, hogy különféle eszközök egyszerűen, külön konfiguráció nélkül csatlakoztathatóak legyenek egy hálózathoz. A protokollok lehetővé teszik hálózati elemek automatikus felderítését, irányítását és eseménykezelését, olyan nyílt internetes szabványokra alapozva, mint a TCP/IP, az UDP vagy a SOAP; illetve lehetővé teszik tűzfalak és routerek megkerülését NAT segítségével, és az eszközök webböngészőn keresztül való kezelését. A technológia a számítógépre közvetlenül kapcsolódó eszközöknél használt Plug-and-play protokoll továbbfejlesztése hálózati eszközökhöz

Az UPnP protokoll architektúrálisan megengedi az egyenrangú hálózatok létrehozását PC-k, egyéb hálózati készülékek és vezetéknélküli eszközök között. Ez elfogadott szabványok alapján egy nyitott architektúra, mint például a TCP/IP, az UDP, a HTTP és az XML.

Az UPnP támogatja a konfigurálásmentes hálózatot. Egy UPnP kompatibilis eszköz bármilyen gyártótól beszerezhető, amely tud dinamikusan csatlakozni egy hálózathoz, megszerzi az IP-címet, bejelenti a nevét, kérésre közli a képességeit, és más eszközök jelenlétéről értesül, valamint azok képességeiből képes tanulni. A DHCP és DNS-kiszolgálók szabadon választhatóak és csak akkor használhatóak, ha elérhetőek a hálózaton. Az eszközök automatikusan elhagyhatják a hálózatot anélkül, hogy otthagynának bármilyen felesleges állapotinformációt.

További UPnP tulajdonságok:

- Média- és eszközfüggetlenség: az UPnP technológia sok médián futhat, melyek támogatják az IP-t, köztük a villamos távvezetékeket (PLC), Ethernet, IR (IrDA), RF (WiFi, Bluetooth), és FireWire. Különleges eszközmeghajtó-támogatás nem szükséges, közös protokollokat használnak helyette.
- Felhasználói felület (UI) kontrollja: az UPnP architektúra gyártói kontrollt tesz lehetővé az eszköz felhasználói felülete fölött a webböngészőn keresztül.
- Operációs rendszer és programozási nyelv függetlenség: bármely operációs rendszer/programozási nyelv képes használni az UPnP-alapú termékeket, ui. az UPnP nem határozza meg vagy korlátozza az ellenőrzési (kontroll) pontokon futó alkalmazások API kialakítását, ezt az operációs rendszer készítőire hagyja. A gyártók kontrollálhatják az UPnP eszközt a webböngészőn keresztül úgy, mint ha az egy hagyományos programozható alkalmazás lenne.
- Programozhatósági kontroll
- Bővíthetőség

1.3.3. Adathalászat

Adathalászatnak (*phishing*) azt az eljárást nevezzük, amikor egy internetes csaló oldal egy jól ismert cég hivatalos oldalának láttatja magát és megpróbál bizonyos személyes adatokat, például azonosítót, jelszót, bankkártyaszámot stb. illetéktelenül megszerezni.

A csaló általában e-mailt vagy azonnali üzenetet küld a címzettnek, amiben ráveszi az üzenetben szereplő hivatkozás követésére egy átalakított weblapra, ami külsőleg szinte teljesen megegyezik az eredetivel.

Az ilyen támadások elleni védelem alapjai megtalálhatók a webböngészőkben: a biztonsági csomagokból ismert *phishing szűrő* vészjelzést adhat a gyanús weboldalak meglátogatásakor. A három legelterjedtebb webböngészőről (Internet Explorer, Opera, Firefox) elmondható, hogy felveszi a harcot az adathalászok ellen. Amennyiben aktiváljuk az ez irányú védelmet, a böngészők ellenőrzik az előhívott oldalakat, jellemző *phishing* tulajdonságok után kutatva. Ezt elősegíti a megbízható oldalak fehér listája és az ismert phishing oldalak fekete listája, amely listák automatikusan frissülnek a számítógépen. Mivel folyamatosan újabb és újabb adathalász-oldalak jelennek meg a weben, ezért ha a böngésző olyan oldallal találkozik, amely még nem szerepel a listán, akkor kapcsolatba lép egy Update szerverrel (ez a kapcsolódás Internet Explorer esetén nem kapcsolható ki, a Firefox ezen a területen együttműködik a Google-lal, a kérdéses URL-t a keresőgép szervere ellenőrzi). Az Internet Explorerrel szemben a Firefox esetén ez a szolgáltatás kikapcsolható, vagy bekapcsolt állapot esetén választható, hogy a böngésző a helyi adatbázist ellenőrizze-e, vagy a Google szerverét használja.

Az új High Assurance SSL Certificate-et mindhárom böngésző támogatja, így a jövőben az SSL esetéhez hasonlóan, nem csak a böngésző és a szerver közötti kapcsolat fog kódolódni, hanem egy független ellenőrzőhelyet is engedélyeztek. Így a böngészők a címsorban csak az így engedélyezett, biztonságos oldalakat jelenítik meg zöld színben. Ezt azonban egyelőre még egyetlen webhely-üzemeltető sem alkalmazza.

1.3.4. Brute force-támadás

A brute force-támadás (nyers erő), más néven a teljes kipróbálás módszere, egy, a titkosító rendszerekkel szemben alkalmazott támadási mód, ami elvileg mindig eredményes.

Működésének lényege, hogy a rejtjelező rendszer ismeretében az összes lehetséges *kulcsot* kipróbálva határozza meg az alkalmazott *kulcsot*. Eredményességét csak a műszaki (informatikai) háttér és a rendelkezésre álló idő határozza meg. Gyors és nagy kapacitású hardverre (célhardverre) van szükség. A törési idő függ a lehetséges *kulcsok számától,* azaz

kulcs méretétől (hosszától) és bonyolultságától (választható karakterek száma). Nehézséget okoz, hogy a kipróbált kulcsról eldöntsük jó-e vagy rossz.

$$\text{Lehetséges kulcsok száma} = (\text{karakterek száma})^{\text{kulcs hossza}}$$

Védekezésként a kulcs gyakori cseréjével a támadót folyamatos kulcskeresésre kényszeríthetjük. Vagy könnyen készíthetünk olyan kulcsot, amely feltörése akár több évet is igénybe vesz. Az ilyen titkosítás gyakorlatilag feltörhetetlen.

1.4. Belső támadások

1.4.1. Fork bomba

A fork bomba egy olyan program, amely a CPU időt és memóriát terheli túl, ezáltal jelentősen lassítja a számítógép működését. Védekezni ellene erőforrás limitek beállításával lehet. Az újabb Linux disztribúciók és Windows operációs rendszerek már alapból PAM-ot használnak, így ez ott nem probléma.

1.4.2. Programhibák

Az adott kiszolgáló program vagy az operációs rendszer hibája segítségével akadályozható a program rendeltetésszerű működése. A hiba lehet például egy távolról kihasználható buffer overflow, amelynek következtében a program illegális műveletet hajtana végre vagy számára nem hozzáférhető memóriaterületre írna, így az operációs rendszer a futását megszakítja.

Elképzelhető olyan hiba is, amit felhasználva a program egy része vagy egésze lefagyasztható vagy hibás működésre bírható. A kernel hibája is veszélyeztetheti a stabilitást.

Védekezni a programok helyes beállításával és a megfelelő upgrade-ek elvégzésével lehet.

1.4.3. Buffer overflow (stack overflow)

Ha programozási hiba folytán egy adott buffer a méretén 'túlírható', akkor a tömböt tartalmazó függvény visszatérési címe (a stack-en a buffer vége után tárolódik) átírható.

Ennek oka az, hogy a C compilerek fordításkor nem figyelik a tömbhatár-túllépés lehetőségeit, illetve léteznek olyan függvények, melyek a bufferbe írás során nem ellenőrzik annak méretét. Ez nem lenne gond, mert minden ilyen függvénynek létezik biztonságos (a tömbhatárokat ellenőrző) változata, csakhogy könnyű a rosszabb megoldást választani.

Így azokon a rendszereken, ahol a stack futtatható, érdekes dolgokat művelhetünk (pl. a futó program jogaival shell hívható a visszatérési cím egy 'exec("/bin/sh"...)'-ra állításával). Ha a program setuid/setgid-es, akkor az kód, amire a visszatérési cím mutat, örökli a futó program jogait, tehát pl. az előbbi esetben egy setuid-es shell-t kaphatunk.

Súlyosabb a helyzet, ha az adott program egy hálózati kiszolgáló, mert bizonyos esetekben a hiba távolról is kihasználható, lehetőséget adva ezzel bárkinek a jogosulatlan hozzáféréshez.

A programozói gondatlanság ellen védelmet nyújt a StackGuard, amely egy GCC (GNU C Compiler) kiterjesztés. Érdemes használni, főként, ha valaki setuid-es programot fejleszt, de ha nem bízunk egy más által fejlesztett programban, az is újrafordítható vele.

A legjobb védekezés setuid/setgid-es programok ellenőrzése. Vannak programok, amelyeknek a korrekt működéshez kell, de néhánynak nem. Ezekről vegyük le. Néhány program futtatása biztonságosabb, ha valamilyen wrapper-t használunk, amely megnehezíti a hibák kihasználását.

1.4.4. Symlink attack

Known tmp filename attack-nak is nevezik. A probléma az, hogy egyes programok world-writable könyvtárba (pl. /tmp) írt tmp file-jainak neve kitalálható, tehát lehetséges a program indítása előtt létrehozni azon a néven egy linket, amely egy másik file-ra mutat. Ez különösen setuid/setgid bites program, illetve több privilégiummal rendelkező felhasználó vagy a rendszer által futtatott program esetén veszélyes, mivel így a symlink segítségével olyan file módosítható, amelyhez a linket létrehozó felhasználónak normális esetben nem lenne joga.

Megoldás lehet a TMPDIR környezeti változó használata és beállítása felhasználói profile-ba. A hibás programot pedig érdemes upgradelni, vagy lecserélni.

1.4.5. Race condition

Ez a symlink attack kiterjesztése arra az esetre, amikor a program ellenőrzi, hogy létezik-e már az adott tmp állomány, illetve nem symlink-e, de nem megfelelő módon nyitja meg (open használatakor az O_EXCL flag nélkül) vagy rosszul állítja be a hozzáférési jogokat. Ekkor a file ellenőrzése és megnyitása közötti időben létrehozhatunk ugyanazon a néven egy symlinket, amely egy általunk nem, de a program tulajdonosa/groupja (setuid/setgid program esetén) vagy használója számára olvasható/írható fájlra mutat, így már nekünk is jogunk lesz azt módosítani, esetleg még a hozzáférési jogai is átíródnak.

Ilyen jellegű hibák ellen hatásos védelem csak a megfontolt programírás és telepítés, illetve a symlink attack-nél említett TMPDIR környezeti változó beállítása lehet.

1.4.6. IFS (Inter Field Separator, mezőhatároló) megváltoztatása

Az IFS az egymás után következő karaktersorok (utasítások, paraméterek, stb.) elválasztására szolgál. Ennek átírása setuid/setgid-es programok esetén használható ki, ha a program írója nem kellő körültekintéssel használta a *system()* vagy az *exec()* függvényhívásokat. Egy *system("/bin/akarmi")* függvényhívás, ha kiadtuk az "*export IFS='/'*" utasítást, a következőképp hajtódik végre: "*bin akarmi*". Így tehát bárki elhelyezheti saját "*bin*" nevű programját a PATH-ban. Szerencsére ez nem egyszerű, mivel a PATH-ban lévő könyvtárak egy közönséges felhasználó számára általában nem olvashatók.

2. Csomóponti védelmi lehetőségek, feladatok, eljárások

Az alábbiakban a számítógépes csomópontokon, munkaállomásokon alkalmazható – elsősorban az operációs rendszerek által kínált és az alkalmazások használatához kapcsolódó – néhány egyszerű, de nem elhanyagolható védelmi lehetőségről lesz szó. Olyan alapvető szabályok ezek, amelyek kéznél vannak (vagy könnyen beszerezhetőek), viszonylag egyszerűen konfigurálhatók és alkalmazhatók, ugyanakkor sok további kellemetlenségtől kímélhetik meg a felhasználót.

A szemléltető ábrák a Windows XP operációs rendszerhez kötődnek, de más operációs rendszerek is hasonló lehetőségeket nyújtanak.

2.1. Jelszavak megválasztása

A legalapvetőbb védekezési módszer minden számítógépes erőforrás esetében abból áll, hogy azonosítjuk azt a személyt, aki az erőforrást használni szeretné; ehhez általában bekérünk egy azonosítót és egy titkos jelszót. Nem szabad megengedni, hogy a jelszó túlságosan egyszerű legyen, mivel ez komoly veszélyt jelent a biztonságra, ugyanakkor a jelszó nem lehet megjegyezhetetlenül bonyolult sem, mivel ez lehetetlenné teszi az adott fiók használatát.

A rövid vagy szótárból kikereshető szavak és szóösszetételek elég könnyen megtalálhatóak, megfejthetőek. Könnyen megfejthető az olyan jelszó is, amely magát a felhasználói nevet is magában foglalja. Ezeket a variációkat percek, de legrosszabb esetben órák alatt dobja ki egy jelszófejtő program. (Amit érdemes néha lefuttatni, ha a password file rossz kezekbe kerül.)

A legtöbb jelenlegi rendszer már alapértelmezésben használ szótárakat és különféle algoritmusokat a megfelelően bonyolult jelszó kiválasztásához, így általában nem is fogadja el a „könnyen megfejthetők” egyikét sem.

A megfelelő jelszó választása mellett gyakori hiba a felhasználók felelőtlensége is. Nagyon fontos a jelszavaik védelme és biztonsága, nem szabad azokat átadni senki másnak. Mindemellett ajánlatos a shadow password használata. Így /etc/passwd file - amely mindenki számára olvasható - nem tartalmaz jelszavakat, azokat a /etc/shadow file-ban tárolja a

rendszer, hozzáférési joga pedig csak a root-nak van.

A különböző rendszerek általában visszafordíthatatlanul titkosított formában tárolják a jelszavakat. A visszafordíthatatlanság itt azt jelenti, hogy az eredeti jelszó nem állítható elő a titkosítottból. Ennek a módszernek az egyik előnye, hogy a kalóz még a titkosító eljárás és a titkosított jelszó ismeretében sem tud egyszerűen hozzájutni az eredeti jelszóhoz. Azonban számos jelszótörésre alkalmas módszer létezik, ha a kalóz ismeri a titkosító eljárást (a legtöbb hálózati szolgáltatást nyújtó programnál ehhez könnyen hozzá lehet férni).

2.2. File-ok változásainak figyelése

A legkörültekintőbb óvintézkedések mellett sem lehetünk biztosak abban, hogy nem lehet betörni gépünkre. A behatoló viszont, - ha kellően ügyes és nem egyenesen rombolni akar - nem hagy feltűnő nyomokat, viszont készít magának egy kiskaput, hogy később könnyedén hozzáférjen a gépünkhöz. Ez lehet egy megváltoztatott 'passwd' file, esetleg egy átírt 'login', stb. Ezeket szűrhetjük ki, ha a file-ok változásait figyelő programot (pl. tripwire) használunk, amely változáskor figyelmeztetést ad. Ajánlott az általa generált file-t lemezre menteni és biztonságos helyen tárolni. Ha bármilyen gyanús dolog történik, elővehető és a gépen lévővel összehasonlítható.

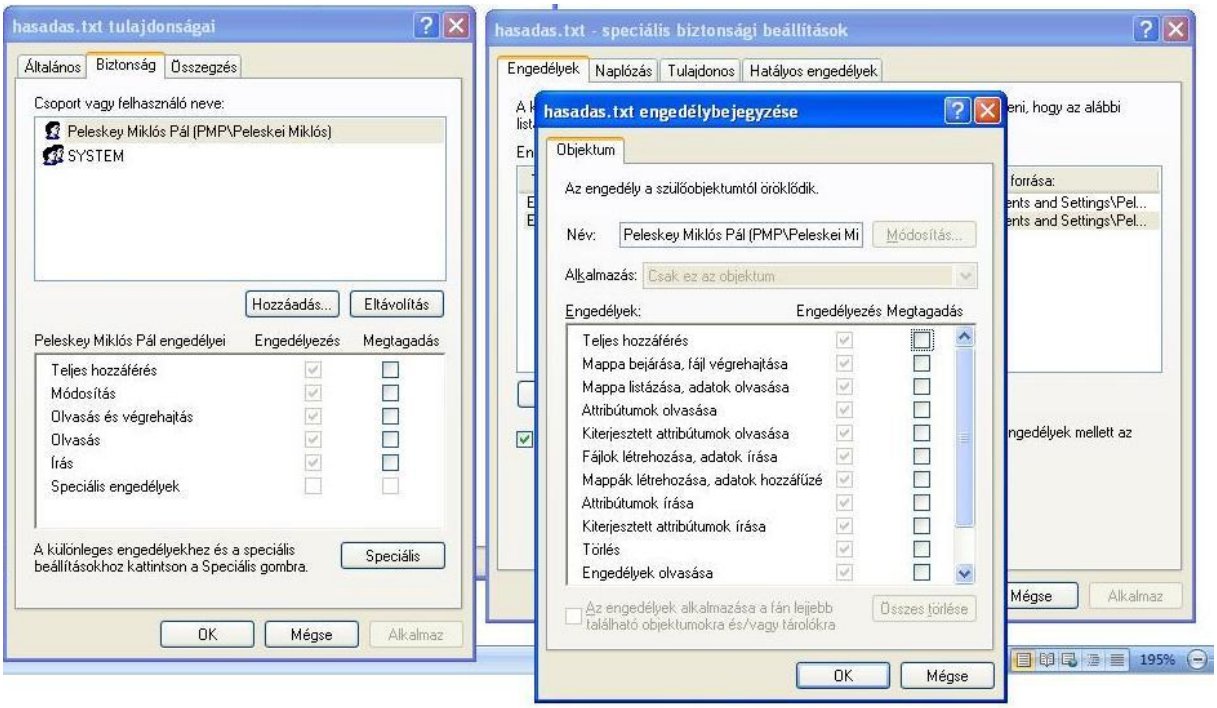
2.3. PAM (Pluggable Authentication Module)

Ez egy egységes authentikációs rendszer, feladata a felhasználók bejelentkezésével, memória és fájlrendszer használatával, valamint az authentikációs szolgáltatásokat nyújtó programokkal kapcsolatos beállítások kezelése. A felhasználókra vonatkozó beállításokat az operációs rendszerek beépített szolgáltatásként kínálják.

2.4. Mappa- és fájljogosultságok, attribútumok

A mappa- és fájl attribútumok (rejtett, írásvédett stb.) megfelelő beállítása és a felhasználók számítógépes erőforrásokhoz való hozzáférési jogosultságainak (olvasási, írási, módosítási, végrehajtás stb. jogok) együttese jelenti adott felhasználó tényleges jogosultsági, hozzáférési rendszerét. További szabályozási lehetőség az egyes feladatokhoz is kötődő felhasználói csoportok kialakítása a megfelelő jogosultsági rendszerrel, és az erőforrások megosztásának szabályozási rendszere.

Az alábbi ábra adott felhasználó (felhasználói csoport) adott objektumhoz (mappa, fájl stb.) való hozzáférési jogosultságainak egyfajta szabályozási lehetőségét szemlélteti.



2.5. Adatvédelem, fájlvédelem

Érzékenyebb adatok használata esetén célszerű használni az egyes alkalmazásokban felkínált jelszavas védelmet, amely kiterjedhet a fájl egészére, vagy annak különböző részeire. A trójaik elleni védelem céljából érdemes a makrók használatát letiltani, illetve ellenőrzötten használni azokat.

2.6. Frissítések

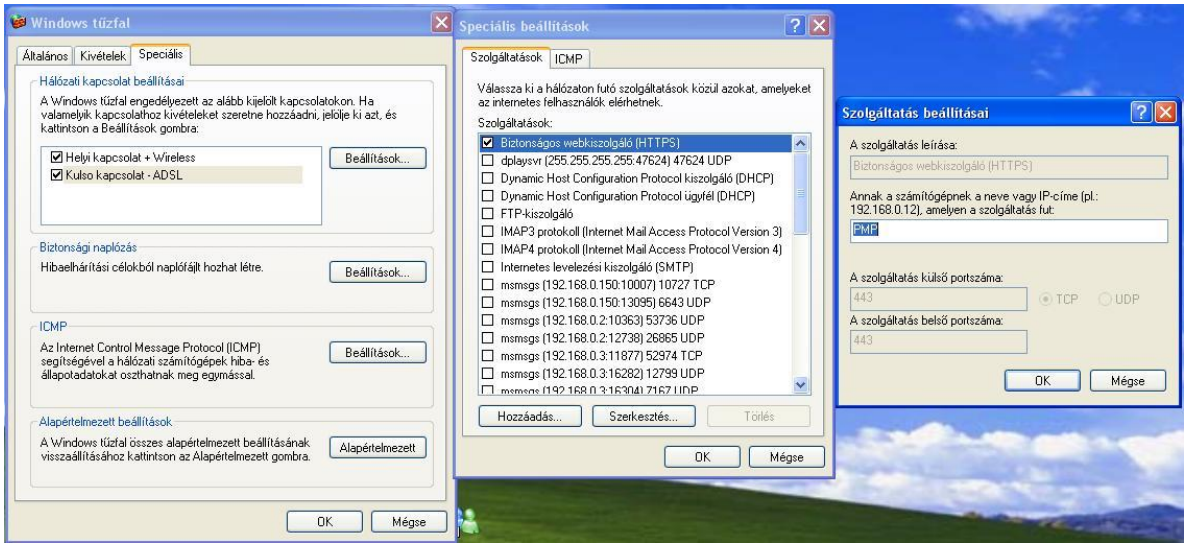
Mind a számítógép operációs rendszere, mind pedig a számítógépre telepített programok, alkalmazások hibái, hiányosságai, biztonsági rései szolgálhatnak támadási felületként. Ezeket a hibákat általában a szolgáltatók, gyártók által készített programfrissítésekkel lehet és érdemes megszüntetni. A frissítések Interneten való keresése és a letöltések automatikussá tehetők.

2.7. Helyi védelmi és biztonsági rendszer

A számítógépen használt operációs rendszerekbe beépített biztonsági és vírusvédelmi rendszer használata, a helyes beállítások alkalmazása nagyfokú védelmet nyújthat a támadásokkal szemben. Vonatkozik ez a fájlvédelmi beállításokra, a felhasználói jogosultságokra, a gépen futtatható szolgáltatások (ftp, web, mail stb.) tiltására (vagy engedélyezésére), kommunikációs portok tiltására (engedélyezésére), biztonságos portok és protokollok használatára egyaránt.

Az operációs rendszer kínálta lehetőség természetesen helyettesíthető egyéb telepített védelmi rendszerekkel.

A Windows XP integrált biztonsági központjának felülete.



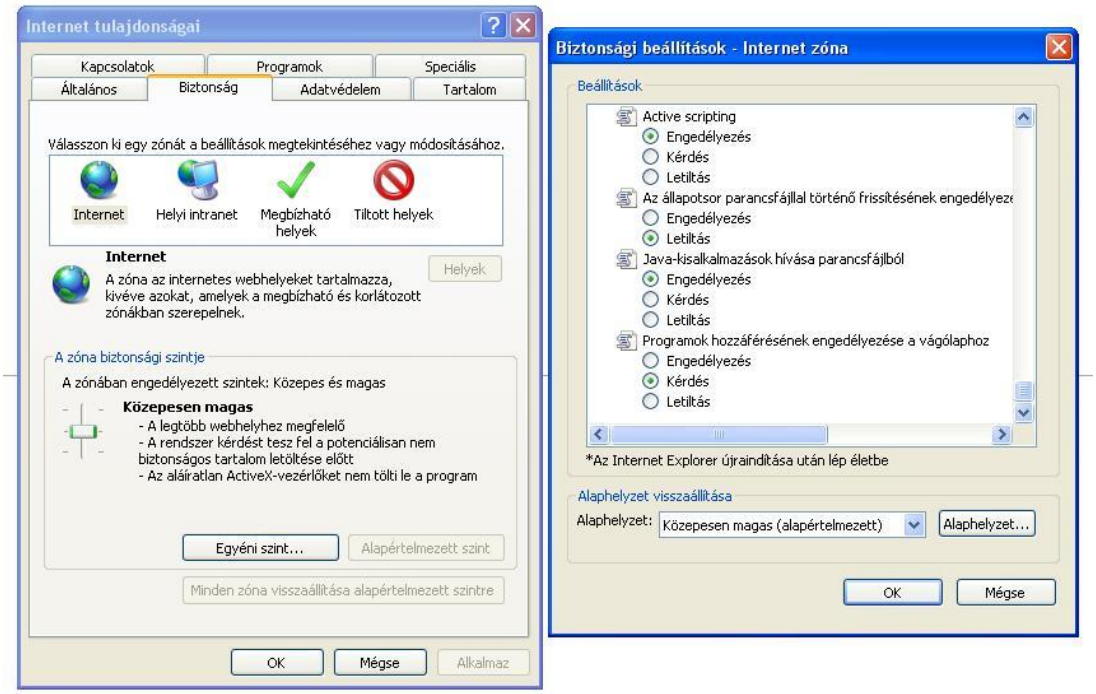
A Windows XP tűzfal beállítási lehetősége.

2.8. Programok telepítése, helyes használata

Nagyon fontos, hogy a számítógépre telepített alkalmazások biztonsági szempontból is megfelelően legyenek konfigurálva. Nagy figyelmet kell fordítani a programok opcióinak helyes beállítására. Különösen is a levelező kliensek és web-browserek helyes beállításával és megfelelő használatával számos kellemetlenségtől kímélhetjük meg a gépünket és magunkat. Az már csak természetes, hogy a nem jogtiszta szoftverek használata nem csak törvénybe ütköző cselekedet, hanem biztonsági szempontból gyakran elégtelen, és nagyban hozzájárul a károkozók terjedéséhez.

2.9. Letöltések, levelezés, böngésző

Talán a legnagyobb veszélyforrás az internetes alkalmazások (levelezés, webböngészés, ftp stb.) használata során a számítógépre kerülő károkozó programok (elsősorban vírusok, kémprogramok). Az ezek ellen való védekezés egyszerű és alapvető lehetősége a felhasználó figyelmén és tájékozottságán is múlik. A számítógépre telepített védelmi rendszer (vírusirtó program, adathalászat szűrő, spamszűrő stb.), valamint e programok megfelelő beállítása elengedhetetlen a biztonságos működés érdekében.



A böngésző biztonsági rendszerének beállítása.

2.10. Hitelesítő, beléptető rendszerek, hardver eszközök

A felhasználói azonosítás és hitelesítés egyre elterjedtebb kiegészítő módszere a különböző beléptető kártyák (intelligens kártyák), tokenek használata, vagy éppen az ujjlenyomat-, hang-, vagy arcfelismerés alkalmazása.

Az intelligens kártyáknak széles a választéka. A hagyományos mágnescsíkos, a memóriachipes és közelítőkártyák mellett legbiztonságosabbak a mikroprocesszorral ellátott kártyák. Ezek az adattárolás mellett az adatok feldolgozására is képesek, jól használhatóak kriptográfiai műveletek támogatására és gyakorlatilag nem nyerhető ki a rajtuk tárolt titkos információ.

A tokeneknek két típusa használatos: szinkron, aszinkron. A szinkron tokenek a token és a számítógép közös titkos kulcsát belső számlálóval elő, vagy a szinkronizált időt használják erre. Az aszinkron tokenek kihívás-válasz (challenge-response) alapon működnek, a számítógép és a token közös kulcsot használ. A szerveren megjelenő kihívást beírva a tokenbe, megjelenik a válasz.

Egyre több helyen használják a személyre jellemző biometrikus tulajdonságok alapján történő hitelesítési technológiát. Ide tartozik az ujjlenyomat, retina, írisz, hang, tenyér, aláírás, arc minta alapján történő felismerés.

2.11. További szoftveres védelmi megoldások

Ebbe a körbe sorolhatók mindazok a számítógépre telepíthető programok, amelyeknek feladata a csomóponti védelem: tűzfal programok, anti-vírus, anti-spam, anti-spyware programok. Ezek ma már nem csak külön-külön telepíthetők és alkalmazhatók, hanem többségükben integrált rendszerként jelennek meg, komplett védelmi technológiákat kínálva. Legtöbbjük rendelkezik központi telepítési és menedzsment lehetőséggel, biztosítva ezáltal az érintett informatikai rendszer, a helyi hálózat gépeinek egységes helyi védelmét, a rendszeres program és adatbázis frissítéseket, a központi felügyeletet.

3. Védelmi technológiák rendszere

A védelmi és biztonsági technológiák alapvetően a számítógépes hálózat védelmére szolgáló hardveres és szoftveres megoldások alkalmazását jelentik, amelyek alkalmazásának célja a támadások kockázatának csökkentése. A számítógépes hálózatok, csomópontok hatékony védelmét többszintű biztonsági rendszerrel érdemes megvalósítani, amelynek része a tűzfal, a behatolás-megelőzés és az átjáró alapú antivírus technológiák. Az egyes technológiák alkalmazása függ a várható veszélyektől, vagyis attól, hogy az adott hálózat milyen védelmi rendszert igényel. Az általuk nyújtott biztonsági megoldások egymást kiegészítik, igazán hatékony megoldást együttes, integrált alkalmazásuk jelent.

3.1. A védelmi technológiák hálózati biztonsági szintek szerinti csoportosítása

- Csomagszintű védelem, pl. az útválasztók hozzáférési listája (ACL - Access Control List) vagy más néven az állapotmentes tűzfalak.
- Kapcsolatszintű védelem, pl. az állapottartó csomagszűrő tűzfalak.
- Alkalmazásszintű védelem, pl. proxy tűzfalak, behatolásmegelőző rendszerek (IPS).
- Fájlszintű védelem, pl. átjáró alapú antivírus rendszerek.

Az alábbi ábra a négy hálózati biztonsági technológiát hasonlítja össze. Az egyes kategóriák védelmi szintje, viszonylagos teljesítménye és aszerinti értékelése, hogy hány protokollt/alkalmazást fednek le, lehetővé teszi, hogy a különféle szervezetek saját hálózatuk védelméhez a megfelelő technológiákat válasszák ki.

	Csomagszintű védelem	**Kapcsolat szintű védelem**	**Alkalmazás szintű védelem**	**Fájl szintű védelem**
Példa	egyszerű csomagszűrés (pl. router ACL listája, állapotmentes tűzfal)	állapottartó csomagszűrő tűzfalak	behatolásmege-lőző rendszerek (IPS) és proxy tűzfalak	átjáró alapú vírusszűrők
Működés	a csomagok fejlécét vizsgálja	a csomagok fejlécét és a vezérlő	a csomagok tartalmát is vizsgálja	a forgalmon belül lévő fájlokat

		információkat vizsgálja		vizsgálja
Protokoll- és alkalmazás védelem	nincs, csak csomagszint	széleskörű	korlátozott körű	szűkkörű (pl. e-mail, web, ftp)
Védelem	kétirányú	kétirányú	jellemzően a szervereket védi a kliensektől	jellemzően a szervereket védi a kliensek felől
Teljesítmény	magas	magas	közepes	lassú

3.2. Csomagszintű védelem

A csomagszűrésként is ismert csomagszintű védelem egyike a hálózathoz való csatlakozás ellenőrzésére szolgáló legelterjedtebb módszereknek. Az elv egyszerű: a csomag fejlécében szereplő alapinformációk összehasonlításával megállapítani, hogy a csomag engedélyezett-e. A Cisco IOS Access Control List (ACL) egyike a legelterjedtebb csomagszűrőknek. Az IPTables (illetve korábban az IPChains) szintén egy népszerű alkalmazás, melyet szinte minden Linux verzió magában is foglal.

A kétirányú kommunikáció a csomagszűrésen alapuló hálózati biztonsági rendszereknek nagy feladatot jelent. Ha valaki blokkolja a teljes bejövő forgalmat, ezzel megakadályozza, hogy a kimenő üzenetekre érkező válaszok beérkezzenek, és a kommunikáció megszakad. Ezért két rést is szabadon kell hagyni, egyet a kimenő, egyet pedig a bejövő forgalom számára az arra való legkisebb törekvés nélkül, hogy a bejövő forgalom a már meglévő hálózati kimenő kapcsolatokhoz társuljon. A csomagszűrés ekképpen beengedhet létező kapcsolatok részének álcázott rosszindulatú csomagokat, melyek így a védett erőforrásokban is kárt tehetnek.

A csomagszűrő eszközök nem követik a dinamikus protokollokat, amelyeknél a szerver és a kliens véletlenszerűen változó adatátviteli portokat használ az adatátvitelhez. Dinamikus portokat használó protokoll pl. az FTP, az RPC és a H.323. Hogy ezek az alkalmazások csomagszűrés mellett is működhessenek, igen nagy rést kell szabadon hagyni,

ami a csomagszűrés nyújtotta védelem szintjét jelentősen csökkenti. Például egy szabványos FTP működéséhez engedélyezni kell minden 1023-nál magasabb számú (1023 - 65500) portra irányuló és minden 20-as portról érkező forgalmat, ami igen nagy rést jelent a hálózat biztonsági rendszerén.

3.3. Kapcsolat szintű védelem

A kapcsolat szintű biztonsági technológiák a kapcsolat állapotát követve ellenőrzik a két vagy több hálózat között az adatforgalmat és az olyan csomagokat, amelyek az előre felállított biztonsági előírások szerint engedélyezett kapcsolatnak nem részei, kidobják. A kapcsolatszintű védelmet alkalmazó tűzfalak minden egyes hálózati kapcsolatról megőrzik az állapotinformációkat és egy kapcsolati állapottáblázat alapján lehetővé teszik az engedélyezés/tiltás eldöntését. A kapcsolatszintű védelem leginkább közismert rendszerei az állapottartó csomagszűrő tűzfalak.

Megjegyzendő, hogy a kapcsolatszintű védelmi technológiák "kapcsolat alapúak", tehát a tűzfal túlmegy az egyedi TCP kapcsolatokon, hogy sok ilyen kapcsolatot ellenőrizhessen. A kapcsolatszintű alkalmazások támogatják a dinamikus protokollokat, ehhez a kliens-szerver kommunikációban azonosítják a port változtatására vonatkozó utasításokat és összehasonlítják a megbeszélt kapu forgalmát a kapcsolat későbbi állapotával. Például az FTP kapcsolatok követésekor a tűzfal figyeli a parancsok kiadására és a dinamikus portok váltásának megbeszélésére szolgáló kontrollkapcsolatot, ennek alapján engedélyez az adat- és fájlforgalom számára különféle kapcsolatokat.

Mivel a kapcsolatszintű védelem a csomagszintű védelem minden előnyével bír annak hátrányai nélkül, általa a legtöbb hálózaton szükségtelen a csomagszintű védelem alkalmazása.

3.4. Alkalmazásszintű védelem

Az alkalmazásszintű védelmi technológiák a hálózat forgalmát figyelik és támadások vagy behatolások jelei után kutatva elemzik azt. A hálózat biztonsági infrastruktúráján belül két közismert technológia szolgálja az alkalmazásszintű védelmet: a proxy tűzfal és a behatolásmegelőző rendszer (IPS).

A proxy tűzfalak olyan hálózati rendszerek, amelyek a hálózati szolgáltatáshoz kapcsolódó kliens "nevében" dolgoznak, és meggátolják, hogy a kliens és szerver közvetlen fájlcserélő kapcsolatba lépjen. A kliens létrehoz egy kapcsolatot a proxy szerverrel, a proxy szerver pedig létrehozza a kapcsolatot a célszerver felé. Ezt követően a proxy továbbítja az adatokat a felek között.

Az IPS olyan hálózati eszközt jelent, amely az IP cím, protokoll/szolgáltatás vagy az alkalmazásszintű elemzés és ellenőrzés alapján engedélyezi vagy tiltja az adatforgalmat. Az IPS fogadja a hálózat forgalmát, összegyűjti és újraegyesíti az adatfolyamokat; átvizsgálja az alkalmazások szabványszerűségét és parancsait, hogy felderítse azokat a gyanús jeleket, melyek az előre meghatározott intézkedéseket indokolttá teszik. Az intézkedések a gyanús esemény naplózásától a kapcsolat teljes bontásáig terjedhetnek.

A proxy tűzfal és az IPS ellenőrzésképpen megvizsgálja az alkalmazásfolyamon belüli vezérlő- és adatmezőket, hogy az egyes műveleteket a biztonsági rendszabályok megengedik-e, és hogy nem jelentenek-e valamilyen veszélyt a védett rendszerre nézve. Az alkalmazásszintű parancsok és formátumok megértésével mindkét eszköz meg tudja különböztetni a legitim forgalmat a rosszindulatú tartalmaktól. A proxy tűzfalak és az IPS rendszerek IP töredezettség-mentesítést és TCP adatfolyam újraegyesítést hajtanak végre, valamint egyértelművé teszik az adatforgalmat, ezáltal megakadályozzák, hogy az IP protokoll forgalmi sajátosságait a rosszindulatú felhasználók tevékenységük leplezésére használják ki.

A proxy tűzfalak rendszerint támogatják az olyan általános internet alkalmazásokat, mint pl. a HTTP, az FTP, a telnet, az rlogin, az e-mail és az news. Azonban új proxyt kell fejleszteni minden egyes új alkalmazás és protokoll bevezetésekor, hogy azok átjuthassanak a tűzfalon, továbbá minden egyes alkalmazáshoz egyedileg testre szabott szoftverekre és eljárásokra van szükség.

Az IPS általában az alkalmazások és protokollok szélesebb körét támogatja, ezek közé tartoznak a hálózatot az internet felől érkező támadások ellen védő alkalmazások is. Az IPS révén úgy lehet új alkalmazásokat engedélyezni, hogy ehhez nem kell a felhasználó munkaállomásán változtatásokat eszközölni. Ily módon az IPS a belső hálózat felé

"átlátszóbb", mint a proxy tűzfalak - sok IPS ráadásul L2 (Layer 2) üzemmódban is képes működni, ami tovább egyszerűsíti integrációjukat.

A proxy tűzfalak és az IPS egyes vírusokat és trójai falovakat az alkalmazások szervizmezőit áttekintve képesek felderíteni. Például az IPS az ismert vírusok jellemzőinek felderítéséhez meg tudja nézni az e-mail forgalom tárgy mezőjét, a csatolmány nevét és típusát. Mindazonáltal az alkalmazásszintű védelem nem végez részletes fájlszintű elemzést, ami pedig szintén szükséges sok vírus felderítéséhez.

3.5. Fájlszintű védelem

A fájlszintű védelem képes az adatforgalomban résztvevő fájlokat kitömöríteni és megvizsgálni, hogy nem tartalmaznak-e valamilyen rosszindulatú programkódot, pl. vírust, férget, trójai falovat. A fájlszintű védelem egy közismert technológiája az átjáró alapú antivírus rendszer.

Az antivírus rendszerek vírusmintákat - a vírust azonosító egyedi bájtsorozatokat - keresnek, és megtisztítják a fájlt a vírustól. A legtöbb antivírus rendszer nem csupán az eredeti vírust, hanem annak számos változatát is felismeri, mivel maga a minta rendszerint változatlan marad. Emellett pedig heurisztikus módszereket is alkalmaznak a még ismeretlen vírusok felderítésére - sok esetben rendkívül hatékonyan.

Az átjáró alapú antivírus rendszerek a hálózati forgalomba beágyazott fájlokat vizsgálják át, pl. a HTTP forgalom (letöltések) fájljait, e-mailek csatolmányait. Amennyiben az átjáró oldali antivírus rendszer fertőzött fájlt talál, kiveszi az adatforgalomból, így az nem befolyásolja a többi felhasználót. A hálózati forgalomban részt vevő fájlok átvizsgálásához az átjáró alapú antivírus rendszernek a kódolási protokollok (pl. MIME, unicode, Base64) és tömörítési algoritmusok széles skáláját kell ismernie.

Mivel az ismert vírusok száma több mint 80 000-re rúg, az átjáró oldali antivírus rendszernek igen alapos vizsgálatokra kell képeseknek lennie. Az új vírusok elleni hatékony védelemhez a vírusleíró fájlok folyamatos frissítésére van szükség, hiszen folyamatosan jelennek meg új vírusok.

Mivel a vírusok miatt átvizsgált alkalmazásfolyamokat az átjárói antivírus rendszernek teljes egészében újra kell egyesítenie, amikor a forgalom átmegy a hálózaton, a felhasználók vagy szerverek az átvizsgált folyamokban rövidebb-hosszabb késleltetést észlelhetnek. A rendszergazdák rendszerint részletekbe menően kontrollálhatják a forgalmat és azt, hogy mely fájlok vizsgálata indokolt.

Az antivírusok főként az e-mail- és hálózati forgalom fájljait ellenőrzik, rendszerint a szervertől a kliens felé történő kommunikációt figyelik. A vírusok célja a végfelhasználói rendszerek károsítása, de terjedésükhöz számos e-mail- és hálózati szervert használnak. Ebből adódóan fontos a vírusokat már a szerverről/re való le/feltöltés pillanatában detektálni.

3.6. A védelmi technológiák kapcsolata, egymásra épülése

Az egyes szervezetek a vírusok, férgek és egyéb támadások elleni védekezés különböző rétegeiként használják a hálózati biztonsági technológiákat hálózati, kapcsolati, alkalmazás és fájl szinten. Mivel a kapcsolatszintű védelem a csomagszintű védelem minden előnyével bír annak hátrányai nélkül, általa a legtöbb hálózaton szükségtelen a csomagszintű védelem alkalmazása.

Az alábbi ábra mutatja a technológiák egymásra épülését, illetve hogy milyen ellenőrző funkciók lépnek életbe:

- kapcsolatszintű védelemnél az állapottartó tűzfal elemzi a csomagokat
- alkalmazásszintű védelemnél a behatolásmegelőző rendszer (IPS)
- fájlszintű védelemnél az átjáró alapú antivírus látja el a védelmet

<table>
<tr><td>fájlszintű védelem</td><td>átjáró védelem</td><td colspan="2">fájlok vizsgálata: fájl1, fájl2, fájl3, …</td></tr>
<tr><td rowspan="2">alkalmazásszintű védelem</td><td rowspan="2">behatolás megelőzés (IPS)</td><td>protokoll szabványszerűség</td><td>alkalmazás elemzés</td></tr>
<tr><td colspan="2">összeállítás, normalizálás, értelmezés</td></tr>
<tr><td>kapcsolatszintű védelem</td><td>állapottartó csomagszűrés</td><td colspan="2">kapcsolatok figyelése: kapcs1, kapcs2, kapcs3, …</td></tr>
<tr><td>hálózat</td><td>csomagok</td><td colspan="2">csomag1 |csomag2 | csomag3 | …</td></tr>
</table>

A csomagok az Internet Protokoll sajátosságai miatt szinte teljesen tetszőleges sorrendben érkezhetnek. Ideális esetben minden csomagot hozzá tud a biztonsági rendszer rendelni egy kapcsolathoz, vagy különben eldobásra kerül. Ideális esetben minden kapcsolatban használt protokollt felismer a biztonsági rendszer, hogy az alkalmazásszintű védelem megfelelő legyen, illetve a nem felismerhetőeket letiltja. Azon protokollokat pedig, amelyek jellemzően fájlokat tartalmaznak, a fájlszintű elemzésnek is aláveti az átjáró védelem. Azon fájlok, amelyek nem megfelelő (vagy nem szabványos) tartalmat hordoznak, megtisztításra vagy eldobásra kerülnek.

4. Biztonsági területek – védelmi technológiák

4.1. Hálózatközeli biztonság

A hálózatközeli biztonság az informatikai adatátviteli közegek és a rajtuk továbbított adatok biztonságát és védelmét jelenti. Bele tartozik ebbe a lehallgatásbiztos üvegszálas kábelezés, a virtuális LAN-ok kialakítása, a felhasználóhoz kötött hálózati azonosítás vagy a titkosított adat- és hangátvitel, a biztonságos wireless (WIFI) kommunkáció vagy a távmunka-támogatás.

4.2. Határvédelem, szegmentálás

Az informatikai rendszerek (számítógépek, LAN-ok, VLAN-ok, szegmensek) kapcsolódási felületeik (hálózati interface-k, switchek, routerek) révén kerülnek egymással összeköttetésbe, a rendszerek közötti adatátvitel pedig vezetékes vagy vezeték nélküli adatátviteli közegen át történik. A kapcsolódási felületek jelentik a rendszerek fizikai határát, amely védelmi szempontból stratégiailag is fontos terület. Az informatikai rendszer határvédelmét ellátó eszköz a tűzfal, melynek feladata a belső hálózat és a külvilág – speciális esetben a LAN hálózati szegmensek – között történő forgalom szabályozása, ellenőrzése, szűrése. A számítógép közvetlen határvédelmét ellátó eszköz lehet a gépen használható hardveres vagy szoftveres tűzfal, valamint a különböző autentikációs és IDS/IPS megoldások együttese.

A tűzfal-technológia régi múltra tekint vissza, ennek ellenére a mai napig képes érdekes szakmai feladatot jelenteni egy-egy komplexebb tűzfalrendszer megtervezése, a központi adminisztrálás vagy éppen különböző gyártók termékeivel való együttműködés biztosítása.

Nagyobb vagy biztonsági szempontból kritikus lokális hálózatokban az egyes hálózatrészek egyszerű tűzfalas elválasztása helyett célszerűbb a komplexebb, szegmentációs technikákkal történő szeparáció alkalmazása. A szegmentálás lényege, hogy a különböző biztonsági szintű részrendszerek egymástól – lehetőleg az ISO/OSI réteg egész mélységében – történő elválasztásával ellenőrzött, és csak a szükséges mértékű hozzáférést, illetve adatkommunikációt biztosítunk az egyes elemek között.

Az egyes zónák közti információáramlás szűrhető, korlátozható, megfigyelhető és szabályozható a megfelelő informatikai biztonsági eszközök alkalmazásával. Ezek az eszközök lehetnek maguk a routerek, switch-ekbe helyezett tűzfalkártyák, önálló tűzfalak, behatolásérzékelő és – megelőző (IDS/IPS) eszközök.

4.3. Behatolásvédelmi rendszerek (behatolásdetektálás és -megelőzés)

A külső támadások egy része ellen tűzfalakkal lehet védekezni, de a belső hálózaton történő támadások, illetve illetéktelen hozzáférési kísérletek érzékelésére és elhárítására mindenképpen szükség van behatolásérzékelő, illetve – megelőző (IDS/IPS) eszközökre.

Az ilyen rendszerek biztonsági szempontból nélkülözhetetlen információt biztosítanak a leterheléses támadásokról, a hálózaton kommunikálni próbáló trójai falovakról, a jogosulatlan belépési kísérletekről és egyéb, az alkalmazások vagy az operációs rendszer sérülékenységének kihasználására irányuló tevékenységről.

Az eszközök gyanús tevékenység érzékelése esetén képesek automatikusan beavatkozni a folyamatba (pl. adatkapcsolat megszakításával, vagy a felhasználói account felfüggesztésével), ezáltal a hálózatnak valósidejű védelmet biztosítanak.

A különböző gyártók termékei széles operációsrendszer-támogatottsággal rendelkeznek, így heterogén hálózati környezetben is implementálható teljes körű behatolásérzékelési és -védelmi rendszer.

4.4. Vírusvédelem, tartalomszűrés

A hagyományos, kizárólag a kliensoldali vírusvédelemre, így a felhasználókra támaszkodó megközelítés vállalati szinten nem képes hatékonyan biztosítani a vírusmentes informatikai környezetet. Az egyre kifinomultabb, rendkívül gyorsan terjedő és sokoldalú vírusok ellen csak a többlépcsős, minden veszélyeztetett belépési pontot felügyelő rendszer nyújthat elégséges védelmet. Ezért a vírusvédelmi rendszereknél minden potenciális sérülékenységi pontot le kell fedni a munkaállomásoktól a fájlszervereken keresztül a különböző internetszolgáltatások, különösen is a levél- és webforgalom szűréséig.

Bár a vírusok jelentik a legkézzelfoghatóbb fenyegetést, a tartalomszűrési funkció jól kiegészítheti a hagyományos vírusszűrést. A vírusok és egyéb rosszindulatú programok elleni

védekezés mellett így lehetőség nyílik a nem kívánt, bizalmas tartalmú vagy titkosított levelek, a túl nagy vagy nem engedélyezett típusú csatolt állományok ellenőrzésére, a nem produktív honlapok és a webes forgalom korlátozására.

A különböző vírusvédelmi és tartalomszűrő rendszerek széleskörű beállítási lehetőséget biztosítanak, a központi menedzselhetőséget biztosító modullal rendelkeznek, módosíthatóak, beállított időszakokra, a felhasználói csoportokra külön érvényesíthetők, és a felhasználói manipulációktól védettek.

4.5. Központi felhasználó- és jogosultságadminisztráció (IDAM)

Minden heterogén informatikai környezetben felmerül a felhasználók adminisztrálásának, a jogosultságok átláthatatlanságának problémája. A megoldást jelentő központosításnak elsősorban erőforrás-megtakarítási és biztonságpolitikai motivációja van, de a pénzintézeti szektorban ez törvényi előírás is.

Az IDAM rendszerek bevezetése két jól elkülöníthető, önmagában is értéket létrehozó fázisból áll. Az előkészítő szakaszban mérjük fel a vállalat felhasználó-adminisztrációval kapcsolatos folyamatait és szabályzatait, az informatikai rendszereket. A konszolidáció a folyamatok egységesítésével, a felhasználók jogosultsági szerepkörökbe sorolásával, az alkalmazások jogosultságprofiljainak kialakításával folytatódik. A második szakaszban történik az IDAM termék konkrét implementációja, a workflow folyamatok leprogramozása, a háttérrendszerekhez csatlakozást biztosító konnektorok és ügynökök kifejlesztése.

4.6. Elektronikus aláírás, PKI rendszerek

Az e-ügyintézés, a hiteles elektronikus dokumentumok kezelésének nélkülözhetetlen eleme az elektronikus aláírások, illetve az ezt megvalósító nyilvános kulcsú infrastruktúra (PKI) használata. Ez az infrastruktúra nemcsak elektronikus aláírások készítésére és ellenőrzésére, hanem erős azonosításra, levelek, állományok titkosítására, a dokumentum létrejöttét hitelesen igazoló időpecsételésre egyaránt használható.

4.7. Távoli hozzáférés, távmunka

Az adatkapcsolatok egy részében – jellemzően a telephelyek közötti bérelt vonalas kommunikációnál vagy a távoli dolgozó internetes belépésekor – „privát”, azaz védendő hálózatrészek nyilvános hálózaton keresztül kommunikálnak.

A virtuális magánhálózatok (VPN, Virtual Private Network) kialakításával el lehet érni, hogy ezek a kapcsolatok biztonságosak legyenek, az átvitt bizalmas információ ne kerüljön illetéktelenek kezébe. Virtuális magánhálózat kialakítására tehát olyan esetben van szükség, amikor a bizalmas hálózatokat úgy vagyunk kénytelenek összekötni, hogy az adat közben nem bizalmas hálózatokon is áthalad.

A VPN-hálózatok kialakítását érdemes integrálni a többi biztonsági elemmel (tűzfal, IDS/IPS, IDAM), és lehetővé tenni a központosított adminisztrálást, felügyeletet.

4.8. Naplófeldolgozás, biztonsági események kezelése

A legtöbb rendszer bőséges naplóállomány készítésére konfigurálható. Különösen igaz ez a biztonsági rendszerekre. A napló rengeteg eseményéből először ki kell válogatni a biztonsági eseményeket. Az ezek közötti kapcsolatok vizsgálatával lehet biztonsági incidensre utaló összefüggéseket keresni. Ez manuális eszközökkel szinte lehetetlen feladat, különösen, ha figyelembe vesszük, hogy több rendszerből származó naplóesemények korrelációját is érdemes vizsgálni (pl. gyanús esemény a tűzfalon, majd ezt követően egy szerveren).

Olyan megoldásokat célszerű alkalmazni, amelyekben a naplófeldolgozás nem csak a log állományok rendszeres archiválását jelenti, hanem a folyamatok automatizálását, a korrelációk programozott keresését. A biztonsági esemény-kezelés, illetve a naplózás kiterjedhet a felhasználók és adminisztrátorok munkájának biztonsági felügyeletére is, biztonsági auditok végzésére.

4.9. Biztonsági rendszerek távfelügyelete

A biztonsági rendszerek felől érkező riasztások javarésze nem fenyegetettséget vagy támadást jelent, hanem egy-egy ritka, de veszélytelen informatikai eseményt, amely semmilyen beavatkozást nem igényel. Az igazi támadások esetén viszont gyakran csak a legsürgősebb beavatkozás mentheti meg a rendszer működésképességét vagy az abban tárolt

adatok biztonságát. A nagy felelősséggel járó döntés a fenyegetettségek, sérülékenységek, rosszindulatú programok, új támadási módszerek naprakész ismeretét feltételezi.

A mai rendszerek bonyolultságára és a fenyegetettségek változási sebességére tekintettel általánosságban kijelenthető, hogy nem várható el a cég vagy szervezet informatikájának napi üzemeltetését ellátó, sokszor eleve túlterhelt szakemberektől a biztonsági kockázatok ilyen szintű ismerete.

Megoldás lehet erre biztonsági távfelügyeleti szolgáltatás igénybe vétele, amelyben erre a feladatra specializálódott szakembereink végzik el a kockázatok elemzését, javasolnak megelőző intézkedéseket a felügyelt hálózatban (pl. patch-elés, upgrade, stb.), döntenek a riasztások kezeléséről, az azonnali beavatkozás szükségességéről. A felügyelt biztonság a "csak riasztás" szintjétől a teljes kiszervezésig (felügyelet és menedzsment) jelent megoldást.

5. Tűzfal-technológiák

A tűzfal (*firewall*) az informatikai rendszerek határvédelmi eszköze, célja az informatikában annak biztosítása, hogy a hálózaton keresztül egy adott számítógépbe ne történhessen illetéktelen behatolás. Szoftver- és hardverkomponensekből áll. Hardverkomponensei olyan hálózatfelosztó eszközök, mint a router vagy a proxy. A szoftverkomponensek ezeknek az alkalmazási rendszerei tűzfalszoftverekkel, beleértve ezek csomag- vagy proxyszűrőit is. A tűzfalak általában folyamatosan jegyzik a forgalom bizonyos adatait, a bejelentkező gépek és felhasználók azonosítóit, a rendkívüli és kétes eseményeket, továbbá riasztásokat is adhatnak.

5.1. A tűzfal működése

A tűzfal megpróbálja a számítógépet, a privát hálózatot, illetve a hálózati szegmenst a nemkívánt támadásoktól megóvni. Szabályozza a különböző megbízhatósági szintekkel rendelkező számítógép-hálózatok közti forgalmat. Tipikus példa erre az internet, ami semmilyen megbízhatósággal nem rendelkezik és egy belső hálózat, amely egy magasabb megbízhatósági szintű zóna, azon belül pedig az egyes számítógépek, amelyek még magasabb megbízhatósági szintű rendszerek. A hálózaton belül egy közepes megbízhatósági szintű zóna az úgynevezett határhálózat, vagy demilitarizált zóna (DMZ), amit az internet és a megbízható belső hálózat között alakítanak ki. Alapjában minden szolgáltatást, amely a külső

hálózat felhasználóit látja el a DMZ-n belülre kellene helyezni. A leggyakoribbak ezek közül a web szerverek, mail szerverek, ftp szerverek és DNS szerverek.

A tűzfal megfelelő működéséhez helyes beállítás szükséges. A biztonsági szabványok „alapértelmezett-letiltás" tűzfal-szabálycsoportot határoznak meg, amelyben csakis azok a hálózatok vannak engedélyezve, amelyeket már külsőleg engedélyeztünk. Egy ilyen beállításhoz részletesen ismerni kell a hálózati eszközöket és azokat a végpontokat, amelyek az informatikai rendszer mindennapi működéséhez szükségesek. Sok vállalatnál hiányzik ez az ismeret, és ezért egy „alapértelmezett-engedélyezés" szabályt alkalmaznak, amiben minden forgalom engedélyezve van, amíg konkrétan nem blokkolják. Az ilyen beállítások kéretlen hálózati kapcsolatokat és rendszer veszélyeket okoznak. A szabálymegszegéseket leszámítva, egy tűzfal funkciója nem abból áll, hogy veszélyeket felismerjen és akadályozzon. Főleg abból áll, hogy a meghatározott kommunikációs kapcsolatokat engedélyezze, a forrás- vagy célcímek és a használt szolgáltatások alapján. A támadások felkutatásáért a behatolás-felismerő rendszerek a felelősek, amelyet akár a tűzfalra is lehet telepíteni, de ezek nem tartoznak a tűzfalhoz.

5.2. Tűzfalak csoportosítása

- *Külső tűzfal*: a teljes helyi hálózatot részben elválasztja az internettől
- *Belső tűzfal*: a helyi hálózatnak egy különösen védendő részét zárja el annak többi részétől (így az internettől is)
- *Személyes tűzfal*: adott számítógépen elhelyezett tűzfalszolgáltatás

5.3. Tűzfalak feladatai, típusai funkciók szerint

5.3.1. Csomagszűrés *(packet-filter firewall)*

Az adatcsomagok egyszerű szűrése a forrás- és cél-port, forrás- és célcím, hálózati interface, protokoll alapján. Ez egy, a tűzfal-adminisztrátor által definiált szabályrendszer alapján történik. Ez minden hálózati-tűzfal alapfunkciója. A vizsgálat eredményeképp a csomagokat megsemmisíti vagy továbbítja. Ez egy gyors és univerzális megoldás, viszont jelentős háttérismeretet, a hálózati és alkalmazási protokollok ismeretét igényli. Ez a tűzfalak leggyakrabban használt fajtája, ezekkel az alapvető szűrésekkel rendelkezik manapság a legtöbb router és switch.

Néhány általános gyakorlati szabály:

- internetről a DMZ-ben levő levelező-szerverhez engedélyezett a mail-szolgáltatás (SMTP, POP3 és IMAP)
- a levelező-szervernek szabad a DMZ-ből az internetre az SMTP által mail-t küldeni és DNS-lekérdezéseket végrehajtani
- a lokális hálózatból a levelező-szerverhez engedélyezett az adminisztrációs szolgáltatások végrehajtása (SSH, Remote Desktop, Backup)
- minden más csomag a demilitarizált zónába vagy abból kifelé egy logfile-ba íródik, és utána eldobásra kerül

5.3.2. Állapot szerinti szűrés

A csomagszűrő tűzfalakhoz hasonlóan az állapottartók is szabályláncokkal dolgoznak, de jelentős különbségként, már nem csak az adott csomagból nyert információkat használják fel döntéseik meghozatalához, hanem a csomagok közti kapcsolatokat is figyelembe veszik. Ez a plusz kiegészítés alapvetően a kapcsolatorientált protokollok esetén nyújt plusz lehetőségeket, nagyobb biztonságot. Ilyen protokollra tipikus és a leggyakoribb példa a TCP. A TCP protokoll tűzfalazása esetén az állapottartó tűzfalak képesek a TCP kapcsolatok állapotának követésére, azaz képesek megkülönböztetni a kapcsolat kiépülését végző csomagokat (3-way handshake), a kapcsolat kiépülése után adatot közvetítő csomagokat, valamint a kapcsolat megszakítását, lezárását végző csomagokat. A csomagok megkülönböztetése mellett a tűzfal figyelembe veszi, hogy adott csomag csak adott helyen jelenhet meg a kommunikációban. Például adatot tartalmazó csomag nem előzheti meg a kapcsolat kiépülését, és nem érkezhet a kapcsolat lezárását követően sem. A több csatornás, vagy több porton kommunikáló kapcsolatok, protokollok átviteléhez az állapottartó csomagszűrők ugyanazt az elvet alkalmazzák, amit a csomagszűrő tűzfalak is és adatértelmező képességükre is ugyanazon limitációk vonatkoznak. Az állapottartó-csomagszűrők segítségével könnyebbé válik az átmenő forgalom nyomon követése és szűrése. Már nem csak atomi szinten kerülnek ellenőrzésre a csomagok, hanem a kapcsolatok egésze, az azok közti összefüggések alapján van lehetősége a tűzfalnak az ellenőrzésre. Például TCP válaszcsomagok esetén nem csak az ACK flag megléte, vagy hiánya szolgáltat alapot a döntéshez, hanem a teljes TCP kapcsolat nyomon követése (seq-num, ack-num, window size,

stb...) adja meg a segítséget a tűzfalnak. Ugyanígy az új kapcsolat kezdeményezése sem csak a SYN flag meglétén múlik.

Az állapottartó szűrés tehát a csomagszűrés egy kibővített formája, ami a 7. OSI-rétegen egy rövid vizsgálatot hajt végre, hogy minden hálózati csomagról egyfajta állapottáblát hozzon létre. Ezáltal felismeri ez a tűzfal a csomagok közti összefüggéseket és az aktív kapcsolathoz tartozó munkafolyamatokat leállíthatja. Így sikerül ennek felismerni egy kapcsolat felépítése után, hogy a belső kliens a külső célrendszerrel mikor kommunikál, és csak akkor engedélyezi a válaszadást. Amikor a célrendszer olyan adatokat küld, melyeket a belső kliens nem kért, akkor a tűzfal már önmaga blokkolja az átvitelt a kliens és a célrendszer között fennálló kapcsolatnál. Ez különbözteti meg ezt a tűzfalat egy szokásos csomagszűréstől. Egy proxy-val ellentétben a kapcsolat itt önmagában nem befolyásolt.

5.3.3. Alkalmazás szintű tűzfal

Egy alkalmazás-szintű tűzfal a tisztán csak a forgalomhoz tartozó, mint a forrás, cél és szolgáltatás adatokon kívül a hálózati csomagok tartalmát is figyeli. Ez lehetővé teszi a dedikált proxy-k alkalmazását is, amik egy specializált tartalomszűrést vagy egy Malware-szkennelést is lehetővé tesznek. Egy alkalmazás szintű tűzfal alapszintű feladata nem abból áll, hogy meghatározott alkalmazások (programok) hálózathoz való hozzáférését engedélyezze vagy megtiltsa. Egyébként egy áramkör szintű proxy-t lehet egy ilyen tűzfalra létesíteni, ami egy protokollfüggetlen port- és címszűrés mellett egy lehetséges hitelesítés a kapcsolat felépítésének támogatásához. E nélkül egy alkalmazás számára nem lehetséges egy külső hálózattal (internettel) történő kommunikálás.

5.3.4. Proxy / Anonymous proxy

A proxy tűzfalak működési elve nagyon egyszerű. A kliensek és a kiszolgálók között nem épül fel közvetlen kapcsolat, hanem mindketten a tűzfalon futó proxy alkalmazással kommunikálnak. A proxy egyik hálózati csatolójával az ismeretlen hálózat kiszolgálóihoz kapcsolódik, a másikkal pedig a belső hálózatban található kliensekhez. A kapcsolat kettősségéből kifolyólag a proxy tűzfalak minden különösebb beállítás nélkül képesek kivédeni a csomagszintű támadásokat.

A számítógép hálózatokban proxy szerverek a kliensek kéréseit köztes elemként más szerverekhez továbbítják. A kliens csatlakozik a proxyhoz, valamilyen szolgáltatást (fájlt, csatlakozást, weboldalt vagy más erőforrást) igényel, amely egy másik szerveren található. A proxy szerver a kliens nevében eljárva csatlakozik a megadott szerverhez, és igényli az erőforrást a számára. A proxy esetlegesen megváltoztathatja a kliens kérését vagy a szerver válaszát, és alkalomadtán kiszolgálhatja a kérést a szerverhez való csatlakozás nélkül is. Az olyan proxy szervernek, ami változtatás nélkül továbbítja a kérelmeket és a válaszokat külön neve is van: ez a gateway, vagy néha tunneling proxy.

Az alkalmazás-szintű tűzfal integrált proxyt használ, ami a munkamenetének helytállósága alapján építi fel a kliensekkel és a célrendszerekkel a kapcsolatot. A szervernek csak a proxy IP-címe lesz látható, mint feladó, nem pedig a kliensé. Így a helyi hálózat struktúrája nem lesz felismerhető az Internet felől. Tehát megakadályozza a közvetlen kommunikációt a külső és a védett hálózat között. Közvetítő szerepet játszik a kettő között: a belülről érkező kéréseket feldolgozza, majd azokkal azonos értelmű kérést küld a külső szerver felé, az azokra érkező válaszokat pedig ugyanilyen módon továbbítja a belső hálózat felé. Elég biztonságosnak mondható és általában egyszerűen konfigurálható. Hátránya viszont, hogy kizárólag olyan kommunikációra használható, melynek értelmezésére képes. Magukba foglalhatnak tartalmi gyorsítótárat, így néhány esetben jelentős mértékben csökkenthetik a kifelé irányuló forgalmat. Minden magasabb kommunikációs protokollnak (HTTP, FTP, DNS, SMTP, POP3, MS-RPC stb.) van egy saját, dedikált proxy-ja. Egyetlen alkalmazás-szintű tűzfalon több dedikált proxy is futhat egyszerre.

Anonim proxy: Az eredeti webező identitásának elrejtésére, a webszerver és a böngésző közti kommunikációba harmadik félként beépül olyan módon, hogy valójában ő tölti le a kiszolgálóról a kliens által kért weblapokat. Ezeket továbbítja, így a tényleges kliens identitása (IP címe) a szerver elől rejtve marad.

5.3.5. Transzparens Proxy tűzfal

A proxy tűzfalak konfigurációs igényeiből fakadó kényelmetlenségnél nagyobb problémát jelent az ennek következtében jelentkező emberi hibaforrás hangsúlyosabb jelenléte. (A túl sok és gyorsan változó konfiguráció, mely beállítása a felhasználók alkalmazásait is érinti vagy követhetetlenné válik sok felhasználós hálózatokban, vagy egyre

lazább, minden körülmények között működő, de túlságosan "nyitott" szabályrendszereket eredményez.) A változásokat tovább sürgette, hogy egyre több olyan protokoll jelent meg, melyek nem voltak felkészítve a proxy-s működésre. A cél tehát a transzparens működés megvalósítása, és a proxy funkciókat nem támogató protokollok kezelése lett.

A megoldás adott volt. A tűzfalon - elhelyezkedéséből adódóan - minden forgalom áthalad (a tűzfal, mint a szervezet hálózatának internetes, alapértelmezett átjárójaként szerepel). Ebből adódóan lehetőség van a tűzfalon a csomagszűrőkhöz hasonló funkciókat ellátni. A klienseken semmilyen proxy beállítást nem kell megtenni, azaz a kliensek közvetlenül próbálnak kapcsolódni a szerverhez. A csomagszűrökkel szemben azonban a csomagok nem jutnak át a tűzfalon, hanem a tűzfal a csomagokat, kapcsolatokat – beépített csomagszűrőjének segítségével - "elkapja", és magára irányítja. Az átirányított kapcsolatokat pedig a proxy program fogadja, a nem-transzaparens proxyk működéshez hasonlóan. Természetesen a transzparens proxy használatával is lehetőség van nem tarnszparens működésre.

A tarnszparens proxyk tehát transzparenciájuk megvalósítása céljából kiemelten támaszkodnak az alacsonyabb szintű csomagszűrőre. A csomagszűrő és a proxyk megfelelő együttműködése eredményezi a sima proxyknál kényelmesebb használati módot. A transzparens proxyk használatával a felhasználók számára a hálózati erőforrások tűzfalon keresztüli elérése kényelmesebbé válik, adminisztrációja áttekinthetőbb.

A proxy tűzfalak megoldást nyújtanak a kliensek kényelmes és biztonságos kiszolgálására, látszólag megoldva a tűzfalak által keltett problémákat. Azonban az újabb protokollok fejlődésének, összetettebbé válásának eredményeként napjainkban egy újabb problémával kellett a tűzfalaknak, és az őket felhasználó szakembereknek szembenézniük. Ez a merőben új probléma, az összetett protokollok kezelésének kérdése.

Napról napra több alkalmazás támaszkodik összetett protokollokra, amelyek megfelelő, biztonságos kezelésére, a már meglévő technológiák nem nyújtanak kielégítő megoldást. Ilyen összetett protokollhasználatra jó példa a napjainkban az e-business keretében elengedhetetlen HTTPS protokoll, ami egy SSL (titkosítást és authentikációt megvalósító) protokoll és egy HTTP protokoll kombinációjából született meg. A megoldás egy új tűzfaltechnológia megjelenését eredményezte.

5.3.6. Moduláris proxy tűzfalak

A moduláris proxy tűzfalak megszületése, más tűzfalaktól eltérően, kizárólag a nagyobb biztonság elérése céljából történt meg. A tűzfalak ezen ága nem kényelmi okokból fejlődött ki, a tervezési szempont a megfelelően biztonságos működés elérése volt. A moduláris proxy tűzfalak rendelkeznek a transzparens tűzfalak minden jó tulajdonságával, azaz képesek az átmenő adatfolyam alkalmazásszintű szűrésére, csomagszűrő kiegészítőt tartalmaznak, valamint transzparensek a kliens számára. Az alapvető különbség a hagyományos transzparens tűzfalak és a moduláris tűzfalak között, hogy míg a transzparens tűzfalak minden protokoll értelmezésére, elemzésére különálló tűzfal komponenssel rendelkeznek, - amelyek nem képesek együttműködésre, valamint sok esetben bizonyos funkciókat mindegyik komponens megvalósít (kapcsolat fogadása, kapcsolódás a szerverhez, stb.) -, addig a moduláris proxy részei, moduljai képesek együttműködni, valamint a különböző feladatok ellátását más-más modul végzi, csökkentve ezzel a felesleges redundanciát.

A modularitás több szinten valósul meg egy moduláris tűzfalon. A leggyakoribb és általában a legtöbb helyen szereplő példa az egyes proxy modulok együttműködésén alapszik. Napjainkban egyre több az összetett alkalmazásszintű protokollt használó program, erre a legjobb példa a HTTPS. A HTTPS különlegessége, hogy nem csak összetett protokoll, hanem az egyik része titkosított, úgynevezett kripto protokoll, amelynek a szűrése, visszafejtése még bonyolultabb. A moduláris proxy esetében egy SSL proxy és egy HTTP proxy kerül kombinálásra. Az SSL proxy kapja meg az átmenő forgalmat, azt dekódolja - azaz a kliens felé, mint a szerver jelenik meg, végrehajtva ezzel egy Man-In-The-Middle támadást -, majd a dekódolt forgalmat átadja a HTTP proxynak. A HTTP proxy már a sima HTTP kérést kapja meg, mintha azt egy sima klienstől kapná. A kérés feldolgozása után a HTTP proxy a kérést továbbküldi, mintha a szervernek küldené, de a kérés az SSL proxyhoz kerül, amely a kérést újra titkosítja és elküldi a szervernek. Ezzel a módszerrel lehetőség nyílik a HTTPS forgalom transzparens http szintű szűrésére, amely fontos részét képezheti a trójai programok, valamint más nem kívánt programok elleni védekezésnek.

A moduláris tűzfal bizonyos proxy moduljai fel vannak készítve arra, hogy a rajtuk átmenő forgalom egy részét képesek legyenek egy másik proxynak további elemzésre átadni, azaz más elemző proxy modult a bevonni, beágyazni a forgalom elemzésébe. Természetesen,

ha egy proxy modul esetén lehetőség van beágyazásra, akkor az bármelyik másik proxy modult képes beágyazni. (Persze elképzelhetőek olyan kombinációk, amelyek esetében nincs értelme a beágyazásnak.) Lehetőség van továbbá több szintű beágyazásra is, például SSL proxyba ágyazott HTTP proxyban tartalomszűrésre, ahol a tartalomszűrést egy tartalomszurok modul valósítja meg, vagy SSL proxyba ágyazott POP3 proxyban a letöltendő levelekben víruskeresésre. Látható, hogy a beágyazás segítségével az egyre több kombinált protokoll széles palettájának elemzésére, szűrésére van lehetőség, valamint lehetőség van nem protokoll specifikus modul alkalmazására is. A moduláris tűzfalak másik oldala, amely általában kevésbé kerül előtérbe, ám szintén nagyon fontos, teremt lehetőséget a különleges igények kielégítésére, valamint ez veszi át az egyes proxy moduloktól a közös feladatok ellátását. Ilyen modul például a kapcsolatok fogadásáért, vagy kiépítéséért felelős modul. A proxy moduloknak nem feladata többé a kapcsolatok kezelése, csak azok forgalmának elemzése, szűrése, a kapcsolatok kezelésére külön modulok állnak rendelkezésre. Azaz egy speciális modul feladata a kapcsolatok fogadása, majd kapcsolódás esetén, a megfelelő ellenőrzések után, a kiépült kapcsolatot átadja a proxy modulnak, amely csak a forgalommal foglalkozik. Ugyanígy, ha szükség van a kapcsolat szerver oldali részére, az szükséges új kapcsolat kiépítése, akkor azt szintén nem a proxy végzi, hanem az ezért felelős modul. Az architektúra következtében így lehetőség van bizonyos esetekben például az alapértelmezettől eltérő új kapcsolat létrehozását végző modul használatára, amely a szerverhez való sikertelen kapcsolódás esetén megpróbál egy másik szerverhez kapcsolódni.

Egy moduláris tűzfal segítségével lehetőség nyílik az eddigi tűzfal által megoldhatatlan problémák biztonságosabb, flexibilisebb és gyorsabb megoldására, növelve ezzel a védendő rendszer biztonságát. A modularitás révén a tűzfal képessé válik az átmenő forgalom még részletesebb értelmezésére, ezzel segítve az adminisztrátor munkáját az IBSZ hálózati határvédelemre vonatkozó szabályainak még pontosabb betartatását illetően.

5.3.7. Mély-protokollelemzés és content vectoring

A proxy tűzfalak alkalmazásszintű jelenlétükből kifolyólag elvileg képesek lehetnek a teljes átmenő adatforgalom elemzésére és befolyásolására. Ehhez két dolgot kell a proxy-nak teljesítenie: egyrészt ismernie kell a protokoll összes szabványos utasítását és metódusát, másrészt képesnek kell lennie a protokollban átvitt adat elemzésére. Az előbbit hívjuk mély protokollelemzésnek, míg az utóbbi a content vectoring megvalósítását jelenti.

- **Mély-protokollelemzés**

  Sokak számára meglepő lehet, de hiába léteznek protokollok, azok betartását alap esetben egy hálózati eszköz sem ellenőrzi. Ez nagy teret ad a rosszindulatú támadóknak, hiszen sok hálózati eszközben és alkalmazásban vannak olyan biztonsági rések, amiket, a protokollt sértő metódusokkal ki lehet játszani.

  Amennyiben a proxy alkalmazás a teljes szabványt megvalósítja, tehát ismeri az összes utasítást és attribútumot, egyfajta hálózati rendészként minden szabványt sértő kommunikációs próbálkozást megtagadhat. További előnye a mély protokollelemzésnek, hogy segítségével a tűzfal „élesebben lát". A hálózati kommunikációban jóval részletesebben tud eseményeket megkülönböztetni egymástól, aminek következtében a reakciója is kifinomultabb lehet.

- **Content vectoring**

  A tartalomelemzés tipikusan a moduláris tűzfalak sajátja. Önálló modulként épülnek be az architektúrába, így képesek valamennyi proxy-val együttműködni. A megoldandó feladat általában vírusszűrés a tartalomban. Ritkább esetben kulcsszavak alapján történő szűrés is előfordulhat.

5.3.8. Tartalomszűrés

Egy tűzfal a tartalomszűrő használatával egy kapcsolat hasznos adatait kiértékelni, illetve az áthaladó adatokat ellenőrizni tudja.

Jellegzetes példái:

- az URL-szűrés és a vírusfigyelés. Mindkét feladathoz többnyire kiegészítő programokra (URL-szűrőre, víruskeresőre) van szükség, a tűzfalak általában nem tartalmazzák ezeket a lehetőségeket.
- a lekért weboldalakról az ActiveX és/vagy JavaScript kiszűrése
- bizalmas információk kiszűrése
- kulcsszavak alapján nem kívánt weboldalak zárolása
- nem kívánt alkalmazás-protokollok (például: fájlmegosztási) blokkolása

A legtöbb rendszer csak a nagyon egyszerű szabály-definíciókat engedi meg. Az elsődleges probléma nagyon bonyolult és előfordulhat, hogy a koncepció meg sem valósítható technikailag. Mert, ha például tényleg teljességgel ki kell szűrnie az engedélyezett rendszereknek az adatforgalomból a bizalmas információkat, ehhez először meg kellene oldani azt a technikai problémát, hogy bizalmas szteganográfiai (rejtjelezett) vagy kódolt információkat fel lehessen ismerni és ki lehessen szűrni. Az aktuális tűzfalrendszerekben létező egyszerű szabályok ellenére ezek kivitelezése sokrétű lehet. Gyakran külön csomagokat kell összefűzni, amivel a vizsgált adatforgalom egészként felismerhető, átvizsgálható és alkalmanként megváltoztatható. Végül az adatforgalmat ismét különálló csomagokra kell bontani és továbbküldeni.

5.3.9. Behatolás felismerő és behatolás megelőző rendszerek

A „behatolás felismerő rendszer”-t (IDS) és „behatolás megelőző rendszer”-t (IPS) manapság már egyre gyakrabban integrálják a tűzfalakban. Mindkettő felismer egy behatolási próbát a kommunikációs minták alapján. A különbség az, hogy egy IDS a támadást csak felismeri, míg az IPS megpróbálja blokkolni. Az egyes rendszerek ideiglenes tűzfal-szabályt hoznak létre, ami egy támadó IP-cím felől érkező összes további kapcsolódási próbálkozást blokkolja. Ha viszont a támadó hamis küldő-címmel ellátott csomagokat küld a rendszernek, akkor ezzel el tudja érni, hogy ne legyen hozzáférés a hamis című klienshez. Ezzel egymás után le tudja választani az összes címet a rendszerről, amelyekre épp szükség lenne a működéshez (DNS-szerver stb.).

5.3.10. Hálózati címfordítás

A hálózati címfordítás (*Network Address Translation*, NAT) lehetővé teszi belső hálózatra kötött saját nyilvános IP cím nélküli gépek közvetlen kommunikációját tetszőleges protokollokon keresztül külső gépekkel. Vagyis, hogy több számítógépet egy routeren keresztül kössünk az internetre. Az elsődleges cél ez esetben az, hogy egy nyilvános IP-címen keresztül több privát IP-című (privát címtartomány: RFC 1918) számítógép csatlakozhasson az internethez. A belső gépekről érkező csomagok feladójaként saját magát tünteti fel a tűzfal (így elrejthető a védett host igazi címe), a válaszcsomagok is hozzá kerülnek továbbításra, amelyeket – a célállomás címének módosítása után – a belső hálózaton elhelyezkedő eredeti

feladó részére továbbít. Egy proxy-val ellentétben itt a csomagokat csak tovább küldik, és nem analizálják a tartalmukat.

5.3.11. Port szűrés, port kezelés

A tűzfalnak figyelnie kell az egyes portokon folyó forgalomra. Érzékelnie kell, ha valaki végigpásztázza a nyitott portokat (port scanning), képesnek kell lennie az egyes portok lezárására, valamint fel kell tudni figyelnie az egyes portokon jelentkező „gyanús" forgalomra is.

5.4 Tűzfal architektúrák

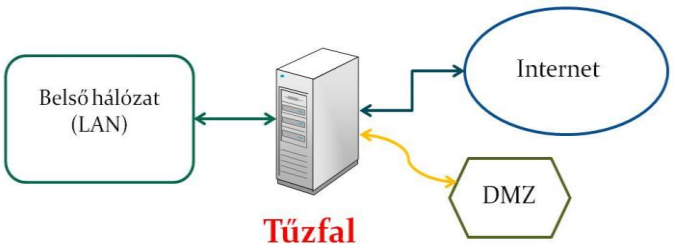
5.4.1. Egyszerű tűzfal-architektúra

Minden be- és kiáramló forgalom áthalad a tűzfalon, amely a védendő belső hálózat és az Internet kapcsolatot biztosító router (útválasztó) között helyezkedik el. Már maga a router is mintegy megerősítő előszűrőként jelent védelmi vonalat. Beállított ACL (Access Control List, hozzáférési lista) segítségével szűri a csomagokat. Ebben az architektúrában a nyilvános szolgáltatások (mail, web, dns stb.) elhelyezése kétféle lehet: vagy magán a tűzfal szerveren, vagy pedig a belső hálózatban. Egyik sem igazán jó megoldás. A tűzfal szerveren elhelyezett szolgáltatások terhelik a tűzfalat, míg a belső hálózatban elhelyezve növelik a belső hálózat gépeinek támadásokkal szembeni kiszolgáltatottságát.

5.4.2. Egytűzfalas DMZ architektúra

A DMZ (demilitarizált zóna) személyes vagy vállalati hálózatok megbízhatatlan külső, és a megbízható belső része között elhelyezkedő terület. A benne elhelyezkedő hálózati eszközökhöz és erőforrásokhoz mind a megbízható belső, mind a megbízhatatlan külső területről engedélyezi a hozzáférést, de megakadályozza, hogy a külső területről bármilyen kérés vagy hozzáférési kísérlet eljusson a belső hálózatra. A DMZ célja, hogy egy plusz biztonsági réteget biztosítson a szervezet helyi hálózatának. Így egy külső támadónak csak a DMZ-ben található berendezésekhez lehet hozzáférése, mintsem az egész hálózathoz. Egy hálózatban azok a hostok a legsebezhetőbbek, amelyek a LAN-on kívüli felhasználóknak nyújtanak szolgáltatásokat, mint a mail, web és DNS szerverek. Az ezen hostok megnövekedett fenyegetettsége miatt ezeket egy saját alhálózatba helyezik. Ezzel védik a

hálózat többi részét abban az esetben, ha valakinek sikerülne behatolnia. A DMZ-ben lévő hostoknak nem szabad közvetlen kapcsolatban lenniük a belső hálózatba tartozó egyik hosttal sem. A kommunikáció más hostokkal a DMZ-n belül és a külső hálózatba engedélyezett. Ez teszi lehetővé, hogy a DMZ-n belüli hostok szolgáltatást nyújthassanak mind a belső, mind a külső hálózatba. A forgalmat pedig egy közbenső tűzfal irányítja a DMZ szerverei és a belső hálózat kliensei közt.



5.4.3. Kéttűzfalas DMZ architektúra

Ebben a topológiában mind a DMZ-t, mind pedig a belső hálózatot külön tűzfal védi. Ez a megoldás jobb terhelésmegosztást tesz lehetővé, különösen nagy mennyiségű adatforgalom esetén hasznos. Lehetővé válik egy hatékonyabban működő heterogén tűzfalas megoldás alkalmazása: egy nagy forgalmat bonyolító külső csomagszűrő tűzfal és egy alkalmazásátjáróként (proxy) használt belső tűzfal. Ezáltal a DMZ-ben elhelyezett nyilvános szolgáltatásokat nyújtó szerverek optimálisabban érhetők el kívülről, a belső részen lévő privát rendszerek pedig hatékonyabban szigetelhetők el.

5.5. Ismertebb tűzfal megoldások

5.5.1. Szoftveres megoldások

- Internet Connection Firewall (ICF)

  A Windows XP, Vista és a Windows Server 2003 beépített tűzfalprogramja, amely az Internetre kapcsolódó számítógépeket védi a külső támadások ellen.

- Microsoft ISA szerver

  Windows 2003 vagy Windows 2007 operációs rendszerre telepíthető tűzfal szerver.

- Netfilter

  Linuxos rendszerek beépített csomagszűrő tűzfalrendszere, alkalmazása proxy szerverrel (squid) együtt használva igazán megbízható és hatékony megoldást nyújt.

Csomagszűrő, kapcsolat nyomkövető, hálózati címfordító stb. alrendszerei jól konfigurálható és menedzselhető kezelést tesznek lehetővé, hatását szabály-láncokon keresztül fejti ki.

- Border Manager

  A Novell Netware hálózati operációs rendszeren futó kényelmesen konfigurálható tűzfal-rendszer. A Novell BorderManager az iparág legelső, valódi címtárra épülő hálózati szolgáltatáscsomagja, amelynek tagjai felügyelik, biztosítják és felgyorsítják a felhasználók hozzáférését az információhoz minden hálózati határon – azokon a pontokon, ahol bármely két hálózat találkozik. A termékcsalád része a továbbfejlesztett proxy-gyorsítótárkezelést, a védőgátak szűrőfunkcióit, a felhasználói szintű hozzáférésvezérlést, hálózatcím-fordítást, útirányítást és távoli hozzáférési szolgáltatásokat tartalmazó BorderManager, a BorderManager FastCache és a BorderManager Authentication Service. A biztonsági rendszerét integrálták az NDS-vel (Netware Directory Service, hálózati címtárszolgáltatás) és IP-címek helyett felhasználókat kezel.

- A csomópontokra (munkaállomásokra, szervergépekre, routerekre) külön telepíthető tűzfal szerverprogramok.

5.5.2. Hardveres megoldások

- Csomagszűrő routerek

  Mivel a routerek a hálózatok bejáratánál állnak, így ezen az egy ponton a teljes átmenő forgalom ellenőrizhető. A routerek - amelyek addig csak a csomagok célba juttatásáért voltak felelősek - új feladatot kaptak: adott szabályok alapján egyes csomagokat engedjenek át, míg másokat dobjanak el. A döntéseket az úgynevezett csomagszűrő routerek - vagy csomagszűrő tűzfalak - a továbbítani kívánt csomag fejlécében található adatok és a beprogramozott szabályok összehasonlítása alapján hozzák meg. A csomagszűrők a fejlécet azonban különböző szinten vizsgálják. Az IP szint minden esetben kiértékelésre kerül, azaz a döntést befolyásolja a csomag forrás és cél címe, esetleges fragmentálási adatai, illetve ritka esetben még az IP opciók értékei is. A legtöbb implementáció esetében kiértékelésre kerül a következő (adatkapcsolati) réteg is (TCP/UDP). Ezek plusz információt jelentenek a döntési szabályok

megfogalmazásakor, lehetőség nyílik a forrás, illetve cél portra való szűrésre, illetve TCP esetén a TCP flagek figyelembe vételére is. A szűrés döntési mechanizmusa szabálylistákon alapul.

- Routerek, switchek, aktív eszközök védelmi megoldásai

  A hálózatok, alhálózatok, virtuális hálózatok összekapcsolását biztosító menedzselhető hálózati aktív eszközök védelmi beállítási lehetőségei között alapvetően megtalálhatók a megfelelő hálózati rétegbeli szűrések: routereknél a Layer3 rétegbeli IP cím szerinti szűrés, az ACL (Access Control List) segítségével szabályrendszer használata, switcheknél a Layer2-beli MAC (hardver cím) szerinti szűrés. Ezen kívül további biztonsági beállítások is elvégezhetők.

- Alkalmazás szűrő védelmi eszközök

  Ezek a hardver eszközök többnyire integrált levelezés-, web-, spam- és spyware szűrést végeznek, kombinálják a megelőzést, az elhárítást, és a proaktív funkciókat. Általában az alábbi védelmi szolgáltatásokat nyújtják:

  - Felhasználó azonosítást követően korlátozza az egyes tartalmakhoz való hozzáférést
  - Blokkolja a pornót, drogot, erőszakot tartalmazó oldalakat
  - Korlátozza a magánjellegű web hozzáféréseket munkaidőn túlra
  - Blokkolja egyes fájltípusok letöltését
  - Blokkolja egyes alkalmazások internet hozzáférését (Skype, Messenger, Zene letöltés stb.)
  - Leállítja a spyware letöltéseket (drive-by downloads)
  - Leállítja a vírus letöltéseket
  - Blokkolja a spyware oldalakhoz történő hozzáférést
  - Detektálja a spyware hozzáférést az Internethez
  - Megkönnyíti a spyware eltávolítást
    Blokkolja az offenzív oldalakat

6. Komplex, széleskörű integrált védelem és biztonság

6.1. Biztonsági elvárások és megoldások

Az informatikai rendszerek működésétől alapvetően elvárt követelmény a bennük tárolt, feldolgozott és továbbított adatok biztonságos rendelkezésre állása. Kulcsfontosságú kérdés tehát a rendszerek biztonságos, hatékony, gazdaságos és professzionális működtetése, üzemeltetése, beleértve a gyors hibaelhárítást, valamint a felhasználói támogatást. Ezeknek a követelményeknek és elvárásoknak egy jól szervezett, megfelelően adminisztrált és működtetett széleskörű, integrált védelmi és biztonsági rendszer képes megfelelni.

A széleskörű védelem magába foglalja a hálózat egészének, az aktív és passzív eszközök, a felhasználói csomópontok, és egyéb informatikai berendezések fizikai, logikai és teljes adminisztratív védelmi eszközrendszerét.

Az integrált védelem egyesíti a biztonsági és védelmi eljárások teljes skáláját, a hardver- és szoftvermegoldások együttesét, az informatikai biztonsági szabályzatnak megfelelő működést, a felhasználói támogatást, segítségnyújtást, valamint az erre irányuló kutatási-fejlesztési tevékenységet. Átfogja az informatikai rendszer különböző szintjeit, egymást kiegészítő-összehangoló feladatokat lát el az egyes rétegekben.

6.2 Átfogó védelmi rendszer előkészítése

A hatékony és átfogó védelmi rendszer kialakítása komoly előkészületeket, helyzetfelmérést, kockázatelemzést és tervezést igénylő feladat, melynek során figyelembe kell vennünk, hogy

- melyek a védendő területek: hálózati eszközök, csomópontok, a rajtuk tárolt adatok, futtatott alkalmazások
- melyek a veszélyforrások, ki és milyen mértékben jelent veszélyt: vírusok, spywarek, betörések, adatlopás, felhasználók tájékozatlansága stb.
- milyen mértékű védelemre van szükségünk: indokolt, szükséges és optimális védelem
- alkalmazható, illetve szóba jöhető védelmi technológiák vizsgálata, elemzése
- jövőbeni tendenciák

6.3 Integrált védelmi rendszer elemei, kialakítása

Az előbbiekben vázolt elemzéseknek megfelelően lehet kialakítani a védelmi stratégiát, megválasztani az alkalmazásra kerülő technológiákat. A fentiekben tárgyalt védelmi technológiák bár külön-külön is lehetnek hatékonyak, de önmagukban nem sokat érnek. Például a tűzfal egy olyan szoftveres vagy hardveres eszköz, amely felügyeli a számítógépek egymás és az internet közötti adatcsere folyamatait, képes a bejövő és kimenő adatok ellenőrzésére, szűrésére. Azonban a tűzfalak általában nem nyújtanak kellő védelmet a kémprogramoktól, trójai falovaktól, különböző reklámoktól, levélszemetektől. Egy olyan rendszeren pedig, amely tele van biztonsági résekkel, teljesen felesleges tűzfalat használni. Célszerű naponta ellenőrizni az aktuális frissítéseket az operációs rendszerhez, böngészőkhöz, és minden olyan alkalmazáshoz, amelynek szüksége van internetes kapcsolatra. A védelmi rendszer további elemei sem elegendőek magukban, hiszen például a vírusvédelem még nem jelent biztonságot a betörések, támadások, vagy éppen a kémprogramok (spyware) vagy kéretlen levelek (SPAM) ellen. A számítógépes csomópontok biztonságos üzemeltetéséhez legalább az alábbi eszközök együttes használata szükséges:

- antivírus
- antitrójai
- reklám- és kémprogram tisztító
- személyes adatok védelmére szolgáló eszköz
- tűzfal
- programfrissítés

E megoldások akár külön is vagy integráltan is megtalálhatóak, és telepíthetők a számítógépre. A megfelelő hatékonyság feltétele még a helyes konfigurációjuk és használatuk, rendszeres ellenőrzésük.

Másfelől nem elegendő a csomópontok közvetlen védelméről gondoskodni, védeni kell magát a helyi hálózatot és virtuális hálózatokat. E védelem is csak akkor lehet igazán hatékony, ha teljes körű, kiterjed a biztonsági területek egészére és mindenirányú, vagyis védelmet jelent mind a bejövő, mind a kimenő forgalom és adatok számára. Nem elegendő csupán a szűrés, a hibákat fel kell deríteni, meg kell találni, el kell hárítani, a kiváltó okokat pedig meg kell szüntetni.

6.4. Az Informatikai Biztonsági Szabályzat

A hálózatra kötött számítógépek számos visszaélésre adnak lehetőséget, a számítógépeken futó alkalmazásokból, az alkalmazások által használt protokollokból és az emberi gondatlanságból kifolyólag. A kockázat mértéke pedig az ugyanazon hálózatra csatlakozó felhasználók számával arányosan nő. Ennek csökkentésére tett kísérleteket (ideértve a szoftveres és adminisztratív megoldásokat is) többnyire az aktuális problémák elhárítása motiválja, amiből két dolog következik:

- IT biztonságtechnikában a kockázatot csak minimalizálni lehet, teljesen kiküszöbölni általában nem.
- az IT security folyamatosan fejlődik, a biztonság nem állapot, hanem folyamat, fenntartásához állandó felügyeletre, rendszerfrissítésekre van szükség.

Az adminisztratív intézkedések sorába tartozik a kötelességek és jogok szabályozása, azok ismertté tétele; a szervezeti felépítés kialakítása, amelyben az adatvédelemért felelős szakemberek megfelelő helyet kapnak; valamint a hálózatban dolgozók egyértelmű munkaköri leírásának elkészítése.

Ha a fenti intézkedések megtörténtek, akkor lehet felmérni, hogy hol és milyen technikai eljárásokat célszerű alkalmazni. A leggyakrabban használható alaptechnikák a jelszó, valamint a fizikai hozzáférés és a rejtjelezés különböző fajtái.

Az informatikai biztonsághoz meg kell teremteni a teljes szabályozási rendszert. Az informatikai rendszer biztonsági, védelmi szabályzata az Informatikai Biztonsági Szabályzat (IBSZ), amely kiterjed az informatikai eszközök fizikai, logikai és adminisztratív védelmére. Az IBSZ külön fejezetei szabályozzák a különböző alkalmazott védelmi technológiákat, a határpontokon keresztül elérhető távoli erőforrások használatát, az ehhez kapcsolódó felhasználói jogosultságok rendszerét, az egyes hálózati protokollokra vonatkozó előírásokat valamint bizonyos logikai szabályokat, felelősségi és intézkedési jogköröket, rendelkezéseket. Az informatikai rendszer megfelelő működtetése és védelmének biztosítása nagymértékben függ az IBSZ-ben előírt, jól megtervezett és alkalmazott védelmi stratégia maradéktalan és következetes végrehajtásától.

7. Összegzés

A fenti áttekintés célja az alapvető védelmi technológiák ismertetése volt. A gyakorlatban azonban a technológiák sokszor összemosódnak és a helyes döntés meghozását még tovább nehezíti, hogy külsőre gyakran nehéz megítélni az egyes termékek technológiai hovatartozását és színvonalát. A technológiák kiválasztásakor az adott informatikai rendszer biztonsági igényeink alapos felmérésén keresztül vezet az út. Mérlegelni kell azt is, hogy az IT biztonságtechnikában a vírusvédelem, a behatolás védelem és a hálózati határvédelem külön fogalmakat jelentenek és általában külön eszközök segítségével valósítják meg feladataikat.

Bármely informatikai rendszer biztonságát illetően rendkívül fontos a termék támogatása, supportja vagy távmenedzsmentje. A biztonság nem állapot, hanem folyamat. A legjobb megoldás sem képes hozzáértő kezelés nélkül tényleges biztonságot nyújtani. Lehetetlen felhívni minden buktatóra a figyelmet, hiszen nap-mint-nap találkozhatunk új fogalmakkal, kihívásokkal, amelyekre figyelni és számítani kell. Tisztában kell lenni a védelmi rendszer tulajdonságaival, képességeivel és működésével, biztosítva annak folyamatos felügyeletét.

A hardveres és szoftveres megoldások választásakor világosan kell látni, hogy a hardver alapú megoldások sem működnek szoftver nélkül. Az, hogy egy megoldás hardveres, azt jelenti, hogy valamilyen speciális konfigurációra optimalizálnak egy megoldást, ezzel növelve a teljesítményt és rendelkezésre állást. Ugyanakkor sok esetben éppen a kötött hardver kiépítés flexibilitási problémákhoz vezethet. Bármilyen változtatás, mely a fizikai felépítését érinti ezeknek az eszközöknek, többnyire költséges és erősen behatárolt.

A szoftveres védelmi rendszerek mindig valamilyen operációs rendszeren futnak. Fontos, hogy ne csak a védelmet ellátó szoftver maga, de az azt kiszolgáló operációs rendszer és minden egyéb kiszolgáló alkalmazás napra készen frissítve legyen. A védelem fokozása érdekében célszerű az alap operációs rendszert csak a lehető legcsekélyebb és a védelmi szoftver futtatásához feltétlenül szükséges szolgáltatásaival konfigurálni, működtetni, minimálisra csökkentve ezzel is a támadási felületet.

A rendszer teljesítményét összességében a hardver és a rajta futó szoftverek teljesítménye határozza meg. A megfelelő teljesítményű védelmi rendszer tervezésekor, kialakításakor

természetesen messzemenően figyelembe kell venni a várható igénybevételt, terhelést. Egy informatikai biztonságot növelő intézkedés, beruházás csak akkor lehet hatékony, ha nem okoz aránytalanul nagyobb többletmunkát, költséget vagy kényelmetlenséget, és ha azt a felhasználók nem csak megértik, de maradéktalanul be is tartják. A biztonság és a használhatóság, az követelmények és elvárások egyensúlyának megtalálása mindig az adott helyzettől függ.

Mellékletek

1. Használt fogalmak definíciója

- **ARP**: Address Resolution Protocol. Feladata egy adott címhez tartozó hardver cím lekérdezése: a kérdező küld egy broadcast üzenetet, amelyben elküldi azt a címet, amelyhez tartozó hardver címet tudni akarja. (A broadcast üzenet lényege, hogy minden interface elfogadja a helyi hálózaton, de csak az fog válaszolni, akinek a címét az üzenet tartalmazza.)
- **ARP cache**: az ARP által lekérdezett hadver címek egy előre definiált ideig itt tárolódnak, ezért nem kell minden alkalommal újra lekérdezni őket.
- **certificate**: a hálózaton erre felhatalmazott szerverek által kiadott tanúsítvány. Ha egy kliens kapcsolatot kezdeményez, a szerver elküldi a certificate-jét, amit a kliens ellenőriz. Ha rendben találja, megkezdődhet az információ átvitele.
- **chroot**: használatával egy program bezárható egy adott könyvtárba. Ha jól állítjuk be, nincs lehetősége, hogy a megadott könyvtáron kívül mást is lásson.
- **exploit**: biztonsági hiba kihasználása, általában a kihasználására írt programot illetik e névvel.
- **hardver cím**: a hálózati interface fizikai címe, MAC address. Ethernet hálózatok esetén egy 32 bites hexadecimális szám, amely alapesetben minden ethernet interface-nél különböző.
- **ICMP**: Internet Control Message Protocol. A hálózatba kapcsolt intelligens eszközök - a hálózattal kapcsolatos - üzenetcseréjére találták ki.
- **ident:** a kapcsolatot kezdeményező felhasználó nevének azonosítására szolgál.
- **noexec**: a mount által használt opció. Az így felmountolt fájlrendszeren lévő programok nem futtathatók.
- **nosuid**: a mount által használt opció. Az így felmountolt fájlrendszerben nincs hatása a setuid/setgid bitnek.
- **PAM**: Pluggable Authentication Module. Egy egységes authentikációs rendszer, melynek feladata a felhasználók bejelentkezésével, memória és fájlrendszer használatával, valamint az authentikációs szolgáltatásokat nyújtó programokkal kapcsolatos beállítások kezelése.

- **promiscuous mód**: az ethernet interface olyan módja, amelyben nem csak a neki címzett csomagokat fogadja el, hanem mindent.
- **routing tábla**: megadja egy adott célcím esetén az eléréséhez szükséges útvonalat.
- **rsh, rcp, rexec, rlogin**: a kernel RPC (Remote Procedure Call) szolgáltatását használják. Az első egy shell-t hív a távoli gépen, az második file-okat másol a két gép között, a harmadik távoli programfuttatást tesz lehetővé, az utolsó beléptet a távoli gépre. A hálózaton kódolatlanul kommunikálnak, így az átvitt információ lehallgatható.
- **setuid/setgid**: a futtatható binárisokon lévő "s" bit. Ha a file tulajdonosának (setuid), illetve group-jának (setgid) jogai között van, akkor az adott program futtatója a file userének/groupjának jogaival rendelkezik a program futásának ideje alatt. A programok betöltése előtt a linker kitisztítja az environment veszélyes részeit, lefutásuk után pedig az általuk használt memóriaterületeket.
- **SNMP**: Simple Network Management Protocol. Az intelligens hálózati eszközök távoli managementjének megkönnyítésére készült protokoll.
- **SOAP:** Simple Object Access Protocol. Egy üzenetküldésre használt, XML-alapú formátum. Az SOAP formátumot használó üzenetek különböző protokollok segítségével továbbíthatók, de erre a célra általában a HTTP protokollt használják. A SOAP a WSDL és a UDDI mellett a webszolgáltatások harmadik alapvető elemét képezi.
- **spoof(ing)**: általában a hálózati kommunikáció valamely részébe harmadik fél részéről történő beavatkozás.
- **spyware**: olyan programokat nevezzük spyware-nek, ami egy számítógépén futva a számítógép felhasználójáról információt gyűjt és továbbít egy külső félnek. A szabadon letölthető szoftverek között gyakran előfordul az ilyen rejtett funkcióval ellátott program.
- **syslogd**: a rendszer és az egyes programok üzeneteinek kezelésére írt daemon.
- **TCP**: Transmission Control Protocol. A TCP réteg felelős a csomagok nagy részének veszteségmentes átviteléért. Kapcsolat-orientált protokoll, azaz várja az elküldött csomag nyugtázását. Ha ezt nem kapja meg egy meghatározott időintervallumon belül, újraadja a csomagot.

- **trójai (faló):** látszólag hasznos program, amelynek titkos funkciói vannak. Ezek a funkciók általában az adott rendszer védelmi mechanizmusát megkerülve hajtják végre az előírt feladatot, pl. adatokat küldenek a gépről.
- **UDDI:** Universal Description, Discovery, and Integration, azaz univerzális leírás, felfedezés és integrálás rövidítése – egy platformfüggetlen, XML-alapú nyilvántartó rendszer.
- **UDP**: User Datagram Protocol. A feladata hasonló, mint a TCP-nek, de nem kapcsolat-orientált.
- **visszatérési cím**: (return address) a következő végrehajtandó utasítás címe.
- **world-writable**: a "t" (sticky) bit be van állítva a könyvtáron vagy bárki számára írható/olvasható. A "t" bit jelentése: mindenki számára írható/olvasható, de mindenkinek csak a saját tulajdonú file-okhoz vannak jogai.
- **WSDL**: Web Services Description Language. Webszolgáltatás leíró nyelv, amely webszolgáltatások leírására szolgál XML formátumban.
- **XML:** Extensible Markup Language (Kiterjeszthető Leíró Nyelv), általános célú leíró nyelv, speciális célú leíró nyelvek létrehozására.

2. Néhány fontosabb protokoll

Az Internet kommunikációs protokollja a **TCP/IP** (Transmission Control Protocol/Internet Protocol). Nemcsak a nagyterületű hálózatok használják, hanem a legtöbb lokális hálózat is ezzel kommunikál. Kétszintű protokoll, a hálózati rétegben (Layer3, L3) működő IP-ből és a transzport rétegben (Layer4, L4) működő TCP-ből áll.

A **TCP** a magasabb szint, feladata az Interneten továbbítandó üzenetek, adatállományok kisebb csomagokká (packet) való szétdarabolása, és a csomagokhoz információ hozzáfűzése (TCP header - például tartalmazza a csomagok sorszámát.). Szintén a TCP szint feladata a célgépen a csomagok összefűzése. A TCP kétirányú protokoll, ami azt jelenti, hogy a feladó a csomagok megérkezéséről nyugtát vár. Ilyen protokollon alapszik pl. az FTP, HTTP, SMTP, POP3.

Az alsóbb szintű protokoll, az **IP** a csomagok címzési adatokkal való ellátását végzi (mintha egy borítékot készítene, ráírva a címzett és a feladó adatait). Ez a protokoll használja a jólismert IP címeket.

További ismertebb protokollok:

ICMP (Internet Control Message Protocol): mint a neve is mutatja, üzenetkezelésre, vezérlésre szolgáló protokoll. Az ICMP protokollal leginkább a 'ping' parancs forrt össze, amellyel egy IP cím elérhetőségét vizsgáljuk. A válasz szintén egy ICMP üzenet, az 'Echo'.

UDP (User Datagram Protocol): A TCP-ez hasonlóan az IP feletti protokoll, de annál jóval egyszerűbb. Az UDP az üzenetet nem darabolja fel, ahogy a TCP teszi. Ez a protokoll egyirányú, ami azt jelenti, hogy az UDP csomag (azaz a datagramm) megérkezéséről a feladó nem vár nyugtát. Legismertebb alkalmazása a domain név feloldása IP címmé, vagy a ping használata.

PPTP (Point-to-Point Tunneling Protocol): lehetővé teszi, hogy a nyilvános hálózaton egy saját kommunikációs csatorna (az ún. tunnel) létrehozásával két magánhálózat biztonságosan összeköthető legyen. (Ez a fajta összeköttetés a VPN – Virtual Privat Network Protocol).

3. A portokról

A portok egy hálózati kommunikációs csatorna végpontjai. A portok használata teszi lehetővé, hogy egy adott számítógépen futó alkalmazások ugyanazt a hálózati erőforrást használva, a beérkező csomagokból csak a nekik szóló csomagokat kapják meg. (Például az egyik gépen lévő browser (Netscape vagy Internet Explorer) lekér egy másik gépen futó web szerverről egy html lapot, akkor a két gép között létrejövő kommunikációs csatornát az egyes gépek IP címei, valamint a browserhez illetve a web szerverhez tartozó portszám határozza meg.)

A protokollok, illetve az egyes szolgáltatások (szerver) és a szolgáltatásokat igénybevevő kliensek a kapcsolatfelvételhez, valamint a kommunikációhoz az IP cím mellett használják a portokat. A portokat számokkal (is) azonosítjuk, értékük 0-65535 között lehet. A portok és a hozzájuk tartozó protokollok/szolgáltatások azonosításával a IANA (Internet Assigned Numbers Authority) foglalkozik.

A portszámokat felhasználásuk szerint három csoportba osztják:

- jól ismert portok (well known ports),

- regisztált portok,
- dinamikus vagy privát portok.

A jól ismert portok a 0-1023-as sávban lévő portok, amelyeket általában csak rendszerprocesszek vagy rendszerprogramok használnak, és ezek szorosan kötődnek valamilyen szolgáltatáshoz: a 20 és 21-es port az FTP-hez, 22-es az SSH-hoz, 25 az SMTP-hez, 32 a telnet-hez, a 80-as a HTTP-hez stb.

A regisztrált portok (1024-49151) sokkal kevésbé kötődnek egy-egy szolgáltatáshoz, ilyen portszám többféle célra is felhasználható. A privát portokhoz (49152-65535) nem kapcsolódik semmilyen szolgáltatás.

A portok biztonsági szempontból is osztályozhatók, attól függően, hogy alapértelmezésben biztonságos, vagy nem biztonságos protokollokhoz, illetve szolgáltatásokhoz kapcsolódnak.

3. A kriptográfia lényege, feladatai

Rejtjelezés minden olyan tevékenység, eljárás, amelynek során valamely adatot abból a célból alakítanak át, hogy annak eredeti állapota, a megismerésére illetéktelenek számára rejtve maradjon. A rejtjelezés részét képezi a rejtjelezett adat eredetivé való visszaállítása is (kriptoanalízis).
A kriptográfia az algoritmusok (matematikai módszerek, eljárások), a hozzájuk kapcsolódó kulcsrendszerek és a védelmi rendszerek együtteséből áll.

A kriptográfia alapvető céljai:

- **Titkosítás:** Célja a lehallgatás megakadályozása, pontosabban annak biztosítása, hogy egy lehallgatást végző támadó ne értse meg a lehallgatott üzenetek tartalmát.
- **Integritásvédelem:** Célja annak biztosítása, hogy egy üzenet vevője megbízhatóan meg tudja állapítani a vett üzenetről, hogy annak tartalma az átvitel során módosult-e vagy sem.
- **Hitelesítés:** Célja az üzenetfabrikálás és a megszemélyesítés detektálása. A hitelesítés folyamata során tehát a hitelesítést végző résztvevő megbízhatóan meggyőződik egy

vett üzenet feladójának vagy egy párbeszédben részvevő másik félnek a kilétéről. Előbbit üzenethitelesítésnek, utóbbit partnerhitelesítésnek nevezzük.

- **Letagadás elleni védelem:** Célja annak elérése, hogy egy üzenet küldője ne tudja letagadni az üzenetküldés tényét és az időbélyegző révén az idejét, vagy más szavakkal, az üzenet vevője bizonyítani tudja, hogy ki az üzenet küldője, ugyanakkor senki ne tudja észrevétlenül változtatni, módosítani az üzenet tartalmát és a továbbítás adatait.

A fenti célok elérése különböző kriptográfiai mechanizmusok (algoritmusok és protokollok) alkalmazásával lehetséges. Az adatátviteli protokollok által használt néhány ismertebb kriptográfiai algoritmus

A. Szimmetrikus (ugyanaz a kódoló és dekódoló kulcs): DES, IDEA, CAST, RC5

B. Nyilvános kulcsú (aszimmetrikus: a kódoló és dekódoló kulcs más*):* RSA, ECC

5. Digitális aláírás

A digitális aláírás olyan eljárás, mely biztosítja az üzenetek integritását és hitelességét, valamint az üzenetek eredetének letagadhatatlanságát, aszimmetrikus tulajdonsága miatt digitális aláírás sémák a nyilvános kulcsú technikákra épülnek.

Tulajdonságai nagymértékben hasonlítanak a hagyományos aláírás tulajdonságaihoz. Egy fontos különbség a digitális és a hagyományos aláírás között az, hogy a digitális aláírás nem az üzenet anyagi hordozójához (például papír) kötődik, hanem magához az üzenethez. Így nemcsak az üzenet eredetére vonatkozóan nyújt garanciát, hanem segítségével az üzenet tartalmában az aláírás generálása után bekövetkezett módosításokat is detektálni lehet.

Egy digitális aláírás séma két összetevője:

- aláírás generáló algoritmus
- aláírás ellenőrző algoritmus.

Az *aláírásgeneráló algoritmus* az aláíró fél titkos aláíró kulcsával van paraméterezve, míg az ellenőrző algoritmus az aláíró fél nyilvános aláírásellenőrző kulcsát használja paraméterként. Az aláírásgeneráló algoritmus bemenete az aláírni kívánt m üzenet, kimenete pedig egy szigma digitális aláírás.

Az *aláírásellenőrző algoritmus* bemenete egy m üzenet és egy szigma aláírás, kimenete pedig 1, ha szigma az aláíró fél érvényes aláírása m-en, és 0 egyébként. Értelemszerűen, az ellenőrzést végző fél akkor fogadja el az aláírást hitelesnek, ha az ellenőrző algoritmus kimenete 1. Természetesen az ellenőrzés végrehajtásához az ellenőrző félnek ismernie kell az aláíró fél nyilvános kulcsát, mert ez szükséges az ellenőrző algoritmus helyes paraméterezéséhez. Az aszimmetrikus kulcsú rejtjelezéshez hasonlóan, itt is biztosítani kell tehát a nyilvános kulcsok hitelesítését.

Gyakorlati okokból célszerű, ha a digitális aláírás mérete nem függ az aláírt üzenet méretétől. Ezt úgy érhetjük el, hogy nem magát az üzenetet írjuk alá, hanem annak egy rögzített hosszúságú lenyomatát, amely praktikus értelemben megbízhatóan reprezentálja magát az üzenetet. Ehhez egy kriptográfiai lenyomatképző függvényt, vagy más néven hash függvényt használunk, amely tetszőleges hosszúságú bináris sorozatot rögzített hosszúságú (rövidebb) bináris sorozatba képez le. Annak érdekében, hogy a lenyomat valóban megbízhatóan reprezentálja az üzenetet, egy kriptográfiai hash függvénytől megköveteljük, hogy az egyirányú és ütközés ellenálló legyen.

6. Titkosítás, biztonságos protokollok használata

Kriptográfiai eljárások számos alkalmazásával találkozhatunk a mindennapi életben. Kriptográfiai módszerekkel történik például a

- híváskezdeményező hitelesítése a GSM rendszerben
- a bankkártyák mágnes csíkján tárolt adatok védelme az illetéktelen módosítások ellen
- fizetős műholdas adások kódolása
- Biztonsági protokollok vezetékes számítógépes hálózatokban
    - hálózati rétegben megvalósított biztonság (IPSec);
    - szállítási rétegben megvalósított biztonság (SSL/TLS);
      az SSL protokoll hitelesített és titkos kommunikációt tesz lehetővé a böngésző és a webszerver között, illetve a levelező szerver és kliensek között
    - alkalmazási rétegben megvalósított biztonság (SSH, PGP);
      az SSH (Secure Shell) protokoll az RSA algoritmust használja a fájlátvitel, illetve távoli gépre történő biztonságos bejelentkezés során
- Biztonsági protokollok vezeték nélküli számítógépes hálózatokban

- Mobil IP biztonsága;
- WiFi biztonsági protokollok (WEP, WPA, WPA2);
- cellás mobil rendszerek biztonsága (GSM és UMTS);
- Bluetooth biztonsági protokoll

7. A védelmi megoldások és feladatok rövid áttekintése

- ***Vírusirtó programok***

  A vírusirtó program megakadályozza, hogy rosszindulatú programok kárt tegyenek a rendszerben/fájlokban illetve hogy a hackerek ún. hátsó kapukat és trójai falovakat telepítsenek a gépre adatok és jelszavak ellopásához

- ***Tűzfalak***

  A hardveres (pl. router) és szoftveres tűzfal szabályozza az adott informatikai rendszeren a be- és kimenő hálózati forgalmat.

- ***Programok frissítése***

  A gyártók gyakran adnak ki frissítéseket és hibajavításokat programjaikhoz, melyek befoltozzák a felfedezett biztonsági réseket.

- ***Kém- és reklámprogramirtók***

  Jelszavainkat és egyéb fontos adatokat lophatnak el tőlünk a kémprogramok, a reklámprogramok pedig kéretlen reklámokat jelenítenek meg és információt gyűjtenek internetezési szokásainkról.

- ***Hozzáférések korlátozása***

  Jelszavak korlátozzák a hozzáférést legtöbb számítógéphez, szolgáltatáshoz, bár manapság egyre inkább terjedőben vannak az alternatívák, mint a biometrikus azonosítás, vagy a smart card.

- ***Rendszeres biztonsági mentések***

  Az adatvesztés megelőzésére redundáns adattárolást érdemes bevezetni, mint például a RAID. Egy meghajtó élettartamára a SMART adatokból következtethetünk. Biztonsági mentést végezhetünk külső merevlemezre, szalagos adattárolóra vagy akár egy távoli szerverre is, így biztosítva a földrajzi elhatároltságot. Ugyanakkor a

megfelelő backupot nem helyettesíti a raid, hiszen egy helyen vannak a merevlemezek, így nem védenek például tűz, földrengés, és sok esetben a táp meghibásodásától.

- ***Titkosítás***

  A titkosítás során fájlokat, virtuális meghajtókat, merevlemezeket, pendrive-okat, telefonhívásokat, üzeneteket, online tranzakciókat védhetünk meg az illetéktelen hozzáféréstől egyszerre biztosítva a hitelességet, a bizalmas kezelést, az információ integritását és ezek tanúsítását egy harmadik félnek.

- ***Adatvédelem az Interneten***

  Böngészés során sok személyes adatnak minősülő adatot hagyunk hátra a böngészőben és a meglátogatott honlapokon. Egyes szoftvercsomagok komplett megoldásokat kínálnak ezekre az adatvédelmi kihívásokra (nyomkövető sütik, private header, szülői felügyelet, jelszavak küldésének szabályozása stb.).

- ***Megfelelő lépések és kockázatbecslés***
  - Jogosultságok beállítása (lehetőleg csoportok és nem felhasználók szintjén)
  - Védelmi eszközök karbantartása
  - Házirendek létrehozása és ellenőrzése, a felhasználók oktatása
  - Forgatókönyvek készítése vész- ill. katasztrófahelyzetekre

8. Rövidítések, mozaikszavak listája

ACL	Access Control List
API	Application Programming Interface
ARP	Address Resolution Protocol
CPU	Central Processing Unit
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarized Zone
DNS	Domain Name Service
ECC	Edge Card Control
FTP	File Transfer Protocol
GCC	GNU C Compiler

GNU	GNU Not Unix
GSM	Global System for Mobile Communication
HTTP	Hypertext Transfer Protocol
HTTPS	Secure Hypertext Transfer Protocol
IANA	Internet Assigned Numbers Authority
ICF	Internet Connection Firewall
ICMP	Internet Control Message Protocol
IDAM	Identity & Access Management
IDS	Intrusion Detection System
IMAP	Internet Message Access Protocol
IOS	Input Output System
IP	Internet Protocol
IPS	Intrusion Prevention System
IPsec	IP security
IrDA	Infrared Data Association
IRC	Internet Relay Chat
ISA	Internet Security and Acceleration
ISO	International System Organisation
L2, L3	Layer2, Layer3
LAN	Local Area Network
MAC	Media Access Control
MIME	Multipurpose Internet Mail Extensions
NAT	Network Address Translation
NDS	Netware Directory System
OSI	Open System Interconnection
PAM	Pluggable Authentication Module
PGP	Pretty Good Privacy
PKI	Public Key Infrasturcture
PLC	Packet Line Carc
POP3	Post Office Protocol 3
PPTP	Point to Point Tunnelling Protocol
RF	Radio Frequency

RFC	Request For Comments
RPC	Remote Procedure Call
PPTP	Point-to-Point Tunneling Protocol
RAID	Redundant Array of Inexpensive Disks
RSA	Rivest Shamir Adelman (algorithm)
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDDI	Universal Description, Discovery, and Integration
UDP	User Datagram Protocol
UI	User Intrerface
UMTS	Universal Mobile Telecommunications Systems
UPnP	Universal Plug and Play
URL	Universal Resource Location
USB	Universal Serial Bus
VLAN	Virtual Local Area Network
VPN	Virtual Privat Network Protocol
WEP	Wired Equivalent Privacy
WiFi	Wireless-Fidelity
WPA	WiFi Protected Access
WSDL	Web Services Description Language
XML	Extensible Markup Language
XSS	Cross Site Scripting

Felhasznált irodalom

- Michael D. Bauer: Szerverek védelme linuxszal (O'Reilly Kossuth, 2003.)
- Tony Bautts, Terry Dawson, Gregor N. Purdy: Hálózati adminisztrátorok kézikönyve (O'Reilly Kossuth, 2005.)
- Othmar Kyas: Számítógépes hálózatok biztonságtechnikája (Kossuth Kiadó, 2000.)
- Kónya László: Számítógép-hálózatok (Inok Kiadó, Budapest 2006.)
- MTA SZTAKI 2004-es tanulmánya:Az informatikai hálózati infrastruktúra biztonsági kockázatai és kontrolljai
- http://www.szabilinux.hu/security/bizt/node1.html
- http://www.itb.hu/ajanlasok/a8/html/a8_m3_1.htm
- http://www.openssh.com/hu/index.html
- http://www.biztostu.hu
- http://www.safesoft.hu
- http://www.avisys.hu
- http://www.humansoft.hu
- http://www.balabit.hu
- http://www.utimaco.com
- http://phserver.phy.georgiasouthern.edu/rldeal/m685/wnic/Docs/HUN/security.htm
- http://www.gentoo.org/doc/hu/security/security-handbook.xml
- https://www.icsalabs.com/icsa/product.php?tid=fghhf456fgh
- http://www.iana.org/assignments/port-numbers
- http://www.wikipedia.hu