AZ INFORMATIKAI HÁLÓZATI INFRASTRUKTÚRA BIZTONSÁGI KOCKÁZATAI ÉS KONTROLLJAI

Készítő:	MTA SZTAKI
Státusz:	*Második mérföldkő lezárása; nyilvános!*
	2005. június

A tanulmány elkészítésében és belső lektorálásában részt vettek:
Becz Tamás, Kincses Zoli, Lakatos György, Pásztor Szilárd, Rigó Ernő, Tiszai Tamás, Tóth Beatrix

Tartalomjegyzék

Táblázatjegyzék

Ábrajegyzék

1. Bevezető

Jelen tanulmány az IHM-MTA Kutatási Program keretében végzett *„Internet védelmi rendszer struktúrájának kidolgozása”* című kutatási projekt (Sorszám: *E4*, Iktatószám: *4671/4/2003*) részét képezi. A projekt fázisait magába foglaló mérföldkövek és a hozzájuk kapcsolódó határidők (kiemelve a jelen tanulmány által lefedni szándékozó részt):

> 1. mérföldkő: 1-2 kutatási fázis (Informatikai hálózati infrastruktúra biztonsági kockázatainak elemzése, és a kockázat-kezelési lehetőségek feltárása), lezárás: 2004. június 30.
>
> **2. mérföldkő: 3 fázis (Biztonsági mintarendszerek kidolgozása), lezárás: 2005. június 30.**
>
> 3. mérföldkő: 4 fázis (A biztonsági rendszerek üzemeltetési módszertanának kidolgozása), lezárás: 2006. február 28.

Az Informatikai Hálózati Infrastruktúra Biztonsága (továbbiakban – IHIB) magában foglalja a hálózat működéséért felelős hardver és szoftver elemeket, de ezeken felül számításba veszi a humán faktort és az egészet körülvevő adminisztratív jellemzőket is.

A mostani fázisban elkészült mintarendszert ToReS-nek neveztük el, ami az angol Tools Related to Security elnevezés rövidítése. Az rendszer a tanulmányhoz mellékelve található, míg tartalmáról és annak felhasználási módjáról több különböző fejezetben is szó lesz.

1.1. A tanulmány felépítése

A tanulmány a bevezető részek (Bevezető, Biztonsági politika...) után kitér a figyelembe veendő szabályokra (Adatvédelem, Segítséggel élőkre vonatkozó szabályok, Szerzői jog és szabadalmak, Igazságügyi eljárás). A kockázatelemzéssel az első kötet [Tan_1] foglalkozott részletesebben, ebben a kötetben az általános topológia → biztonságos topológia átalakításától indulva a szolgáltatások biztonságán át a hosztok biztonságáig terjedően tárgyaljuk az egyes témaköröket. Ezeket kiegészíti az olyan ritkábban használt, de egyre terjedőben lévő technikák említése, mint az intelligens kártya vagy a biometria.

A ToReS mintarendszerben lévő és a biztonsághoz szükséges eszközök nagy része a mellékelt CD-n megtalálható, míg az eszközök beállításait, és minden ehhez kapcsolódó információt is taglalunk. A melléklet egyik fejezete a CD indításával és használatával foglalkozik.

A szoftverek és csomagok listája megtalálható a ToReS_csomaglista fájlban (SXW és PDF formátum) a CD-n. A biztonsági eszközök főleg az admin és utils csomagokban találhatók, de a fejlesztők (devel, interpreter stb.) és a felhasználók (editor, graphics, net, sound stb.) is megtalálják a számukra fontos és hasznos eszközöket.

A Windows felhasználók számára elérhető a Sysinternals programjai mellett HASH programok, jelszótároló alkalmazás vagy digitális aláírás alkalmazás (Linuxra és más rendszerekre is elérhetők!)

A mellékletek között találhatunk még összefoglaló biztonsági fenyegetettségeket taglaló toplistákat, vagy olyan néhány oldalas összefoglalókat, melyek biztonsági problémák felléptekor, reményeink szerint csak ritkán, kerülhetnek felhasználásra. Ezekkel az a cél, hogy amennyiben mégis szükség lenne rájuk, akkor egyszerűen és gyorsan lehessen használni a tömören összefoglalt információkat.

1.2. Röviden a tartalomról és a célokról

Az anyagban foglaltak a non-profit szervezetek (akadémiai intézmények, önkormányzatok) számára ad egy használható koncepciót és megoldásgyűjteményt. Ezek magukba foglalják az egyes alkalmazások és rendszerelemek javasolt telepítését, beállításait és működtetését is.

A felhasznált eszközök elsősorban szabad szoftverek, néhány (elterjedtsége, szolgáltatásai, célnak való megfelelősége miatt) kiemelkedő pénzes eszköz, illetve saját fejlesztések. Az egyes eszközökhöz tárgyalt beállítások és rendszerelemek a mellékelt CD-n találhatók meg teljes terjedelemben, ezért ebben az anyagban csak a lényegi részek kerülnek bemutatásra.

Míg a tanulmány előző része [Tan_1] bővebben ismertette az elméleti alapokat és az ezeken nyugvó gyakorlati alkalmazásokba adott betekintést, addig ez a rész sokkal gyakorlatiasabb, így „szűkszavúbb", ezáltal informatikai képzettséget vár el és feltételez a tárgyalt megoldások telepítése, alkalmazása és beállítása során.

A tanulmány célja, hogy az egyes hálózatokért, részhálózatokért vagy néhány gépért felelős helyi rendszergazdák, vagy ilyen feladatot ellátó személyek segítséget kapjanak az általuk felügyelt rendszerek biztonságosabbá tételéhez, és szükség esetén a hozzájuk forduló kollégák munkájának segítéséhez. Amennyiben a rendszerek topológiájában is átalakításra van szükség, ehhez is segítséget nyújt ez a tanulmány, valamint a mellékelt ToReS CD.

1.3. Szerzői jogi nyilatkozatok

A szerzői jogról szóló törvényi szabályok [Szerzői_jog] szellemében kell a tanulmánnyal eljárni úgy a készítőkre, mint átvevőkre és az olvasókra vonatkozóan. Hasonló módon az Adatvédelmi törvény [AVT] ide vonatkozó paragrafusait is alkalmazni kell.

A szerzők nem vállalnak semmilyen felelősséget az anyagok téves felhasználásából, részben kiragadott, vagy jogszerűtlen felhasználásából eredő károkért, és az általuk készített anyagokkal kapcsolatban is csak azt tudják vállalni, hogy legjobb szakmai tudásuk szerint állították össze azokat.

A tanulmány számos olyan linket (kapcsolódási pontot) tartalmaz, amelyek az Internet más és más oldalaira vezetnek. Ezen oldalak tartalmáért és szolgáltatóik adat-, valamint információvédelmi gyakorlatáért a szerzők nem vállalnak felelősséget.

A tanulmányban említett konkrét rendszerek a védjegyet birtokló cég tulajdonában vannak, a példák csak az adott témakör szemléltetésére szolgálnak, azokból általános következtetéseket nem érdemes levonni.

A mellékelt CD csak egy példányban készült a tanulmányban használt fontosabb hivatkozások archiválására és az ajánlott vagy bemutatott eszközök összegyűjtésére. A CD nem másolható, és tartalma nem tehető nyilvánossá, csak a tanulmányt olvasó használhatja segítségként, amennyiben az anyagban előforduló fontosabb hivatkozások nem lennének elérhetők a megadott címen, vagy nincs Internet-kapcsolata. Az összegyűjtött eszközökkel szemben a mindenkor érvényes[1] szerzői jogi szabályokat kell betartani!

[1] Előfordulhat, hogy egy termék szabadon elérhető, de később a készítő változtat a licencpolitikáján, így a felhasználás időpontjában érvényes szabályokat kell figyelembe venni.

2. Biztonsági politika, fogalmak, írott szabályzatok

Az informatikai és az információs biztonság bevezetése, fenntartása, betartása és számonkérése csak vezetői elkötelezettség, felügyelet és támogatás esetén lehet hatékony. Amennyiben e vezetői hozzáállás nem biztosított, úgy még van esély a biztonságos működésre, de az állapotok nem nevezhetők ideálisnak.

Minden biztonsági rendszer akkor a leghatékonyabb, ha írott szabályzatokon alapul, amit mindenki egységesen ismeri, és be is tartja ezeket a szabályokat. A szabályok elsősorban nem megkötéseket tartalmaznak, hanem a felelősségek, kötelességek, lehetőségek és tiltott cselekmények leírását. A katasztrófatervet is nyugodt időszakban, jól végiggondolt módon kell kidolgozni, a biztonsági szabályokat is az események bekövetkezte előtt kell bevezetni, hogy a problémák felléptekor mindenki ismerje szerepét és dolgát.

Ezt a vezetői elkötelezettséget és hozzáállást jelenti a „Biztonsági politika", melynek nagyon fontos szimbolikus szerepe van a szabályzatok és a biztonsági megoldások kialakításában.

A szabályozás kapcsán fontos tisztázni, hogy adott fogalmakon mindenki ugyanazt érti-e, ezért szükséges az alapvető kifejezéseket röviden meghatározni.

2.1. Fogalmak

A hálózatbiztonság területén léteznek alapfogalmak, melyekből az első kötetben többet is használtunk, definiáltunk, idéztünk olyan forrásokból, melyek közös hivatkozási alapot jelenthetnek szakemberek és laikusok számára egyaránt. Néhány példa a fontosabbakra, melyeket ennek az anyagnak a megértéséhez fontos meghatározni.

Hálózat biztonság:

Minden, a hálózattal kapcsolatba hozható dolog (adat, adathordozó, felszerelés stb.) megfelelő védelmezésére utal. Magában foglalja az adminisztratív funkciókat, mint pl. a fenyegetettség vizsgálatát, a technikai eszközöket és berendezéseket, mint a kriptográfiai termékek, és hálózati hozzáférést szabályozó termékek (pl. tűzfalak). Tartalmazza továbbá a hálózat erőforrásainak a szabályozásban előre megfogalmazott módon, és kizárólag az arra felhatalmazott személyek általi használatának kikényszerítését.

Biztonsági management:

Szabályozások és eljárások, amelyek csökkentik a sikeres betörés esélyét, és növelik a mégis bekövetkezettek felfedezésének esélyét.

Biztonsági szabályzat:

Szabályok és gyakorlatok olyan halmaza, amelyek meghatározzák vagy szabályozzák azt, hogy egy rendszer vagy szervezet hogyan nyújt biztonsági szolgáltatásokat erőforrásainak védelme érdekében. Részét képezi a biztonsági architektúrának.

Az egyes fogalmak a tanulmányban kerültek kifejtésre előfordulási helyükön. Ezen kívül ajánljuk a következő magyarázó szótárakat és fogalomtárakat:

- a [Fogalomtár] használatát, illetve a PDF változatot az IHM honlapjáról
- a szabadon használható és le is tölthető szótárt a [Dictionary] címről.
- az [Infosec] szótár on-line vagy PDF verzióját,

melyek közül az első magyar, a többi angol nyelvű.

2.2. Szabályzatok felépítése

A mintarendszer alapeleme a széles körben elismert szabványok és ajánlások figyelembevétele, és a rendszer működését valamint a működtetők munkáját szabályozó írott dokumentumok rendszere.

A biztonsági szabályzatok felépítésénél fontos megemlíteni, hogy egy minőségbiztosítási rendszer léte (pl. ISO 9001:2000) nagyban segíti a dokumentációs rendet és a szabályzatokhoz való megfelelő hozzáállást. Az információbiztonság (menedzsment és technikai rész) egyik legelterjedtebb szabályozási módszertana a BS 7799, mára részben ISO szabvánnyá vált, és magyar fordításban is megjelent:

- MSZ ISO/IEC 17799:2002. Az informatikai biztonság menedzselésének eljárásrendje
- MSZE 17799-2:2004. Az információvédelem irányítási rendszerei. Előírás és használati útmutató

A minőségbiztosítási és információbiztonsági szabványok közötti harmonizáció miatt a következő táblázat állítható fel:

	ISO 9001:2000 (MB – Minőség Biztosítás)	ISO 17799 (IB – Informatikai Biztonság)	Megjegyzés
0	Alkalmazási terület, Rendelkező hivatkozás, Szakkifejezések	1. Alkalmazási terület 2. Meghatározások- informatikai biztonság- kockázatbecslés- kockázatkezelés	A szabályzat célját és a fogalmak definiálását már az elején meg kell ejteni.
1	4. Minőségirányítási rendszer - általános követelmények - dokumentálás követelményei	3. Biztonsági szabályzat - informatikai biztonsági szabályzat	A szabályzat tartalmi meghatározása.
2	5. Vezetőség felelősségi köre - elkötelezettség - vevőközpontúság - minőségpolitika - tervezés - felelősség és hatáskörök, kommunikáció - átvizsgálás	4. Szervezetbiztonság - informatikai biztonság infrastruktúrája - harmadik fél hozzáférése - erőforrás-kihelyezés, alvállalkozók	Az MB-ben megfogalmazott elvárások részben tartalmazzák az IB követelményeit is, így IB ⊂ MB relációban vannak.
3	6. Gazdálkodás az erőforrásokkal - gondoskodás erőforrásokról - emberi erőforrások - infrastruktúra - munkakörnyezet	5. Vagyonosztályozás és ellenőrzés - vagyoni felelősségre vonás - információ minősítés 6. Személyzet biztonsága - munkakör és erőforrások biztosítása - képzés - véletlen biztonsági eseményekre és zavarokra adott válasz	A MB-ban megfelelő módon szabályozott kérdések után az IB itt is kiegészítő rendelkezéseket valósíthat meg a biztonságra koncentrálva.
4	7. Termék előállítása - előállítás megtervezése - vevőfolyamatok - tervezés és fejlesztés - beszerzés - előállítás és szolgáltatás - megfigyelő és mérőeszközök kezelése	7. Fizikai és környezeti biztonság - biztonságos zónák - berendezés biztonsága - általános óvintézkedések 8. Kommunikáció és üzemeltetés - eljárások és felelősségek - tervezés és átvétel - rosszindulatú kód elleni védelem - házirend (sértetlenség és rendelkezésre-állás fenntartása, naplózás) - hálózat-, közegek, információcsere és szoftverváltás menedzselése	Mivel az IB-ban a biztonság a termék, ezért itt ezek minősége = biztonságukkal. A MB-ban a „vevő érdeke nem sérülhet”, míg az IB-ban a szolgáltatás alá vontakra érvényes ugyanez az érdekvédelem.

	ISO 9001:2000 (MB – Minőség Biztosítás)	ISO 17799 (IB – Informatikai Biztonság)	Megjegyzés
5	8. Mérés, elemzés, fejlesztés - útmutatás - figyelemmel kísérés és mérés - nem megfelelő termék kezelése - adatok elemzése - fejlesztés	9. Hozzáférés-ellenőrzés - üzleti követelmények - felhasználói hozzáférés és felelősség, hálózati-, operációs rendszer-, alkalmazás-, rendszerhasználati hozzáférés és hozzáférés figyelése - mobil- és távmunka 10. Rendszerfejlesztés és karbantartás - biztonsági követelmények - alkalmazási rendszerek biztonsága - kriptográfiai óvintézkedések - rendszerfájlok biztonsága - fejlesztő és támogató folyamatok biztonsága 11. Üzletmenet-folytonosság 12. Előírások betartása - jogi követelmények - biztonsági szabályzat és műszaki megfelelőség felülvizsgálata - rendszer audit	A MB 8. fejezet részletezése az IB 9-11. fejezetei. Az IB 12. fejezete részletesebb kiegészítése a MB-ban alapnak számító auditnak, mely tulajdonképpen a minősítő eljárás. Belső audit szerepel a MB-ban is (8.2.2), és a termék megfelelőségének figyelése, elemzése valamint a helyesbítő és megelőző tevékenységek is az IB területén kontrollként ismert eljárások alapjait adják.

1. Táblázat: ISO 9001:2000 – ISO 17799 összehasonlítása

Az egyes szabványok alapján mindenki elkészítheti magának is a megfelelő szabályzatokat, valamint a részletes technikai szabályzatokat is. A szabványok megvásárlása után ennek az eljárásnak a költséghatékony módja a külföldi vagy hazai forrásból elérhető mintaszabályzatok[2] felhasználása

A folyamatok érzékeltetésére ismertetünk két listát, melyek címszavai alapján már „érezhető", hogy mire és hogyan kell figyelni a szabályzatok elkészítésében.

Javasolt gyakorlati teendők a hatékony információbiztonsághoz (Sarah D. Scalet and Scott Berinato's: „The 10 Key Components of Good Information Security" anyaga alapján):

1. Azonosítsd a veszélyeket! Ebben segíthet az első tanulmány [Tan_1]
2. Vond be a főnököt! Szükséges, hogy mindenki megértse, hogy ez nem önmagáért van, hogy a biztonságnak napi gyakorlattá kell válnia, és ehhez segít a „hatalmi szó". Arról nem is beszélve, hogy a mindenkibe a főnök is beletartozik!
3. Nevezz ki felelőst! Ha nincs ilyen, minden el fog sikkadni (a felelősnek határidőt is tanácsos adni).
4. Fejlessz ki, és vezess be egy biztonsági szabályzatot! Ebben segítségedre lesz ez a tanulmány is.
5. Képezd az alkalmazottakat, és növeld figyelmüket a téma iránt! A leggyengébb láncszem mindig az ember. Fontos hogy tudja, mire és mért kell figyelnie.

[2] ld. a fejezet végi hivatkozásoknál Berényi Melinda szakdolgozatát.

6. Csináltass biztonsági auditot! Ki fog derülni, hogy sok gyenge pont volt. Ezek nyilvánosságra hozatala segíthet az 5. pont megvalósításában.
7. Ne feledkezz meg a fizikai biztonsági kérdésekről! Bármely gép feltörhető, ha fizikailag hozzáférnek. Bármi ellopható, ha szabadon áramolnak az adathordozók a zárt hálózatban.
8. Ne feledkezz meg a belső veszélyekről!
9. Légy naprakész! A biztonság nem egyszeri feladat. Frissíteni kell. Tisztában kell lenni az újdonságokkal.
10. Készülj fel a legrosszabbra! A biztonsággal foglalkozóknak a paranoia kívánatos állapot.

Egy hosszabb lista szerint 16 pontban foglalhatók össze a teendők („Practice List for Information Security Management" GAO – General Accounting Office lista).

1. Azonosítsd az információs erőforrásokat és szervezeti vagyont, melyet meg kell védeni!
2. Fejlessz ki gyakorlati veszély-értékelési eljárásokat, melyek összekötik a biztonságot az üzleti érdekekkel!
3. Tartsd meg a program és üzleti menedzserek felelősségét!
4. Folyamatosan foglalkozz a veszélyekkel!
5. Hozz létre egy központi csoportot a kulcsfeladatok végrehajtására!
6. Biztosíts független elérést a központi csoportnak a vezető tisztségviselők felé!
7. Jelölj ki dedikált személyzetet és finanszírozást!
8. Növeld a személyzet szakértelmét és technikai képzettségét!
9. Kösd össze a szabályokat az üzleti kockázatokkal!
10. Tégy különbséget a szabályok és az irányelvek, útmutatók között!
11. Támogasd a szabályok betartását a központi csoporton keresztül!
12. Folyamatosan képezd a felhasználókat (és mindenki mást is) a veszélyekről és a kapcsolódó szabályozásról!
13. Használj figyelemfelkeltő és felhasználóbarát technikákat!
14. Figyeld azokat a tényezőket melyek befolyásolják a veszélyeket, és visszajelzést adnak a biztonsági hatékonyságról!
15. Használd az eredményeket, hogy irányíthasd a jövőbeni törekvéseket, és megtarthasd a menedzserek felelősségét!
16. Légy figyelemmel az új megfigyelési és monitorozási technikákra!

A szabályzatok fontosságát már az előző tanulmányban is kiemeltük, de az általánosságok listáin túlmenően álljon itt néhány cím, melyről a szabályzatkészítéshez kaphatunk útmutatást, és mintaszabályzatok is elérhetők (a harmadik címen magyar nyelven is):

http://www.secinf.net/policy_and_standards/

http://www.sans.org/resources/policies/

http://www.cert.hu/ismert/?cat=a9szakdolgozat

3. Egyéb figyelembe veendő szabályok

A szabályzatok megírásánál ügyelni kell arra, hogy sem a rendszergazdai sem a felső vezetői diktatúra leképezésének nem szabad engedni. Kiemelten értendő ez akadémiai körben, ahol a kutatói és oktatói (és sok esetben a hallgatói) szabadság vélt és valós jogai közepette még fontosabb a felhasználók közreműködése a szabályok betartásában.

Ezen felül más intézményekben sem szabad a helyi szabályzatokban a magasabb rendű jogszabályoknak ellentmondani, vagy azokat felülszabályozni, ezért a következő területekre és azok alapszabályaira kell figyelemmel lenni.

3.1. Privacy / Adatvédelem

Az Országgyűlési Biztosok Hivatala honlapjáról [Adv_HB] elérhető az Adatvédelmi Biztos hivatala és az Adatvédelmi törvény is, mellyel kapcsolatban fontos tudni, hogy:

- az adatalanyt védi, ezáltal az a szemlélet érvényesül, hogy az adatokat előbb címkézni kell, és utána szabályozni azt, ami nem következik a törvényből (pl. milyen adatok kinek a tulajdonát képezzék)
- kiemelten fontos az adatok célhoz kötöttségének elve, tehát nem fogadható el a „minden adatot mindenkiről határozatlan ideig" szélsőségek egyike sem, ha ezekhez nem társul ezt alátámasztó indok és kezelő szabályzat.
- létezik adatvédelmi audit is, amely során felderíthető, hogy az adatvédelem területén milyen szabályzási, eljárási vagy egyéb megoldandó gondok merülnek fel.

3.2. Deficiency / Segítséggel élők

A MITS-en (Magyar Információs Társadalom Stratégia) belül több részstratégia[3] létezik, melyek közül az egyik az esélyegyenlőségi (e-Esély) részstratégia. Ez röviden – nem részletekbe menően – foglalkozik a digitális világban fontos esélyegyenlőséggel, és ennek megteremtési módjaival. A konkrétumok szintjén több területre kell figyelemmel lenni, ha az esélyegyenlőség szellemében akarunk eljárni. Ezek a területek röviden a következő pontokba foglalhatók össze:

- fizikai hozzáférés (közlekedés, megközelítés) biztosítása;
- logikai hozzáférés (Internet, számítógép, mobiltelefon, e-szolgáltatások);
- a hozzáférések alternatíváinak és specifikumainak kidolgozása szabványok és ajánlások szerint, melyek angol nyelven elérhetők a http://www.tiresias.org lapról, de magyar nyelven is léteznek segítő leírások. Ezek közül néhány példa:
    - Weblapok tervezése és kialakítása pl. vakbarát módon [Paramédia], melyhez ellenőrző program is elérhető a megfelelőség ellenőrzéséhez,
    - Terminálok elhelyezése, adatbeviteli és adatközlő csatornáinak kialakítása (pl. Tiresias fontkészlet, tolókocsis hozzáférés, mobiltelefon bevonása) [Tiresias],
    - Kiegészítő megoldások alkalmazása (pl. hangjelzés a vakoknak, fényjelzés a siketeknek, intelligens kártyán tárolt fogyatékosság-típusokhoz rendelt kiegészítő szolgáltatások) [Tiresias].

[3] ld. http://www.itktb.hu/ oldalon

A segítséggel élők számára kialakított megoldások sok más ember számára is előnyösek lehetnek!

3.3. Copyright / Szerzői jog, szabadalmi törvény

A szerzői jogi törvény, és egyéb szabályok elérhetők az Artisjus – Magyar Szerzői Jogvédő Iroda Egyesület – honlapján [Artisjus]. Fontos tisztában lenni az egyes oltalom alatt álló művek közzétételének szabályaival, kétes esetben az egyesület jogsegélyszolgálatát lehet és érdemes felkeresni, mielőtt jogi problémák adódnának.

Az Európai Szoftver-szabadalmi törvény – 2005-ben fellángolt vitájának fejleményeit – is figyelembe kell venni az informatikai rendszerben alkalmazott és elérhető szoftverek tekintetében. Az állásfoglalás nem tartozik a tanulmány kereteibe, de mindenki eldöntheti, hogy akadémiai vagy non-profit intézményként melyik jogi háttér jelent számára előnyt.

3.4. Forensics / Törvényszéki bizonyítás

Amikor a nemkívánatos esemény bekövetkezik, rendelkeznünk kell a megfelelő eszközökkel és módszerekkel ahhoz, hogy rögzítsük az esemény minden nyomát a későbbi eljárás sikerességéért. Ezt a területet részletezzük a következőkben, hiszen előfordulhat, hogy ilyen eljárásba bevonnak, vagy ilyen eljárás alá vonnak egyéneket vagy intézményeket, ezért nem árt tudni, hogy mit követel meg a szakértőtől az eljárás.

A kapcsolódó törvényekre és eljárási rendekre részletesen nem térünk ki, a jogszabályokról a http://www.cert.hu/szabaly/ címen lehet tájékozódni.

Fontos még az is, hogy az adott országban mit fogadnak el az eljárások során. Magyarországon az igazságügyi szakértőkre nagy felelősség hárul, hiszen a bírói karnak kisebb az informatikai szakértelme, így nagyban hagyatkoznak a szakértők véleményére, akik között szintén fennáll a tudás és tapasztalatbeli különbség. Az informatika térhódításával és az informatikai eszközöket alkalmazó bűnözés növekedésével egyre nagyobb szükség lesz a megfelelő szintű szakértői munkákra és az ezt végző szakemberekre.

Az alkalmazható eszközök és módszerek, valamint egy példaeset részletes tárgyalásra kerül a 7.8 fejezetben.

4. Kockázatelemzés

A kockázatelemzéssel sokat foglalkoztunk a [Tan_1] dokumentumban, így ebben a kötetben csak ismétlés jelleggel tekintjük át a fontosabb szabályokat.

4.1. Előkészület[4]

Az átlagember nincs tisztában azzal, hogy mi az, ami biztonságos, és mi az, ami csak annak látszik. A látszat pedig sokszor csal, és a hamis biztonságérzet sokkal rosszabb lehet, mint a tudatos biztonság hiánya. A biztonságérzet eredménye a figyelem (és sokszor a fegyelem) lankadása, ami a sikeres támadás egyik pillére lehet.

Amikor arról kell döntenünk, hogy mire is van szükségünk a megfelelő biztonság eléréséhez, akkor menjünk végig a következő öt lépésen (az „5M kérdés"), hogy kiderüljön, egyáltalán érdemes-e elemezni a megoldás kockázatait.

Első lépés: Milyen problémát old meg a biztonsági intézkedés?

Azt hinnénk, hogy ez egyszerűen megválaszolható, de sokszor találkozhatunk olyan biztonsági megoldással, mely nem rendelkezik egyértelműen megnevezett céllal. Amennyiben megfogalmazható a cél, úgy érdemes írásban is rögzíteni, így mások (partnerek, társintézmények, első körben nem érintett felhasználók stb.) vagy az utódok (új rendszergazda, új biztonsági felelős, új vezető stb.) is érteni fogják egy adott biztonsági megoldás célját.

Amennyiben létezik az egyes biztonsági célokat összefoglaló írott és közzétett biztonsági politika, úgy még hatékonyabb a rendszer, hiszen az egyes megoldásokon túl azok összefüggései is felismerhetők.

Második lépés: Mennyire megfelelő a biztonsági intézkedés a probléma megoldására?

Sok esetben az elemzések a probléma megnevezése után az elméleti megoldásokra térnek anélkül, hogy megvizsgálnák a megoldás hatékonyságát. A probléma és az előzőben említett cél megnevezése után azt kell vizsgálni, hogy a felmerülő megoldások mennyire hatékonyak. Amennyiben sokféle megoldás létezik, úgy elágazásokat építhetünk fel a következő lépés előtt.

Harmadik lépés: Milyen újabb biztonsági problémákat vet fel az adott megoldás?

A biztonság egy összetett és összefüggő rendszer, melyben egy elem megváltoztatása hullámokat gerjeszt. A hullámok lehetnek jók, de új problémák is felmerülhetnek az inkompatibilitástól a heterogenitást erősen követelő gondokon át a skálázhatóság elvesztéséig sok területen.

Negyedik lépés: Mibe kerül a biztonsági intézkedés?

Természetesen nem kell végigmenni az összes lépésen, ha előbb is kiderült, hogy az adott ág vagy választás nem jó (pl. az új felvetődő problémák semmiképpen sem vállalhatók).

Nemcsak anyagi költségekre kell gondolni, hanem minden egyéb közvetett költségekre is, mint a bevezetés, képzés, támogató eszközök ára, licencek, szükséges többlet munkaerő stb. költségeire is, legalábbis az arányosan minderre eső többletköltségeket kell összeszámolni.

[4] Ennek az írásnak az alapja Bruce Schneier „How to Think About Security" című cikke.

Ötödik lépés: A 2-4 lépések megválaszolása után megéri-e a költségeket a biztonsági intézkedés?

Ez a legegyszerűbb lépés, de a leggyakrabban ez keveseket zavar. Nem elégséges, ha egy biztonsági rendszer hatékony. A társadalomban nincs korlátlan anyagi erőforrás, és nincs korlátlan türelem. A társadalomnak azt kell tennie, aminek a legtöbb értelme van, ami a leghatékonyabb a megfizetett pénzért.

Összefoglalva, az öt lépéses eljárás minden biztonsági megoldásra alkalmazható a múltban, jelenben és a jövőben is:

1. Milyen problémát old meg?
2. Mennyire jól oldja meg?
3. Milyen új problémákat vet fel?
4. Melyek a gazdasági és egyéb költségei?
5. A fentiek tükrében megéri-e?

Sokszor használat közben derül ki egyes biztonsági megoldásokról, hogy mennyire nem hatékonyak. Például 2001. szeptember 11-e óta két légi közlekedésben alkalmazott biztonsági megoldás volt hatékony: a pilótakabin ajtajának megerősítése és az utasok meggyőzése, hogy vegyék fel a harcot a fedélzeten a terroristákkal. Minden más megoldás a kismértékű biztonsági megerősítés és a placebo közé esik.

A biztonság nem egy termék, hanem egy eljárás, ezért fontos, hogy az egyes termékekre ne úgy tekintsünk, mint minden problémánkat megoldó eszközökre, hanem egy probléma megoldására kitűzött cél érdekében, meghatározott eljárás alapján végrehajtott feladatok közben megfelelően alkalmazott segítőkre.

4.2. Kockázatelemzés módszere

Amennyiben nagyobb léptékben kell gondolkodnunk, és nem egy adott problémára kell megoldást találnunk, hanem fel kell mérni a lehetséges kockázati tényezőket, akkor kockázatelemzést kell végezni. Az 1. kötetben [Tan_1] már említett elméleti alapok (2.1 fejezet) és gyakorlati segítség (6.1.1 rész) alapján a kockázatelemzés egyénileg is elvégezhető. Külső szakemberekkel mindez már komoly összegekbe kerülhet, de elérhető hazai piacon is.

Ennél részletesebben itt nem térünk ki a módszerre, de kiemeljük, hogy valamilyen formában fel kell mérni az egyes elemek által jelentett kockázatot, és ez alapján kell kialakítani a védelmi intézkedéseket.

5. Topológia (strukturális hálózati biztonság)

Egy hálózat biztonsága érdekében mindenek előtt fontos a hálózati eszközök megfelelő elhelyezése és összekötése, azaz a rendszer topológiája. A topológia kialakítása többféle szempont szerint is lehetséges, de mindenek előtt a fizikai védelmet kell kialakítani.

A mintarendszer alapjául az MTA SZTAKI, mintegy 500 gépet számláló, publikus internetes hálózata szolgált. Ezt modellként való megfelelése szempontjából az alábbi szempontok szerint vizsgáltuk.

5.1. Funkcionalitás

Mivel a kutatóintézet fő profiljában a számítástechnika jelentős szerepet foglal el, hálózata is meglehetősen sokszínű, találhatók rajta kiöregedett SPARC mainframe-ek, Novell hálózati kiszolgálók, Apple gyártmányú gépek, a Windows talán összes verziója, Unixok, ősi, BNC hálózatra csatlakozó nyomtatószerverek, de nem ritka a WLAN elérést PDA-val igénybe vevő munkatárs sem.

Ez a sokszínűség alkalmassá teszi e hálózatot arra, hogy szolgáltatási szempontból modellként szolgáljon az átlagos magyar intézményi hálózatok számára: véleményünk szerint nem várható, hogy egy önkormányzat, iskola, vagy más kutatóintézet ennél bonyolultabb igények kiszolgálására kényszerüljön. A hálózati kliensek nagy része Windows rendszerű PC, ami szintén megfelel az általános magyarországi körülményeknek. Ezek alapján kutatási célterületként funkcionális szempontból a vizsgált hálózatokat a SZTAKI hálózati szolgáltatásainak részhalmazaként definiáljuk.

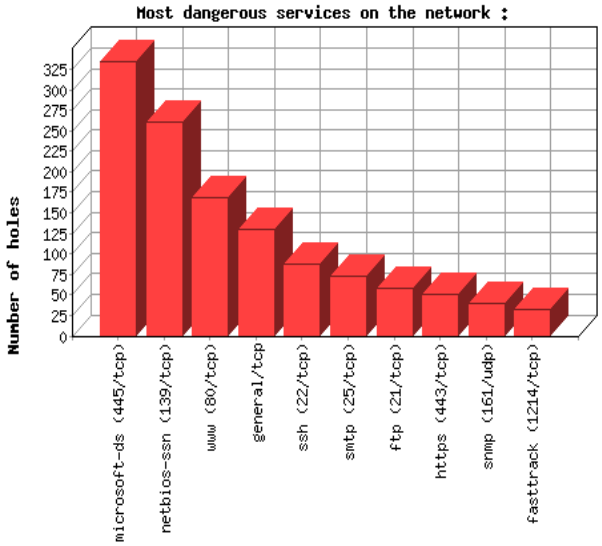
5.2. Hálózat-menedzsment

Az intézeti hálózat funkcionális sokszínűségét jól kiegészíti az adminisztratív megosztottság. A két épületre kiterjedő hálózat két, üvegszálas összeköttetésben álló Cisco Catalyst 4506 típusú, összesen több, mint 500 portos router-switchének, valamint az épületek szobáiban elhelyezkedő falicsatlakozóknak az üzemeltetését egy külső cég végzi, a hálózatra csatlakozó számítógépek telepítését, karbantartását mindenki egyénileg, vagy osztályszinten oldja meg, a hálózat IP szintű kezeléséért és védelméért a Hálózatbiztonsági Osztály felelős, az alkalmazásszintű szolgáltatásokat, beleértve a DNS szolgáltatást is, viszont ismét egy másik osztály végzi.

Az épületekben jócskán találhatók folyamatosan üzemelő szerver és kliens szerepű számítógépek, de előfordulnak csak ritkán bekapcsolt prezentációs célú munkaállomások is. Az intézet nem rendelkezik egységesített és univerzálisan használt adminisztratív célú hálózati szolgáltatásokkal (LDAP, Windows Domain Services stb.).

5.3. Hálózatbiztonság

Az előző két pont ismertetése alapján az intézet publikus hálózati felületének biztonsága nem tekinthető kielégítőnek. A [nessus] hálózati penetrációs mérőeszközzel elvégzett felmérő vizsgálatok egyértelműen rámutattak a SZTAKI Internet hálózatára csatolt rendszerek biztonsági hiányosságainak súlyosságára. A súlyos hálózati sebezhetőségek alapján a szolgáltatásokat „veszélyességi” sorrendbe állító, a 5.1. ábrán látható, grafikonról több következtetés is levonható:



5.1. Ábra: a hálózati szolgáltatások kockázatai (SZTAKI)

- Az előzőleg említett Windows-os PC-k nagy számával összefüggésben, az operációs rendszer hibáinak foltozására szolgáló frissítések rendszeres letöltésének hiánya látható, melyet a NETBIOS-alapú szolgáltatások sebezhetőségeinek első helyezései igazolnak.
- A második fontos következtetés az általános internetes szolgáltatások (WWW, SMTP, HTTPS) indokolatlanul nagy száma, mely abból a tényből következik, hogy viszonylag kevesen veszik igénybe az intézet által központilag nyújtott, megbízható, jó minőségű és megfelelően felügyelt levelező- és webtárhely-szolgáltatásokat, inkább – látható kockázati forrásként szolgáló – saját megoldásokat telepítenek.
- Harmadsorban kiemelhető a tipikusan UNIX rendszerekre jellemző SSH szolgáltatás hibáinak magas száma, melyből arra következtethetünk, hogy a hálózatra csatlakozó szolgáltatók, Linuxos kliensek frissítése szintén kívánnivalókat hagy maga után.
- Az utolsó pozíciót elfoglaló, de még mindig elég nagy számú sebezhetőséget jelző – általában illegális fájlcserére használatos – fasttrack alkalmazási protokoll jelenlétéből további, részben biztonsági, részben jogi problémák jelenlétére következtethetünk.
- A publikus használatra napjainkban kétségek közt ajánlható SNMP és FTP protokollok sebezhetőségeinek nagy száma is megfontolandó, de ezen szolgáltatások jelenléte önmagában – a felderített konkrét sebezhetőségektől eltekintve – még nem egyértelmű biztonsági kockázat.

A fentiek által érzékelhető hálózati biztonsági problémák alapján megállapítható, hogy a kiválasztott hálózat a funkcionális és hálózat-menedzsment szempontokon felül a hálózatbiztonsági aspektusból is megfelelő vizsgálati alap.

5.4. Hálózati struktúra gyengeségeinek kihasználása

A 5.2. ábrán látható hálózat egy Magyarországon (de a világon is) általánosan jellemző, funkcionális alapokon felépített teljes elérésű hálózat, mely minden igényt kiszolgál a biztonsági szempontokon kívül.

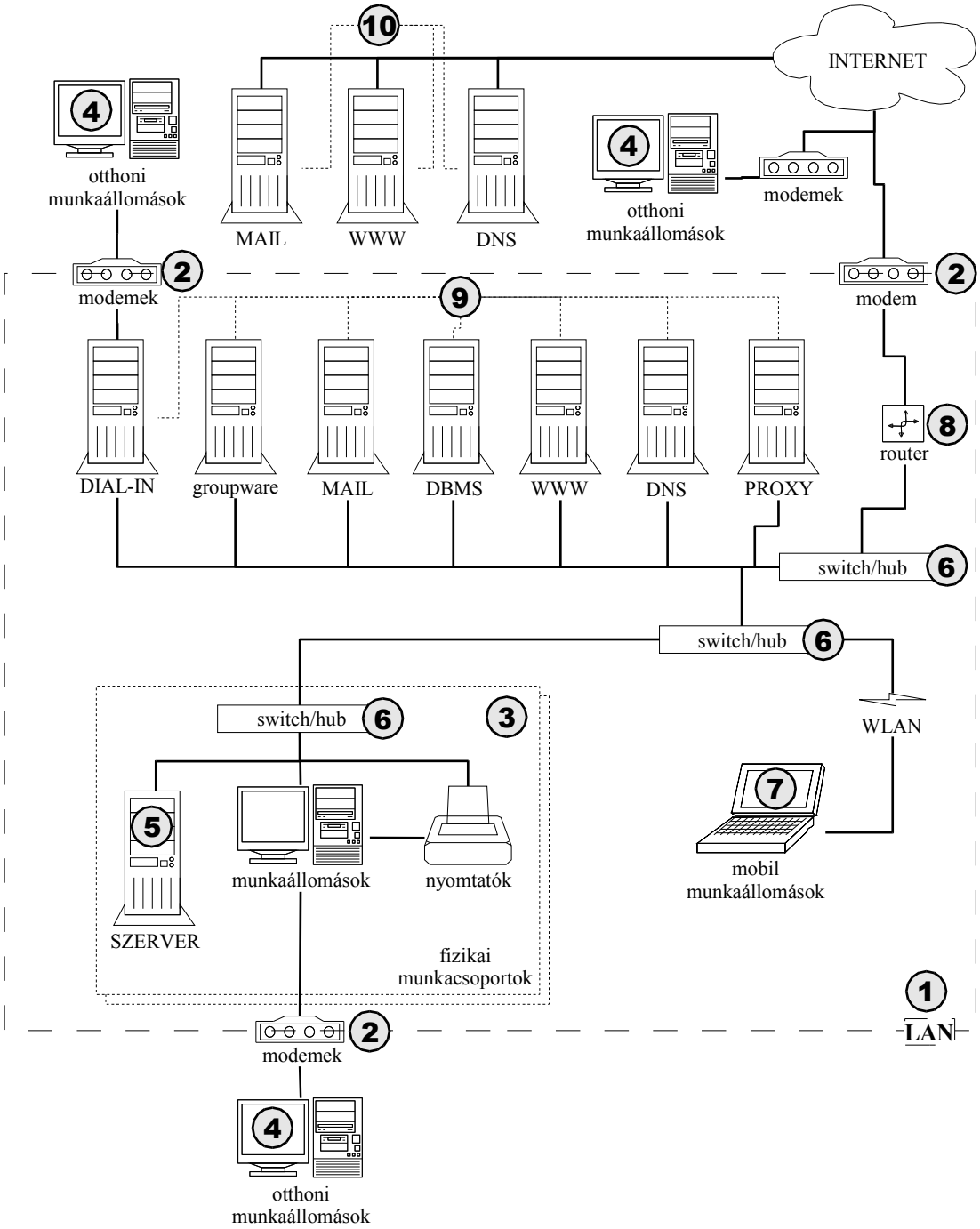
5.4.1. Fizikai struktúra

A strukturálásnál elsődleges szempontként a fizikai adottságok és az elsődlegesen megoldandó feladatok játszanak szerepet. A hálózat az elsődlegesen fizikai elhelyezkedési alapokon kialakított munkacsoportokban (3), vagy ezeken praktikus okok miatt kívül eső (4, 7) munkaállomásokból, az ezek szoftveres működtetéséhez szükséges kiszolgálókból (9, 10), valamint a hálózati elérést biztosító aktív és passzív elemekből (2, 6, 8) áll.

A komponensek fizikai elhelyezkedését tekintve különböző megoldásokkal találkozhatunk:

- Legrosszabb esetben a kiszolgálók és a munkaállomások vegyesen helyezkednek el, ezt az esetet szemlélteti a példán az „5" jelölésű kiszolgáló megjelenése a fizikai munkacsoportban. Ebben az esetben a hálózati struktúra kis hálózatok esetén egyszerűen egy Ethernet hub, vagy switch segítségével központosított, de sok esetben egy ilyen központosított megoldásból kapacitáshiány, vagy helyszűke miatt új aktív elemek aktuális igények szerinti bevezetése során kialakított kusza megoldásokkal is találkozhatunk. Talán magyarázatot sem igényel, hogy egy ilyen szerkezet nem csak a biztonság javítását, de a hibakeresést is jelentősen nehezíti, ezáltal az üzembiztonságot is csökkentheti. Az rendszertelen szerkezetben épített hálózatok kezelhetetlenségük miatt általában nem nőnek túl a 20-30 kiépített végpontos méreten.
- Általánosan elterjedt megoldás a kiszolgálók és a munkaállomások különválasztása, a kiszolgálók központosítása elhelyezkedés szempontjából. Általában ez egy, vagy – adottságoktól függően – több dedikált, jó esetben zárható, légkondicionált „szerver szoba" megjelenésével jár. Ez a helyi különválasztás szükségszerűen magával vonja legalább a kiszolgálógépek hálózati rendeződését is, melynek eredményeképp a kritikus szolgáltatások biztonsági védelmére és üzembiztonságának növelésére jóval több lehetőség nyílik, így ez a struktúra jelentősen jobban skálázható, a gyakorlatban 100-150 végpontos rendszerek is előfordulnak.
- A kiszolgálók és munkaállomások szétválasztásával a felhasználói hálózat és az aktív hálózati elemek rendezése nem feltétlenül következik be, mivel általában ez több munkát és megfelelő, előrelátó tervezést és részben ezekből eredően a kezdetekben magasabb anyagi befektetést igényel, előnyeit ezen tényezők tükrében pedig nem minden esetben könnyű belátni. A fizikai hálózati eszközök központosított elhelyezésének nem csak karbantartási és hibakeresési szempontból van jelentősége, hanem a központosított elhelyezésből származó fizikai védelem és a fizikai rendezéssel együtt célszerűen megjelenő logikai rendeződés biztonsági szempontból is előnyhöz juttatja a rendszert.

A hálózat fizikai központosításának igénye általában annyi végpontig elégíthető ki teljes mértékben, amennyit a kábelezés költsége és a felmerülő megbízhatósági igények lehetővé tesznek. Praktikus határokon felül a központot helyileg több részre osztják és az így felosztott központokat a forgalomtól függő gráfszerkezetben (fa, teljes gráf) rendezve kötik össze nagy teljesítményű vonalakkal.



5.2. Ábra: általános, funkcionálisan felépített, nyilvános, IP alapú hálózat

5.4.2. Logikai struktúra

A komponensek logikailag csak a funkcionális szempontoknak felelnek meg. Az „1” jelölésű LAN egy Ethernet szegmenst alkot, az összes szereplő valós publikus IP címmel rendelkezik, vagyis minden kiszolgáló és munkaállomás minden kiszolgálót és munkaállomást teljes felületen elérhet. A hálózat a „2” jelölésű elérési pontokon biztosít elérést a külvilág (WAN) számára. A „3” jelölésű helyi (LAN kapcsolat) munkaállomások, a „4” jelölésű otthoni (WAN és pont-pont elérés) munkaállomások és a „7” jelölésű mobil (WLAN) számítógépek egyenlő jogú felhasználói a hálózati szolgáltatásoknak.

Az „5” jelölésű helyi használatú szerverek szintén a hálózat egész területéről elérhetőek, bár csak lokálisan, az adott munkacsoport kiszolgálását végzik. A „8” jelölésű router nem végez címfordítást és tűzfalszerű funkciókkal sem rendelkezik, hisz ezek egyike sem szükséges a hálózat üzemeltetéséhez. A hálózati kiszolgáló funkciókat a „9” jelölésű, a munkaállomásokkal, munkacsoportokkal fizikailag azonos hálózatra kötött gépek végzik, esetlegesen együttműködve a „10” jelölésű, az Internet más részén elhelyezkedő kiszolgálókkal.

5.4.3. Támadási lehetőségek

A példában szereplő komponensek nagy része elhagyható, vagy más komponensekkel fizikailag összevonható funkciókat jelenít meg, vagyis konkrét esetekben ritkán találkozhatunk az ábrán vázolt teljes felépítéssel.

5.4.3.1. Megszemélyesítéses támadások

A vázolt hálózatban lehetőség nyílik szinte az összes megszemélyesítési technika megvalósítására, valamint ezzel összefüggésben a LAN számítógépei közti esetleges bizalmi viszony egyszerű kihasználására.

Ha a „6” jelölésű eszközök hubok, a hálózat teljes forgalmát minden rácsatlakozó munkaállomás megfigyelheti, illetve bármelyik csatlakozó munkaállomás bármelyik másik állomás, vagy – a „8”-as jelölésű router gyenge beállításai esetén – bármely internetes cím nevében adhat fel üzeneteket. Ha a „6” jelölésű eszközök switch üzemmódban működnek, szintén lehetőség van a fent említett műveletek elvégzésére, ám ehhez a switchek ARP tábláját is el kell árasztani.

5.4.3.2. Wlan hozzáférés-védelem

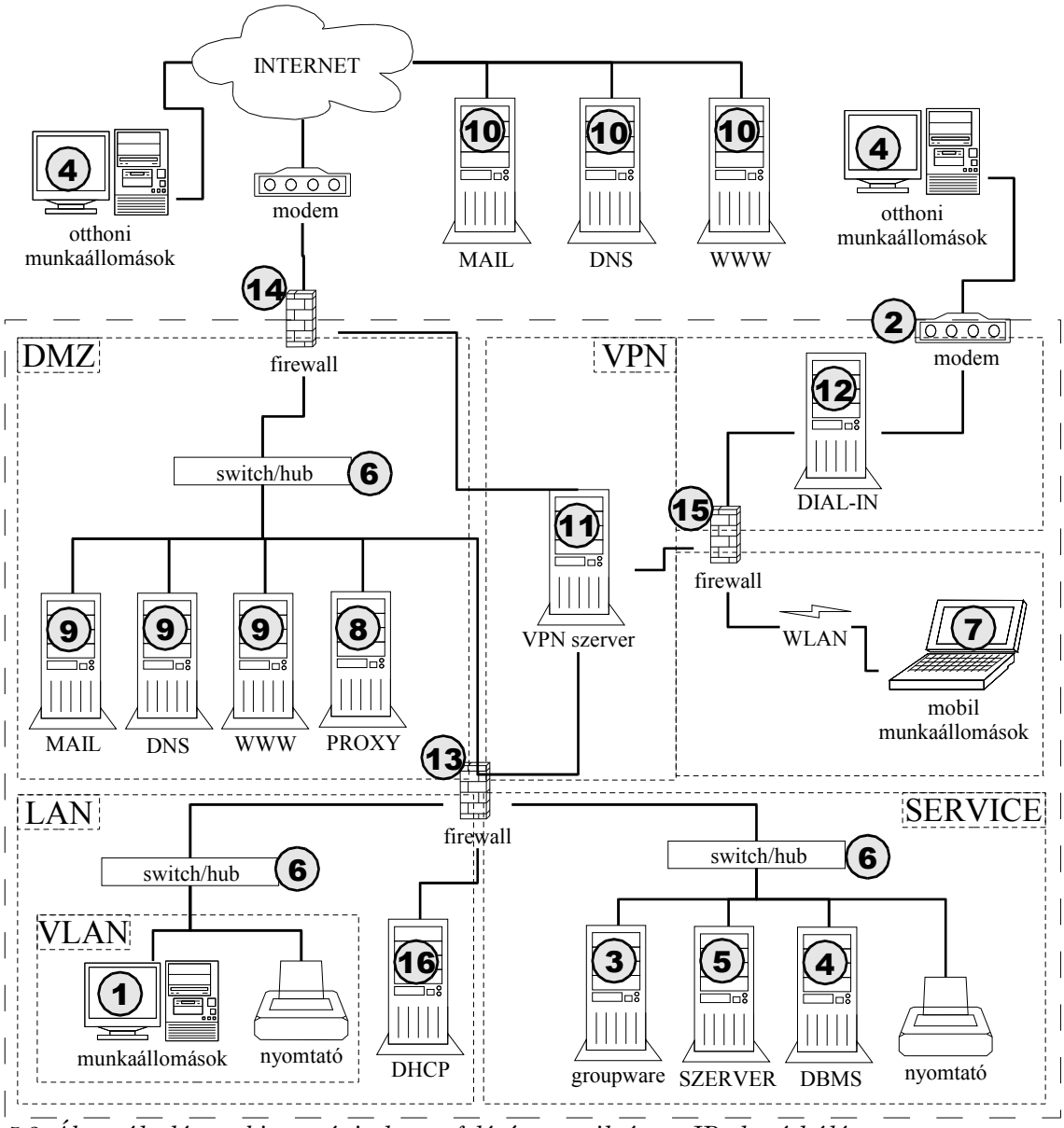
A vezeték nélküli hálózat védelmére jelen dokumentum írásakor nem létezik megfelelő szoftveres megoldás, a Wlan hálózatokban használatos WEP titkosítás feltörése a jelenleg rendelkezésre álló eszközök segítségével akár percek alatt megtörténhet, így a Wlan hálózat és az „1” jelölésű LAN fizikai, vagy legalábbis IP alhálózat szintű elválasztásának hiánya virtuálisan elkerülhetetlenné teszi a teljes hálózat lehallgathatóságát és kompromittálódását.

5.4.3.3. VPN-ek, behívások gyengeségei

Az otthoni munkaállomások lokális munkaállomásként való csatlakozása a hálózatra betárcsázás, vagy IP csatornán kialakított VPN segítségével számos problémát vethet fel. Az otthoni munkaállomások rendszeresen gyengébb biztonsági beállításokkal üzemelnek, mint a munkahelyi gépek, illetve egyéb személyek által való hozzáférhetőségük is egyenes következményeként bővíti az „1” jelölésű LAN-t. Emiatt a vezeték nélküli hálózathoz hasonlóan súlyos hiányosság ezen gépek fizikai, illetve IP alhálózati leválasztásának hiánya.

5.5. Hálózati architektúra megváltoztatása

A 5.2. ábrán látható hálózat 5.4. fejezetben említett biztonsági problémáinak megoldására a következő átalakítás javasolható:



5.3. Ábra: általános, biztonsági alapon felépített, nyilvános, IP alapú hálózat

5.5.1. Logikai struktúra

A mintahálózat nyilvános, az Internet-felhőre kapcsolódó, komponensei változatlanok, a „4“ jelölésű otthoni munkaállomások közvetlen munkacsoportba történő betárcsázása letiltásra került a közvetlen LAN hozzáférés lehetősége miatt.

A funkcionális működést támogatva, de a biztonsági szempontokat szem előtt tartva az eredeti LAN-ból négy fő csoportot alakíthatunk ki:

5.5.1.1. DMZ

A demilitarizált zóna, röviden DMZ azokat a gépeket („9“) tartalmazza, melyek szolgáltatásait egyaránt igénybe veszik az Interneten és a helyi hálózaton elhelyezkedő gépek is. A DMZ gépei legfőképp abban különböznek a hálózat többi zónájától, hogy az Internet irányába közvetlen kiszolgáló tevékenységet végeznek.

A DMZ védelmét a „14“ jelölésű tűzfal látja el, mely, az ábra egyszerűsítése végett – ahogy az ábrán látható többi tűzfal is – router szerepet is vállal. A DMZ hálózat klasszikusan publikus, statikus IP című gépeket tartalmaz, de a „14“ jelölésű tűzfal port-átirányító szolgáltatásai esetén akár privát címek hozzárendelése is lehetséges. Ez a tűzfal a zóna funkcióinak megfelelően mindkét irányban végez forgalomtovábbítást.

5.5.1.2. LAN

A LAN az eredeti hálózat fizikai munkacsoportjait – vagyis a hálózat java részét kitevő kliens szerepű munkaállomásokat („1“) -- tartalmazza. Ellentétben az eredeti fizikai elhelyezkedésen alapuló elkülönítéstől, az új megoldás a hálózatot logikailag, a munkacsoportok alapján csoportosítva layer 2 szinten szeparált virtuális LAN-okra bontja.

A LAN védelmét a „13“ jelölésű központi tűzfal látja el, a LAN gépeinek korlátozott elérést biztosítva a többi főbb csoportban található számítógépek szolgáltatásaihoz. A LAN és virtuális felbontásai privát IP címtartományokat foglalnak el, mivel ezek a számítógépek csak kliens szerepű felhasználói a rendszernek. A tűzfal csak a LAN gépei által kezdeményezett forgalom továbbítását végzi a LAN irányába, ez megoldható a hálózati címfordítást alkalmazó NAT technológia alkalmazásával is.

A LAN esetében dinamikus címkiosztás (DHCP) is célszerűvé válhat. Ilyen esetben, mivel a DHCP protokoll alapvetően nem route-olható, problémát jelenthet a kiszolgáló („16“) VLAN-okhoz viszonyított elhelyezése. A problémára több megoldás adható:

- Minden VLAN-ba külön DHCP kiszolgálót kell telepíteni. Ennek csak akkor van értelme, ha az aktív hálózati eszközök („6“) támogatják a címkiosztást, de ez gyakran így van.
- Egy DHCP kiszolgálót kell telepíteni, de a kiszolgálónak támogatnia kell a VLAN protokollt, így megoldható, hogy fizikailag egy gép több VLAN-t is ellásson dinamikus címekkel.
- Az előző megoldáshoz hasonlóan egy kiszolgálót kell telepíteni, azonban a VLAN-ok kiszolgálása megoldható az aktív hálózati eszköz („6“) DHCP-proxy szolgáltatás támogatásának segítségével.

5.5.1.3. SERVICE

A SERVICE zóna a DMZ privát megfelelője. Itt találhatók azok a nem kliens szerepű gépek, melyek szolgáltatásokat nyújtanak a DMZ és a LAN számítógépei számára. Ilyen szerep juthat egy intranetes groupware-nek („3“), egy általános – a 5.2., funkcionális ábrán egy, vagy több logikai munkacsoport kiszolgálását végző – általános szerepű szervernek („5“), és általában minden adatbázis-kiszolgálónak („4“).

A SERVICE zóna védelmét a „13“ jelölésű tűzfal látja el, a DMZ és LAN zónák számítógépeinek korlátozott elérést biztosítva a zóna szolgáltatásaihoz. A SERVICE zóna számítógépei a belső elérhetőség biztosítása érdekében célszerűen statikus privát IP címmel rendelkeznek, az Internetet a LAN gépeihez hasonlóan csak kliens szerepben érhetik el.

5.5.1.4. VPN

Külön zónaként szerepel a LAN-on fizikailag kívül, de logikailag belül elhelyezkedő, a LAN-hoz hasonló szolgáltatásokat igénybe venni kívánó, úgynevezett „szatellit“ munkaállomásokat kiszolgáló VPN zóna. A VPN zóna az általánosan elterjedt három külső csatlakozási pont felől a következőképpen érhető el az otthoni („4“) és mobil („7“) szatellit munkaállomások számára:

- Az internet kapcsolattal rendelkező szatellitek, a hálózat egyéb internetes forgalmával együtt, a „14“ jelölésű tűzfalon keresztül juthatnak el a „11“ jelölésű VPN szerverhez.
- A telefonos behívók a „2“ jelzésű modemen keresztül a „12“ jelölésű, elkülönített terminálszerverrel veszik fel a kapcsolatot, ahonnan a „15“ jelölésű tűzfalon keresztül juthatnak el a VPN szerverhez.
- A „7“ jelölésű mobil munkaállomások egy WLAN elérési ponton keresztül szintén a „15“ jelzésű tűzfalhoz kerülnek, melyen keresztül a VPN szerverhez csatlakozhatnak.

Mint látható, a szatellit gépek egymástól és a hálózat többi részétől layer 2 szinten elkülönítve, tűzfalas szűrés után juthatnak el az azonosítást és hitelesítést végrehajtó, valamint a biztonságos csatornát közvetlenül kiépítő központi VPN szerverhez. Az így autentikált forgalom a „13“ jelzésű tűzfal szigorú szűrőszabályain keresztül logikailag a LAN egy VLAN-jából származó forgalomként kezelhető a továbbiakban, de a szolgáltatásokat az útba eső tűzfalak segítségével célszerű minimálisra korlátozni.

A szatellit gépek biztonságos kezelésére – mivel ezek a gépek a védeni kívánt hálózat számára ellenőrizhetetlen adatforgalmat is folytathatnak – nincsen teljes értékű megoldás, csak abban az esetben ha teljesen a hálózatkezelő felügyelete alá vonhatók, ám erre a gyakorlatban kevés lehetőség adódik.

5.5.2. Fizikai struktúra

Az ábrán látható struktúra fizikai szempontból összevonásokra ad lehetőséget, a „6“, „13“, „14“, „15“ jelölésű aktív hálózati eszközök akár egyetlen VLAN kompatibilis routing switchben is helyet kaphatnak, fontos azonban, hogy az apró szaggatott vonallal elkülönített hálózatok (DMZ, LAN és részei, SERVICE, VPN és részei) fizikailag, vagy virtuálisan is különböző (Ethernet) szegmensekre essenek. Az ábrán vázolt logikai struktúra a 5.4. fejezetben említett fizikai központosítást is könnyebbé téve az eszközök behatolás elleni védelmét is biztosítja.

5.6. ToReS hálózati infrastruktúra alkalmazási esetei

A felhasználási módok a legtöbb esetben a hálózati struktúra megváltoztatását is igénylik. A jelen fejezetben kiemelt változtatásokon kívül még számos más megoldás is található, de itt csak a feltételezett tipikus alkalmazási esetekre szorítkoztunk.

5.6.1. Publikus Internet hálózat

A már kiépített hálózat teljes átszervezésére az esetek túlnyomó többségében egy lépésben nincs lehetőség. Példánkban már az is jelentős előrelépést jelenthet, ha a hálózat kerületét, azaz a „2” jelölésű belépési pontokat, de minimálisan a „8” jelölésű routert a ToReS rendszer által biztosított csomagszűrő tűzfallal védhetjük, melynek segítségével legalább a triviálisan káros forgalom megakadályozható.

Szintén egyszerűbb, alacsony befektetésű, változtatás lehet a „9” jelölésű kiszolgálók közül minél többet a ToReS rendszer biztonságilag felkészített funkcióival kiváltani. Ilyen szolgál-

tatás lehet a DHCP, a levelező (MAIL), a web (WWW), a DNS és a PROXY szolgáltatás, melyek konfigurációja alapesetben egyszerű, a ToReS CD pedig mindegyikre magas színvonalú, biztonsági szempontokat figyelembe vevő, lokális tűzfallal védett ingyenes megoldást szolgáltat. Hasonlóképp lecserélhetők a „10" jelölésű publikus szerverek, azzal a kiegészítéssel, hogy a „9" és „10" rendszerek egyidejű cseréje esetén lehetőség nyílhat a ToReS által nyújtott VPN szolgáltatások igénybevételével legalább a szerverek közti kommunikáció titkosítására.

A kritikusabb szolgáltatásokat nyújtó, bonyolult szoftveres konfigurációjú szerverek kis anyagi ráfordítással szintén védettebbé tehetők, ugyanis, akár régebbi, leselejtezett PC-k felhasználásával ToReS tűzfalas előtéteket kaphatnak. Az ilyen, olcsó, nem megbízható hardveres megoldások esetén azonban fel kell készülni a megnövekedett meghibásodási valószínűségre is.

5.6.2. Kliens hálózat (NAT)

Főként egyszerűbb, de a magyarországi Internet elterjedésének kezdeti korszakában kiépített hálózatokra jellemző lehet a csak kliens jellegű felhasználási mód mellett a publikus internetes elérhetőség. Ilyen esetben gyakorlatilag a kliensek számára semmi szükség a publikus IP címek megtartására, vagyis az egész hálózat, egy ToReS IP címtranszformációs tűzfal mögé elhelyezve, az 5.3. ábrán „NAT" jelöléssel ellátott alhálózatnak megfelelő pozícióba tolható. Ezzel a megoldással részben kiválthatóak a kliensekre telepített alkalmazástűzfalak, hiszen a hálózat kívülről indított kapcsolatfelvételek ellen védetté válik, a NAT hálózat privát IP címei nem érhetők el az Internet irányából.

5.6.3. Biztonsági szemszögből felépített hálózat

Az 5.3. ábrán látható, biztonsági szempontokat figyelembe vevő módon kiépített hálózat számos pontján használhatók fel a ToReS rendszer által nyújtott megoldások. Természetesen az alább felsorolt megoldások nem mindegyike tekinthető kötelezőnek, hisz a ToReS által nyújtott funkciók bármelyikére található más – számos esetben költséges – megoldás is. A felsorolás inkább áttekintésként szolgál a rendszer teljes felhasználási körét illetőleg:

- A kliens számítógépek („1", „7") esetében a ToReS rendszer nehezebben alkalmazható, de megfelelő módosításokkal, kiegészítésekkel a live CD jó alapot szolgáltathat egy képzettebb rendszergazda számára egy asztali munkaállomás kialakításához is. A rendszer által alapszolgáltatásként nyújtott, biztonságos bejelentkezést, VPN kapcsolatot, TLS/SSL titkosított kommunikációs csatornákat elérhetővé tevő kliens alkalmazások jól működnek együtt a ToReS által megvalósított kiszolgálókkal, tűzfalakkal, de más termékek hasonló szolgáltatásaival is.
- A kliens számítógépekhez hasonlóan a Linux operációs rendszert támogató groupware („3"), adatbáziskezelő („4") és általános kiszolgáló („5") megoldások számára is megfelelő alapot biztosíthat a ToReS által nyújtott RBAC szolgáltatással kialakítható védett futtatási környezet.
- A ToReS rendszer által nyújtott, biztonságos PROXY („8") szolgáltatások jól felhasználhatók egyéb tűzfalrendszerek kiegészítőiként, és biztonsági beállíthatóságuk mellett a korlátozott sebességű Internet eléréssel rendelkező hálózatok esetében minőségi javulást is eredményezhetnek az Internet felhasználásában.
- A „9" jelzésű DMZ kiszolgálók és a „10" jelölésű publikus kiszolgálók cseréjekor, az előző fejezetben említett megjegyzések mellett, még ki kell kötni azt is, hogy egy biztonsági szempontokat figyelembe vevő rendszer telepítése és használata esetén a konfigurációt és

felügyeletet végző rendszergazdán is sok múlik, vagyis a beállítások hibás, vagy túl lazára történő megváltoztatása esetén a ToReS rendszer által nyújtott szolgáltatások a lecserélt rendszernél kevésbé biztonságos helyzetet is eredményezhetnek.

- Mivel egy általános hálózatban (5.2. ábra) nem feltétlenül található a 5.3. ábrán található, „11" jelzésű VPN szerver, a ToReS rendszer kézenfekvő felhasználási lehetőségei közé tartozik ennek a rendszernek a megvalósítása. A ToReS által támogatott IPSec, MPPE és OpenVPN protokollok a VPN felhasználási lehetőségek nagy részét lefedik.
- A DIAL-IN szolgáltatás („12") az esetek többségében már a ToReS rendszer bevezetése előtt is megtalálható, hiszen egy funkcionális igényforrásból származó, ráadásul kivesző-félben levő, megoldásról van szó. Ennek ellenére a ToReS rendszer megfelelő szoftveres támogatást nyújt egy behívóközpont szolgáltatásainak ellátására. A speciális hardveres eszközök, modem-poolok vezérléséhez a rendszermag újrakonfigurálása, kiegészítése válhat szükségessé.
- A biztonsági szempontokat is figyelembe vevő rendszerek által alkalmazott tűzfal-megoldások („13", „14") esetében a ToReS rendszer magas szintű szolgáltatásokat, és kényelmes konfigurációs felületet nyújt. Képességeit tekintve a rendszer megelőzi a kereskedelmi forgalomban kapható kompakt tűzfalmegoldásokat, azzal a megjegyzéssel, hogy a ToReS rendszer nem csak tűzfalként, hanem egyben routerként és forgalomanalizátorként is üzemeltethető, ami további költségmegtakarítás mellett az egyszerű, áttekinthető karbantarthatóságot is lehetővé teszi.

6. Szolgáltatások biztonsága

A felhasználók azok, akik a szolgáltatásokat igénybe veszik, működtetik vagy éppen veszélyeztetik, tehát az emberi tényezők a legfontosabbak a biztonság kialakításában, és csak ennek figyelembevételével alakítható a szolgáltatások biztonsága.

6.1. Szolgáltatás szerinti felosztás

Az itt felsoroltakon kívül elképzelhetők még olyan szolgáltatások, melyeket kisebb számú felhasználó igényel vagy használ, de ebben a részben csak a főbb területeket foglaltuk össze.

6.1.1. Web / Böngészés

Kockázati tényezők, általános sebezhetőségek

A webszerverek és a kliensek összetett, bonyolult programok sok szolgáltatással, amelyek érzékeny területet érintenek. A kockázatokat három csoportba sorolhatjuk:

Hibák a szerverben vagy konfigurációjában

Ilyen hiba esetén egy távoli felhasználó számára a következőkre nyílhat lehetőség:

- Bizalmas információkat szerezhet meg, amelyeknek nem szabadna nyilvánossá válniuk;
- Programokat futtathat a szerveren, amelyek akár az operációs rendszert vagy valamelyik beállítását is módosíthatják;
- Információkat nyerhet ki a webszervertől, amelyek segítségül szolgálhatnak ahhoz, hogy illetéktelenül hozzáférést szerezzen a géphez;
- DoS-támadást (túlterheléses támadást) indíthat, megbénítva ezzel a szervert és ideiglenesen alkalmatlanná téve azt feladata ellátására.

Kliensoldali kockázatok

- Olyan tartalom letöltése, amely megbénítja a böngészőt, megsérülhet a kliensgép operációs rendszere, sérülhetnek biztonsági követelmények, vagy egyszerűen csak idegesítő;
- Olyan személyes információk helytelen kezelése, amelyeket – akarva vagy akaratlanul – a felhasználó adott meg.

Védelmi lehetőségek

Fontos kiemelni, hogy a „biztonságos” kliensek és szerverek csak a hálózaton küldött, ill. fogadott adat megvédésére képesek, a biztonságos működéshez szükség van mind kliens-, mind szerveroldali rendszerszintű megoldásokra is.

Kliensoldali biztonság

Noha a lényegesebb megfontolások a szervernél szükségesek, a kliens oldalán is szükség van bizonyos irányelvek betartására. Ne felejtsük el, hogy például az űrlapokban elküldött adatok alapértelmezés szerint titkosítás nélkül haladnak a hálózaton, erre a böngészők figyelmeztetnek is. Bevett megoldás az SSL használata, amelyet azonban szükségszerűen szerveroldalon is támogatni kell: a http biztonsági célú kiegészítése, a https a 443-as porton működik. Ha a szerver ezt nem támogatja, akkor SSL használatára nincs lehetőség, más megoldást kell keresni. Ilyen lehet például a titkosítás Javascript segítségével. A Java mindenkori implementációja azonban szintúgy tartalmazhat biztonsági hibákat, amelyeket a kliensgép megszerzéséhez adott esetben ki lehet használni.

Szerveroldali biztonság

Minél több szolgáltatása van a webszervernek, annál nagyobb a biztonsági kockázata is. Az egyik legfontosabb, hogy ne mindig root-ként fusson a szerver, hanem azonnal mondjon le a rendszergazdai jogairól, amint nincsen már azokra szüksége. Emellett lényeges, hogy megfelelően legyenek beállítva a szerver által szolgáltatott könyvtárstruktúra gyökerének és részeinek jogai. Írási joga természetesen senkinek se legyen, lehetőleg még annak a felhasználónak sem, akinek a nevében a webszerver fut: ha sikerül is egy támadónak a szerver felhasználója nevében egy programot vagy utasítást lefuttatnia a gépen, annak minél kisebb esélye legyen a károkozásra. Az sem árt, ha a fájlok és könyvtárak a root tulajdonában vannak, ekkor a legkisebb az esély arra, hogy ezeket bárki illetéktelen módosítani tudja, arra viszont ügyelni kell, hogy setuid-os bináris semmiképp se legyen közöttük.

Sem a naplófájlok, sem a szerver konfigurációs állományai ne legyenek módosíthatók, de lehetőleg még olvashatók sem olyan jogokkal, amilyenekkel a webszerver a root jogról való lemondása után fut. Mindig azt kell szem előtt tartani, hogy a lefuttatott CGI-programok azzal a joggal futnak, amilyen jogú szerver indította őket, és az egyik legfontosabb szempont, hogy ezeknek a programoknak minél kevesebb állományhoz legyen hozzáférési lehetőségük. A CGI-k, mint a futtatható programok általában, gyakran hoznak létre ideiglenes állományokat, ezek alapértelmezett helye a /tmp könyvtár. Ez azonban biztonsági kockázatot is hordoz magában, mert egy esetleges puffertúlcsordulásos vagy egyéb hiba esetén lehetővé válhat, hogy a támadó ide a CGI segítségével programot töltsön és azt elindítsa. Ennek elkerülése érdekében célszerű megfontolni, hogy milyen könyvtárakra adunk a szerveroldalon írási jogosultságot.

Alapvető biztonsági követelmények

Példaként az igen népszerű és nagy tudású Apache webszervert vizsgáljuk. Számtalan konfigurációs direktívája szolgál az elérés korlátozására, azonban egy általános célú webszervernél ennek a jelentősége csekély, lévén, hogy ez a szolgáltatás többnyire az egész külvilágnak szól. Néhány hasznosabb direktívát azonban érdemes sorra venni.

A szerver által szolgáltatott tartalmakat tároló könyvtárstruktúrát már említettük, ennek gyökerét az alábbi konfigurációs bejegyzéssel tudjuk megadni:

```
DocumentRoot "/var/www/html"
```

Ez a direktíva beállítja azt a legfelső szintű könyvtárat, amelyet a szerver a külvilágnak szolgáltathat. Amikor tehát a hosztot bármiféle kiegészítés nélkül URL-ként megadjuk, akkor az itt megnevezett könyvtárban keresi a szerver – alapértelmezés szerint – az index.html állományt, amit aztán a böngészőnek elküld. Ezen a könyvtáron kívülre, rendeltetésszerű működés esetén, sohasem kerülhet a szerver, kivéve, ha a szimbolikus hivatkozásokat engedélyezzük és van ezen a területen olyan szimbolikus hivatkozás, ami kimutat az adott könyvtárból.

Másik hasznos beállítási lehetőség például a CGI-k futtathatóságát korlátozó direktíva. Mindenek előtt azt kell beállítanunk, hogy melyik könyvtár(ak)ból engedélyezzük szkriptek futtatását a szerveren:

```
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
```

Ez a beállítás az URL-ben kapott cgi-bin könyvtár esetén a /var/www/cgi-bin tartalmában keres. Noha nem jelent kezelhetetlen problémát, ha CGI-k több helyről is futhatnak, viszont tekintettel arra, hogy a CGI-k az egyik legnagyobb biztonsági kockázatot jelentik, sokkal egyszerűbb, ha jól áttekinthető és kézben tartható, hogy hol vannak. Maga a CGI-k futása jelenti az igazi kockázatot, ezt csökkenthetjük, ha igen kis jogú felhasználó nevében

futtatjuk őket. Erre az Apache is lehetőséget ad a `suEXEC` kiegészítés használatával, ennek kiaknázásával megtehetjük, hogy a webszervert futtató felhasználónál is kisebb joggal rendelkezzenek a CGI-ként futó programok.

A szerveroldali kiegészítések, mint például a PHP, további kockázatot jelentenek. Legelső veszélyük, hogy jelentősen meg tudják növelni a szerver terhelését, hiszen a HTML-be ágyazott hívások tipikusan az adott oldal minden kiszolgálásakor lefutnak. Ezt nehezen kerülhetjük el, ha a szolgáltatást igénybe kívánjuk venni. Itt is alkalmazható azonban a CGI-knél is használt suEXEC kiegészítés, így a szerveren kívül futó binárisok által hordozott biztonsági kockázat csökkenthető.

Egyéb előnyös biztonsági beállítások

Sokféle támadás ellen véd, ha ún. chroot jail-be zárjuk a szervert. Ekkor, ha jogot is szerez a támadó a szerveren keresztül a géphez, csak nehezen tud a belső könyvtárstruktúrából kitörni és ezzel az uralmat a gép fölött átvenni.

Az Apache rendelkezik olyan jogosultság-ellenőrzési mechanizmussal, amely lehetővé teszi, hogy a rendszerszintű beállításokat bizonyos könyvtárak esetében felülírjuk, illetőleg felülírja az a felhasználó, akitől ezt a lehetőséget nem vesszük el. Egy .htaccess nevű állomány az ezt tartalmazó könyvtárra a globális beállításoktól eltérő, lokális engedélyezéseket tehet. Ha el akarjuk kerülni, hogy ilyen módon a felhasználók átnyúljanak a fejünk fölött, az Apache konfigurációjában ezt megtehetjük, ezzel kényszerítve mindenkire a globális szabályokat:

```
<Directory />
AllowOverride None
</Directory>
```

Ha a root felhasználót ki szeretnénk zárni az egyébként a szerver által kiszolgáltatható adatok gazdái köréből, akkor a következő direktíva használata javasolt:

```
UserDir disabled root
```

Megfigyelés és elemzés

Hogy nyomon tudjuk követni webszerverünk tevékenységét, a naplózást természetesen meg kell oldanunk és a naplófájlokat figyelnünk kell. Ugyan a naplók csak a már megtörtént eseményekről tájékoztatnak, az általuk nyújtott átfogó kép mégis jó útmutatást ad arra vonatkozóan, hogy milyen típusú támadások várhatók, és így ellenőrizhetjük, hogy a szükséges óvintézkedéseket életbe léptettük-e. Az access_log a sikeres kiszolgálásokat, az error_log pedig a külvilág által kért, de a szerveren nem talált fájlokat, visszautasított klienseket és egyéb hibákat naplózza. Ezek rendszeres figyelemmel kísérése után az igényelt változtatásokat a konfigurációban megtehetjük.

6.1.2. E-mail / Elektronikus levelezés

Az elektronikus levelezés mára több formában is elérhető, levelezőkliens programmal belépve egy adott szerverre, böngészőhöz kapcsolódóan, interaktív belépést követően a szerveren lévő klienssel stb. így a biztonsági problémák is szerteágazók lehetnek.

Kockázati tényezők, általános sebezhetőségek

Az elektronikus levelezést – a web böngészése mellett – az Internet leggyakoribb felhasználási módjának tekinthetjük. Az Internettel kapcsolatba kerülő felhasználók közül nagyon kevesen maradnak ki az e-mail forgalomból, vagyis meglehetősen széles közönséget érintő szolgáltatásról van szó. Ennek tükrében nem meglepő, hogy az elektronikus levélforgalom továbbítását végző levelezőszervereknek több kockázati tényezővel is számolniuk kell.

Ezek egy része követlenül a szerverszoftver ellen irányul. Ezek azok a támadások, melyek minden publikus szolgáltatást nyújtó rendszer esetén hasonló sémát követve a különböző szintű kommunikációs rétegek szabványos formátumát megtörve, vagy épp szabványos, de speciális tartalommal kártékony hatásúvá tett üzenetekkel, végső soron mindig a kiszolgáló *implementációs gyengeségét* használják ki, a szolgáltatást futtató számítógép általános támadását (DoS, puffer túlcsordulás, stb...) célozva.

Az ilyen, általános kockázati tényezőt jelentő, támadási formák mellett kiemelhetőek azok a problémák is, melyek jellegzetesen a levelezőkiszolgálókat érintik. Az egyik legkézenfekvőbb probléma az elektronikus levelezés nyilvános voltából adódik. Sok más korai protokollhoz hasonlóan az e-mail továbbításra használatos protokoll, az SMTP (Simple Mail Transfer Protocol) és a rá épített infrastruktúra is *optimista felépítésű*, vagyis az egyszerűség, technikai megfelelőség és működőképesség követelményeit tartotta fontosnak, azonban az idők folyamán ezek a követelmények eltolódtak, így jelenleg már nem fogadható el az a korai megoldás, mely szerint a mail szerverek feladata volt bármilyen címzett és feladó paraméterekkel ellátott levelek átvétele és továbbítása (mail relay) a címzett elsődleges kiszolgálójának irányába.

Látható, hogy ez a felépítés – a feladót jellemző információk megfelelő beállítása mellett – milyen egyszerűen használható ki hamisított levelek célbajuttatására. Egy tetszőleges forrásból származó levelek továbbítására rábírható, ún. *open relay* levelezőkiszolgáló így nem csak a rossz szándékú személyek számára biztosít visszaélési lehetőséget, hanem lehetővé teszi a kéretlen leveleket, vagy vírusokat küldő automaták számára is nemcsak a címzett, de a levelesládáját védő mechanizmusok megtévesztését is.

Ezzel elérkeztünk a levelezőkiszolgálókat érintő másik fontos területre, a spam- és vírusszűréshez. A kéretlen levelek különböző tiltólistákon, vagy statisztikai módszereken alapuló szűrése vagy megjelölése teljesítménybeli, kényelmi, vagy produktivitási kérdésként vizsgálható. Az e-mail klienseket támadó vírusok kiszűrésére azonban már a kiszolgálón sort kell keríteni, hisz itt egy pontban megvizsgálható a teljes áthaladó forgalom, vagyis itt van lehetőség a szűrőprogramok határvédelmi eszközként való alkalmazására.

A *vírusellenőrzés* megvalósításához szükség van a vírusszűrő program és a levelezőkiszolgáló integrálására, melynek különböző megoldásai különböző problémákat vethetnek fel. E problémák közös pontjaként említhető a szűréssel járó mintaillesztés, formátumkonverzió és az esetlegesen csomagolt mellékletek kezeléséből származó jelentős terhelésre alapozható DoS támadás lehetősége.

A szolgáltatásbénításra a spam- és vírusszűrés teljesítményigényét kihasználó célzott támadásoknál is nehezebben kezelhető, indirekt és elosztott, kimondottan a levelezőrendszereket érintő lehetőséget ad az ún. *backscatter* támadás, melynek során – az SMTP protokoll eredendően jóhiszemű felépítését kihasználva – a bénításra kiszemelt kiszolgáló által fogadott domain-névbe tartozó hamisított feladóval és hamisított címzettel küldenek leveleket (általában spam-eket) rengeteg különböző levelezőszervernek, amelyek ennek hatására kézbesítési hibaüzenetekkel árasztják el az áldozatot. Megjegyezendő, hogy a backscatter „támadás" sok esetben a spam küldő robotok által – a kéretlen levelek hitelesebbé tételét célozva – elhelyezett valódi forráscímek miatt, mintegy melléktermékként jelenik meg.

A fent leírt problémák mind a levelek továbbításával, vagy a levelezőszerverre való fogadásával kapcsolatosak, azonban figyelmet kell szentelnünk a felhasználók levélfogadását elősegítő, elterjedt POP (Post Office Protocol) és IMAP (Internet Message Access Protocol) protokollokra is. Felhasználási módjuknál fogva ezek a protokollok általában több fajta felhasználó-azonosítási metódussal rendelkeznek, de e módszerek elsődlegesen a felhaszná-

lói fiók azonosítását célozzák és nem, vagy nem elegendő mértékben foglalkoznak a kommunikációs csatorna lehallgatásából eredő problémákkal. Ezen felül e protokollok kiszolgálóit is érintik az SMTP esetén említett implementációs gyengeségekből eredő kockázati tényezők.

Védelmi lehetőségek

Az előző szakaszban részletesebben kifejtett kockázati tényezőket összefoglalva a következő általános védelmi lehetőségekkel élhetünk.

Implementációs gyengeségek

Az implementációs gyengeségek elleni védekezésképp célszerű „tiszta előtörténettel" rendelkező, dokumentáltan és célzottan a biztonsági követelményeket és a vonatkozó kódolási előírásokat figyelembe vevő módszerekkel elkészített szerverszoftver alkalmazásával lehet leginkább védekezni. Egy ilyen szempontok szerint választott, modern szoftver esetén is fontos az általános frissítési alapelvek figyelembevétele és betartása.

Optimista felépítés

Az SMTP protokoll és a rá épülő infrastruktúra optimista felépítése, a hitelesítést és titkosítást célzó megoldások hiányosságai komoly fejtörést okozó architekturális kockázati tényezők forrásaként szolgálnak, melyekre nincs megfelelő alternatív intézkedés.

A kötelező azonosítási lépés miatt a POP és IMAP protokoll ilyen szemszögből nem tekinthető nyilvános elérésűnek, vagyis az implementációs gyengeségek kiküszöbölése és a lehallgatás elleni védekezés, például egy SSL biztosítású csatornával megoldást szolgáltathat a protokollok gyengeségeivel szemben.

Open relay

Az open relay gyakorlatilag konfigurációs problémaként kezelhető, mivel az általános igényeket kielégítő levelezőszerverek mindegyikénél lehetséges a nyílt levéltovábbítás korlátozása. Célszerű olyan szoftvert választani, mely a korlátozások széles skáláját támogatja, hiszen gyakran fordulhat elő, hogy különböző felhasználási igényeknek megfelelően a kiszolgálón nem elegendő a csatlakozó IP címe, vagy a kapcsolatot fogadó hálózati interfész alapján történő szűrés. A tetszőleges forráscímről való levélfogadáshoz általában külön azonosítási lépés szükséges. Mivel az SMTP protokoll alapvetően nem támogatja a kommunikációs partnerek azonosítását és hitelesítését, a problémára több eltérő megoldás is született.

Az egyik megoldás, az ún. *„POP before SMTP"*, a levelezőkliensek azon tipikus (de nem garantált) viselkedését próbálja kihasználni, hogy a felhasználói postafiókok tartalmának megtekintése, letöltése (például az elterjedt POP protokoll segítségével) általában rövid idővel megelőzi a kliens kimenő irányú levélforgalmának megindulását. Ez alapján az SMTP szerver a kötelező POP azonosítási lépés alapján a kliens IP címére rövid időre (például 15 percre) engedélyezi a mail relay funkciót. A megoldás előnye, hogy a kliensben semmilyen átalakítást nem igényel, és a szerver a korlátozás eredményeképp végülis nem tekinthető open relaynek. Az a tény azonban, hogy a szerver a kliens viselkedésére felállított feltételezések alapján korlátozza és időzíti szolgáltatásait, gyenge lábakon álló szolgáltatásbiztonságot eredményez, ráadásul az SMTP szerver számára kommunikációs lehetőséget kell biztosítani a levél letöltését lehetővé tevő POP vagy IMAP szerverrel, mely további kialakításbeli korlátozásokhoz, és új kockázati tényezők megjelenéséhez vezethet.

Elegánsabb – de a kliens támogatását is igénylő – megoldást szolgáltat az SMTP protokoll kibővítésével lehetővé tett azonosítási lépés bevezetése. Erre elterjedten az *SASL* (Simple Authentication and Security Layer) protokoll bevezetésével került sor. A legtöbb modern e-

mail kliens és szerver már támogatja a kimenő SMTP kapcsolatok autentikációját ezen a módon. Meg kell jegyeznünk azonban, hogy az SASL csak azonosítást biztosít, és titkosítást nem, viszont az *SSL* vagy *TLS* csatornán keresztül felépített SMTP kapcsolatot még nem minden kliens támogatja.

Spam- és vírusszűrés

A spam szűrésére általános, heterogén hálózati környezetben nem adható teljes megoldás, mivel sem a kézzel, akár elosztottan karbantartott tiltólisták, sem pedig a statisztikai módszerek nem garantálják a teljes biztonságot, minden esetben értékes levelek elvesztésére is számítani kell. Ez ugyan egyes esetekben még megengedhető, az ezzel kapcsolatos döntés azonban már nem technológiai jellegű, inkább a hálózathasználati policy hatáskörébe tartozik. Gyakran előfordul, hogy a levelezőkiszolgáló az említett okok miatt nem végez tényleges spamszűrést, csak a spam-nek érzékelt levelek megjelölésével segíti a felhasználókat.

Az említett statikus tiltó- és engedélyező listák – idegen szakkifejezéssel *blacklist* és *whitelist* – mellett elterjedőben van egy köztes, intelligensebb spamszűrési megoldás, az ún. *graylist* (vagy *greylist*), mely azt próbálja kihasználni, hogy a kéretlen leveleket küldő automaták a tömeges levélküldés minél gyorsabb megoldására törekedve általában egyszerűsített SMTP rendszerrel rendelkeznek, és nem készültek fel a legitim kliensek által ismert és kezelt összes hibaesetre, például az „átmeneti hiba" kategóriába tartozó helyzetek kezelésére.

A graylist megoldást alkalmazó kiszolgáló egy – kézzel is bővíthető – whitelist és egy blacklist automatizált karbantartását végzi. A bejövő kapcsolatokat – ha a feladó, vagy a kliens szerepel az engedélyező listán – a szűrő teljesen átlátszó módon kezeli. A még ismeretlen kapcsolatokat azonban a szerver először „átmeneti hiba" üzenettel elutasítja, majd a kliens újracsatlakozására várakozik, miközben a feladó azonosítására különféle ellenőrzéseket is foganatosíthat. A kliens későbbi újracsatlakozása azt jelzi, hogy ott is összetett levelezőszerver üzemel, mire – ha az időközben lefuttatott egyéb ellenőrzések is sikeresek voltak – a whitelist az új kliens címével bővül.

A blacklist automatikusan bővülhet, ha egyazon klienstől több független – nem újracsatlakozás jellegű – kézbesítési kísérlet indul, későbbi újracsatlakozás nélkül, de alapvetően a whitelist bővítésének manuális karbantartására szolgálhat. A megoldás hátránya, hogy több legitim feladó – jellegzetesen egyes automaták – sem rendelkeznek helyes SMTP hibakezeléssel, ezért a graylist megoldás sem tekinthető biztonságosnak, nem beszélve arról, hogy szélesebb körű elterjedése bizonyosan magával fogja vonni a kéretlen levelek küldőinek alkalmazkodását is.

Lehetséges alternatívaként említhető a graylist rendszer mechanizmusaira emlékeztető, jóváhagyáson alapuló szűrési megoldás, amely ugyancsak egy whitelist automatikus bővítését végzi. Ez a beérkező leveleket ugyan minden esetben átveszi, de egy kriptográfiai tokennel bővített, visszaigazolást kérő levelet küld a feladónak, melyben arra kéri, küldje vissza a tokent. A token visszaérkezésekor az eredeti – addig késleltetett levelet – átengedi, és egyidejűleg a visszaigazolást elküldő felet az engedélyező listára helyezi. A megoldással természetesen ismét az a gond, hogy a visszaigazolni képtelen automaták, például értesítő szolgáltatások és levelezőlisták, nem tudnak a whitelistre kerülni, leveleik a címzett számára jelzés nélkül eltűnhetnek.

A spamszűréssel ellentétben a vírusszűrés meglehetősen nagy biztonsággal elvégezhető feladat. Figyelembe kell azonban venni, hogy amint a vírusok, úgy a vírus adatbázisok is napról napra változnak, így kiemelt, és az általános levelezési követelményektől jellemzően eltérő figyelmet igényel a rendszer folyamatos felügyelete és karbantartása. A levelezőszer-

verek általában önmagukban nem támogatják a vírusszűrést, így külön vírusszűrő programcsomag alkalmazása szükséges.

A vírusszűrés a levelezési folyamat különböző pontjain valósítható meg. Első megközelítésben célszerűnek tűnhet a vírusszűrés beépítése az SMTP protokollba, hisz így már a kommunikáció során elutasításra kerülhet a vírusosnak észlelt levél, ezzel a probléma lekezelését a feladóra lehet bízni.

E megoldás problémája az, hogy erőforrás igénye nagy. Összetet szerkezetű levelek esetén a vizsgálat jelentős méretű tárterületet igényelhet, ugyanakkor a szűrési folyamatnak a kommunikációs csatorna lezárása előtt végig kell futnia, vagyis nincs mód arra, hogy a – általános levelezőszerver-arhitektúrában jellegzetesen löketekben érkező – levélforgalmat majd átmeneti tárolás után kötegelten dolgoztassuk fel.

A gyors kapcsolatfeldolgozás utáni kötegelés lehetőségének elvetése még jóindulatú üzemkörnyezetben is kisebb-nagyobb torlódásokat, elutasított kapcsolatokat eredményezhet, míg szándékos DoS támadás esetén elsődleges felületet nyújthat a rendszer túlterhelésére.

A kötegelt feldolgozással végrehajtott vírusszűrésnél nem merülnek fel ilyen csúcsterhelési problémák, de az átlagos szerverterhelés jelentős, akár több nagyságrendbeli megnövekedésével itt is számolni kell.

Backscatter és egyéb DDoS lehetőségek

Mivel DDoS támadások ellen meglehetősen nehéz védekezni, csak néhány alapvető korlátozó eszközzel élhetünk a probléma kezelésének érdekében. A spam- és vírusszűrésnél is hasznosnak bizonyulhat – de az említett heterogén környezet miatt levélvesztéshez vezethet – a vonatkozó levéltovábbítási és levélformátum szabványok szigorú betartatása, mellyel az ellenőrizhetetlen, vagy hiányos feladójú, érvénytelen domain-névvel bejelentkező, a formátumot vagy a protokoll szabályait akárcsak kis mértékben is megszegő kapcsolatokat azonnal eldobhatja a kiszolgáló, így növelve áteresztő képességét.

Alapvető biztonsági követelmények

A mintarendszer levelező szerverének alapjául az IBM cég által – az akkor közkeletű és defacto szabványnak számító `sendmail` programnak megfelelő alternatívaként („drop-in") -- kifejlesztett `postfix` (technikai és konfigurációs leírását lásd: http://www.postfix.org) levelezőrendszer szolgál. A `postfix` rendszert rugalmas és hatékony spam- és vírusszűrési tulajdonságokkal látja el a `MailScanner` program (bővebb információ: http://www.mailscanner.info), valamint az általa támogatott és felhasznált vírusszűrők hosszú listájáról a kiterjedt és ingyenes frissítési támogatás alapján kiválasztott `ClamAV` antivírus program (http://www.clamav.net), végül az egyik legelterjedtebben használt spamszűrő, az egyéb kisebb programokat is integrálni képes `spamassassin` (http://spamassassin.apache.org). Az alábbiakban a felsorolt programrendszer egyes, külön konfigurálandó, részeinek alapvető és előnyös biztonsági beállításait mutatjuk be.

Postfix beállítások

A `postfix` konfigurációs állományai a `/etc/postfix` könyvtárban találhatók. Két alapvető fájlt kell megemlítenünk: a `main.cf` a rendszer viselkedését alapvetően befolyásoló konfigurációs beállítások helye, a `master.cf` pedig a rendszer kommunikációs komponenseinek leírására szolgál.

A `master.cf` beállítási lehetőségei közül biztonsági szempontból fontos flageket állíthatunk be a különböző részfeladatokkal rendelkező daemon szálak számára:

– Az `unpriv` jelző „`yes`”-re állításának hatására az adott processz nem root jogosultságokkal fog futni. Ez a beállítás segít az esetleges implementációs problémák hatásának csökkentésében. Mivel a levelező szolgáltatás egy privilegizált porton, a standard 25-ös TCP csatornán hallgat a bejövő üzenetekre, az `smtp` komponenst mindenképp privilegizáltan kell futtatnunk. A lokális felhasználók levelezési könyvtárainak írása szintén root jogosultságot igényel az adott felhasználó személyazonosságának felvételéhez, ezért a kézbesítést végző `local` komponens is „`no`” jelzőt kap.

– A `chroot` jelző igenlő beállításának hatására az adott szolgáltatás számára csak a levelezőrendszer technikai területei lesznek láthatóak. Ez célszerűen minden szolgáltatási szálat érint, kivéve azokat, melyek például a felhasználói könyvtárak eléréséhez, vagy más speciális feladatok végrehajtásához a teljes fájlrendszer elérését igénylik. Ilyen az előzőleg említett `local` komponens, azonban az `smtp` komponens nyugodtan futhat chroot beállítással.

– A `maxproc` beállítás szabályozza az egyes szolgáltatási szálak maximális párhuzamos futásának lehetőségét. A levelezőrendszert futtató számítógép teljesítményétől függően célszerű lehet a DDoS támadási felületet nyújtó `smtp` komponens terhelésfüggő korlátozása.

A `main.cf` által nyújtott lehetőségek közül a következők helyes megadását alapvető biztonsági követelménynek tekinthetjük:

– A `mynetworks` beállítással adhatjuk meg, melyek azok a forrás IP címek, melyeket a levelezőkiszolgáló saját címnek tekint. A beállítás alapértelmezése, a `127.0.0.0/8`, csak a `postfix` rendszert futtató számítógépről fogad leveleket, azonban bizonyos esetekben a tartomány bővítése válhat szükségessé, például egy intranetes kiszolgáló esetében. Fontos, hogy sose állítsuk a `mynetworks` értékét olyan IP címek tartalmazására is, melyek fölött nem rendelkezünk közvetlen befolyással, vagyis ne szolgáljunk „open relay”-ként nem megbízható hosztok számára.

– A `mydestination` paraméter a fogadni kívánt levelek címzettjének domain-neveit szabályozza. A spam és vírus automaták címgyűjtésének hatékonyságát ronthatja, ha csak a használatban levő domainek kiszolgálását engedélyezzük, vagyis például a más célra használatos `www.`, `ftp.`, `ns.` kezdetű címeken nem veszünk át leveleket.

– A `content_filter` egy alternatív spam- és vírusszűrő megoldás támogatására szolgál. Ez a megoldás az SMTP protokollt használja a levelezőszerver és a tartalomszűrő szolgáltatás közti kommunikációra, a bejövő leveleket a `content_filter` opció által meghatározott címre küldi.

– Az `inet_interfaces` határozza meg, a levelezőkiszolgáló mely hálózati interfészeken fogadjon leveleket (a levélküldés az opciótól független döntéseken alapszik). Egyes esetekben kézenfekvő korlátozó tényező lehet. Az „`all`” kulcsszó a számítógép összes interfészének felel meg.

– A `message_size_limit` opció segítségével byte pontossággal lehet meghatározni a fogadni kívánt levelek méretét. Ennek a beállításnak egyensúlyt kell szolgáltatnia a levelezőszerver funkcionalitása és teherbírása között, hisz az Internet kapcsolatok sebességének és a (multimédia) fájlformátumok átlagos méretének növekedésével meglehetősen nagy méretű üzenetek is e-mailbe kerülhetnek, ráadásul a levelekben bináris tartalom továbbítására használatos BASE64 kódolási mód még egyharmadával növeli a továbbított fájlok méretét. A fájlméretet korlátozatlanul hagyva a szerver levélforgalmát egyszerűen

leállíthatja egy „végtelen” méretű üzenet, de a méret növekedésével az általános DoS támadások hatékonysága is növekedhet.

– A már említett SASL azonosítási funkciók bekapcsolására a `smtpd_sasl_auth_enable` változó igenlőre állítása szolgál. A SASL belépési kötelezettség alól felmentést kapnak az `smtpd_sasl_exceptions_networks` változóban szereplő címtartományok. Az egyes kliensek által hibásan támogatott SASL implementációkkal való együttműködéshez érdemes lehet a `broken_sasl_auth_clients` változó igenlő beállítása, azonban ezt a lehetőséget csak akkor célszerű választani, ha biztosan tudjuk, hogy szükség van rá.
– A SASL azonosítást kiegészítendő célszerű a TLS támogatás engedélyezése is, ehhez az `smtpd_use_tls` változó igenlő beállítása szükséges, emellett meg kell adni a nyílt kulcsú titkosításhoz elengedhetetlenül szükséges kulcsok elérhetőségét is (`smtpd_tls_CA-file`, `smtpd_tls_cert_file` és `smtpd_tls_key_file` változók). Mivel, ahogy említettük, a TLS (vagy SSL) protokollt nem minden SASL kliens támogatja, felmerülhet a TLS opcionálissá tétele is, amit az `smtpd_enforce_tls` igenlő beállításával érhetünk el.

Egyéb előnyös biztonsági beállítások

A fenti, kötelező érvényűnek tekinthető biztonsági beállításokon felül az alábbi hasznos opciókra érdemes még figyelmet fordítani.

Postfix beállítások

– Az `smtpd_banner` opció segítségével korlátozhatjuk, mennyi és milyen információval lássa el a szerver a csatlakozó klienseket. Alapértelmezésben ez a változó az operációs rendszer és a `postfix` verzióját is közli, azonban elegendő lehet egy egyszerű „ESTMP service” üzenet is (az ESMTP az SMTP kibővített parancskészlettel rendelkező változata).
– A `header_checks` és `body_checks` változók segítségével olyan szűrőfeltételeket tartalmazó állományokat adhatunk meg, melyekben például reguláris kifejezésekkel állíthatók fel alapvető szabályok az átvenni kívánt, vagy épp elutasítandó levelek fejlécére és törzsére. Ilyen szűrőfeltételek segítségével tilthatjuk például a teljes rendszerre az `.exe`, és egyéb veszélyesnek tekintett kiterjesztésű, vagy MIME típusú fájlok fogadását. Mivel mintarendszerünkben külön vírusszűrő rendszer üzemel, melynek szolgáltatásaival rugalmasabban szabályozhatjuk a fogadni kívánt tartalmat, ezt a lehetőséget csak mint gyorsító alternatívát említjük.
– A `strict_rfc821_envelopes` és `smtpd_helo_required` változók igenlő beállítása a backscatter, spam- és vírustámadások hatásának csökkentését célozza az SMTP kommunikáció szabványosságának kényszerítésével. Mint említettük, az ilyen szigorú követelmények azonban egyes legitim levélforrások kitiltásához is vezethetnek.
– Az `smtpd_sender_restrictions` változó egy tűzfal-jellegű sorozatos szabályillesztést tesz lehetővé a küldő fél korlátozására, a szabályok folyamatosan egymás után kerülnek kiértékelésre. A `postfix` dokumentáció részletesebb információt szolgáltat az egyes paraméterekről, itt csak a fontosabbakat emeljük ki:
  – A `permit_mynetworks` szabály hatására, a `mynetworks` konfigurációs változóban megadott hálózati címek engedélyezésre kerülnek, vagyis akármilyen feladót megadhatnak.
  – A `reject_unknown_sender_domain` az ismeretlen domain-nevet megadó feladók elutasítását eredményezi.

  - A `reject_non_fqdn_sender` a nem teljesen specifikált (például: „www.sztaki.hu" helyett csak „sztaki") nevet megadó feladók elutasítását eredményezi.
  - A `check_sender_maps` változó által megadott, feladókra vonatkozó mintaillesztéseket tartalmazó fájl alapján engedélyezi, vagy tiltja az elérést.
- Az `smtpd_recipient_restrictions` változó az előző változóhoz hasonló szabályillesztést tesz lehetővé a címzett tekintetében. A fontos paraméterek:
  - A `permit_mynetworks` az előzőekhez hasonlóan a `mynetworks` változóban megadott hostok számára akármilyen címzett megadását lehetővé teszi.
  - A `permit_sasl_authenticated` változó hatására a SASL azonosítást elvégző kliensek számára akármilyen címzett megadását lehetővé teszi.
  - A `reject_unknown_recipient_domain` és `reject_non_fqdn_recipient` az előzőekhez hasonlóan az ismeretlen vagy nem teljesen specifikált címzett domainek esetén tiltja a hozzáférést.
  - A `check_recipient_maps` változó által megadott, címzettekre vonatkozó mintaillesztéseket tartalmazó fájl alapján engedélyezi, vagy tiltja az elérést.
- Az `smtpd_helo_restrictions` az SMTP protokoll kezdeti lépését jelző HELO parancsban megadott kliens hoszt nevére állít korlátozásokat a következő fontosabb szabályokkal:
  - A `permit_mynetworks` hatására az előzőekben megadottakkal összhangban a szabályok feloldása történik meg.
  - A `reject_invalid_hostname`, `reject_unknown_hostname` és `reject_non_fqdn_hostname` korlátozások rendre a hibás, ismeretlen vagy nem teljesen specifikált kliens hosztnevek tiltását végzi. Itt fontos megjegyezni, hogy az SMTP protokoll pontos követéséhez hasonlóan ezek a szabályok is legitim kliensek elutasítását eredményezhetik.
- Az `smtpd_error_sleep_time` változó nullára való állításának eredményeképp az SMTP protokollt sértő hibák esetén azonnali kapcsolatmegszakítást tesz lehetővé a szerver számára. Ez a beállítás nagyságrendekkel javíthatja a DDoS támadásokkal szembeni ellenállóképességet a fenti egyéb korlátozó szabályokkal kombinálva, azonban egyes legitim klienseket megzavarhat ez a viselkedés.
- Az `smtpd_client_connection_rate_limit` beállítás az azonos forrás IP címről érkező másodpercenkénti kapcsolatfelvételi próbálkozások számát korlátozza. A számot alacsonyan tartva ismét a DDoS támadásokkal szembeni ellenállóképesség növelhető, azonban a túl alacsonyra beállított értékek a levelezőszerver áteresztőképességét csökkenthetik.

MailScanner beállítások

A `MailScanner` egy specializáltan levél-ellenőrzési és szűrési célú szolgáltatásokat nyújtó szoftvercsomag, így a legtöbb alapbeállítás megfelel az alapvető biztonsági követelményeknek. E helyütt csak a fontosabb, vagy speciális viselkedésű konfigurációs fájlokról illetve beállításokról ejtünk szót, további részleteket a megfelelő kézikönyv tartalmaz.

A `MailScanner` konfigurációs állományai a `/etc/MailScanner` könyvtárban találhatók. Két alapvető fájlt kell megemlítenünk: a `MailScanner.conf` a rendszer viselkedését alap-

vetően befolyásoló konfigurációs beállítások helye, a `spam.assassin.prefs.conf` a spamszűrő alrendszer beállításait tartalmazza.

A rendszer a fenti két általánosabb célú konfigurációs fájlon kívül a `virus.scanners.conf` fájlban keresi a telepített víruskereső programok elérési útvonalait. Minden vírusszűrőt azonos, szabványos paraméterekkel hív meg, ezért a legtöbb ilyen egyedi termékhez külön wrapper („burkoló") állományok tartoznak, melyeket a `wrapper` alkönyvtárban találhatunk.

A `MailScanner` a fogadni, vagy elutasítani kívánt fájlnevek és fájltípusok leírását a `filename.rules.conf` és `filetype.rules.conf` fájlokban tárolja, amelyek tartalmát tetszés szerint bővíthetjük. Az alapértelmezett beállítások jószerivel az összes közvetlenül indítható fájltípust tiltják, vagyis kiszűrik a levelező kliensek automatikus futtatási hibáinak kihasználását célzó kártékony programok többségét.

Fontos megemlítenünk az értesítő levelek beállításainak (helyénvaló) alapértelmezéseit (`MailScanner.conf` „`Notifications`" szekció), miszerint *a vírusokat küldő felhasználók levélben való értesítése **nem** történik meg*. Sajnálatos módon még mindig számos e-mail vírusszűrő szoftver küld a – gyakran hamisított – feladói címekre ijesztő, félrevezető tartalmú leveleket, melyekben az esetleg onnan soha el nem küldött levelek alapján az ártatlan felhasználó gépén feltételezett vírusok eltávolítására szólítanak fel. Ez a viselkedés nem csak a vétlen felhasználókat zavarja és a rendszergazdákat kényszeríti folytonos magyarázkodásra, hanem kitűnő lehetőséget ad a backscatter támadáshoz hasonló DDoS megvalósítására is (képzeljünk el egy kezdetben gyorsan terjedő vírust, mely bizonyos kiszemelt forráscímek nevében továbbítja magát szerverről szerverre).

A „`Spam Actions`" konfigurációs váltózó segítségével definiálhatjuk, hogy a rendszer által felhasznált spamassassin által kéretlen levélnek vélt üzenetekkel mi történjék. Ez a beállítás alapesetben „`deliver`", azaz kézbesítést jelöl ki, azonban lehetőség van például arra, hogy egy vállalkozó szellemű felhasználó címére ömlesszük az összes spamet, vagy az eredeti üzenetet egy értesítő formalevél mellékleteként továbbítsuk.

Megfigyelés és elemzés

A postfix és mailscanner rendszerek a Linux syslog alrendszerén keresztül állítható részletességű, igény szerint bőséges naplómennyiséggel támogatják a rendszer levélforgalmának megfigyelését és elemzését. Az elemzést megkönnyíti a logwatch segédprogram, melynek napi összesítő táblázatai alapján a küldött, fogadott, hibás, vírusos és egyéb állapotú levelek számának változásai nyomon követhetők.

Ezen felül a MailScanner „`send notices`" konfigurációs opciójának segítségével lehetőség van egy kiemelt rendszergazdai fiók direkt értesítésére, ez azonban az Internet jelenlegi vírusforgalma mellett könnyen a fiókot figyelő felhasználó gyors elfásulásához vezethet.

6.1.3. Samba / Windows-Linux fájlmegosztás

Kockázati tényezők, általános sebezhetőségek

Mivel a SAMBA kiszolgáló mindig a kliens kérései alapján tényked ik, „magától" nem kezd önálló tevékenységbe, ezért kétféle kockázattal kell számolnunk. Az első az, hogy a program esetleges hibáját kihasználva a rosszindulatú kliens eléri azt, hogy a kiszolgálónk beszünteti a működést, és akár a gazda operációs rendszer rendelkezésre állását is komolyan akadályozza. A második eset, amikor a program vagy a konfigurálás hibáját kihasználva a támadó illetéktelenül adatokhoz fér hozzá, azokat megszerzi, vagy módosítja.

Védelmi lehetőségek

Első körben a SAMBA kiszolgálónkat a hálózat szintjén védhetjük. Megtehetjük ezt a tűzfalunkon is, a SAMBA szerver a 137-es, 138-as UDP és a 139-es és 445-ös TCP portokon kommunikál, de szerencsére maga a SAMBA ilyen szempontból jól konfigurálható. A fejlesztők gyorsan követik a hálózati protokoll változásait, így a szerver a lehetőségekhez képest biztonságos fájl és nyomtatóműveleteket szolgáltat.

Védekezzünk úgy hálózati, mint interfész szinten. Ha a gépünknek több interfésze van, akkor csak a belső hálózati interfészen engedjük meg a SAMBA használatát. Alábbiakban példán mutatjuk be, hogy érhető el ez a működés a /etc/samba/smb.conf fájlban:

```
interfaces = eth1 lo
bind interfaces only = yes
```

Ilyenkor csak az eth1 és loopback interfészen figyel a SAMBA. Hálózati szintű hozzáférés szabályzásra is van lehetőség. Ismét egy példát láthatunk:

```
hosts allow = 127.0.0.1 10.0.21.0/24 10.0.22.0/24
hosts deny = 0.0.0.0/0
```

A fenti konfiguráció hatására a hozzáférés csak a localhost-ról, a 10.0.21.0/24-es és a 10.0.22.0/24-es hálózatokból lehetséges. Védekezhetünk felhasználó szinten is úgy, hogy az „smb.conf" [global] szekciójában felsoroljuk az érvényes felhasználókat, felhasználócsoportokat, így:

```
valid users = makesz, @iroda
```

Ekkor a kiszolgálónkhoz csak a 'makesz' felhasználó és az 'iroda' csoport tagjai kapcsolódhatnak.

Külön szabályozhatjuk a hozzáférést az IPC$ megosztáshoz. Ez az egyetlen megosztás, amit anonymous módon mindenki elérhet, így a korlátozás a felhasználói azonosító és jelszó nélkül próbálkozó támadók ellen jelent védelmet. A többi megosztás természetesen a felhasználók számára hozzáférhető marad. A példa most sem marad el:

```
[IPC$]
hosts allow = 10.0.1.0/24 127.0.0.1
hosts deny = 0.0.0.0/0
```

A példa a 'localhost' és a '10.0.1.0/24' felől csatlakozókat engedi be. A többiek 'acces denied' üzenetet kapnak, nem böngészhetnek a többi megosztás közt és más erőforrásokhoz sem férhetnek hozzá. Ez a módszer csak akkor ajánlott, ha az előtte felsoroltak közül egyik sem célravezető.

Alapvető biztonsági követelmények

Ne engedjünk hozzáférést máshonnan, csak a helyi, privát hálózatról. Ha nem így teszünk, akkor az Internet összes feltört, Windows-t futtató számítógépe potenciális támadóként léphet fel SAMBA szerverünkkel szemben. A védelem második szintje a fájl és nyomtató-hozzáférés jól átgondolt szabályozása.

A témáról a következő lapon olvashatunk részletesen:

http://href.hu/x/d6c

(http://hu.samba.org/samba/docs/man/Samba-HOWTO-Collection/AccessControls.html)

Egyéb előnyös biztonsági beállítások

Az NTLMv2 hitelesítés konfigurálására az alábbi regisztrációs adatbázis kulcsok használhatók:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa]
    "lmcompatibilitylevel"=dword:00000003
```

A gép a 0x00000003 érték hatására csak NTLMv2 válaszokat ad. A kliensek NTLMv2 hitelesítést fognak használni, a kapcsolat is NTLMv2-es lesz, ha a kiszolgáló támogatja. A tartományvezérlők LM, NTLM és NTLMv2 hitelesítést használnak.

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\MSV1_0]
    "NtlmMinClientSec"=dword:00080000
```

A 0x00080000 érték hatására csak NTLMv2 kapcsolatok jöhetnek létre. Amennyiben az NtlmMinClientSec vagy az NtlmMinServerSec közül bármelyiknek az értéke 0x00080000, akkor a kapcsolat meghiúsul, ha nem tudnak megegyezni NTLMv2-ben.

Megfigyelés és elemzés

A SAMBA naplófájlok alapértelmezésben a /var/log/samba/ könyvtárba készülnek. A hálózaton közlekedő csomagok megfigyelésében a tcpdump program segíthet.

6.1.4. FTP

Kockázati tényezők, általános sebezhetőségek

Titkosítatlanság

Az FTP az egyik legrégebbi hálózati protokoll, amit TCP/IP fölött használni szokás. Már csak kora miatt sem tudja maradéktalanul figyelembe venni a mai elvárásokat. Legismertebb és legkönnyebben kihasználható hiányossága az, hogy az adatok a hálózaton nyílt szöveg (plain text) formában közlekednek. Mivel az FTP ugyanakkor felhasználó-alapú azonosítást használ, szükségszerű a jelszóval védett hozzáférés-ellenőrzés. Az iménti két alapvető tulajdonság elkerülhetetlen következménye, hogy az azonosításkor a kliens felől érkező adatok, a felhasználónév és jelszó mindenfajta titkosítás nélkül jutnak el a szerverig és akár hosszú utat is megtehetnek a hálózaton, amelynek bármelyik szakaszában fennáll a lehallgatás veszélye. A lehallgatott adatforgalomból pedig egy igen egyszerű protokollanalizálással máris kinyerhető a felhasználónév és a hozzá tartozó jelszó.

Programhibák és jellemző tulajdonságok

A protokoll egyszerűségéből eredő kockázat mellett nem hanyagolható el a szerverek működéséből és esetleges hibáiból származó kockázat sem. Például gyakorlatilag minden FTP-szerver rootként fut, legalábbis működése egy bizonyos szakaszában. Két alacsony sorszámú – TCP/20-as és TCP/21-es – portot használ, és ahhoz, hogy ezekre a portokra tudja kötni magát, a szervernek rendszergazdai jogosultságra van szüksége. Mindez potenciális hibaforrás, hiszen senki sem garantálja, hogy valamely hibát kihasználva egy támadó nem épp ennek segítségével fogja a gép fölötti uralmat átvenni.

Mint felhasználói azonosítást használó szerver, az FTP-szerver is kockázatot jelent, ha a root felhasználónak van rajta keresztül hozzáférési jogosultsága. A csak root joggal olvasható vagy írható fájlok FTP-n keresztüli manipulálási lehetősége számos veszélyt hordoz magában, például a kliens módosíthat olyan, egyébként rendszergazdai joggal futó programot, amely alkalmas a gép fölötti uralom megszerzésére. További nyilvánvaló hátránya ennek, hogy a hálózaton kódolatlanul közlekedő root-jelszó az egyik legkönnyebben kihasználható biztonsági hiba, ennek megszerzésével azonnali korlátlan uralomhoz juthat a támadó.

Összességében elmondható, hogy az FTP bármiféle valós felhasználói jogosultságú használata be nem látható veszélyeket jelent, ezért, hacsak lehet, ennek a hozzáférésnek a lehetőségét korlátozni, ill. tiltani kell.

Alapvető biztonsági követelmények

Számtalan FTP-szerver érhető el ingyenesen, többségük szereti magáról azt állítani, hogy a legbiztonságosabb. Elhinni persze egyiket sem kell, mindenesetre olyan szervert válasszunk, amelyet rendszeresen frissítenek és karbantartanak. Példaként mi a *proftpd*-t választottuk, amelynek nagy tudása, sok hasznos tulajdonsága van, emellett a tapasztalatok szerint megfelelően védhető, de kisebb funkcionális igény esetén az egyre jobban terjedő *vsftpd* is alkalmas a feladat ellátására. Azoknak a felhasználóknak a listáját írjuk a /etc/ftpusers állományba, akiknek az FTP használatát tiltani szeretnénk. A *proftpd* ennek a fájlnak a tartalmától függetlenül tiltja a root hozzáférését, de ezt az alapértelmezett tiltást a „RootLogin on" direktívával kiiktathatjuk, ha mindenképpen ki szeretnénk szolgáltatni a szerverünket. Rengeteg konfigurációs beállítási lehetősége miatt a *proftpd* szolgáltatásait és azok kezelését nem részletezhetjük, de néhány fontos beállításra kitérünk.

Természetesen korlátozhatjuk és feltétlenül érdemes is korlátoznunk, hogy mely gépek férhessenek hozzá a szerverhez. Az alábbi konfigurációs részlet jelentése intuitív és áttekinthető:

```
<Limit LOGIN>
Order Allow,Deny
Allow 192.168.2.8, mydomain.com, anotherdomain.net
Deny from all
</Limit>
```

A példában szereplő direktíva a LOGIN engedélyezettségét szabja meg, engedélyezés-tiltás a sorrend. A felsorolt címekről engedélyezett a hozzáférés, azaz lehetősége lesz a kapcsolódni próbáló kliensnek a felhasználói azonosításra, más gépekről azonnal visszautasítást kap.

Ugyanilyen fontos természetesen, hogy a szerver ne fusson rendszergazdai jogokkal; csak addig legyenek root jogai, amíg a 21-es portra rákapcsolódik, ezután azonnal mondjon le róla és egy, lehetőleg kis jogosítványokkal rendelkező, közönséges, valódi ember által nem használt felhasználó nevében fusson tovább.

Védelmi lehetőségek

Egy jó útmutató a támadónak, ha pontosan tudja, hogy mivel áll szemben. Például igen nagy segítség az a beszédes *proftpd*, amelyik a kapcsolódási kísérlet elején ezt mondja:

```
ProFTPD 1.2.4 Server (FTP) [ftp.xyz.com]
```

Innen a támadó máris tudja, hogy pontosan melyik FTP-szerver esetleges biztonsági hiányosságainak nézzen utána, ha be akar törni. Egy kézenfekvő elhárítása a problémának, ha a „ServerIdent Off" direktívát beállítjuk a *proftpd* konfigurációs állományában, ekkor a szerver már csak a következő üzenetet fogja a kliensnek küldeni:

```
220 ftp.xyz.com FTP server ready.
```

Egy ugyancsak jó, de kicsit gonoszabb módszer, ha meghamisítjuk a bejelentkező szöveget valamely más, létező szerver üzenetére, így hosszas és fölösleges elfoglaltságot keresve a behatolni kívánó illetőnek. Ez esetben viszont ügyeljünk arra, hogy még véletlenül se legyenek közösen vagy hasonló módon kihasználható biztonsági hibák a színlelt és a valós szerver esetében.

Gyakori, sokféle potenciális támadás ellen védő módszer az ún. *chroot jail* alkalmazása, amely azt takarja, hogy a szerverprocessz gyökérkönyvtára valamelyik belső, tipikusan kizárólag az FTP részére fenntartott alkönyvtár lesz. Ezt konkrétan a *proftpd* esetében a „DefaultRoot" direktívával lehet beállítani, aktivizálása pedig azzal az előnnyel jár, hogy bármi is történjék az FTP-szerverrel, bárki is törjön be programhibán keresztül vagy lopjon el egy jelszót, az illetéktelen behatolónak a mozgástere erősen korlátozva lesz az így megadott alkönyvtárra és az azon belüli könyvtárstruktúrára, illetve állományokra, mert a megadott gyökérnél magasabb szintre csak a jail-ből való nem könnyű kitöréssel kerülhet, enélkül más alkönyvtárakat sem tud elérni még rendszergazdai jogosultsággal sem.

Egyéb előnyös biztonsági beállítások

Az egyik alapvető általános védekezési mechanizmus, hogy egyszerűen tiltjuk a felhasználói szintű hozzáférést, és csak 'anonymous' elérést engedélyezünk. Ekkor nincs probléma sem a valódi felhasználók jelszavainak ellopásával, sem azzal, hogy FTP-ről bárkitől is félteni kellene a szerver felhasználóinak állományait. A legtöbb FTP-szerver, így a *proftpd* is, erre lehetőséget ad, de az is megvalósítható, hogy virtuális FTP-felhasználók legyenek, azaz névvel és jelszóval rendelkező elérési lehetőségek, amelyeknek azokban semmi közük a szerver általános célú felhasználóihoz.

Annak érdekében, hogy a „brute force" módszerrel dolgozó jelszótörők ne próbálkozhassanak hatékonyan az FTP-szerverünkön valamelyik felhasználó jelszavának kitalálásával, célszerű a próbálkozások számát maximálni. Ezt a *proftpd* konfigurációjában a „MaxLoginAttempts 3" direktívával állíthatjuk például 3-ra. A harmadik hibás jelszókísérlet után a szerver eldobja a kapcsolatot, ezen felül pedig a naplózási mechanizmuson keresztül a sikertelen kísérletet a naplóba is bejegyzi, amely jó támpontot nyújthat további biztonsági intézkedések foganatosítására.

Az 'anonymous' FTP-elérés kapcsán érdemes még ügyelni arra, hogy a feltöltés tiltva, ill. korlátozva legyen. A korlátok nélküli engedélyezés esetén könnyen előfordulhat, hogy a kliens megtölti a teljes tárterületet, ezzel a bosszúságon kívül rosszabb esetben a szerver működésének veszélyeztetését, jobb esetben csak más anonymous felhasználók feltöltési lehetőségének megszűnését okozva. Amennyiben a szerver feladata igényli, hogy a feltöltés engedélyezett legyen, azt mindenképpen célszerű egyetlen (tipikusan *incoming*) könyvtárra korlátozni.

6.1.5. NFS

Kockázati tényezők, általános sebezhetőségek

Működési elv

Az NFS használatakor két lépésben történik jogosultság-ellenőrzés. Az első a `mount` parancs kiadásakor, amelynek során a szerver ellenőrzi, hogy a kliens jogosult-e a kívánt alkönyvtár hálózaton keresztüli elérésére. Amikor a kliens jelzi a csatlakozási szándékát, a szerver megvizsgálja a /etc/exports fájl tartalmát, amelyben azon gépek neveinek vagy IP-címeinek listája van, amelyek jogosultak arra, hogy egy adott megosztott pontot elérjenek. Ha ezen az ellenőrzésen túljut a kliens, akkor a fájlrendszer szintjét látja a továbbiakban és a második lépésben már az NFS-nek szorosan a részét nem képező felhasználói jogosultság-ellenőrzés lép életbe, amelynek során a felhasználói- és a csoportazonosítók határozzák meg minden egyes fájlra, hogy elérhetők-e a kliens számára.

Kockázatok

A /etc/exports fájlban található lista név vagy IP-cím alapján azonosít. Ha a támadó hozzáférést szerez a névkiszolgálóhoz és átállítja a mutatót a saját gépére, vagy más módszerrel IP-címet hamisít (spoofing) magának, akkor ezt a mechanizmust kijátszhatja. Könnyű belátni, hogy akkor is illetéktelenek férhetnek hozzá az adatokhoz, ha egy, a bizalmi listában szereplő gép fölött uralmat szereznek. Ha a kiajánlás rw opcióval történt, akkor még írási jogosultsága is lesz a kliensnek.

Példa: 'gyula' felhasználó felhasználói azonosítója a szerveren 999. Gyula létrehoz egy fájlt, ami csak 'gyula' felhasználó számára hozzáférhető (ekvivalens a chmod 600 fájlnév parancs kiadásával). A kliensnek joga van azt az alkönyvtárat elérni, ahol ez a fájl tárolva van. A kliens gépén a 999-es felhasználói azonosító 'pista' felhasználóhoz tartozik. Ennek következtében, mivel a felhasználók az azonosítószámukkal vannak nyilvántartva, a kliens oldaláról 'pista' jogosult lesz elérni azt a fájlt, amit elvileg csak 'gyula'-nak lenne szabad. További gond, hogy ha a kliensgépen egy illetéktelen behatoló rendszergazdai jogosultságot szerez, akkor bármelyik kliensgépi felhasználó nevében tevékenykedhet, így a szerveroldalon is hozzáférhet több felhasználó állományaihoz.

Portmap

Az NFS által is használt szolgáltatás, a *portmap* ahhoz szükséges, hogy a kapcsolódni kívánó gép megtudja a szervertől, hogy mely portokon mely szolgáltatásokat találhat meg. Noha rendszeresen javítják, mégis egy állandó, potenciális biztonsági lyuk a szerveren, és fontos, hogy az NFS-hez hasonlóan csak a szükséges legszűkebb kör számára legyen elérhető.

Szerveroldal

A szerveroldali nfsd és mountd működése további kockázatokat rejt magában. Ha a kliensnek megengedjük, hogy rendszergazdai jogosultsággal kezdeményezzen kapcsolódást a szerver felé, akkor a 0-s felhasználói azonosító birtokában jelentkező kliens az egy-egyes megfeleltetés miatt a szerveren is rendszergazdai jogosultsággal végezhet fájlműveleteket. Könnyű belátni, hogy ha a kliens rossz kezekbe került, a behatoló ezzel a szerveren megosztott részen lévő, csak root által írható vagy olvasható állományokra is jogot szerez.

Kliensoldal

Vannak biztonsági megfontolások, amelyek a klienseknél is figyelmet érdemelnek. Ilyen például az, hogy a szerver sem tökéletesen megbízható. Ha a szerver gazdája egy root tulajdonú, setuid-os (azaz bárki által indítva root effektív azonosítóval futó) programot helyez el a kiosztott területen, és a kliensoldalon ennek a futtatása engedélyezett, akkor ezzel a kliens fölött is átveheti az uralmat. Még biztonságosabb, ha semmiféle szerveren tárolt program futtatását nem engedélyezzük a kliensen, azonban ez sok alkalommal olyan korlátokat építene, amelyek nélkül az NFS a kívánt célra nem használható, ezért indokolt esetekben ettől el lehet tekinteni.

Zárolások

Az NFS újabb megvalósításai teljeskörűen támogatják az állományok zárolását. Ezt az rpc.statd és az rpc.lockd programok kliensoldali futtatásának segítségével valósítják meg, azonban ezek a programok a *portmap*-re támaszkodnak. Így a portmaphez kapcsolódó kockázatok a kliensoldalra is érvényesek.

Védelmi lehetőségek

A kliensoldali root NFS-en keresztüli hozzáférésénél tiltani kell, hogy a felhasználói azonosítója a szerveren is érvényesüljön. Az elfoglalt kliensgép esetleges támadója azonban még akkor sem veszélytelen, ha ezt megtesszük, mert a su parancs használatával bármelyik felhasználó nevében próbálhat csatlakozni és így elérési jogot szerezni. Ezért rendkívül lényeges, hogy minden alapvetően fontos futtatható vagy nem futtatható állomány a root birtokában legyen, mert a kliensoldali root az egyetlen olyan felhasználó, akinek a próbálkozásai ellen be tudjuk biztosítani a szervert.

A felhasználói azonosítás UID-alapú egyeztetéséből adódó problémákat segíthet továbbá kiküszöbölni a 4-es verziójú NFS-ben megjelent új szolgáltatás, az *idmapd*, amelynek mind a kliens-, mind a szerveroldalon futnia kell, és az UID-ek és GID-ek névre fordítását végzi el szükség esetén, oda-vissza.

Ezzel párhuzamosan bizonyos operációs rendszereken futó, így például linux-alapú kliens esetén egyenesen rá is kényszeríthetjük a klienst arra, hogy csak rootként próbálkozzon: ha a kiosztás opciói közé a secure választást is fölvesszük, akkor a szerver nem fogja engedni, hogy az 1024-nél nagyobb számú portokról érkező kérések csatlakozzanak. Ugyanakkor, mivel a közönséges felhasználók csak ilyen portokra kötve tudnak kommunikálni, egyedül a rendszergazdának marad módja arra, hogy kapcsolódjon, őt pedig a fentebb ismertetettek alapján kordában tudjuk tartani.

A kliensoldali biztonság megteremtésének érdekében alkalmazhatjuk a nosuid opciót annak tiltására, hogy a szerverről setuid-os bináris futtatható legyen. Ha minden futtatást tiltani szeretnénk, akkor a noexec opciót használjuk.

Alapvető biztonsági követelmények

Az első és legfontosabb szempont mind a szerver-, mind a kliensprogram frissessége. Ezeken túl lényeges, hogy a *portmap* a *tcp_wrapper*rel védve legyen, azaz figyelembe vegye a *tcpd* által használt /etc/hosts.allow és /etc/hosts.deny állományok tartalmát a jogosultságok ellenőrzésekor. Hogy ez így van-e a mi esetünkben, arról információt nyerhetünk például a

```
strings `/sbin/portmap` | grep hosts
```

parancs kiadásával. Ha ebben a garanciát nem nyújtó módszerben nem bízunk, akkor érdemes valamilyen nyomkövető – például *strace* – programmal vizsgálni, hogy a portmap olvassa-e ezeket az állományokat. Ha az említett fájlnevek szerepelnek az eredményül kapott kiírásban, akkor a portmap korlátozható ezeknek az állományoknak a testreszabásával, mivel a távoli gépek hozzáférését ezeken keresztül jól lehet tiltani és/vagy engedélyezni. A lehetőséget kihasználva jegyezzük be az állományokba, hogy a portmaphez csak azokat a gépeket engedje hozzáférni, amelyeknek valóban szükségük van rá. Ügyeljünk azonban arra, hogy csak IP-címeket tegyünk a portmap soraiba ezekben az állományokban, mert a névfeloldás indirekt módon portmap-aktivitást is kiválthat, amely újabb névfeloldáshoz, majd újabb portmap-aktivitáshoz vezet, ezzel végtelen ciklust okozva.

Egyéb előnyös biztonsági beállítások

Védelem tűzfallal

Amennyiben lehetséges, az NFS-szolgáltatást nyújtó szervert védjük tűzfallal is. Könnyen megoldható, hogy a portmap 111-es és az *nfsd* 2049-es portjait csak azok számára tegyük elérhetővé, akiknek tényleg szükségük van rá. A többi kiszolgáló, a statd, mountd, lockd és rquotad – amelyek a portmap segítségével találnak szabad helyet maguknak – már nem

ennyire egyszerű eset. Érdemes ezeket úgy indítani, hogy fixen megadjuk nekik azokat a portokat, amelyekre ráülnek, ezeket pedig a tűzfalon már könnyen engedélyezhetjük a megfelelő gépek számára, hiszen nyilván az a követendő eljárás, hogy csak azokat a portokat engedjük, amelyeken szükséges a kommunikáció, minden más port a tűzfal által zárva van.

A *statd* és a *mountd* a -p kapcsoló után megadott porton fog figyelni, a *statd* esetében még egy -o kapcsoló és az azt követő portszám is szükséges, mert a kimenő kommunikációt a program az így megadott porton küldi. A *lockd*-t a kernel indítja, itt a modul (ha modulban van) vagy a kernel (ha statikusan van befordítva) betöltési paramétereként lehet megadni, hogy melyik portokat foglalja le. Küldjük a lockd-t példaként mind TCP, mind UDP tekintetében a 32768-as portra. Ha modulokat használunk, a /etc/modules.conf állományba írjuk be a következő sort:

```
options lockd nlm_udpport=32768 nlm_tcpport=32768
```

Ha a kernelbe statikusan van belefordítva a szolgáltatás, akkor adjuk a következő paramétereket a kernel indítási parancssorához:

```
lockd.udpport=32768 lockd.tcpport=32768
```

Az *rpc.quotad* segédprogramnak két verziója létezik, az egyik a *mountd*-hez hasonlóan támogatja a -p opciót. Érdemes ezt használni és a beállítási lehetőséggel élni.

Megfigyelés és elemzés

Ha a hosts.allow, hosts.deny, root_squash, nosuid opciókat használjuk a fixre rögzített porton használt segédprogramokkal együtt, valamint gondoskodunk arról, hogy mindig a lehető legkevesebb hibát tartalmazó, azaz többnyire a legfrissebb verziószámú programokat használjuk, akkor már majdnem biztonságban érezhetjük magunkat, ami az NFS-t illeti. Teljesen biztosak azonban sohasem lehetünk a dolgunkban, az NFS egy bonyolult rendszer, könnyen új hibák kerülhetnek napvilágra, ezért a folyamatos felügyeletről gondoskodni kell. Az NFS-szerver naplózza a sikeres és sikertelen csatlakozási próbálkozásokat, így a logok figyelésével jó képet kaphatunk a próbálkozó kliensekről és ennek megfelelően módosíthatjuk a beállításokat.

6.1.6. DNS

A DNS egy elosztott, hierarchikus tartománynév kiszolgáló rendszer. Lényegében két feladatot hajt végre, válaszol a kliensek kérdéseire, illetve maga is kérdéseket tesz fel a hierarchiában fölötte álló más DNS szervernek, ha a kliens által kért információ nem áll rendelkezésére. Az egyszerű lekérdezések az 53-as UDP porton, a hosszú zónatranszferek az 53-as TCP porton át zajlanak.

Kockázati tényezők, általános sebezhetőségek

A DNS működéséből következik, hogy kétféleképpen támadható. Az első eset az, amikor a kiszolgáló működését próbálják akadályozni úgy, hogy folyamatosan kérdésekkel bombázzák, illetve körmönfontan megfogalmazott kérdésekkel próbálják zavarba hozni, többletmunkára kényszeríteni. A másik lehetőség az, hogy megpróbálják félrevezetni úgy, hogy egy másik, a mi kiszolgálónk által adatforrásként használt DNS-t a támadó hatalmába keríti, vagy szándékosan rosszindulatú DNS-t üzemeltet, így az a mi DNS-ünket és ismereteiket tőlünk beszerző többi klienst is félrevezeti. Előfordulhat az is, hogy DNS-ünk kérdésére nem a valódi DNS válasza érkezik előbb, hanem egy hamis válasz.

Védelmi lehetőségek

A szolgáltatást akadályozó támadások ellen hálózatunk határán álló tűzfal segítségével védekezhetünk úgy, hogy az egy címről érkező kérdések számát korlátozzuk. Érdemes zónatranszfer korlátozást bevezetni, hiszen zónánként konfigurálhatjuk azt, hogy mely gépeknek adunk zónatranszfer jogot. Üzemeltethetünk külön DNS-t a belső, és a külső hálózat számára is.

A DNS félrevezetése, mérgezése ellen a DNSSEC technikával védekezhetünk. Lényegében arról van szó. hogy a zónafájlokat a szolgáltató DNS titkos kulcsával aláírja, így a megkapott zóna integritását a zónatranszfer után a nálunk lévő nyilvános kulccsal ellenőrizhetjük. A DNSSEC alkalmazásának részletes leírását a következő oldalon olvashatjuk: http://www.ripe.net/disi/dnssec_howto/

Alapvető biztonsági követelmények

Kövessük a BIND9 változásait, biztonsági frissítéseit. Ha a DNSSEC technikát is alkalmazzuk, úgy tartsuk be a kulcskezelés szabályait, vigyázzunk, nehogy aláírásra használt kulcsaink illetéktelen kezekbe kerüljenek.

Megfigyelés és elemzés

A BIND9 a /var/log/daemon.log fájlba naplóz. A naplózást a named.conf logging opciójával hangolhatjuk.

6.1.7. DHCP

A DHCP IP és UDP protokollt használ a kommunikációra, ami biztonsági kockázatot hordoz, ugyanakkor a DHCP nem tartalmaz biztonsági szolgáltatásokat. Ez azért komoly probléma, mert a DHCP alapvető konfigurációs adatokat közvetít a kliensek számára.

Kockázati tényezők, általános sebezhetőségek

A kockázati tényezők kétfélék. Az első típus az, amikor valaki jogosulatlan DHCP kiszolgálót telepít hálózatunkra, ami a klienseink kérésére válaszolva hamis adatokkal látja el azokat. Előfordulhat, hogy a támadó DHCP kiszolgálója megtéveszti a kliens gépeket, hogy azok az ő fennhatósága alatt álló routert használják a hálózat többi részével folytatott kommunikációra. A második esetben a támadó jogosulatlan DHCP klienst üzemeltet, amely trükkös DHCP kérésekkel áraszthatja el kiszolgálónkat, így elhasználva az allokálható IP címeket, vagy csak így szerez IP címet saját – rosszindulatú – céljaira.

Védelmi lehetőségek

A lehetséges védelem alapvető módszere az, hogy megakadályozzuk illetéktelenek fizikai hozzáférését hálózatunkhoz. Ez magában foglalja a hálózat alsó két rétegéhez való hozzáférést. Ha Wireless LAN-t is üzemeltetünk, akkor különösen vigyázzunk, nehogy illetéktelen csatlakozhasson rádiós LAN hálózatunkhoz.

Alapvető biztonsági követelmények

Ügyeljünk a hálózat tervezésénél, hogy az ne legyen könnyen hozzáférhető. Használjunk olyan hálózati eszközöket, amelyek éppen nem használt portjait ki lehet kapcsolni, és a csatlakozás legyen regisztrált MAC címhez köthető. A DHCP kiszolgálónk lehetőség szerint ne küldjön konfigurációs adatokat ismeretlen hálózati eszközöknek, csak ismert Ethernet címűeknek. A WLAN hozzáférésünk legyen MAC címhez kötött, és használjuk a rendelkezésre álló forgalomtitkosítási technikát.

Egyéb előnyös biztonsági beállítások

Adott esetben fontoljuk meg az IPSEC protokoll alkalmazását.

Megfigyelés és elemzés

A DHCP működése során a /var/log/messages fájlba naplóz. A kiadott kliens-cím adatokat a /var/lib/dhcp3/dhcpd.leases fájlban tartja, itt nézhetünk utána annak, mely gépek kaptak címet a kiszolgálónktól. A hálózati forgalomba a tcpdump programmal tekinthetünk bele.

6.1.8. LDAP

Az SLAPD címtár kiszolgáló a 389-es, illetve a 636-os TCP portokon fogadja a kéréseket. A 389-es az ldap://, a 636-os az ldaps:// port. A SLAPD támogatja az LDAPv3 protokollt, a Cyrus SASL implementációval erős azonosítási és adatbiztonsági SASL (Simple Authentication and Security Layer) technikával bír, az OpenSSL TLS program segítségével támogatja a tanúsítvány alapú hitelesítést és titkosítást, a címtár adatbázis(ok)hoz történő hozzáférés jól szabályozható és támogatja a Unicode használatát is.

Kockázati tényezők, általános sebezhetőségek

Az SLAPD címtár sok esetben központosított azonosításra használatos, de ha nem, a benne tárolt adatok akkor is fokozott védelmet érdemelnek. A tervezés során jól gondoljuk meg, hogy milyen adatokat teszünk bárki számára hozzáférhetővé, nehogy levélszemét terjesztők férhessenek hozzá kollégáink e-mail címeihez. Érdemes a címtár kívülről történő használatát ennek megfelelően erősen korlátozni, esetleg megtiltani.

Védelmi lehetőségek

A SLAPD használja a TCPD konfigurációs fájlokat, így a /etc/hosts.allow és a /etc/hosts.deny fájlokban a hosts_access(5) manlap alapján szabályozhatjuk a hozzáférést. Például:

```
slapd: 192.168.2.0/255.255.255.0 127.0.0.1 : ALLOW
slapd: ALL : DENY
```

Használjuk az LDAPS protokoll adta SSL titkosítási lehetőségeket, de a SLAPD az LDAP porton is támogatja a STARTTLS technikát. Az azonosításra használhatjuk a SASL keretrendszert is.

Alapvető biztonsági követelmények

Alaphelyzetben a SLAPD gépünk összes interfészén várja a lekérdezéseket. Erre nyilván nincs szükség, főleg kezdetben nem. Csak onnan fogadjunk el kapcsolatokat, ahonnan feltétlenül szükséges. Kísérleti stádiumban az ldap://127.0.0.1 elegendő lesz, ezt a /etc/default/slapd fájlban állíthatjuk be. Távoli elérést csak TLS fölött engedélyezzünk.

Egyéb előnyös biztonsági beállítások

A SLAPD-ben a műveletek engedélyezésénél figyelembe vehetjük a SSF (Security Strength Factor) faktort, így az érzékenyebb műveleteket csak kellőképpen biztonságos hitelesítés után lehet végrehajtani. A témával kapcsolatos részletes dokumentáció a következő címen érhető el: http://www.openldap.org/doc/admin23/slapdconf2.html#Access Control

A SLURPD segítségével az adatbázisainkat tarthatjuk több replikában is más kiszolgálókon, így, a rendszeres mentés mellett, még biztosabbak lehetünk abban, hogy a nehezen felépített adatbázisunk nem fog elveszni. A SLURPD folyamatosan nyomonköveti az adatbázis

változásait, és azokat rendszeresen továbbítja a replika kiszolgálóknak. A témával kapcsolatos részletes dokumentáció a következő címen érhető el: http://www.openldap.org/doc/admin23/replication.html

Megfigyelés és elemzés

A SLAPD a /var/log/syslog fájlba naplóz, működését itt kísérhetjük figyelemmel.

6.1.9. Dialin / Betárcsázás

Telefonvonalon történő modemes behívások fogadására az MGETTY és a PPPD programokat használjuk. Az MGETTY jól illeszkedik a különféle modemekhez, mert kellő rugalmassággal konfigurálható. Az MGETTY-ket az init indítja a /etc/inittab fájl alapján, minden soros vonalhoz egyet-egyet. Alább szemléltetésül az inittab egy részlete látható.

```
# Example how to put a getty on a modem line.
#
#T3:23:respawn:/sbin/mgetty -x0 -s 57600 ttyS3
T1:23:respawn:/sbin/mgetty ttyS1
T2:23:respawn:/sbin/mgetty ttyS2
T3:23:respawn:/sbin/mgetty ttyS3
```

Az MGETTY konfigurációs fájljai a /etc/mgetty könyvtárban találhatók. Az mgetty.config tartalmazza az általános és portonkénti beállításokat, a login.config a bejelentkezéssel kapcsolatos, a dialin.config a hívóazonosítással kapcsolatos beállításokat. Az MGETTY a naplófájljait a /var/log/mgetty könyvtárba készíti, a soros portokhoz tartozó naplót portonként mg_<portnév>.log néven, míg a visszahívás naplóját mg_callback.log néven.

Kockázati tényezők, általános sebezhetőségek

A betárcsázás – mint szolgáltatás – annyiban jelent nagyobb kockázatot, mint más, mondjuk az SSH szolgáltatás, hogy a telefonhálózaton folyó kommunikáció nincs kiegészítő titkosítással védve, így a kezdeti azonosítást esetleg le lehet hallgatni. A PPP kapcsolat kiépülése után a további rétegek gondoskodhatnak az adatforgalom biztonságáról. Értelemszerűen itt is követni kell a megfelelő komponensek, az MGETTY és a PPPD frissítéseket, és ha a felhasználó azonosítás a UNIX felhasználóazonosító-jelszó adatok alapján történik, úgy kellő bonyolultságú jelszavak használatát ösztönözni, illetve a betárcsázó felhasználókat a többiektől elkülönítve kezelni. A betárcsázó felhasználók tagjai a 'dip' csoportnak, hogy hozzáférhessenek a /dev/ppp eszközhöz. A visszahíváshoz az adott felhasználókat fel kell venni a 'dialout' csoportba, ami kivédhetetlen kockázatot jelent, mert így joguk lesz a soros portokat írni. Az mgetty ugyan folyamatosan figyeli a porton folyó forgalmat, de ebből esetleg baj is származhat.

Védelmi lehetőségek

A hívófél-azonosítás (CallerID) kézenfekvő megoldás, ha a helyi telefontársaság, vagy a telefonközpont és a modem támogatja ezt a lehetőséget. Ez esetben a /etc/mgetty/dialin.config fájlban beállíthatjuk, hogy milyen számokról fogadunk el hívást, milyenekről nem, fogadunk-e körzeten kívüli hívásokat, és szeretjük-e azokat a klienseket, akik nem árulják el számukat.

A visszahívás (CALLBACK) jó megoldás, ha a behívót mentesíteni akarjuk a behívás telefonköltsége alól. Ekkor a behívót azonosítás után az mgetty a beállított, vagy – kimondani is szörnyű – tetszőleges, az azonosítás után egy kérdésre válaszul beírt számon visszahívja. Az első megoldás abból a szempontból használható, hogy így majdnem biztosan egy állomásra korlátozhatjuk a behívót. A visszahívást használó felhasználót a 'dialout' csoportba is fel kell venni. Ez kockázattal jár, lásd az előző pontot.

Alapvető biztonsági követelmények

A felhasználói azonosítók kiadásával kapcsolatos szabályokat gondosan tartsuk be. A visszahívó szolgáltatásnál különösen ügyeljünk arra, hogy ne engedélyezzünk tetszőleges számra visszahívást. Ha szükséges, inkább hozzunk létre több azonosítót a különféle visszahívandó számoknak megfelelően.

Egyéb előnyös biztonsági beállítások

További hitelesítésre használhatjuk a PPPD-be épített PAP és CHAP eljárásokat.

Megfigyelés és elemzés

A működés során készülő naplófájlok jól követik az eseményeket. Például, a soros port írására tett kísérletnek ilyen nyoma van az MGETTY naplóban:

```
06/23 14:09:59 yS1  wfr: waiting for ``RING''
06/23 14:10:09 yS1  mdm_read_byte: read returned -1: Interrupted system call
06/23 14:10:09 yS1  wfr: timeout waiting for RING
06/23 14:10:09 yS1  huh? Junk on the line?
06/23 14:10:09 yS1   >>> could be a dial-out program without proper locking -
check this!
```

A PPPD a /var/log/messages fájlba írja naplóját. Egy bejelentkezés üzenetei így néznek ki:

```
Jun 19 10:53:47 ns pppd[5461]: pppd 2.4.1 started by a_ppp, uid 0
Jun 19 10:53:47 ns pppd[5461]: Using interface ppp0
Jun 19 10:53:47 ns pppd[5461]: Connect: ppp0 <--> /dev/ttyS2
Jun 19 10:53:50 ns pppd[5461]: user balu logged in
Jun 19 10:53:50 ns pppd[5461]: kernel does not support PPP filtering
Jun 19 10:53:51 ns pppd[5461]: found interface eth0 for proxy arp
Jun 19 10:53:51 ns pppd[5461]: local  IP address 192.168.2.1
Jun 19 10:53:51 ns pppd[5461]: remote IP address 192.168.2.3
Jun 19 10:53:51 ns pppd[5461]: CCP terminated by peer
Jun 19 10:53:51 ns pppd[5461]: Compression disabled by peer.
Jun 19 11:00:03 ns pppd[5461]: LCP terminated by peer
Jun 19 11:00:03 ns pppd[5461]: Hangup (SIGHUP)
Jun 19 11:00:03 ns pppd[5461]: Modem hangup
Jun 19 11:00:03 ns pppd[5461]: Connection terminated.
Jun 19 11:00:03 ns pppd[5461]: Connect time 6.3 minutes.
Jun 19 11:00:03 ns pppd[5461]: Sent 974724 bytes, received 160115 bytes.
Jun 19 11:00:03 ns pppd[5461]: Exit.
```

6.1.10. Távoli elérés

Kockázati tényezők, általános sebezhetőségek

A távoli elérés segédprogramjai a kezdetek óta sokat változtak. Először a *telnet*, később az *rsh* terjedt el, mára teljesen átvette a helyüket az *ssh*. Akár telnetet, akár rsh-t használni hatalmas biztonsági kockázat, ezeket a programokat mindenképp kerülni kell.

Lévén, hogy a távoli elérés jogosultság-ellenőrzéssel jár – amely általában felhasználói név és jelszó átküldését jelenti a távoli gépre –, alapvető fontosságú a hálózati forgalom titkosítása. Ezt sem a telnet, sem az rsh nem tudja megtenni, az ssh viszont számos szolgáltatást nyújt ezen a téren. Az ssh 1-es verziója azonban elavult és számos ismert támadás ellen nem nyújt kellő védelmet, ezért fontos, hogy a 2-es sorozatot használjuk, a továbbiakban is erre hagyatkozunk.

Mivel az ssh meglehetősen hatékony és tesztelt titkosítási módszereket használ, a vele kapcsolatos biztonsági kockázatok jelenlegi tudásunk szerint gyakorlatilag a kommunikációban részt vevő két gép helyi biztonsági kockázataira redukálódnak. Ha a kliens vagy a

szerver fölött már egy támadó átvette az uralmat, akkor naplózhatja a begépelt vagy hálózaton kapott jelszavakat. Ha az ellenőrzés esetleg nem jelszó-alapú, hanem nyilvános kulcsú titkosítással történik, egy illetéktelen kezekbe került titkos kulcs természetesen szintén aláássa a teljes mechanizmus megbízhatóságát.

Védelmi lehetőségek

A kézenfekvő, illetéktelen behatolás elleni általános hoszt-védelmi módszerek alkalmazása, a nem elavult titkosítási protokollok használata és a titkos kulcsok gondos kezelése esetén az ssh nem igényel további védekezést az ssh-szerver megfelelő beállításán kívül.

Alapvető biztonsági követelmények

Ugyan ki lehet kapcsolni az ssh titkosítását, de nyilvánvaló biztonsági okokból ezt soha ne tegyük meg. Általában megfelelőek a szerver (sshd_config) alapértelmunk kiemelt figyelmet néhány részre. Tiltsuk például a pusztán hoszt-alapú kliensellenőrzést, azaz az .rhosts állomány alapján történő elbírálást:

```
IgnoreRhosts yes
```

Ne legyen továbbá lehetséges belépni üres jelszóval (azaz jelszó nélkül):

```
PermitEmptyPasswords no
```

Természetesen a globális szerverkonfiguráció ne legyen sem írható, sem olvasható senki más, csak a root számára.

Egyéb előnyös biztonsági beállítások

Az ssh esetében nem csak a titkosítás kiemelten fontos, hanem a partnerek azonosítása is. Ezt ún. host key-k segítségével végzi a program, amelyeket az első kapcsolódás alkalmával eltárol az adott felhasználó saját könyvtárában, ha a globális listában nem találja meg. Jól kézben tartható azonban, ha minden olyan gép, amelyhez a felhasználók kapcsolódni kívánnak, a globális listában szerepel és ellenőrzött kulcsa van. Ezért a `/etc/ssh/ssh_known_hosts` fájlba célszerű eltenni az ellenőrzött kulcsokat, ezzel a felhasználók által okozott potenciális bizonytalanságot kiküszöbölhetjük. Ha valamelyik gépnek megváltozik a hostkulcsa (mert például újratelepítették és nem mentették el a régi kulcsot, az ssh pedig újat generált), akkor csak ezen az egy helyen kell frissíteni. Célszerű az itt tartott gépeket annyi alias-szal együtt tárolni, amennyivel csak várhatóan hivatkozni fognak rá a felhasználók, így jó eséllyel megtalálja majd az ssh a globális beállítások között a szükséges kulcsot.

A hoszt-alapú korlátozás az ssh esetében is hasznos eszköz, a /etc/hosts.allow és párja, a /etc/hosts.deny beállításaival az elérést szűkíthetjük a gyakran használt gépekre és ezzel az esetleges, ssh-ban később felmerülő biztonsági lyukak kihasználóinak idegen gépről jövő támadásait jó eséllyel visszaverhetjük.

Megfigyelés és elemzés

Az ssh bőbeszédű logot ír, ezt természetesen elzárva kell tartani a felhasználóktól. Az esetleges gépek, amelyek a napló tanúsága szerint felismerhetően ötletszerű, gyakoribb nevekkel és jelszavakkal próbálkoztak, vagy több alkalommal újrakapcsolódtak, kizárhatjuk a kommunikációból. Amennyiben fölmerül annak a gyanúja, hogy valamelyik gépre betörtek, mindenképpen vizsgáljuk meg az ssh-klienst illetve ssh-szervert, hogy sértetlen-e, mert a betörők egyik kedvelt tevékenysége az sshd lecserélése, majd a kapott jelszavak elnaplózása, így begépeléskor a saját jelszavunk is áldozatul eshet egy ilyen módosított sshd-nek.

6.1.11. Adatbázis-szerver

Kockázati tényezők, általános sebezhetőségek

Hangsúlyoznunk kell, hogy az adatbázisok biztonsági szempontból megkülönböztetett figyelmet érdemelnek, hiszen fontos, részletes adataink tárházai. Az adatok mennyisége és komplexitása okán, a bizalmasság szokványos sérülésének elkerülésén túl fontos szempontot jelent az adatok meglétének biztonsága, vagyis a biztonsági mentések kérdésköre is.

Természetesen egy adatbázis szerverre is jellemzőek azok az általános kockázati tényezők, amelyeknek többé-kevésbé szükségszerűen minden nyilvános szolgáltatás ki van téve, így pl. DoS támadás, vagy illetéktelen hozzáférési kísérletek.

A bizalmasság kérdéskörében külön kiemelendő, hogy – lévén e szerverek képesek egyszerre több felhasználó részére egymástól elkülönült adatokat tárolni – biztosítanunk kell azt, hogy a rendszer bizonyos adataihoz legitim módon hozzáférő felhasználók ne láthassák a többiek információit.

Ugyancsak a bizalmasság kérdéskörébe tartozik annak garantálása, hogy az adatbázis mentései is megfelelően biztonságos körülmények között kerüljenek tárolásra.

Védelmi lehetőségek

Védelmi lehetőségeink egyrészt a hosztok védelméről szóló fejezetben részletesebben taglaltakat foglalják magukban, másrészt – tekintettel arra, hogy az adatbázis rendszerek fejlesztői is tisztában vannak a biztonság fontosságával – az adatbázis programok biztonságos beállításait jelentik. Mivel a minta rendszeren a Postgresql adatbázis-szerver található, ezért a továbbiakban ezen keresztül mutatjuk be a lehetséges beállításokat.

Alapvető biztonsági követelmények

Legfontosabb kiindulási alapnak – mint általában – az tekintendő, hogy lehetőség szerint mindenki csak a számára szükséges jogosultságokkal rendelkezzen. Első lépésben magának a szerverhez való hozzáférésnek a lehetőségét kell a minimumra korlátozni. IP cím alapú korlátozásra ugyan lehetőség van a Postgresben is, mégis – ha tehetjük – ezeket a kapcsolatokat már a tűzfalon korlátozzuk, hiszen tudjuk, bármely korlátozást a lehető legkorábban érdemes bevezetni.

Elterjedt felhasználási mód, amikor az adatbázis-szerver és az azt használó programok egy hoszton helyezkednek el. Ebben az esetben érdemes a TCP/IP alapú hozzáférést teljesen letiltani, és csak a UNIX socketeken való kommunikációt engedélyezni. Ezt a `/etc/postgresql/postgesql.conf` fájlban, a `tcpip_socket = false` értékre való állításával érhetjük el.

A Postgresql hozzáférés szabályozása az `/etc/postgresql/pg_hba.conf` fájlban található. Itt adhatjuk meg, ki, milyen címekről (vagy címtartományokból), milyen azonosítási eljárás után férhet hozzá, melyik adatbázishoz. Figyeljünk arra, hogy a csomagkarbantartáshoz használt, – `postgres` nevű – felhasználó jogait ne korlátozzuk localhost-ról.

A következő szint már az adatbázison belüli egységekhez (object) való hozzáférést szabályozza. Ehhez a GRANT utasítást használhatjuk, amellyel tábla szintű szabályozásra nyílik lehetőség.

Egyéb előnyös biztonsági beállítások

A terheléses támadások elleni védekezéshez segítséget nyújthatnak a `postgesql.conf`-ban található, a kapcsolatok számára, a használt memória méretére, és a kérések optimalizálására vonatkozó beállítási lehetőségek.

Amennyiben a tábla szintnél szűkebb egységhez kívánunk jogosultságokat hozzárendelni (vagyis kiválasztott oszlopokhoz), akkor lehetőségünk van arra, hogy létrehozzunk egy új VIEW-t, majd ehhez adjunk jogosultságokat.

7. Hosztok biztonsága

Hosztok alatt olyan számítógépet értünk, mely lehet csak kliens feladatokra használt gép, de mégis lehetnek rajta szerver funkciók is, akár úgy, hogy a gép felhasználója részéről ez tudatos szolgáltatás, akár úgy, hogy nem (pl. benne volt az alap-telepítésben).

7.1. Általános biztonsági problémák

Mivel az elterjedt operációs rendszerek már szerverként is üzemelhetnek[5], ezért nehéz elválasztani a kliens-szerver vonal mentén az egyes gépek feladatait és beállításait. A felhasználók megtehetik, hogy egy kliensnek szánt gépre szerver programot telepítenek, így ezeket az eseteket már szabályzás szintjén kell rendezni, hogy ki miként teheti meg ezt.

Sok esetben azért fontos a szerver funkciók telepítésének elkerülése vagy a szolgáltatás kikapcsolása, mert amennyiben nem élünk ezekkel a lehetőségekkel (pl. a legfrissebb kutatási eredményeinket akarjuk a cikkíró társaknak elérhetővé tenni – ezt inkább egy dedikált szerveren tegyük), akkor nem szabad ezt a támadási felületet is meghagyni a rendszeren. Valójában nincs olyan elfogadható ok, ami miatt egy helyesen kialakított szolgáltatásokkal rendelkező hálózatban a kliensek 24 órás elérhetősége és szerver szolgáltatása indokolt lenne.

A szerver (szolgáló, kiszolgáló) szolgáltatások azokat is szolgálhatják, akik rossz szándékkal közelítenek, feltörik a szolgáltatást, és „jobbik" esetben tönkreteszik a munkánkat. Rosszabb esetben a gépünkre tiltott tartalmat, jogsértő anyagokat tesznek fel, és tudtunk nélkül szolgáltatunk a külvilág felé, amíg rá nem jönnek az illetékes eljáró szervek. Amennyiben látványosan lelassul a gépünk, hangosabban működik a merevlemezünk, akkor mi is gyanakodhatunk, hogy nem minden folyamat fut a mi érdekünkben.

Meg kell említeni, hogyha jóindulatúan minden szolgáltatást kikapcsolunk, akkor is futnak hálózatról támadási lehetőséget nyújtó rendszerszolgáltatások a Windows rendszerekben (ld. svchost, a Windows RPC felülete), és a Microsoft ezt „border security managementtel", vagyis saját tűzfallal próbálja egyre erőszakosabban (ld. SP2) megoldani.

7.2. Telepítési alapelvek

Nemcsak cégérdekből megfogalmazott szlogen vagy szabad szoftvereket használók jelmondata az, hogy „csak tiszta forrásból", hanem biztonsági és jogkövető szempontból is igaz. A jogi oldallal most nem foglalkozunk részletesen, az idevonatkozó jogszabály elérhető az érdeklődők számára. [Szerzői_jog] Ezen felül az egyes termékek telepítésénél szerepelnek a licenc-feltételek, melyek elfogadása akkor is kötelező ránk nézve, ha sokan nem is olvassák el, hogy mit fogadtak el...

Biztonsági szempontból azért fontos a gyártótól származó eredeti adathordozó és alkalmazás használata, mert az informatikai biztonság történetében sokszor volt rá példa, hogy a másolt adathordozón nem odavaló kártékony vagy hibás adat is volt. A szándékosság kérdése másodlagos, az adathordozó anyaghibás is lehet, de a nem megfelelő licenc miatt több gondunk lehet, mint egy megfelelővel (pl. terméktámogatás hiánya, „összeakadhatnak" a termékek, frissítés korlátozása[6], sérülhet az intézet hírneve stb.).

[5] Ebben segítségükre van a hardverek megnövekedett képessége is, így egy olcsó asztali számítógép is lehet szerver.

[6] Sajnálatos módon ez napirenden van, pedig a nem frissített rendszerek azoknak is gondot okoz, akik frissítik rendszerüket. Egy hasonlattal élve: *szeretné, ha a kóbor kutyákat is beoltanák veszettség elleni védőoltással?*

Nem hologramos (gyári készlet) esetében fontos a telepítőkészlet eredetiségének ellenőrzése (best practice: md5sum, PGP/GPG signature, de minimum a fájlméretek és dátumok összehasonlítása az eredetivel, erre remek eszköz akár a könyvtárlistázó parancs is).

Amennyiben megvan a jogtiszta vagy szabadon elérhető nyílt forrású eredeti telepítő készlet, úgy a legfontosabb telepítési folyamat megkezdése előtt az alapelvek a következők:

Megfelelően előkészített hardverek: új eszközök esetén amúgy is elő kell készíteni a rendszert, legalábbis nagyon ajánlott megtervezni az egyes lépéseket, mint a lemez particionálása (fdisk), formázása (format), melyeket manapság már a CD-ről indítható és telepíthető operációs rendszerek is támogatnak. Particionálásnál fontos, hogy DoS támadást ne tudjanak a felhasználók indítani, ezért a /home, /tmp, /samba külön partícióra kerüljön, ne a rendszerrel egy helyre (multiuser Windows szerverek esetén is!), esetleg quota használatát mérlegelni kell már ennél a lépésnél is.

A célnak megfelelő perifériák és környezet: telepítéskor legyen összerakva a gép minden olyan hardver eleme, amellyel később használni fogják (pl. megfelelő videokártya, meghajtók, nyomtató stb.). Ezzel elkerülhető, hogy később új biztonsági hardver elemmel bővítve a rendszert (USB kulcs, kártyaolvasó, ujjlenyomat olvasó stb.) újra szükség legyen kiegészítő elemek telepítésére, esetleges összeférhetetlenségek felbukkanására.

A rendszer céljának ismerete: tudnunk kell, hogy a leendő gép szerver lesz vagy kliens, milyen feladatokat fog ellátni (esetleg azt is, hogy melyeket biztosan nem, és melyeket lehet, hogy igen, de *még nem* stb.), így ezek alapján milyen az elvárt szoftveres felépítése az operációs rendszertől az alkalmazásokig beleértve a protokollokat azok implementációi által. Az is elképzelhető, hogy egy adott gépen többféle operációs rendszer is helyet kapna, ekkor az első pontban említett lépésnél jól megfontoltan kell a háttértárat felosztani (százalékos arány) vagy beosztani (lehet több darab is, vagy éppen rack-es megoldás is, és ekkor az előző pontra utalunk vissza a megfelelő perifériák fontosságára). meg kell tervezni az egyéb, nem standard erőforrások (CD/DVD író, digitalizáló vagy multimédia eszközök) hozzáférési szabályait is, mert ettől függhet a telepítésük. vannak olyan gépek, pl. biztonságkritikus szerverek, ahol a külső adattárolók (CD/DVD, floppy, pendrive, laplink kábel) csatlakoztatását is korlátozni kell (upload irányban spy/malware, download irányban /etc/shadow, de BIOS szinten is megköthető, hogy pl. a gép boot-olhat CD-ről vagy sem).

Szükséges és elégséges beállítások: alapelv, hogy csak azt és annyit telepítsünk, ami szükséges, és minél kevesebb legyen az olyan elem a rendszerben, ami „jó, ha van, vagy hátha jó lesz majd" alapon kerül bele[7]. Az alaprendszer telepítése során kerülni kell minden olyan extra szolgáltatást, amiről nem tudjuk, hogy mi az, de úgy érezzük, hogy majd kipróbálnánk, mert sokszor nem jutunk el addig, de a rendszeren ott van, helyet foglal, biztonsági problémák lehetnek benne, és ha szükség lesz rá, akkor nehezebb eltávolítani a rendszerből, mint feltelepíteni.

Tudjuk, hogy már tízes nagyságrendű géppark esetén is fontos a kényelmi szempont (pl. az deny,allow order nagyon kényelmetlen, mert minden felhasználó panaszkodik, hogy miért nem tud háttérképet állítani, ezt-azt elérni stb. Az allow,deny order megoldást használják a legtöbb helyen, de tudni kell, hogy az veszélyes, esetleg ez is egy mérlegelési tényezőként szerepelhet (kényelem, adminisztrációs igény vs. biztonság). Már telepítéskor konfigurálandó eszközök: RSBAC (vagy akármilyen access control system. ebben benne van a tűzfaltól kezdve a fájlműveleteken át az erőforrás- és perifériakezelésig minden), konkrét példák: windows NT right management, active directory beállítások, linuxos best practice

[7] Például az MS Office csomag is olyan már, hogy telepítéskor választhatunk a *telepítés*, *első alkalommal telepítendő*, *CD-ről induljon*, vagy a *ne kerüljön telepítésre* opciók közül.

példák: grsecurity (.net), selinux (nsa.gov/selinux), komplett OS: adamantix (debian+selinux).

Alaprendszer mentése, mentésből „telepítése”: Ha egyszer végigmentünk egy testre szabott (custom) telepítésen, és a rendszert megfelelőnek érezzük a kitűzött célokhoz, akkor érdemes elmenteni az egészet egy úgynevezett image-be. Ez felhasználható később a saját rendszer visszaállításához, vagy hasonló rendszerek gyors telepítéséhez is.

Telepítés és használat megfelelő jogkörei: a telepítéseket és a mindennapi használatot más azonosítót használva kell végezni. Az alaprendszer telepítését rendszergazdai jogokkal végezzük el, majd az első teendők egyike egy olyan felhasználó létrehozása, mely a mindennapi használatra való. A rendszergazda azonosítót a rendszer üzemeltetésére, felhasználók kezelésére és a jogosultságok beállításaira használjuk, míg a mindennapi azonosító lesz a személyes tevékenységhez használt azonosító.

Mentés és archiválás: a teljes rendszer vagy annak egy része (pl. felhasználói adatok, rendszerfájlok) rendszeres mentésre kerülhet, így adatvesztés esetén legalább az utolsó mentésig meglesznek az adatok. A mentés lehet ciklikus (napi mentés, a hetedik napon heti mentés, a harmincadik napon havi mentés), és így adott számú adatrögzítővel megoldható. Az archiválás egy adott állapot hosszú távú megőrzésére való (pl. a telepítés végeztével hogyan nézett ki a rendszer), így ez többnyire egyszer írható médiára készül, és dátummal ellátva archiválásra kerül.

7.3. Alapvető biztonsági követelmények

Léteznek olyan szabályok, melyeket ökölszabályoknak is szoktak nevezni, egyes esetekben elérhetők olyan képernyővédők, melyek ezeket ki is írják a képernyőre (persze kérdés a hatásfok, hiszen olyankor éppen nem néz oda az ember...). A teljesség igénye nélkül következzen pár alapszabály:

- „szükséges és elégséges”, azaz a minimalizálás elve, valamint a felesleges szolgáltatások leszedése
- patch kezelés: az egyes operációs rendszereken ez automatikusra is állítható, így egy megjelenő újabb frissítés vagy javítás minimális felhasználói bevonással elvégezhető
- szolgáltatások elkülönítése, izolálása (E-mail, Web) gyakorlat: chroot, uml, grsecurity
- redundáns szerver beállítása (a rendelkezésre-állást erősíti)
- autentikáció (a tudás-tulajdonlás-tulajdonság 3T-ből legalább kettő alkalmazása), hozzáférés-vezérlés (access control)
- gyenge alapbeállítások átállítása, már a tudatos telepítéstől
- accountability (logok figyelése és elemzése)
- hálózati forgalom vezérlése, figyelése, statisztikai elemzése, kiugró/szokatlan események elemzése és az okok kiderítése
- rendszeres mentés (napi, heti, havi)
- fizikai védelem (ld. még topológia is)
- humán védelem (megfelelő humánpolitika és oktatás), és végül:
- megfelelő szabályzat-politika (bevezetés, oktatás, számonkérés, felülvizsgálat stb.)

Más-más ökölszabály listák érhetők el különböző formákban (képernyővédők Windows rendszerhez, magyarázó változat a Biztostű portálon és a CERT.HU oldalain):

```
http://tinyurl.com/2wrcl
      (http://www.microsoft.com/technet/archive/community/columns/security
      /essays/10imlaws.mspx)
http://tinyurl.com/2a4re
      (http://www.microsoft.com/technet/archive/community/columns/security
      /essays/10salaws.mspx)
http://www.biztostu.hu/course/view.php?id=21
http://www.cert.hu/ismert/10altalanos/peldak.html
```

7.3.1. Windows XP megerősítése

A dokumentum célja, hogy a rendszergazdáknak a Windows XP munkaállomás beállításához segítséget nyújtson.

Az alábbi tanulmány a National Institute of Standards and Technology (NIST) 800-68 számú speciális kiadványa alapján készült. Az eredeti dokumentum címe: Guidance for Securing Microsoft Windows XP Systems for IT Professionals: A NIST Security Configuration Checklist. Azonban ez az anyag több vonatkozásban is eltér az eredeti tanulmánytól: részben lényeges fejezetek nem kerültek ide, mint például a telepítéssel, frissítéssel foglalkozó rész, valamint az alkalmazásokat elemző fejezet. A hangsúly itt a Windows XP lehetséges beállításaira került. Ugyanakkor bővebb is ez az anyag, mert bizonyos beállításokhoz magyarázatokat fűztek, s kiegészítették a magyar Windows XP-ben alkalmazott elnevezésekkel.

A dokumentumban lévő információkat különböző környezetben elhelyezkedő Windows XP munkaállomások, mobil számítógépek és telekommunikációs rendszerek megvédésére használhatók. A lehetséges környezeteket a következő rész ismerteti.

A melléklet az egyes környezetekhez tartozó biztonsági mintabeállításokat tartalmazza. A sablonok a National Security Agency (NSA), a Defense Information Systems Agency (DISA) és a Microsoft által kifejlesztett mintabeállításokon alapulnak. A sablonok a Center for Internet Security (CIS), DISA, NSA, Microsoft és a NIST biztonsági szakembereivel történő egyeztetés alapján készültek.

A Windows XP számos lehetőséget nyújt a sablonok telepítéséhez. A Security Configuration and Analysis Microsoft Management Console (MMC) „snap-in" funkciója kiválóan megfelel arra, hogy egy lokális gépen telepítse a sablont. Windows XP tartomány (domain) környezetben a Csoport házirend szerkesztő (Group Policy Editor) az erre alkalmas eszköz. A Microsoft a GPMC-t ajánlja a többszörös tartományban a csoportok beállításainak kezelésére. A biztonsági beállításoknál alkalmazott eszközök listája a 7.3.1.4 fejezetben található.

Az itt leírtakat sokan és alaposan tesztelték, de minden rendszer és környezet egyedinek számít, ezért a rendszeradminisztrátornak saját magának is tesztelnie kell azokat. A NIST Windows XP biztonsági sablonok kifejlesztését az motiválta, hogy egy sokkal biztonságosabb Windows XP munkaállomás konfiguráció szülessen. Azonban egyes beállítások a rendszer működőképességét vagy használhatóságát csökkenthetik, ezért óvatosan alkalmazza azokat. Tesztelési módszernek javasolható, hogy a rendszeradminisztrátor egy frissen telepített rendszerből induljon ki, fokozatosan építse fel a kívánt rendszert.

7.3.1.1. A Windows XP környezetei

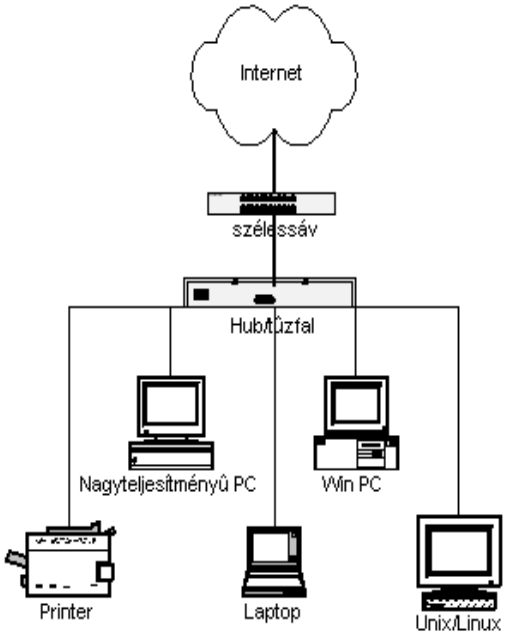
Ez a rész bemutatja azokat a környezeti típusokat, amelyekbe a Windows XP gazdagép kerülhet: otthoni illetve kisvállalkozási környezet (OKV), vállalati környezet vagy fokozott biztonságot igénylő hely. Egy negyedik környezeti típus, az elavult környezet is ide sorolha-

tó, amikor is a Windows XP speciális igények kielégítését vállalja fel, vagyis azt, hogy a korábbi szerverekhez és alkalmazásokhoz történő, időben visszafelé mutató kompatibilitásnak tegyen eleget.

Feltétlenül meg kell említeni, hogy minden környezetben szükséges a biztonságot érintő dokumentumok elkészítése is, ilyenek mint a Elfogadható felhasználás irányelvei (acceptable use policy), a biztonsági tudatosság növelését célzó dokumentumok stb.

Otthoni környezet illetve kisvállalkozások számítógépei (OKV)

Az OKV kis, hétköznapi, egyedülálló számítógépet jelent, amit üzleti célra használnak. Az OKV ugyanakkor sokféle környezetet jelenthet, kezdve a csak alkalomszerűen munkára használt otthoni géptől egy cég kisebb részlegénél munka céljaira alkalmazott gépekig, amelyet technikai vagy üzleti okokból nem távolról kezelnek. Az alábbi ábra egy tipikus OKV környezetet mutat:



7.1. Ábra: Tipikus OKV hálózati architektúra

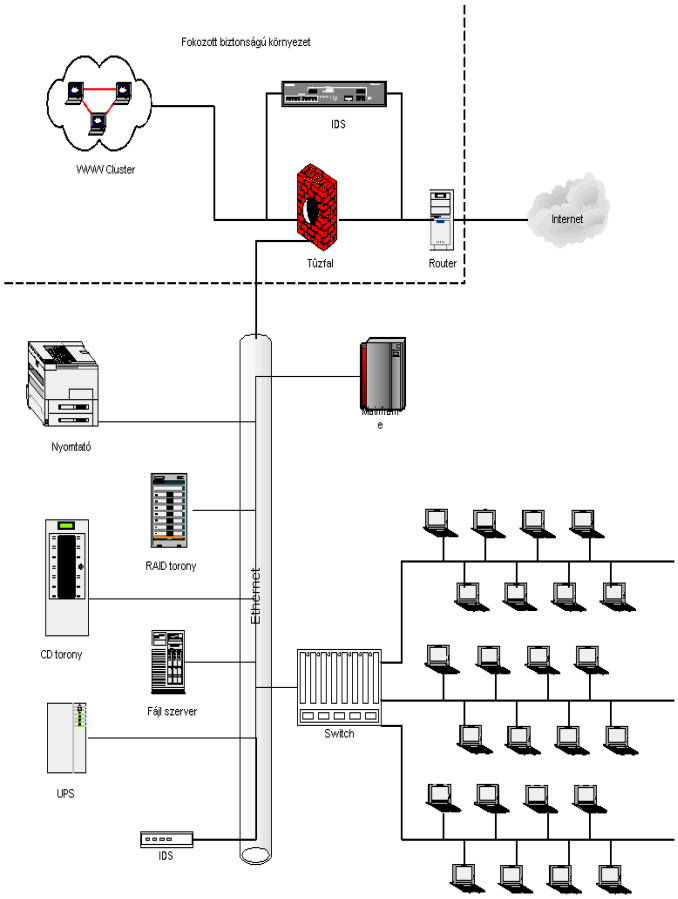
Általában az OKV környezet szokott a legkevésbé biztonságos lenni. Ugyanis az OKV rendszer adminisztrációjával foglalkozók keveset tudnak, keveset foglalkoznak a biztonsággal, sokkal inkább a működőképességet tartják szem előtt. Ilyen környezetben nem mindig alkalmaznak biztonsági szoftvereket, például vírusölőt, személyes tűzfalat. De a hálózatvédelemmel sem mindig törődnek, tehát az OKV-s gépek közvetlenül ki van téve külső támadásoknak. És ezért sokszor támadási célpontokká is válnak, nem a rajtuk lévő információ megszerzése miatt, hanem inkább további támadások kiindulópontjául választják ezeket, vagy férgek által okozott károk mellékszereplőivé válnak.

Az OKV környezetben az elsődleges *fenyegetések* kívülről várhatók, ezért kevésbé szigorú felhasználói házirendet kell megfogalmazni, mint egy vállalati vagy fokozott biztonságú környezetben. A támadások főleg a hálózati szolgáltatások ellen irányulnak, vagy rosszindulatú programoknak (vírusok, férgek) a számítógépekre való felvitele történik. A támadások többnyire a rendszer elérhetőségét bénítják (rendszer-összeomlás, hálózati sávszélesség csökkenése, működőképesség korlátozása), de érintheti a sértetlenséget is (adatfájlok megfertőzése) és a bizalmasságot is (érzékeny adatok, elektronikus levelek távolról történő elérése).

Az OKV biztonsága a kicsi, olcsó, hardver-alapú tűzfal útvonalválasztók (routerek) elterjedésével javulni fog, mivel azok bizonyos mértékig a mögöttük lévő gépeket is védik. Személyes tűzfalak (BlackICE, ZoneAlarm, Internet Connection Firewall) szintén javítanak a helyzeten. Azonban az OKV megerősítésében kulcsfontosságú szerepet játszik a sebezhetőségek megszüntetése a megfelelő javítások telepítésével, a felesleges funkciók kiiktatásával, a beállítások megváltoztatásával.

A vállalati környezet

A vállalati környezet tipikusan egy menedzselt környezet, ami a hardver és szoftver konfiguráció szempontjából is strukturált. Általában egy kijelölt csoport felelős a biztonságért. A gyakorlott munkatársak és a megtervezett környezet biztosíték arra, hogy mind a telepítésnél, mind pedig a működtetésnél a biztonság prioritást kap. A vállalati környezetre a tartományi modell a jellemző, mert abban hatékonyan végezhetők el a beállítások és az erőforrások megosztása (nyomtató, fájl szerver). A vállalatoknál csak a normál működéshez szükséges szolgáltatásokat engedélyezik, az egyéb, kockázatot jelentő alkalmazásokat törlik. A jogosultságot, a jelszavakat, a házirendek meghatározását központilag végzik, így az egész szervezeten belül azonos elvek érvényesülnek.



7.2. Ábra Tipikus vállalati környezet architektúra

A vállalati környezet sokkal inkább korlátozó, mint az OKV környezet. Többszintű védelmi rendszert szoktak kialakítani (tűzfalak, vírusölők, behatolásérzékelő rendszerek). Vállalati környezetben általában nem merül fel az elavult eszközökkel, rendszerekkel való együttműködés kérdése.

Az ilyen rendszereket belső és külső támadás is fenyegeti. Belső támadásra tipikus eset, amikor valaki a másik számítógépét jogosulatlanul használja. A külső támadás nemcsak a hálózaton kívülről érkezhet, hanem belső felhasználótól is, amikor a szervezet hálózatán keresztül egy belső rendszert támad (a strukturáltság miatt lehetséges). A leggyakoribb támadási mód, amikor a külső féltől rosszindulatú csomag (SPAM, kémszoftver, stb.) érkezik. A hálózati szolgáltatások elleni támadás ritkább, de belső és külső fél is okozhatja. Mindkét módszer a biztonság három alapelvét: a rendelkezésre-állást, a titkosságot és az adatok sértetlenségét is érintheti.

Fokozott biztonságú környezet

A fokozott biztonságú környezet az a környezet, amikor a támadás vagy az adatok napvilágra kerülésének kockázata igen nagy. Az ilyen környezet lehet hálózatba kötve, de állhat önmagában is. Tipikusan olyan számítógépek vannak itt, amelyek bizalmas információt tartalmaznak (személyes adatok, orvosi vagy pénzügyi rekordok, stb.) és ezeken valamilyen műveleteket végeznek (számlázás, forgalomirányítás, stb.) Elsősorban külső támadás érheti, de a belső sem kizárt. Az ilyen környezet lehet egy vállalati vagy OKV környezet része is (pl. a pénzügyi osztály rendszere, vagy egy mozgó munkatárs laptopja az otthoni hálózatban).

A fokozott biztonságú környezetnek ugyanazokkal a fenyegetésekkel kell szembe nézniük, mint a vállalatoknál. A kockázatok és az eredményes támadások következményei miatt azonban kevesebb funkciót szabad megengedni, és a beállításoknál szigorúbban kell eljárni. Itt még inkább szükség van tapasztalt biztonsági szakemberekre, aki a korlátozások mögötti tartalmat is érti.

Elavult környezet

Az elavult környezetekben régi rendszerek vagy alkalmazások futnak, amelyek idejétmúlt kommunikációs eljárásokat használnak. Emiatt aztán az ilyen környezetben működő rendszerek – biztonsági szempontból – sokkal nyitottabbak, s egyúttal védtelenebbek is. Ilyen környezet akár vállalati, akár otthoni környezetben előfordulhat, fokozott biztonságot igénylő környezetben nem valószínű.

7.3.1.2. Windows XP mintabeállításainak áttekintése

A biztonsági beállítások kerülnek felsorolásra ebben a fejezetben, melyek a NIST mintabeállításai között kaptak helyet. A beállításokat hét csoportba osztjuk: azonosítók és jelszavak, helyi házirend, eseménynaplózás, kötött csoportok, rendszerszolgáltatások, fájl jogosultságok, rendszerleíró adatbázis.

Minden csoportnál az is megtalálható, hogy milyen eszközökkel végezzék el a beállítást. A javasolt értékek a 7.3.1.3 fejezet táblázataiban vannak. Az eszközök pedig az 7.3.1.4 fejezetben találhatók.

A Windows XP magyar és angol változatában használt kifejezéseket *dőlt betűvel* jelöljük. A '/' a lehetséges választást jelöli.

Azonosítók és jelszavak beállításai

Először a Fiókházirend (Account Policies) beállításait tekintjük át, ezen belül is a jelszavakat (*Jelszóházirend – Password Policy)*

A helyes jelszavak kiválasztására és használatára nemcsak a felhasználókat kell megtanítani, hanem az operációs rendszerben a jelszavakhoz kapcsolódó paramétereket úgy ajánlatos beállítani, hogy az az erős jelszavak használatát segítse elő. Így a jelszó kitalálásának

vagy feltörésének valószínűsége lényegesen csökkenni fog. A következő paraméterek beállítása kívánatos:

- *Jelszó maximális élettartama - Maximum Password Age***:** ez a felhasználót a rendszeres jelszócserére készteti. Minél kisebb ez az érték, annál valószínűbb, hogy gyenge jelszót választanak a felhasználók, mert azt könnyebb megjegyezni. Minél nagyobb az érték, annál inkább veszélyeztetett a jelszó, vagyis nagyobb az esélye, hogy jogosulatlanok megismerjék.
- *Jelszó minimális élettartama – Minimum Password Age*: ez a beállítás arra vonatkozik, hogy a felhasználónak hány napot várnia kell, mielőtt újra megváltoztathatja a jelszavát. Sokan a jelszó maximális élettartamának lejárta után megváltoztatják a jelszavukat, majd azonnal vissza is változtatják a régire. Ezt küszöböli ki a „jelszó minimális élettartamának" beállítása. Hátránya, hogy azokat a felhasználókat is korlátozza, akiknek új jelszava napvilágra került. Ilyenkor adminisztrátori beavatkozás szükséges.
- *Legrövidebb jelszó – Minimális Password Length:* a jelszó minimális hossza karakterekben megadva. Ennek hátterében az a meggondolás áll, hogy hosszabb jelszót nehezebb kitalálni/feltörni. Hátránya viszont, hogy a hosszabb jelszót nehezebb megjegyezni.
- *A jelszónak meg kell felelnie a bonyolultsági feltételeknek – Password Must Meet Complexity Requirements:* a minimális jelszóhosszhoz hasonlóan ez is a jelszó kitalálását nehezíti. A bonyolultság ellenőrzése azt jelenti, hogy a jelszó nem tartalmazza a felhasználó azonosítóját, és különféle karakterek (kis- és nagybetű, szám, speciális karakter) keverékéből áll össze.
- *Előző jelszavak megőrzése – Enforce Password History:* a korábban használt jelszavak megőrizhetők a rendszerben, a megadott számú jelszó kerül tárolásra. Előnye, hogy a felhasználó korábbi jelszavait nem alkalmazhatja. Hátránya, hogy a régi jelszavak – a tárolás miatt – esetleg nyilvánosságra kerülnek.
- *A tartomány összes felhasználója jelszavának tárolása visszafejthető titkosítással – Store Password Using Reversible Encryption for All Users in the Domain:* ha ezt a lehetőséget beállítják, akkor a jelszó visszakódolható formában kerül tárolásra, ami veszélyes lehet. Csak akkor állítsák be ezt a módot, ha egy régebbi autentikálási protokoll támogatása (CHAP – Challenge Handshake Authentication Protocol) szükséges.

Azonosítók (Fiókzárolási házirend – Account Lockout Policy Settings)

A támadók gyakran a felhasználók azonosítóját használják a jelszavak kitalálásához. A Windows XP-ben az azonosító letiltható, ha túl sok sikertelen kísérlet történik egy megadott időn belül. A NIST mintában a következő paramétereket állították be:

- *Fiókzárolás küszöbe – Account Lockout Threshold:* a hibás kísérletek maximális számát mutatja ez az érték, ezután az azonosítót ideiglenes letiltják.
- *Fiókzárolás időtartama – Account Lockout Duration:* az azonosító ideiglenes letiltásának idejét adja meg. Gyakran rövid, de azért lényeges időtartamra állítják be (pl. 15 perc), két okból. A jogos felhasználónak, aki véletlenül zárta ki magát a használatból, csak 15 percet kelljen várnia az újbóli belépésre, ahelyett, hogy az adminisztrátort kellene megkérnie az azonosító újbóli megnyitására. Másodszor, azok, akik a jelszót kívánják felderíteni, általában tömegesen próbálják ki a jelszavakat, és így ők egyszerre csak viszonylag kevés jelszóvak kísérletezhetnek, a folytatáshoz pedig 15 percet várniuk kell. Ez a beállítás csökkenti az ún. „brute force attack", azaz a nyers erő használatát.

- *Fiókzárolási számláló nullázása - Reset Account Lockout Counter After:* ezt a beállítást a „Fiókzárolás küszöbe"-vel együtt alkalmazzák. Ha például a „kísérletek száma" 10, és ez a paraméter 15 perc, akkor 15 percen belüli 10 hibás jelszó megadása után az azonosítót letiltják.

Az azonosítók és jelszavak beállításának nagy kérdése, hogy hogyan lehet egyensúlyt teremteni a biztonság, a működőképesség és a használhatóság között. Például ha egy azonosítót néhány hibás jelszó megadása után kizárunk, ez ugyan megnehezíti a jogosulatlan belépést, de meglehetősen megnöveli az azonosítóval foglalkozó adminisztrátor munkáját a véletlen elütések miatti kizárások visszaállítása miatt. Másrészt emiatt a felhasználók papírra fogják leírni a jelszavukat, illetve könnyen megjegyezhető jelszavakat alkalmaznak. Ezért alaposan végig kell gondolni a beállításokat.

A beállítás módja:

Start -> Vezérlőpult -> Felügyeleti eszközök -> Helyi biztonsági beállítások -> Fiókházirend -> Jelszóházirend illetve Fiókzárolási házirend

Start –> Control Panel -> Administrative Tools -> Local Security Policies -> Account Policies -> Password Policy illetve Account Lockout Policy

Helyi házirend

A *helyi házirend (local policy)* három témával foglalkozik: a naplózással, a felhasználói jogok kezelésével valamint a biztonsági beállításokkal.

A naplózás (Naplórend – Audit Policy)

A Windows XP hatékony eszközt biztosít a rendszer megfigyelésére, ellenőrzésére. A megfigyelés módszere a naplózás (log), a rendszeradminisztrátorok ezeket átnézve kiszűrhetik a jogosulatlan tevékenységeket. A naplók az incidensek felderítésénél is hasznos eszköznek bizonyulhatnak. A naplók rögzíthetik a bejelentkezéseket/kilépéseket, az azonosítók kezelését, a címtár-hozzáférést, a házirendek változtatását, a rendszerjogok módosítását, a folyamatok nyomon követését, stb. amint az az alábbi táblázatban látható. Minden naplózási fajtánál beállítható, hogy a sikeres, a sikertelen vagy mindkét típusú események bejegyzésre kerüljenek, vagy semmi se kerüljön bejegyzésre. A bejegyzéseket az *Eseménynapló (Event viewer)* segítségével nézhetjük meg.

Ellenőrzés beállítása	Leírás (Bejegyzés készülhet, ha...
Bejelentkezés naplózása - Audit account logon events	a felhasználó erről a munkaállomásról egy távoli gépre be- vagy kilép.
Fiókkezelés naplózása - Audit account management	felhasználói vagy csoport-azonosítót hoznak létre, módosítanak vagy törölnek; felhasználói azonosítót átneveznek, letiltanak vagy visszaállítanak; jelszót állítanak be vagy módosítanak.
Címtárszolgáltatás-hozzáférés naplózása - Audit directory service access	a címtár egy olyan objektumához férnek hozzá, amelynek saját rendszer-hozzáférési listája (SACL) van.
Fiókkezelés naplózása - Audit logon events	a felhasználó be- illetve kijelentkezik, vagy a lokális géppel hálózati kapcsolatot épít.
Objektum-hozzáférés naplózása Audit object access	a felhasználó saját rendszer-hozzáférési listával (SACL) rendelkező objektumhoz (fájl, könyvtár, rendszerleíróadatbázis-kulcs vagy nyomtató) fér hozzá. A sikertelen hozzáférések is bizonyos helyzetben normálisak lehetnek. Óvatosan használja!
Házirendváltozás naplózása - Audit policy change	a felhasználói jogokat, naplózást vagy egyéb házirendet érintő változtatás történik.

Ellenőrzés beállítása	Leírás (Bejegyzés készülhet, ha...
Rendszerjogok használatának naplózása - *Audit privilege use*	a felhasználó a jogait gyakorolja. Nagyon sok bejegyzés kerülhet a naplóba!
Folyamatok nyomon követésének naplózása - *Audit process tracking*	egy program elindul, egy folyamat leáll, duplum kezeléskor vagy közvetett módon férnek hozzá egy objektumhoz.
Rendszeresemények naplózása *Audit system events*	a felhasználó a számítógépét indítja/leállítja vagy a rendszer biztonságát és a biztonsági bejegyzéseket érintő esemény történik.

2. Táblázat: Eseménynapló beállításai.

Felhasználói jogok kiosztása – User Right Assignment

A felhasználói jogok kiosztása meghatározza, hogy melyik csoportnak (adminisztrátor, felhasználó stb.) milyen jogosultsága legyen. A cél itt az, hogy minden csoport csak annyi jogot kapjon, amennyi feltétlenül szükséges, a felhasználók pedig abba a csoportba kerüljenek, ahol csak a nekik éppen szükséges jogok vannak. Ez „legkevesebb jog" elve. A jogok sokfélék lehetnek: a helyi vagy távoli rendszerekhez történő hozzáférés, mentések végrehajtása, a dátum/időpont megváltoztatása, naplók kezelése, a rendszer leállítása.

A 6. táblázatban szereplő bejegyzések többsége az elnevezés alapján beazonosítható. Némelyik szerepe talán első olvasatra nem egészen világos. Ezekhez egy rövid magyarázat található itt (a név után a 6. táblázatban lévő sorszám található. A kiválasztás szempontja önkéntes):

- Az operációs rendszer részeként való működés (2): Egy processz számára lehetséges, hogy bármelyik felhasználó nevében azonosíthassa magát, s így gyakorlatilag ugyanazokat az erőforrásait érje el, amit a felhasználó.
- Szülőkönyvtár-jogosultság mellőzése (7): Ez a jog azt biztosítja, hogy a felhasználó végigmehet a könyvtárak fa szerkezetén, anélkül, hogy bármilyen joga lenne a könyvtárakban. Tehát nem listázhatja ki egy könyvtár tartalmát, csak átkelhet rajta.
- Lapozófájl létrehozása (9): Melyik felhasználó illetve csoport hozhatja létre a lapozófájlt, illetve módosíthatja a méretét.
- Token objektum létrehozása (10): Egy hozzáférési token létrehozására szolgáló jog. Az Active Directory-ban a felhasználó hitelesítése után a helyi biztonsági rendszer létrehoz a felhasználó számára egy tokent, azaz egy olyan speciális csomagot, amely tartalmazza a felhasználó nevét, SID-jét, a felhasználót tartalmazó globális csoportok SID-jeit (tartományvezérlőtől származik), a felhasználót tartalmazó lokális csoportok SID-jeit (helyi gépről származik), rendszerszintű jogosultságokat.
- Számítógép- és felhasználói fiókok megbízhatóságának engedélyezése (19): Ez a beállítás lehetővé teszi, hogy egy felhasználó egy másik felhasználót vagy objektumot megbízhatónak minősítsen, azaz a saját jogait továbbadhassa egy másik felhasználónak, objektumnak. (Beállítsa a „Trusted for Delegation" opciót.) Például egy számítógépen futó szerverprocessz esetén – amit a kliens megbízhatónak minősített – a szerverprocessz a kliens számítógépén lévő erőforrásokhoz a kliensre beállított jogokkal hozzáférhet.
- Biztonsági naplózás létrehozása (21): Meghatározza azokat a fiókokat, amelyeket használó processzek képesek a biztonsági naplóba bejegyzéseket tenni.
- Számítógép eltávolítása tokjából (34): Meghatározza, hogy egy felhasználó a laptopját bejelentkezés nélkül kiszedheti-e a dokkoló állomásról.

- Folyamat token lecserélése (35): Ennek a jogosultságnak a tulajdonosa kezdeményezhet egy olyan processzt, amely kicseréli egy másik futó processz hozzáférési tokenjét.
- Könyvtárszolgáltatás-adatok szinkronizálása (38): Meghatározza, hogy melyik felhasználónak illetve melyik csoportnak van joga a directory service adatainak szinkronizálására. Ez „Active Directory" szinkronizálás néven is ismert.

Biztonsági beállítások – Security Options

A helyi házirend kialakításánál a már korábban említettek mellett további beállítások is lehetségesek, ezekkel az ún. biztonsági beállításokkal jobb biztonsági körülményeket tudunk kialakítani, mint az alapértelmezéssel. A NIST minta több tucat ilyen beállítást ismertet, például: az üres jelszó használatának letiltása, a rendszergazda és a vendég azonosító átnevezése, a floppyhoz és a CD-ROM eszközhöz távolról való hozzáférés korlátozása, egy tartományon belül a biztonságos csatorna kódolása, a bejelentkezési képernyő biztonságosabbá tétele (az előző azonosító elrejtése, figyelmeztető felirat megjelenítése, a jelszó lejárati ideje előtt a felhasználó figyelmeztetése), bizonyos típusú hálózati hozzáférés korlátozása, a hitelesítés típusának meghatározása (pl. NTLMv2).

A 7. táblázatban szereplő bejegyzések többsége az elnevezés alapján beazonosítható. Némelyik szerepe talán első olvasatra nem egészen világos. Ezekhez egy rövid magyarázatot talál itt (a név után a 7. táblázatban lévő sorszám található. A kiválasztás szempontja önkéntes):

- Eszközök: Dokkolás megszüntethető bejelentkezés nélkül is (9): Ez a biztonsági beállítás azt határozza meg, hogy egy hordozható számítógépet (laptopot) bejelentkezés nélkül is le lehet-e kapcsolni a dokkoló állomásról. Ha ez a beállítás engedélyezve van, bejelentkezés nélkül egy külső hardveres (kidobó) gomb hatására lekapcsolható a gép. Ha tiltva van, akkor először a felhasználónak be kell jelentkeznie, és a „Számítógép eltávolítása tokjából / Remove computer from docking station" privilégiummal kell rendelkeznie, hogy lekapcsolhassa a gépét.
- Interaktív bejelentkezés: A munkaállomás zárolásának feloldásához tartományvezérlői hitelesítésre van szükség (28): A Windows XP tárolhatja a felhasználók bejelentkezési információit, azért, hogy ha a tartományvezérlő elérhetetlen, a felhasználó akkor is be tudjon jelentkezni. A fenti beállítástól függően – a tartományvezérlő kiesése esetén is – sikeres lehet a bejelentkezés.
- Interaktív bejelentkezés: Viselkedés intelligens kártya eltávolításakor (31): Ez a beállítás meghatározza, hogy mi történjen, ha egy bejelentkezett felhasználó az intelligens kártyaolvasóból eltávolítja a kártyát. Lehetséges változatok: nincs művelet, munkaállomás zárolása, kényszerített kijelentkezés.
- Hálózati hozzáférés: A névtelen helyazonosító-/névfordítás engedélyezése (39): Ez a biztonsági beállítás megadja, hogy egy névtelen felhasználó lekérdezheti-e egy másik felhasználó azonosítójának (biztonsági azonosítójának - SID) attribútumait. Ha beállítják, akkor az a felhasználó. aki ismeri az adminisztrátor SID-jét, a számítógéppel kommunikálhat és a SID segítségével az adminisztrátor nevét lekérdezheti.
- Hálózati hozzáférés: SAM fiókok névtelen felsorolása nem engedélyezett (40): Meghatározza, hogy egy névtelen felhasználó kilistázhatja-e a SAM fiókokat. A Windows bizonyos tevékenységeket, mint pl. a tartományfiókok neveinek listázását és a hálózaton történő megosztását, még a névtelen felhasználóknak is megengedheti. Erre például akkor lehet szükség, ha egy adminisztrátor egy másik megbízható tartományban a felhasználóknak jogokat szeretne adni, s ez a tartomány nem tekinti megbízhatónak a másik tartományt.

(SAM: Minden Windows szerveren vagy munkaállomáson van egy helyi SAM (Security Account Manager) adatbázis, amely a helyi felhasználók fiókjait illetve a csoportfiókokat tartalmazza.)

- Hálózati hozzáférés: A névtelen felhasználókra a Mindenki csoportra vonatkozó engedélyek vonatkoznak (43): Meghatározza, hogy milyen további jogosultságokat biztosítanak a számítógéphez csatlakozó névtelen felhasználóknak.
- Hálózati hozzáférés: Névtelenül elérhető Named Pipe-ok (44): Meghatározza, hogy melyik kommunikációs munkamenetnek (pipe) vannak olyan tulajdonságai illetve engedélye amely lehetővé teszi a névtelen felhasználói hozzáférést.
- Hálózati hozzáférés: Távolról elérhető rendszerleíró elérési utak (45): Ez a biztonsági beállítás meghatározza, hogy a rendszerleíró adatbázis melyik ágát lehet hálózaton keresztül elérni, függetlenül attól, hogy a felhasználó vagy a csoport szerepel-e a winreg rendszerleíró kulcs hozzáférési listáján.
- Hálózati hozzáférés: Névtelenül elérhető megosztások (46): Ez a biztonsági beállítás megadja, hogy melyik megosztást érhetik el a névtelen felhasználók.
- Hálózati hozzáférés: Megosztási és biztonsági modell helyi felhasználói fiókok számára (47): Meghatározza, hogy a helyi fiókokat használó hálózati bejelentkezések hogyan legyenek naplózva. Ha klasszikus beállítást használnak, akkor a helyi fiók hitelesítő adatait használó hálózati bejelentkezést a hitelesítő adatokkal naplózzák. Ha csak vendég beállítást alkalmaznak, akkor a helyi fiókot használó hálózati bejelentkezést a vendég fiókhoz kötik. A klasszikus beállítás az erőforrásokhoz való hozzáférés finom vezérlését teszi lehetővé. A klasszikus modell használatával ugyanahhoz az erőforráshoz különböző felhasználók számára más-más típusú hozzáférést tud biztosítani.

A beállítás módja:

Start -> Vezérlőpult -> Felügyeleti eszközök -> Helyi biztonsági beállítások -> Helyi házirend -> Naplórend / Felhasználói jogok kiosztása / Biztonsági beállítások

Start –> Control Panel -> Administrative Tools -> Local Security Policy -> Local Policies -> Audit Policy / User right assignments / Security Options

Eseménynaplózás

A Windows XP a lényeges eseményekre vonatkozó információkat három naplóba jegyzi fel: az Alkalmazás naplóba, a Biztonsági naplóba és a Rendszernaplóba. A napló hibaüzeneteket, naplózási információkat és a rendszer tevékenységével kapcsolatos egyéb bejegyzéseket tartalmaz. A napló nemcsak a gyanús vagy rosszindulatú viselkedés beazonosítására és a biztonsági incidensek kivizsgálására szolgál, hanem a rendszerrel kapcsolatos hibaelhárítást és az alkalmazások problémáinak megoldását is segíti. Ezért fontos, hogy mindhárom naplót használják. A NIST mintabeállítás mindhárom naplófajtát minden környezetben alkalmazza, és a maximális naplóméretet is megadja. Ha ez utóbbi érték nagyon alacsony, akkor nem lesz elég hely arra, hogy a rendszertevékenységgel kapcsolatos információ tárolásra kerüljön. Bizonyos szervezeteknél előírhatják a naplózási házirendet és központi naplózást, ekkor a mintabeállítást ahhoz kell igazítani.

A beállítás módja:

Start -> Vezérlőpult -> Felügyeleti eszközök -> Eseménynapló -> Alkalmazás/Biztonság/Rendszer (jobb kattintás) -> Tulajdonságok

Start –> Control Panel -> Administrative Tools ->Event viewer -> Application/Security/System (rigth click) -> Properties

Kötött csoportok – *Restricted users*

Minden rendszerben és minden környezetben a *távoli felhasználók (remote users)* csoportjából minden felhasználót távolítsunk el, kivéve azokat, akiknek valóban oda kell tartozniuk! Így csökkenthető annak a valószínűsége, hogy valaki távoli felhasználóként jogosulatlanul a rendszerhez férjen. A *kiemelt felhasználók (power users)* csoportjában lévőket is szűrjük meg, mivel jogosultságuk majdnem megegyezik az *rendszergazdák (administrators)* csoport jogosultságaival. Ha egy felhasználónak további – de nem teljes adminisztrátori – jogosultságra van szüksége, akkor ahelyett, hogy a kiemelt felhasználók csoportjába betennék, egyedi jogosultságokat kell számára biztosítani. Alapértelmezésben a NIST mintabeállításai a kiemelt és a távoli felhasználók csoportjából mindenkit eltávolít.

A beállítás módja:

Start -> Vezérlőpult -> Felügyeleti eszközök -> Számítógép kezelése -> Rendszereszközök -> Helyi fiókok és csoportok -> Csoportok -> Kiemelt felhasználók / Távoli felhasználók

Start –> Control Panel -> Administrative Tools -> Computer Management -> System Tools -> Local Users and Groups -> Groups -> Power Users / Remote Desktop Users

Rendszerszolgáltatások

A Windows XP számos szolgáltatást nyújt, ezek többsége a rendszer indításával együtt automatikusan elindul. A szolgáltatások különféle erőforrásokat vesznek igénybe és a gyengeséget, sebezhetőséget is jelenthetnek a gazdagép számára. Minden felesleges szolgáltatást le kell állítani, hogy a rendszer elleni támadások számát csökkenthessék. Menedzselt környezetben a Csoport Házirend Objektumban kell a rendszeren lévő szolgáltatásokat konfigurálni; egyéb környezetben pedig minden rendszerben egyenként kell beállítani a szolgáltatásokat. Mindkét konfigurálási eljárás esetén minden szolgáltatás indítására három eset lehetőség közül lehet választani:

- *Automatikus – Automatic*: A szolgáltatás automatikusan elindul. Ez azt jelenti, hogy a rendszer felállása után a szolgáltatás fut.
- *Kézi – Manual*: Csak akkor indul a szolgáltatás, ha szükséges. A gyakorlatban ez nem azt jelenti, hogy ha igénybe kívánják venni a szolgáltatást, akkor az automatikusan elindul, hanem kézzel el kell indítani.
  Megjegyzés: ha egy szolgáltatás egy másik szolgáltatástól függ, akkor az első – helytelenül – azt feltételezi, hogy a másik szolgáltatás már fut.
- *Letiltva – Disabled:* A szolgáltatást nem lehet elindítani.

A NIST javaslata szerint az alábbi szolgáltatások mindegyikét minden környezetben tiltsuk le, hacsak valamilyen speciális ok nincs a futtatásukra.

- Riasztás
- Vágókönyv
- FTP publikációs szolgáltatás
- IIS rendszerszolgáltatás
- Üzenetkezelő
- Netmeeting távoli asztalmegosztás
- Útválasztás és távelérés
- SMTP

- SNMP szolgáltatás
- SNMP csapda
- Telnet
- WWW publikációs szolgáltatás

A NIST mintabeállítások mindegyike letiltja ezeket a szolgáltatásokat. Ezenkívül – bizonyos környezetekben – még további szolgáltatásokat is letilt, mint a Számítógép-tallózó, Távoli rendszerleíró adatbázis, Feladatütemező és Terminálszolgáltatás. Különösen vállalati környezetben nagy kihívást jelent a szolgáltatások biztonságos beállítása.

A szolgáltatások letiltása:

Start -> Vezérlőpult -> Felügyeleti eszközök -> Szolgáltatások -> (Szolgáltatás nevére kétszer rákattintani) -> Indítás típusa: Automatikus / Kézi / Letiltva

Start –> Control Panel -> Administrative Tools -> Services -> (Double click the service name) -> Startup Type: Automatic / Manual /Disable

Az Universal Plug and Play eszközök (Universal Plug and Play) letiltásánál még két szolgáltatás le kell tiltani: SSDP keresőszolgáltatás (SSDP Discovery Services) és a Universal Plug and Play Device Host Services.

A távolról történő hozzáférés lehetőségét másképpen tiltják le. Bár ez egy nagyon hathatós tulajdonsága a rendszernek, sajnos ekkor a számítógép nagyon ki van téve a hálózati támadásoknak.

Saját gép (jobb klikk) -> Tulajdonságok -> Távoli használat -> (pipa mellőzése)A számítógépről lehet távsegítséget kérni / Távolról is kapcsolódhatnak a felhasználók ehhez a számítógéphez

My Computer (jobb klikk) -> Properties -> Remote (tab) -> (uncheck) Allow Remote Assistance invitation to be sent from this computer / Allow users to connect remotely to this computer

Fájl jogosultságok

Ebben a részben a Windows XP fájlrendszeréhez tartozó hozzáférési szabályok (ACE – access control entries) és hozzáférés-szabályozási listák (ACL – access control lists) általános beállításáról lesz szó. A NIST mintabeállítása csak 25 végrehajtható, de jogosulatlan módosításoktól és jogosulatlan felhasználástól védeni kívánt fájlra korlátozódik. Egy adott környezetben működő Windows XP rendszerhez további beállítások is szükségesek lehetnek.

Egy adott erőforráshoz (fájl, könyvtár) tartozó ACL lista módosítása háromféleképpen történhet:

- Az *Intézőben (Windows Explorer)* – vagy az asztalon,... - az erőforráshoz tartozó *Tulajdonságok (Properties)* elnevezésű ablak megnyitása után a *Biztonság (Security)* fülre rákattintva látható, hogy az erőforráshoz melyik felhasználónak és melyik csoportnak milyen jogosultsága van. A *Speciális (Advanced)* –t megnyitva még finomabb beállítás végezhető, például az erőforrás tulajdonosának beállítása vagy a naplózás.
  Megjegyzés: Ha a Windows XP operációs rendszer nem tartozik tartományhoz, valamint NTFS fájlrendszere van, akkor a Biztonság (Security) fül alapértelmezésben nem látható. Bekapcsolása (ne legyen pipa ennél):

  Intéző -> Eszközök -> Mappa beállításai -> Nézet -> Egyszerű fájlmegosztás használata

  Windows Explorer -> Tools -> Folder Options -> View -> Use simple file sharing

- a %SystemRoot%\system32 –ben található cacls.exe-vel. Ez a parancs módban használható program a fájlok ACL-jeit állítja, de nem módosítja a Windows XP „Se-

curity descriptor”-ait.
Megjegyzés: az MFT (Master File Table)-ben minden fájlhoz tartozik egy rekord, s a rekord mindenféle attribútumot tartalmaz, többek között az ún. „security descriptor”-t is, ami a fájlhoz tartozó ACL-eket tárolja.

– az MMC segítségével egy biztonsági mintasablon (*security template*) betöltésével.

A Windows XP a hozzáférési szabályok öröklési modelljét alkalmazza, vagyis egy objektum ACL-je tartalmazza a szülő-objektumtól örökölt ACE-t. Például az NTFS fájlrendszerben lévő fájl az őt tartalmazó könyvtár ACE-jét örökli. Ugyanakkor a fájlrendszer egy objektumára vonatkozó közvetlen ACE beállítás magasabb prioritású az örökölt ACE-nél.

Rendszerleíró-adatbázis engedélyezése

A Windows XP rendszerleíró-adatbázisának módosítása engedélyhez van kötve. A NIST mintabeállításai a legtöbb rendszerleíró-kulcs és érték beállítását szabályozza, hogy megvédje azokat a jogosulatlan hozzáférésektől és módosításoktól. Alapértelmezésben a teljes rendszerleíró-adatbázissal való műveletek is csak korlátozottan végezhetők el, de nagyon fontos annak ellenőrzése, hogy ez valóban így is van:

Start -> Futtat ->Regedit -> HKLM\SYSTEM\CurrentControlSet\Control\SecurePipeServers\winreg (jobb klikk) -> Permissions

Start -> Run -> Regedit -> HKLM\SYSTEM\CurrentControlSet\Control\SecurePipeServers\winreg (right click) -> Permissions

Gondoskodjon arról, hogy csak a Rendszergazdák rendelkezzenek teljes joggal, a Backup operátor csoportnak alig legyen valamiféle joga (bizonyos értékek lekérdezése, alkulcsok felsorolása, értesítés, olvasás) és a „Helyi szolgáltatás” pedig csak olvasási joggal rendelkezzen.

A következőkben néhány rendszerleíró-kulcs neve, elérési útja, feladata kerül felsorolásra, valamint a javasolt beállítás. Az alkalmazott rövidítések: HKLM = HKEY_LOCAL_MACHINE, HKCU = HKEY_CURRENT_USER, HKU = HKEY_USERS.

Nyomkövetéshez kapcsolódó rendszerleíró kulcsok

HKLM\Software\Microsoft\DrWatson\CreateCrashDump

Értékét 0-ra állítva a Dr. Watson nyomkövető program nem készít a memória tartalmáról másolatot (dump). A memória ugyanis érzékeny információkat is tartalmazhat, mint például jelszavak.

HKLM\Software\Microsoft\Windows NT\CurrentVersion\AEDebug\Auto

Értékét 0-ra állítva a Dr. Watson nyomkövető program nem működik.

Automatikus funkciók

HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun

HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun

HKU\.DEFAULT\ Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun

Az autorun (automatikus indulás) funkció megpróbálja a CD tartalmát azonnal lefuttatni, mihelyt a CD-t a meghajtóba teszik. Ha a CD tartalma kétes, akkor ez a beállítás nem helyes. A kulcs értékét 255-re állítva az automatikus indulás funkció egyik meghajtón sem fog működni.

HKLM\System\CurrentControlSet\Services\Cdrom\Autorun

Értékét nullára állítva a CD automatikus indulás funkciója nem működik.

HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\AutoAdminLogon

Ha ezt engedélyezik (azaz az értéke 1), akkor a rendszerleíró-adatbázisban tárolt jelszóval – jelszó megadása nélkül – be lehet jelentkezni a rendszerbe. Ez nagyon veszélyes, ugyanis a rendszerleíró-adatbázisban nyílt szövegként jelenik meg a jelszó, és a helyi felhasználók is láthatják azt. Másrészt bárki, aki fizikailag hozzáfér a rendszerhez, ugyancsak – jogosultság ellenőrzése nélkül – beléphet a rendszerbe.

Megjegyzés: A jelszó a „DefaultPassword" bejegyzésben található. Ha akár a „DeafultPassword", akár az „AutoAdminLogon" bejegyzés nincs meg az adatbázisban, akkor azok létrehozhatóak.

HKLM\System\CurrentControlSet\Control\CrashControl\AutoReboot

Az „AutoReboot" engedélyezése esetén egy hiba vagy fennakadás esetén a rendszer automatikusan újraindul. Egyesek szerint ez biztonsági és működési szempontból nem kívánatos. Például, ha egy hiba történik és a rendszer automatikusan újraindul, nem lehet tudni, hogy működési hiba történt vagy a rendszert feltörték. Kikapcsolása az érték 0-ra állításával megy.

Hálózati munka

HKLM\System\CurrentControlSet\Services\LanManServer\Parameters\AutoShareWks

Ha a Microsoft hálózaton belül fájl- vagy nyomtatómegosztást használnak, a Windows XP minden fix helyi meghajtót, mint rejtett adminisztratív erőforrást (pl. C$, D$) megoszt. Hacsak nem okvetlenül szükséges ez a megosztás, szüntessük meg! Szükség lehet erre például olyan alkalmazásoknál, amelyek számítanak az ilyen megosztásokra; továbbá a távolról karbantartott rendszerek igénylik az ilyen megosztásokat. A megosztást az érték 0-ra állításával szüntetik meg.

HKLM\System\CurrentControlSet\Services\MrxSmb\Parameters\RefuseReset

Ezt a paramétert 1-re állítva elérhető, hogy a rendszer a ResetBrowser frame-t mellőzi. Ez utóbbit arra lehet használni, hogy a NetBIOS-t és az eddigi master browsert leállítsa, és egy másik számítógépet nevezzen ki master browsernek. A Windows korábbi változataiban ez biztonsági lyuk volt.

Megjegyzés: a Browse Service külön hálózati szolgáltatás, mely még manapság is alapvető fontosságú a hálózati erőforrások megtalálásában (Network Neighborhood). Ez az úgynevezett Master Browser friss, illetve nagyobb hálózatok esetén a Backup Browserek esetleg elavult erőforráslistájában való kattintgatással kérdezhető le. A Browser a lekérdezéshez az UDP szolgáltatást használja.)

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\DisableIPSourceRouting

Ezt a paramétert 2-re beállítva lehetetlenné válik a küldő által beállított útvonalválasztás, az ún. source routing. Tulajdonképpen nincs is rá igazán szükség, de arra felhasználható, hogy egy támadó egy IP csomagot egy közbülső hoszton átküldjön, s így megnézze vagy módosítsa a hálózati kommunikációt.

Megjegyzés: Source routing: A feladó által megadott útvonalon, állomások megadott listáján halad végig a csomag. Két válfaja van, a szigorú (strict) és a laza (loose). Az első esetben csak a listán felsorolt állomásokon haladhat végig a csomag és ha két szomszédosnak felsorolt állomás nem szomszédos, akkor a csomag elvész és egy „Source routing failed" ICMP csomag küldődik a feladóhoz. A második esetben ha a listán két szomszédosnak feltüntetett állomás a valóságban nem szomszédos, akkor is továbbmegy a csomag a lista következő eleméhez, de a routerek által kijelölt útvonalon.)

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\EnableDeadGWDetect

Ha ennek értéke 1, akkor a TCP észleli a nem működő átjárót. A TCP kérheti az IP-t, hogy álljon át a tartalék átjáróra, ha adott számú kapcsolatnál problémát észlel. A támadó ezt a lehetőséget arra tudja felhasználni, hogy a rendszer egy rosszindulatú átjáró felé irányítsa a forgalmat, s így megnézze, módosítsa az adatokat vagy szolgáltatásmegtagadást idézzen elő. A helyes beállítás: 0.

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\EnableICMPRedirect

Ha ez a beállítás érvényben van, akkor a Windows XP a hálózati eszközöktől (pl. router) kapott ICMP Redirect üzenet hatására az útvonalválasztó tábláját átírja. A támadó ekkor egy hamisított ICMP Redirect üzenettel elérheti, hogy az ő rendszerébe (vagy bárhová máshova) irányítsák a csomagokat, s így érzékeny információkat foghat el, betörhet a rendszerbe vagy szolgáltatásmegtagadást idézhet elő.

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\EnablePMTUDiscovery

Ha ez a paraméter be van állítva, akkor a TCP megpróbálja felderíteni az MTU –t (Maximal Transmission Unit – a legnagyobb fizikai csomag mérete bájtban, amit a hálózat még át tud vinni). A NIST javaslata alapján kapcsoljuk ki ezt a lehetőséget, azaz állítsuk 0-ra az értéket, és így minden kapcsolatban 576 bájtos csomagot küld a lokális hálózaton kívüli hosztokra.

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\KeepAliveTime

Az itt beállított érték azt mutatja meg, hogy a TCP milyen gyakran küldjön egy ún. keep-alive csomagot egy éppen nem forgalmazó kapcsolatra, hogy megnézze, hogy még él-e a kapcsolat. Élő kapcsolat esetén a távoli hoszt nyugtát küld. Alapértelmezésben ilyen csomagot nem küld a rendszer. A NIST javaslata 500000 msec, azaz 5 perc.

HKLM\System\CurrentControlSet\Services\Netbt\Parameters\NoNameReleaseOnDemand

Ez a paraméter határozza meg, hogy a hálózatról érkező kérésre megadja-e a számítógép a NetBIOS nevet. Ha 1-re állították, akkor nem adja meg a nevet, ez a rosszindulatú, névvel kapcsolatos támadásoktól védi a rendszert. A NIST mintabeállításaiban ez nem található meg.

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\PerformRouterDiscovery

Ez a paraméter ellenőrzi, hogy a rendszer az RFC 1256 szerint keresi-e az útválasztóját (routerét). Minden NIST mintabeállításban 0 az értéke. (A RFC 1256 az ICMP üzenet kibővítését tartalmazza, vagyis amikor a rendszer egy multicast vagy broadcast hálózatban a szomszédos routerek IP címét keresi meg.)

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\SynAttackProtect

Az ún. SYN elárasztás (Synflood) ellen véd ez a paraméter. Két további paraméter játszik még itt szerepet: az ugyanitt található TcpMaxHalfOpen (ld. következő) és a TcpMaxHalfOpenRetried (ld. innen a második) Ha a kísérletek száma eléri e két paraméter értékének bármelyikét, és a SynAttackProtect értéke 1 vagy 2, akkor a Syn-elárasztás ellen védelem életbe lép. (A NIST a 2-es értéket javasolja, mert az erőteljesebb védelmet biztosít.)

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\TcpMaxHalfOpen

Ez a megengedett SYN-RCVD állapotban lévő kapcsolatok számát határozza meg (vagyis a félig felépített TCP kapcsolatok számát), ha ezt az értéket eléri a rendszer, akkor a SynAttackProtect életbe lép. A NIST alapértelmezésben 100-t javasol.

HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\TcpMaxHalfOpenRetried

Ez azon SYN-RCVD állapotban lévő kapcsolatok számát határozza meg, amelyekre legalább egy válaszcsomag már elment. A NIST által javasolt érték 80. (Érthetően ez kisebb érték, mint az előző.)

HKLM\System\CurrentControlSet\Services\IPSEC\NoDefaultExempt

Alapértelmezésként a Windows XP-ben az IPsec a házirendben előírt szűrések alól kivételt képezhet. (ld. http://support.microsoft.com/?id=810207) A paraméter értékét 1-re állítva ez a kivételezés megszűnik a Kerberosra és a RSVP forgalomra.

HKLM\System\CurrentControlSet\Services\Lanmanserver\Parameters\Hidden

A paraméter 1-re állítva megakadályozható, hogy a rendszer egy ún. „browser" bejelentést tegyen, tehát így a hálózaton lévő browserek elől rejtve marad. Ezáltal csökken annak a valószínűsége, hogy a Microsoft hálózaton keresztül hozzáférjenek ehhez a géphez.

HKLM\System\CurrentControlSet\Control\Session Manager\SafeDllSearchMode

A Windows XP a könyvtárakat megadott sorrendben keresi végig, amikor egy végrehajtandó fájlt keres. Alapértelmezésben először a kurrens könyvtárban keres, majd a Windows és rendszerkönyvtárakban. Ha ezt 1-re állítjuk, akkor a kurrens könyvtárban csak az előbbiek után keres. Biztonsági szempontból ez jobb megoldás, mert a kurrens könyvtár kevésbé védett lehet. Például ha valaki egy trójait helyez el egy megosztott könyvtárban, az alapértelmezésként használt keresési sorrendnél a trójait a megosztott könyvtárat elérő felhasználó – a névazonosság miatt – véletlenül lefuttathatja, míg a második esetben ez nem történik meg.

Javaslatok összefoglalása

- Készítsen jelszó-házirendet, hogy a jelszó kitalálásával vagy feltörésével történő jogosulatlan hozzáférést megakadályozza! A házirendnek a biztonság, a működőképesség és a használhatóság között kell egyensúlyoznia.
- Készítsen napló-házirendet, hogy bizonyos típusú tevékenységet rögzíteni lehessen, így a rendszergazda megnézheti a naplót és észlelheti a jogosulatlan aktivitást!
- A „legkevesebb jog" elvének figyelembevételével adjon a felhasználóknak jogokat!
- A nagyobb biztonság eléréséért az alapértelmezések állítsa át, ahol szükséges; például a jelszó nélküli belépés ne engedélyezzen, a „Rendszergazda" és a „Vendég" azonosítót nevezze át, és határozza meg, milyen típusú azonosítást alkalmaz!
- Állítsa be az „Alkalmazás", a „Biztonság" és a „Rendszer" naplózást!
- Törölje ki a távoli felhasználók és a kiemelt felhasználók csoportjából az összes olyan felhasználót, akinknek nem kell ott szerepelniük!
- Töröljön minden felesleges szolgáltatást!
- Töröljön minden Univerzális plug and play eszközt és Távoli segítséget!
- Korlátozza a hozzáférési lista (ACL) beállításait és a rendszerleíró-adatbázis bejegyzéseit!

7.3.1.3. Biztonsági mintabeállítások

Az alábbi, a NIST által készített mintabeállítások a CIS, DISA, NIST, NSA és a Microsoft szakembereinek egyetértésével készült el. Az itteni beállítások az előző fejezetben leírt csoportosításon alapulnak.

Fiókházirend

Jelszóházirend

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	Jelszó maximális élettartama Maximum password age	90 nap				
2.	Jelszó minimális élettartama Minimum password age	1 nap				
3.	Legrövidebb jelszó Minimum password length	12 karakter	8 karakter			*
4.	A jelszónak meg kell felelnie a bonyolultsági feltételeknek Password must meet complexity requirements	engedélyezve				
5.	Előző jelszavak megőrzése Enforce password history	24 jelszó megőrzése				
6.	A tartomány összes felhasználója jelszavának tárolása visszafejthető titkosítással Store password using reversible encryption for all users in the domain	letiltva				

3. Táblázat: Fiókházirend – jelszóházirend

* Jelszó helyett inkább jelmondatot használjanak!

Fiókzárolási házirend

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	Fiókzárolás küszöbe Account lockout threshold	10 kísérlet	50 kísérlet			
2.	Fiókzárolás időtartama Account lockout duration	15 perc				
3.	Fiókzárolási számláló nullázása Reset account lockout counter after	15 perc				

4. Táblázat: Fiókházirend – fiókzárolási házirend

Helyi házirend

Naplórend

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	Bejelentkezés naplózása Audit logon events	sikeres/sikertelen				
2.	Fiókkezelés naplózása Audit account management	sikeres/sikertelen				
3.	Címtárszolgáltatás-hozzáférés naplózása Audit directory service access	nem definiált				
4.	Fiókkezelés naplózása Audit account logon events	sikeres/sikertelen				

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
5.	Objektum-hozzáférés naplózása Audit object access	sikeres/sikertelen	sikertelen			
6.	Házirendváltozás naplózása Audit policy change	sikeres				
7.	Rendszerjogok használatának naplózása Audit privilege use	sikertelen				
8.	Folyamatok nyomon követésének naplózása Audit process tracking	nem definiált				*
9.	Rendszeresemények naplózása Audit system events	sikeres				

5. Táblázat: Helyi házirend – Naplórend

* Nagy sok esemény kerülhet a naplóba, csak ha feltétlenül szükséges, akkor használja!

Felhasználói jogok kiosztása

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	A számítógép elérése a hálózatról Access this computer from the network	Rendszergazdák			felhasználó, Rendszer-gazdák	
2.	Az operációs rendszer részeként való működés Act as part of the operating system	nincs beállítás				
3.	Munkaállomás felvétele a tartományba Add workstation to domain	nem definiált (nem alkalmazható)				
4.	Memóriakvóták beállítása folyamat számára Adjust memory quotas for a process	nem definiált				
5.	Be lehessen jelentkezni terminálszolgáltatások használatával Allow logon through Terminal Services	nincs be-állítás	Rendszergazdák			
6.	Biztonsági másolat készítése fájlokról és könyvtárakról Back up files and directories	Rendszergazdák				
7.	Szülőkönyvtár-jogosultság mellőzése Bypass traverse checking	felhasználó				
8.	Rendszeridő megváltoztatása Change the system time	Rendszergazdák				
9.	Lapozófájl létrehozása Create a pagefile	Rendszergazdák				
10.	Token objektum létrehozása Create a token object	nincs beállítás				
11.	Globális objektumok létrehozása Create global objects	Rendszergazdák				
12.	Állandó megosztott objektumok létrehozása Create permanent shared objects	nincs beállítás				
13.	Programok hibakeresése Debug programs	egyik sem	Rendszergazdák		egyik sem	

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
14.	A számítógép hálózati elérésének megtagadása Deny access to this computer from the network	vendég				
15.	Kötegelt munka bejelentkezésének megtagadása Deny logon a a batch job	nem definiált				
16.	Szolgáltatásként bejelentkezés megtagadása Deny logon as service	nem definiált				
17.	Helyi bejelentkezés megtagadása Deny logon locally	nem definiált				
18.	Ne lehessen bejelentkezni terminálszolgáltatások használatával Deny logon through Terminal Services	nem definiált				
19.	Számítógép- és felhasználói fiókok megbízhatóságának engedélyezése Enable computer and user accounts to be trusted for delegation	nem definiált (nem alkalmazható)				
20.	Távirányított rendszerleállítás Force shutdown from a remote system	Rendszergazdák				
21.	Biztonsági naplózás létrehozása Generate security audits	Helyi szolgáltatás, Hálózati szolgáltatás				
22.	Ügyfél megszemélyesítése hitelesítés után Impersonate a client after authentication	nem definiált				
23.	Ütemezési prioritás növelése Increase scheduling priority	Rendszergazdák				
24.	Szolgáltatók betöltése és eltávolítása Load and unload device drivers	Rendszergazdák				
25.	Memórialapok zárolása a memóriában Lock pages in memory	nincs beállítás				
26.	Bejelentkezés kötegfájlfolyamatként Log on as a batch job	nem definiált				
27.	Bejelentkezés szolgáltatásként Log on as a service	nem definiált				
28.	Helyi bejelentkezés Log on locally	felhasználó, Rendszergazdák				
29.	Naplózás felügyelete Manage auditing and security log	Rendszergazdák				
30.	Beégetett programok környezeti értékeinek megváltoztatása Modify firmware environment values	Rendszergazdák				
31.	Kötetkarbantartási feladatok végrehajtása Perform volume maintenance tasks	Rendszergazdák				
32.	Folyamatok teljesítményadatainak figyelése Profile single process	Rendszergazdák				
33.	A rendszer teljesítményadatainak figyelése Profile system performance	Rendszergazdák				
34.	Számítógép eltávolítása tokjából Remove computer from docking station	felhasználó, Rendszergazdák				

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
35.	Folyamat token lecserélése Replace a process level token	Helyi szolgáltatás, hálózati szolgáltatás				
35.	Fájlok és könyvtárak helyreállítása Restore files and directories	Rendszergazdák				
37.	A rendszer leállítása Shut down the system	felhasználó, Rendszergazdák				
38.	Könyvtárszolgáltatás-adatok szinkronizálása Synchronize directory service data	nem definiált (nem alkalmazható)				
39.	Fájlok és egyéb objektumok saját tulajdonba vétele Take ownership of files or other objects	Rendszergazdák				

6. Táblázat: Felhasználói jogok kiosztása

Biztonsági beállítások

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejét-múlt	
1.	Fiókok: A Rendszergazdák fiókok állapota Accounts: Administrator account status	nem definiált				
2.	Fiókok: A vendég fiók állapota Accounts: Guest account status	letiltva				
3.	Fiókok: Az üres jelszó használatának konzolbejelentkezésekre korlátozása a helyi fiókoknál Accounts: Limit local account use of blank passwords to console logon	engedélyezve				
4.	Fiókok: A rendszergazdai fiók átnevezése Accounts: Rename administrator account	nem definiált				
5.	Fiókok: a vendégfiók átnevezése Accounts: Rename guest account	nem definiált				
6.	Naplózás: Globális rendszerobjektumokhoz való hozzáférés naplózása Audit: Audit the access of global systems object	nem definiált				
7.	Naplózás: A biztonsági mentés és helyreállítás jogának naplózása Audit: Audit the use of Backup and Restore privilege	nem definiált				
8.	Naplózás: A rendszer azonnali leállítása, ha nem lehet naplózni a biztonsági eseményeket Audit: Shut down system immediately if unable to log security audit	engedélyezve	nem definiált			
9.	Eszközök: Dokkolás megszüntethető bejelentkezés nélkül is Devices: Allow undock without having to log on	tiltva	nem definiált			
10.	Eszközök: Cserélhető adathordozó formázása és kiadása a következő csoportok tagjainak engedélyezve Devices: Allowed to format and eject removable media	Rendszergazdák	Rendszergazdák, interaktív felhasználó			

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejét-múlt	
11.	Eszközök: A felhasználók nem telepíthetnek nyomtatókat Devices: Prevent users from installing printer drivers	engedélyezve	nem definiált			
12.	Eszközök: A CD-ROM használatához kötelező bejelentkezni a helyi számítógépre Devices: Restrict CD-ROM access to locally logged-on user only	engedélyezve	nem definiált			
13.	Eszközök: A hajlékonylemez használatához kötelező bejelentkezni a helyi számítógépre Devices: Restrict floppy access to locally logged-on user only	engedélyezve	nem definiált			
14.	Eszközök: Viselkedés nem aláírt illesztőprogram telepítésekor Devices: Unsigned driver installation behavior	értesítés után engedélyezve				
15.	Tartományvezérlő: A kiszolgálófelelősök ütemezhetnek feladatokat Domain controller: Allow server operator to schedule tasks	nem definiált (nem alkalmazható)				
16.	Tartományvezérlő: LDAP-kiszolgáló aláírási követelményei Domain controller: LDAP server signing requirements	nem definiált (nem alkalmazható)				
17.	Tartományvezérlő: Számítógépfiók jelszómódosításának visszautasítása Domain controller: Refuse machine account password changes	nem definiált (nem alkalmazható)				
18.	Tartományi tag: A biztonságos csatornák adatainak digitális titkosítása vagy aláírása (mindig) Domain member: Digitally encrypt or sign secure channel data (always)	engedélyezve			tiltva	
19.	Tartományi tag: az adatok digitális titkosítása (ha lehet) Domain member: Digitally encrypt secure channel data (when possible)	engedélyezve				
20.	Tartományi tag: az adatok digitális aláírása (ha lehet) Domain member: Digitally sing secure channel data (when possible)	engedélyezve				
21.	Tartományi tag: Számítógépfiók jelszómódosításainak tiltása Domain member: Disable machine account password changes	tiltva				
22.	Tartományi tag: Számítógépfiók jelszavának maximális élettartama Domain member: Maximum machine account password age	30 nap				

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejét-múlt	
23.	Tartományi tag: erős (Windows 2000 vagy frissebb rendszerű) munkamenetkulcs megkövetelése Domain member: Require strong (Windows 2000 or later)session key	engedélyezve			tiltva	
24.	Interaktív bejelentkezés: Ne jelenjen meg a legutóbb bejelentkezett felhasználó neve Interactive logon: Do not display last user name	engedélyezve				
25.	Interaktív bejelentkezés: Nincs szükség CTRL+ALT+DEL billentyűkombinációra Interactive logon: Do not require CTRL+ALT+DEL	letiltva				
26.	Interaktív bejelentkezés: Bejelentkezési üzenet a felhasználónak Interactive logon: Message text for users attempting to log on	<jogi részleg jóváhagyásával>				
27.	Interaktív bejelentkezés: Bejelentkezési üzenet címe Interactive logon: Message title for users attempting to log on	<jogi részleg jóváhagyásával>				
28.	Interaktív bejelentkezés: Tartományvezérlő nélküli bejelentkezés helyileg tárolt információval Interactive logon: Number of previous logons to cache (in case domain controller is not available)	0	1	2		
29.	Interaktív bejelentkezés: A felhasználó figyelmeztetése a jelszó lejárta előtt Interactive logon: Prompt use to change password before expiration	14 nap				
30.	Interaktív bejelentkezés: A munkaállomás zárolásának feloldásához tartományvezérlői hitelesítésre van szükség Interactive logon: Require Domain Controller authentication to unlock workstation	nem definiált	engedélyezve	letiltva	nem definiált	
31.	Interaktív bejelentkezés: Viselkedés intelligens kártya eltávolításakor Interactive logon: Smart card removal behavior	munkaállomás zárolása				
32.	Microsoft hálózati ügyfél: Kommunikáció digitális aláírása (mindig) Microsoft network client: Digitally sign communications (always)	engedélyezve			nem definiált	1
33.	Microsoft hálózati ügyfél: Kommunikáció digitális aláírása (ha a kiszolgáló egyetért) Microsoft network client: Digitally sign communications (if server agrees)	engedélyezve				
34.	Microsoft hálózati ügyfél:Titkosítatlan jelszavak küldése egyéb SMB kiszolgálóknak Microsoft network client: Send unencrypted password to third-party SMB servers	letiltva				2

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejét-múlt	
35.	Microsoft hálózati ügyfél: Ügyfelek leválasztása a nyilvántartási idő végén Microsoft network server: Amount of idle time required before suspending session	15 perc				
36.	Microsoft hálózati kiszolgáló: Kommunikáció digitális aláírása (mindig) Microsoft network server: Digitally sign communications (always)	engedélyezve			nem definiált	
37.	Microsoft hálózati kiszolgáló: Kommunikáció digitális aláírása (ha az ügyfél egyetért) Microsoft network server: Digitally sign communications (if client agrees)	engedélyezve				
38.	Microsoft hálózati kiszolgáló: Ügyfelek leválasztása a nyilvántartási idő végén Microsoft network server: Disconnect clients when logon hours expired	engedélyezve		letiltva	engedélyezve	
39.	Hálózati hozzáférés: A névtelen helyazonosító-/névfordítás engedélyezése Network access: Allow anonymous SID/Name translation	letiltva				
40.	Hálózati hozzáférés: SAM fiókok névtelen felsorolása nem engedélyezett Network access: Do not allow anonymous enumeration of SAM accounts	engedélyezve				
41.	Hálózati hozzáférés: SAM fiókok és –megosztása névtelen felsorolása nem engedélyezett Network access: Do not allow anonymous enumeration of SAM accounts and shares	engedélyezve				
42.	Hálózati hozzáférés: Nem lehet hitelesítő adatokat vagy .NET passportot tárolni hálózati hitelesítéshez Network access: Do not allow storage of credentials or .NET Passports for network authentication	engedélyezve			nem definiált	
43.	Hálózati hozzáférés: A névtelen felhasználókra a Mindenki csoportra vonatkozó engedélyek vonatkoznak Network access: Let Everyone permissions apply to anonymous users	letiltva				
44.	Hálózati hozzáférés: Névtelenül elérhető Named Pipe-ok Network access: Named Pipes that can be access anonymously	nincs beállítás			nem definiált	
45.	Hálózati hozzáférés: Távolról elérhető rendszerleíró elérési utak Network access: Remotely accessible registry paths	nem definiált				

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
46.	Hálózati hozzáférés: Névtelenül elérhető megosztások Network access: Shares that can be accessed anonymously	nincs beállítás				
47.	Hálózati hozzáférés: Megosztási és biztonsági modell helyi felhasználói fiókok számára Network access: Sharing and security model for local account	klasszikus				
48.	Hálózati biztonság: A következő jelszómódosításkor ne tárolja a LAN-kezelő üzenetkivonatát Network security: Do not store LAN manager hash value on next password change	engedélyezve			nem definiált	
49.	Hálózati biztonság: Kötelező kijelentkezés a nyitvatartási óra lejártakor Network security: Force logoff when logon hours expire	engedélyezve		nem definiált		
50.	Hálózati biztonság: LAN-kezelő hitelesítési szintje Network security: LAN Manager authentication level	NTLMv2 küldés LM és NTLM elutasítva	NTLMv2 küldés, LM elutasítva		NTLMv2 küldés	3
51.	Hálózati biztonság: LDAP ügyfél aláírási követelményei Network security: LDAP client signing requirements	aláírás megkövetelve				
52.	Hálózati biztonság: Minimális biztonság az NTLM SSP alapú (a biztonságos RPC is) ügyfelekkel Network security: Minimum session security for NTLM SSP based (including secure RPC) clients	az üzenet sértetlensége, titkossága, az NTLMv2 munkamenet biztonság, 128-bites titkosítás megkövetelve			nem definiált	3
53.	Hálózati biztonság: Minimális munkamenet-biztonság az NTLM SSP (a biztonságos RPC is) kiszolgálókkal Network security: Minimum session security for NTLM SSP based (including secure RPC) servers	az üzenet sértetlensége, titkossága, az NTLMv2 munkamenet biztonság, 128-bites titkosítás megkövetelve			nem definiált	3
54.	Helyreállítási konzol: Automatikus rendszergazdai bejelentkezés Recovery console: Allow automatic administrative logon	letiltva				
55.	Helyreállítási konzol: Hajlékonylemez másolása és hozzáférés minden meghajtóhoz és mappához Recovery console: Allow floppy copy and access to all drives and all folders	nem definiált				
56.	Leállítás: A rendszer leállítható bejelentkezés nélkül Shutdown: Allow system to be shutdown without having to log on	letiltva				
57.	Leállítás: Virtuális memória lapozófájljainak törlése Shutdown: Clear virtual memory pagefile	engedélyezve				4

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
58.	Rendszerkriptográfia: FIPS szabványnak megfelelő algoritmus használata titkosításhoz, kivonatoláshoz és aláíráshoz System crypthography: Use FIPS compliant algorithms for encryption, hashing and signing	engedélyezve	nem definiált			
59.	Rendszerobjektumok: A Rendszergazdák csoport tagjai által létrehozott objektumok alapértelmezett tulajdonosa System objects: Default owner for objects created by members of the Administrators group	az objektum létrehozója				
60.	Rendszerobjektumok: A nem-Windows alrendszerek esetén a kis- és nagybetűk megkülönböztetésének megkövetelése System objects: Require case sensitivity for non-Windows subsystem	engedélyezve	nem definiált			
61.	Rendszerobjektumok: A belső rendszerobjektumok (pl. szimbolikus hivatkozások) alapértelmezett engedélyezésének megerősítése System objects: Strengthen default permissions of internal system objects (e.g. Symbolic Links)	engedélyezve			nem definiált	

7. Táblázat: Biztonsági beállítások

1 Windows 2000 előtti szerverekkel történő kommunikációt megakadályozza

2 Windows NT előtti szerverekkel történő kommunikációt megakadályozza

3 Bizonyos kliensekkel és kiszolgálókkal való kapcsolattartást megakadályozza

4 Emiatt az újrabootolás hosszabb ideig tarthat, különösen a több RAM-mal rendelkező rendszereknél

Eseménynapló házirend

	Házirend	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	Maximális alkalmazási napló fájlméret Maximum application log size	16 MB				
2.	Maximális biztonsági napló fájlméret Maximum security log size	80 MB				
3.	Maximális rendszernapló fájlméret Maximum system log size	16 MB				

8. Táblázat: Eseménynapló házirend

Kötött csoportok

	Kötött csoportok	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	Kiemelt felhasználók Power Users	nincs beállítás				
2.	Távoli felhasználók Remote Desktop Users	nincs beállítás				

9. Táblázat: Kötött csoportok

Rendszerszolgáltatások

	A szolgáltatás neve	Javasolt érték				
		Fokozott bizt.	**Vállalat**	**OKV**	**Idejétmúlt**	
1.	Riasztás Alerter	letiltva				
2.	Vágókönyv Clipbook	letiltva				
3.	Számítógép tallózó Computer Browser	letiltva	nem definiált	letiltva	nem definiált	
4.	Faxszolgáltatás Fax Service	letiltva	nem definiált			
5.	FTP publikációs szolgáltatás FTP Publishing Service	letiltva				
6.	IIS rendszerszolgáltatás IIS Admin Service	letiltva				
7.	Indexelő szolgáltatás Indexing Service	letiltva	nem definiált			
8.	Üzenetkezelő Messenger	letiltva				
9.	Hálózati bejelentkezés Netlogon	nem definiált				
10.	Netmeeting távoli asztalmegosztás Netmeeting Remote Desktop Sharing	letiltva				
11.	Távoli asztal súgó-munkamenet-kezelő Remote Desktop Help Session Manager	letiltva	nem definiált		letiltva	
12.	Távoli rendszerleíró adatbázis Remote Registry	letiltva	nem definiált	letiltva	nem definiált	
13.	Útválasztás és távelérés Routing and Remote Access	letiltva				
14.	Egyszerű levélátviteli protokoll (SMTP) Simple Mail Transfer Protocol (SMTP)	letiltva				
15.	Egyszerű Hálózatkezelő protokoll (SNMP) szolgáltatás Simple Network Management Protocol (SNMP) Service	letiltva				
16.	Egyszerű hálózatkezelő protokoll (SNMP) csapda Simple Network Management Protocol (SNMP) Trap	letiltva				
17.	Feladatütemező Task Scheduler	letiltva	nem definiált			
18.	Telnet Telnet	letiltva				
19.	Terminálszolgáltatások Teminal Services	letiltva	nem definiált			
20.	Univerzális Plug and Play eszközök Universal Plug and Play Device Host	letiltva			nem definiált	

	A szolgáltatás neve	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
21.	World Wide Web publikációs szolgáltatás World Wide Web Publishing Services	letiltva				

10. Táblázat: Rendszerszolgáltatások

Fájl engedélyek beállítása

	Fájlnév	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	%SystemRoot% \system32\at.exe	Adminisztrátorok: teljes; Rendszer: teljes				
2.	%SystemRoot% \system32\attrib.exe	Adminisztrátorok: teljes; Rendszer: teljes				
3.	%SystemRoot% \system32\cacls.exe	Adminisztrátorok: teljes; Rendszer: teljes				
4.	%SystemRoot% \system32\debug.exe	Adminisztrátorok: teljes; Rendszer: teljes				
5.	%SystemRoot% \system32\drwatson.exe	Adminisztrátorok: teljes; Rendszer: teljes				
6.	%SystemRoot% \system32\drwtsn32.exe	Adminisztrátorok: teljes; Rendszer: teljes				
7.	%SystemRoot% \system32\edlin.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				
8.	%SystemRoot% \system32\eventcreate.exe	Adminisztrátorok: teljes; Rendszer: teljes				
9.	%SystemRoot% \system32\eventtriggers.exe	Adminisztrátorok: teljes; Rendszer: teljes				
10.	%SystemRoot% \system32\ftp.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				
11.	%SystemRoot% \system32\net.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				
12.	%SystemRoot% \system32\net1.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				
13.	%SystemRoot% \system32\netsh.exe	Adminisztrátorok: teljes; Rendszer: teljes				
14.	%SystemRoot% \system32\arcp.exe	Adminisztrátorok: teljes; Rendszer: teljes				
15.	%SystemRoot% \system32\reg.exe	Adminisztrátorok: teljes; Rendszer: teljes				2
16.	%SystemRoot% \system32\regedit.exe	Adminisztrátorok: teljes; Rendszer: teljes				2
17.	%SystemRoot% \system32\regedt32.exe	Adminisztrátorok: teljes; Rendszer: teljes				2
18.	%SystemRoot% \system32\regsvr32.exe	Adminisztrátorok: teljes; Rendszer: teljes				
19.	%SystemRoot% \system32\rexec.exe	Adminisztrátorok: teljes; Rendszer: teljes				
20.	%SystemRoot% \system32\rsh.exe	Adminisztrátorok: teljes; Rendszer: teljes				
21.	%SystemRoot% \system32\runas.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				3
22.	%SystemRoot% \system32\sc.exe	Adminisztrátorok: teljes; Rendszer: teljes				
23.	%SystemRoot% \system32\subst.exe	Adminisztrátorok: teljes; Rendszer: teljes				4
24.	%SystemRoot% \system32\telnet.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				
25.	%SystemRoot% \system32\tftp.exe	Adminisztrátorok: teljes; Rendszer: teljes; interaktív[1]: olvas és végrehajt				
26.	%SystemRoot% \system32\tlntsvr.exe	Adminisztrátorok: teljes; Rendszer: teljes				

11. Táblázat: Fájl engedélyek beállítása

1 Interaktív: bejelentkezett felhasználó

2 Mivel a felhasználók ezeket az eszközöket már nem használhatják, a beállítások hátrányosan befolyásolhatják a műveleteket.

3 Néhány szervezet inkább a felhasználóknak biztosítja a runas.exe futtatási jogát, mintsem az adminisztrátorok kapjanak teljes jogosultságot.

4 Alkalmazás-kompatibilitási kérdéseket vethet fel; például egy nem-privilegizált felhasználó egy login szkripten keresztül behívhatja.

Rendszerleíró-adatbázis beállításai

	A rendszerleíró-adatbázis kulcs	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
1.	HKLM\Software	Rendszergazdák:teljes, Rendszer: teljes, Létrehozó tulaj: teljes, Felhasználó: olvasás	nem meghatározott			
2.	HKLM\Software\Microsoft\Windows\CurrentVersion\Installer	Rendszergazdák:teljes, Rendszer: teljes, Felhasználó: olvasás	Rendszergazdák: teljes, Rendszer: teljes, Felhasználó: olvasás			
3.	HKLM\Software\Microsoft\Windows\CurrentVersion\Policies	Rendszergazdák:teljes, Rendszer: teljes, Jogosult felh.: olvasás	Rendszergazdák: teljes, Rendszer: teljes, Jogosult felh.: olvasás			
4.	HKLM\System	Rendszergazdák:teljes, Rendszer: teljes, Létrehozó tulaj: teljes, Felhasználó: olvasás	nem meghatározott			
5.	HKLM\System\CurrentControlSet\Enum	Rendszergazdák:teljes, Rendszer: teljes, Jogosult felh.: olvasás	Rendszergazdák: teljes, Rendszer: teljes, Jogosult felh.: olvasás			
6.	HKLM\System\CurrentControlSet\Services\SNMP\Parameters\PermittedManagers	Rendszergazdák:teljes, Rendszer: teljes, Létrehozó tulaj: teljes	Rendszergazdák: teljes, Rendszer: teljes, Létrehozó tulaj: teljes			
7.	HKLM\System\CurrentControlSet\Services\SNMP\Parameters\ValidCommunities	Rendszergazdák:teljes, Rendszer: teljes, Létrehozó tulaj: teljes	Rendszergazdák: teljes, Rendszer: teljes, Létrehozó tulaj: teljes			
8.	HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\Ratings	Rendszergazdák:teljes, Felhasználó: olvasás	nem meghatározott			
9.	HKLM\Software\Microsoft\MSDTC	Rendszergazdák:teljes, Rendszer: teljes, Hálózati szolgáltatás: érték lekérdezés, - beállítás, alkulcs generálás, alkulcs felsorolás, értesítés. olvasás Felhasználó: olvasás	nem meghatározott			
10.	HKU\.Default\Software\Microsoft\SystemCertificates\Root\ProtectedRoots	Rendszergazdák:teljes, Rendszer: teljes, Felhasználó: olvasás	Rendszergazdák: teljes, Rendszer: teljes, Felhasználó: olvasás			

	A rendszerleíró-adatbázis kulcs	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
11.	HKLM\Software\Microsoft\ Windows NT\Current-Version\SecEdit	Rendszergazdák:teljes, Rendszer: teljes, Felhasználó: olvasás	Rendszergazdák: teljes, Rendszer: teljes, Felhasználó: olvasás			

12. Táblázat: Rendszerleíró-adatbázis beállításai

Rendszerleíró-adatbázis értékei

	A rendszerleíró-adatbázis kulcs	Javasolt érték				
		Fokozott bizt.	Válla-lat	OKV	Idejét-múlt	
1.	HKLM\Software\Microsoft\\|DrWatson\CreateCrashDump	0				
2.	HKLM\Software\Microsoft\Windows NT\CurrentVersion\AEDebug\Auto	0				
3.	HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun	255				
4.	HKU\.DEFAULT\ Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\NoDriveTypeAutoRun	255				
5.	HKLM\Software\Microsoft\Windows NT\CurrentVersion\WinLogon\AutoAdminLogon	0				
6.	HKLM\System\CurrentControlSet\Control\CashControl\AutoReboot	0				
7.	HKLM\System\CurrentControlSet\Services\Cdrom\Autorun	0				
8.	HKLM\System\CurrentControlSet\Services\LanmanServer\Parameters\AutoShareWks	0	nem definiált	0	nem definiált	
9.	HKLM\System\CurrentControlSet\Services\MrxSmb\Parameters\RefuseReset	1				
10.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\DisableIPSourceRouting	2				
11.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\EnableDeadGWDetect	0				
12.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\EnableICMPRedirect	0				
13.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\EnablePMTUDiscovery	0				
14.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\KeepAliveTime	300000				
15.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\PerformRouterDiscovery	0				
16.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\ynAttackProtect	2				
17.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\TcpMaxHalfOpen	100				
18.	HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\TcpMaxHalfOpenRetired	80				
19.	HKLM\System\CurrentControlSet\Services\IPSEC\NoDefaultExempt	1				
20.	HKLM\System\CurrentControlSet\Services\Lanmanserver\Parameters\Hidden	1				

	A rendszerleíró-adatbázis kulcs	Javasolt érték				
		Fokozott bizt.	Vállalat	OKV	Idejétmúlt	
21.	HKLM\System\CurrentControlSet\Control\Session Manager\SafeDllSearchMode	1				

13. Táblázat: Rendszerleíró-adatbázis értékei

7.3.1.4. A Windows XP biztonsági beállításainál alkalmazott eszközök

(konfigurálás, üzemeltetés, monitorozás)

Az eszköz neve	Feladata	Helye
Automatikus frissítés Automatic Update	Ellenőrzi, hogy van-e Microsoft szolgáltatógépen frissítés; letölti és telepíti.	Windows XP rendszerben
Kódolás Cipher	A diszkek üres területeiről véglegesen letörli az adatokat (/w opció)	cipher.exe Windows XP rendszerben
Eseménynapló Event Viewer	Az alkalmazási, a biztonsági és a rendszernapló bejegyzéseit mutatja meg.	eventvwr.exe Windows XP rendszerben
Csoportházirendobjektum-kezelő konzol MMC modul Group Policy Management Console (GPMC) MMC snap-in	Csoport házirend kezelése többszörös tartományban	http://www.microsoft.com/windowsserver2003/gpmc/default.mspx
Csoportházirend modellező varázsló MMC modul Group Policy Modeling Wizard MMC snap-in	Egy adott felhasználóra vagy számítógépre alkalmazott csoportházirend-kombináció hatását határozza meg.	http://www.microsoft.com/windowsserver2003/gpmc/default.mspx
Csoportházirend objektum szerkesztő MMC modul Group Policy Object Editor MMC snap-in	A biztonsági mintabeállításokat a csoportházirend objektumba importálja.	Windows XP rendszerben
HFNetChk.exe	A rendszerhez tartozó biztonsági javítások telepítését ellenőrzi.	http://www.microsoft.com/technet/security/tools/hfnetchk.mspx
Helyi biztonsági házirend Local Security Policy	A helyi biztonsági házirendet mutatja meg és a rendszergazda meg is változtathatja.	Windows XP rendszerben (Vezérlőpult -> Felügyeleti eszközök)
Microsoft Baseline Security Analyzer (MBSA)	Biztonsági szempontból végignézi a számítógépet.	http://www.microsoft.com/technet/security/tools/mbsahome.mspx
Felügyeleti konzol Microsoft Management Console (MMC)	Modulok tárolóhelye	mmc.exe Windows XP rendszerben
Port Reporter	A TCP és UDP portok használatát naplózza	http://tinyurl.com/c3l2e
Rendszerleíró-adatbázis szerkesztő Registry Editor	A rendszerleíró-adatbázis bejegyzéseit a rendszergazda megnézheti és módosíthatja.	regedit.exe és regedit32.exe Windows XP rendszerben

Az eszköz neve	Feladata	Helye
Távoli telepítés Remote Installation Services	A Windows XP automatikus telepítése távoli rendszerből.	Windows 2000 és Windows 2003 rendszerekben
Biztonsági konfiguráció és elemzés MMC modul Security Configuration and Analysis MMC snap-in	Összehasonlítja a rendszer jelenlegi biztonsági beállításait a mintabeállításokkal.	Windows XP rendszerben
Biztonsági mintabeállítás MMC modul Security Template MMC snap-in	A biztonsági mintabeállítások megtekintését, megváltoztatását és alkalmazását teszi lehetővé a rendszergazdák számára.	Windows XP rendszerben
Rendszerelőkészítő-eszköz Sysprep	Az XP képet más rendszerekre klónozza.	sysprep.exe Windows XP rendszerben
Windows Update	Megnézi, hogy van-e frissítés, lehozza és telepíti azt.	http://windowsupdate.microsoft.com
Windows Firewall	Tűzfal	Windows XP rendszerben
Microsoft AntiSpyware	Kémszoftverek elleni védelem	

14. Táblázat: A Windows XP biztonsági beállításainál alkalmazott eszközök

7.3.1.5. A Windows XP rendszerben használt portok

Port	Protokoll	Szolgáltatás	Leírás
21	TCP	FTP	File Transfer Protocol service
23	TCP	Telnet	Telnet service
68	UDP	DHCP	Dynamic Host Configuration Protocol client
80	TCP	HTTP	HyperText Transfer Protocol service
123	UDP	NTP	Network Time Protocol client (Windows Time)
135	TCP	epmap	DCE Endpoint Resolution (remote procedure call)
137	UDP	NetBIOS-ns	NetBIOS Name Service
138	UDP	NetBIOS-dgm	NetBIOS Datagram Service
139	TCP	NetBIOS-ssn	NetBIOS Session Service
161	UDP	SNMP	Simple Network Management Protocol
213	UDP	IPX over IP	Client Service for Netware Service
443	TCP	HTTPS	HTTP over SSL server
445	TCP, UDP	microsoft-ds (SMB)	Microsoft Common Internet File System (CIFS)
500	UDP	IKE	Internet Key Exchange (gyakran az IPSec-kel együtt alkalmazzák)
515	TCP	LPR	Print Spooler service
522	TCP		NetMeeting client
1503	TCP		NetMeeting client
1701	UDP	L2TP	Layer 2 Tunneling Protocol client
1720	TCP		NetMeeting client
1723	TCP/UDP	PPTP	Point-to-Point Tunneling Protocol client
1731	TCP		NetMeeting client
1900	UDP	SSDP	Simple Service Discovery Protocol
2001 2120	UDP		Windows Messenger voice calls
2869	TCP	UpnP	Universal Plug and Play
3002	TCP		Internet Connection Firewall/Sharing
3003	TCP		Internet Connection Firewall/Sharing
3389	TCP	RDP	Remote Protocol Desktop service

Port	Protokoll	Szolgáltatás	Leírás
5000	TCP	UpnP	Universal Plug and Play
6801	UDP		Windows Messenger voice calls
6891 6900	TCP		Windows Messenger file transfers
6901	TCP/UDP		Windows Messenger voice calls

15. Táblázat: A Windows XP rendszerben használt portok

7.3.2. Linux megerősítése

Először a Linux rendszerekre jellemző biztonsági alap problémákat elemezzük ki. Mivel a Linux gyakorlatilag csak egy rendszermag neve, pontosabb, ha a GNU/Linux elnevezést használjuk, mely a Linux kernelre épülő GNU (GNU's Not Unix – http://www.gnu.org) rendszerbe, vagyis a szabad szoftverek családjába tartozó programok összessége.

A GNU/Linux rendszerek hálózati problémái az Internet protokolljainak nyitottsága miatt nem mutatnak különös jellegzetességeket, sőt elmondható, hogy napjaink (az alternatív BSD rendszerek mellett) egyik legpontosabban implementált hálózati protokollverméről van szó.

A rendszer alapvető biztonsági problémáinak forrásaként az Unix rendszerek örökségeként felfogható POSIX megfelelőségi követelményeket kielégítő alrendszerek jelölhetők meg. Ezek közül a legfontosabb, alapvető korlátokat bevezető rendszer a hozzáférés ellenőrzési mechanizmusok és az ezekhez tartozó információk struktúrája. (Itt jegyzendő meg, hogy a Windows NT és az arra épülő operációs rendszerek is a POSIX konformancia terheit nyögik biztonsági szemszögből)

A GNU/Linux központosított biztonsági modellel rendelkezik, a rendszer központja a minden művelet elvégzésére jogosult, 0 azonosítójú, általában root elnevezésű felhasználó. A root felhasználó birtokában vannak a rendszer állományai és leírói, vagyis a root jogosultságainak megszerzése az egész rendszert alkotó állományhierarchia fölötti összes jog megszerzését jelenti. Futás közben a rendszer központi irányító processze, az init, szintén a root jogosultságaival fut, emiatt a teljes rendszer minden tényezőjének módosítására képes, a minden esetben közvetetten vagy közvetlenül belőle származó egyéb processzek nem kötelesek lemondani jogaikról.

A GNU/Linux a központi root jogosultságait két szintű szerkezetben próbálja felaprózni. A legalsó alanyi szintre kerülnek az alapvető, de még mindig meglehetősen tág, jogosultsági körrel rendelkező felhasználók. A felhasználók által futtatott processzek számára a rendszer kommunikációs képességei bizonyos korlátozások alá kerülnek, vagyis a processzek közti, valamint a processzek és a külvilág, a processzek és a fájlrendszer között nem minden hozzáférési próbálkozás lesz sikeres (például: privilegizált portok megnyitása, bizonyos IPC műveletek, mások tulajdonában levő fájlok manipulálása). A felhasználói jogosultságok kibővítésére egyetlen szint, a csoportrendszer szolgál, mely lehetővé teszi egyes felhasználói alanyok egy nevesített halmazba való rendelését, az így létrejött összetett alanyokhoz azután a felhasználókéhoz hasonló felbontással rendelhetők jogosultságok.

A fenti jogosultságrendszer problémái, a rendszer erőforrásainak kisfelbontású hozzáférésszabályzása mellett (például: egy csoportos vagy egyedi alany és egy fájlrendszer-objektum között csak három jogosultság: olvasás, írás és a túlterhelt jelentéskörrel bíró futtatás jogosultság teremt kapcsolatot) bevezetett kiegészítő megoldások, például a szintén a jogokról való lemondásra és a korlátozások öröklésére épülő képességek (POSIX capabilities) rendszere, vagy a fájlrendszer objektumaihoz rendelt alanyok számának 3-ról (tulajdonos, csoport, mások) változtatható mennyiségűre növelése (POSIX ACL), a felhasználható lemezte-

rület, vagy egyéb erőforrások egyszerű limitálása (quota, user limits), különböző kiegészítő privilégium-emelő és korlátozó (például: suid, chroot, sticky bit) megoldások nem oldják meg az alapvető gondokat, hanem inkább a felbukkanó problémák helyi orvoslásával egy részben átfedéseket tartalmazó foltrendszert alkotnak, mely összességében áttekinthetetlen és elosztott adminisztrációjával még ronthatja is a biztonsági szintet.

A klasszikus POSIX-szerű jogosultságrendszer problémái leginkább a rendszer felhasználóinak, processzeinek egymástól történő elválasztásában és korlátozásában jelentkeznek. Nehezen befolyásolható, hogy egy futó szolgáltatásnak, vagy egy távoli terminálról bejelentkező felhasználónak pontosan milyen tevékenységeket legyen szabad elvégeznie, emiatt általánosan jellemző, hogy a közvetlen és egyszerű károkozási lehetőségeket, valamint az előzőleg említett, helyileg kezelni próbált egyéb speciális problémákat kizárva a rendszeren futó processzek a szükségesnél jóval nagyobb jogosultsági körrel rendelkeznek.

Azt, hogy ez a megközelítés milyen káros lehet egy több felhasználós, hálózati szolgáltatásokat nyújtó rendszer esetében, példaként az alapvető biztonsági gondokkal küzdő Windows9x rendszerek bizonyíthatják, azonban a Linux rendszer esetén is számos, távolról is kihasználható súlyos sebezhetőség működése épül a támadási felületet szolgáltató processzek elszigeteltségének hiányára és a szükségtelen járulékos jogosultságok, magasan privilegizált futtató felhasználók jelenlétére. A problémára csak részleges és nehézkesen adminisztrálható megoldást szolgáltat a futási környezet chroot rendszerű elkülönítése, ráadásul a chroot börtönből való kitörésre számos lehetőség nyílik a root jogosultságainak segítségével.

7.3.2.1. Alapvető biztonsági követelmények

Ahogy láttuk, a biztonsági problémák javarészt a biztonságtechnikában alapelvként alkalmazható, a minimálisan szükséges jogosultságok biztosításán alapuló, rendszer hiányából származik, így célszerű a lehetőségekhez mérten a rendelkezésre álló POSIX-jellegű jogosultságrendszer eszközeivel élve a lehető legnagyobb elkülönítési szintet elérni, különösen akkor, ha a rendszer nem megbízhatónak tekinthető felhasználók számára is shell, vagy azzal egyenértékű, például a felhasználó által szerkeszthető dinamikus, scriptelhető (például `PHP`) weboldal szolgáltatást nyújt.

A konkrét beállításokat a telepítés során a fájlrendszer felépítésével kezdhetjük. A nem ismerten korlátos tárigényű processzek (például feltöltést is engedélyező fájlkiszolgálók, leveleket fogadó rendszerek, terminálszerverek) esetén a `quota` rendszerű és a particionálásos helykorlátozási megoldások felhasználása szükséges, hisz a korlátlan növekedés a hely elfogyásával, a hely elfogyása pedig a DoS támadás lehetőségével, vagy egyéb, nem üzemszerű működésből származó biztonsági problémák felmerülésével járhat.

A `quota` rendszer esetén egyes felhasználók, vagy csoportok számára minden partíción külön szabályozhatjuk a maximálisan foglalható blokkok és inode bejegyzések számát. Ez a módszer könnyű változtatásokat tesz lehetővé és a soft limitek segítségével a helyproblémák rugalmasabb kezelésére alkalmasak, a VFS különböző pontjain felcsatolt, az igények szerint méretezett partíciók pedig a funkciók szerinti differenciálást teszi lehetővé (például a `/var/mail` könyvtárba mailbox tárolási rendszer szerint csak kettő fájl tartozhat felhasználónként).

A felhasználók szétválasztására célszerűen minden egyéni home könyvtár csak a tulajdonos számára legyen olvasható, illetve a felhasználók által írható könyvtárak a célszerűen leválasztott merevlemez partíciókon elhelyezkedő fájlrendszerek „`noexec`“ (nem indítható fájlokat tartalmazó) és „`nodev`“ (nem speciális eszközöket tartalmazó) opciókkal történő

csatolását teheti célszerűvé, azonban ez bizonyos legitim rendszerfolyamatok hibamentes működését, valamint például a felhasználói profilok testreszabását, saját bináris programok használatát is megakadályozhatja.

Törekedni kell arra, hogy a rendszer folyamatosan futó, kiszolgáló tevékenységet folytató, komponensei is a lehető legalacsonyabb privilégiumszinten fussanak, egy erre a célra elkülönített felhasználó neve alatt, lehetőség szerint `chroot` környezetben, így a felhasználók és processzek szétválasztása gyakorlatilag egy feladattá olvad össze. Az ennek megfelelő módon felépített szoftverek az emelt privilégiumszintet igénylő műveletekhez külön, egyszerű és könnyen auditálható segédprogramokat tartalmaznak (a felhasználók számára elérhető példaként a privilegizált műveleteket végrehajtó passwd programot említhetjük meg).

Egyértelműsége mellett is kiemelendően fontos biztonsági követelményként kell foglalkoznunk a szolgáltatások minimalizálásával is: ne telepítsünk rendszerünkre olyan – különösen magasan privilegizált – programokat, melyek használata nem alapvető feltétele a rendszer helyes működésének, ha lehetőség van rá, ne csoportos, hanem egyéni korlátozásokat vezessünk be rendszerünkben.

Elmondható, hogy az alapvető biztonsági követelmények a need-to-know elv követését és betartását jelentik, azonban emellett nem szabad megfeledkeznünk a rendszeres szoftverfrissítésről és a biztonsági figyelmeztetéseket publikáló listák rendszeres figyelemmel kíséretől sem.

7.3.3. Digitális aláírás alkalmazása

Az előző fejezetben sok szó esett a digitális aláírásról és a törvényi háttérről, ami azóta változott is itthon, és 2004-ben, majd 2005-ben egyre több alkalmazás jelenik meg a technológia előnyeinek kihasználására. A legkorszerűbb megoldások az intelligens kártyán vagy USB tokenen hordozható és az eszköz memóriájából a titkos kulcsot ki nem adó rendszerek, de ezek ára már az igényelt hardver eszközök miatt sem mindig megfizethető egy intézmény számára.

Ebben a részben egy nem kereskedelmi célokra szabadon elérhető alkalmazást mutatunk be a digitális aláírás használatára.

7.3.3.1. Digitális aláírás Windows rendszeren (PGP)

A digitális aláíráshoz használt PGP szoftver (a szabadon elérhető 8.0.2 verzió Windows2000 rendszerre, mivel a jelenleg elérhető 9.0 verzió már csak 30 napos kipróbálásra érhető el ingyenesen) képernyőképekkel lépésről-lépésre bemutató telepítését és használatát a következő címen lehet elérni: http://www.cert.hu/ismert/3jelszo/pgp_win.html

7.3.3.2. Digitális aláírás Linuxos rendszeren (PGP, GnuPG)

Manapság már Linux alatt is egyre ritkább, hogy a PGP-t vagy a szabadon elérhető GnuPG-t parancssoros módban használnák, de amennyiben erre lenne szükség, úgy a megfelelő man gpg oldalak illetve a program saját súgója (pl. gpg –h) elérhetők ehhez.

Grafikus e-mail kliens esetén (pl. Kmail) elérhető olyan leírás, melyből a rendszer alaplogikája világossá válhat. Magyar nyelven a SuSe rendszer felhasználói leírásából ajánlható az ezt leíró rész: http://href.hu/x/cpr

7.3.4. Öntesztelés

Több olyan alkalmazás létezik, mely az adott rendszert átnézi, és a biztonsági problémákra felhívja a figyelmet. Ezek rendszeres használata segíti, hogy azokat a biztonsági problémá-

kat is felfedjük és kijavítsuk, melyek elkerülték figyelmünket. Nagyon fontos, hogy ne kerülje el figyelmünket az alkalmazás frissítése, mert ezek is csak programok, melyeknek van következő (remélhetőleg jobb) változata.

7.3.4.1. Öntesztelés Windows rendszeren – MBSA

A Windows rendszerekhez a Microsoft által ingyenesen elérhetővé tett MBSA (Microsoft Baseline Security Analyzer) alkalmazás több egységében is képes ellenőrizni az adott rendszert vagy akár több gépet is[8], hogy a rendszer beállításai mennyire biztonságosak és a frissítések mennyire naprakészek. Az MBSA telepítését és használatát írja le lépésről lépésre ez az anyag: http://www.cert.hu/ismert/10altalanos/win2k_mbsa.html

A teszt elvégzésének eredménye többek között egy olyan lista a talált problémákról, melyen szerepel a megoldásokhoz vezető út is. Ezen lépések helyességét az önteszt újrafuttatásával ellenőrizhetjük.

7.3.4.2. Öntesztelés Linux rendszeren

Mivel a Linux rendszeren elérhető megoldások között nem létezik az MBSA-nak megfelelő eszköz, ezért az öntesztelés összetettebb módon a 7.6.1 és 7.7.3 fejezetekben került leírásra.

7.3.5. A rendszerelemek kivonása

Amikor valamilyen okból kifolyólag hardvercsere történik egy intézményben, akkor a cserélni kívánt eszközöket sem szabad csak egyszerűen kidobni. Amennyiben biztonsági és adatvédelmi okokból is megvizsgálták, hogy nincs erre érzékeny adat az eszközökön, úgy funkcionalitását illetően még lehetséges, hogy szükség lesz valamilyen adatra, melynek hiánya közvetve a rendelkezésre-állást befolyásolhatja. Éppen ezért, ha van rá lehetőség, készíteni kell a cserére szánt adathordozóról egy másolatot (ez lemezek esetében egyszerű és olcsó, merevlemez esetében érdemes előszűrni a mentésre szánt adatokat), és ezeket adott ideig még megőrizni, hátha szükség lesz valamire az elmentett adatok közül.

Az ilyen módon tárolt adatokra és adathordozókra azok tartalmának érzékenységétől függően kell alkalmazni a védekezést a tárolás/megőrzés során. Kidolgozható valamilyen jelrendszer is, amivel az egyes lemezeket kívülről is jelölik pl. valamilyen színű matricával.

Amennyiben kikerülnek az intézmény birtokából ezek az adathordozók, úgy a megfelelő törlési eljárásokról gondoskodni kell (bár vannak cégek, akik a többszöri formázás után is vissza tudják hozni az addig tárolt adatokat, de a legolcsóbb és sokak ellen elégséges megoldások egyike a formázás), és csak ezek után bocsáthatók el a saját rendszerből ezek az adathordozók.

7.4. Egyéb előnyös biztonsági beállítások

Minden rendszer biztonsága fokozható a teljes használhatatlanságig – tartja a mondás, ezért óvatosan bánjunk a biztonsági beállítások fokozásával, mert mindig van még mit és hova fokozni, de csak akkor érdemes, ha ez valóban szükséges, és a munkavégzést nem hátráltatja annyira, hogy a fokozás visszaüssön, és a végén a rendszer nem tudja célfeladatát ellátni.

[8] figyelembe kell venni a gépen vagy a többi gépen meglévő tűzfal-jellegű beállításokat, hogy egyrészt sikeresen le tudjon futni az ellenőrzés, másrészt a többi gép ne támadásnak érzékelje a vizsgálatot.

7.4.1. Linux

A GNU/Linux rendszerek számára elérhető számos továbbfejlesztett hozzáférés-védelmi megoldás közül a ToReS mintarendszer számára a grsecurity (http://www.grsecurity.net) biztonsági rendszert választottuk, mivel egy teljes körű, egyszerűen adminisztrálható rendszer képét mutatja, annak ellenére, hogy számos független és céljaiban is elkülönült fejlesztőcsapat munkáinak eredményeit is tartalmazza.

A grsecurity programcsomag alapvetően egy kernel patchből és egy adminisztratív programból áll. A patch számos fontos szigorítást, korlátozási lehetőséget tesz lehetővé, valamint az előre kiszámított címekkel vagy számlálókkal dolgozó támadási módok megnehezítésének céljából számos folyamat véletlenszerűségének mértéket növeli a Linux kernel működése során, egy úttal auditálási lehetőséget is biztosít a rendszerfelügyelet céljaira.

Kiemelendő még az önálló alrendszert alkotó, a klasszikus puffertúlcsordulási hibáktól és egyéb, a memóriavédelem eszközeivel megakadályozható támadási lehetőségektől védelmet biztosító PAX (pageexec) nevet viselő, hazai fejlesztésű módosítás-gyűjtemény, mellyel, igaz, megváltoztatott programfordítási és futtatási mechanizmusok bevezetését megkövetelve, külön védelmi tényezőkkel ellátott bináris állományok készíthetők.

Az adminisztratív program a rendszer processzeinek és felhasználóinak („subject“) a rendszer erőforrásaihoz („object“) és a kernel speciális szolgáltatásaihoz való viszonyát egy ún. szerep alapú hozzáférési rendszer (RBAC – Role Based Access Control) keretein belül. Az RBAC rendszer nagy előnye a kezdő biztonsági adminisztrátorok számára, hogy rendelkezik egy „tanuló“ üzemmóddal is (`gradm -L` parancs), mely során gyűjtött információk alapján a rendszer működését leíró subjectek és objectek automatikusan is meghatározhatók.

7.4.2. Windows

A különböző biztonsági beállításokat és lehetőségeket részletesen ismerteti a 7.3.1 fejezet, így csak azok a kiegészítések szerepelnek, melyek ma már részei az operációs rendszernek (pl. felvásárolt spyware irtó cég, vagy beépített tűzfal) vagy a rendszerhez szabadon elérhetők. A http://www.cert.hu/eszkoz/?cat=1ingyen címen bővebb lista is elérhető, de a népszerűbbeket külön kiemeljük:

- Vírusvédelem (a legtöbb termék 30 napig ingyenesen kipróbálható, de ezek a 30. nap utána is ingyenesek):

  http://www.free-av.com/antivirus/allinonen.html

  http://www.clamwin.com/

- Kémprogramok elleni védelem (AntiSpyware – jelenleg béta verzió, és Spybot):

  http://www.microsoft.com/athome/security/spyware/software/default.mspx

  http://www.safer-networking.org/hu/index.html

- Egyéb biztonsági beállítások (Microsoft Shared Computer Toolkit for Windows XP, jelenleg béta verzió):

  http://tinyurl.com/a8vdm

Azt, hogy milyen biztonsági javító intézkedéseket tehetünk meg a rendszerünkben, egy öntesztelséssel deríthetjük ki (ld. 7.3.4 fejezetben).

7.5. Hozzáférés-védelem (access control)

A hozzáférés lehet fizikai és logikai, és általánosan elfogadott elv, hogy ha egy számítógéphez fizikailag is hozzá lehet férni („a konzolhoz”), akkor a logikai védelem másodlagos.

7.5.1. Hardver

Egyetemi környezetben is létezik már beléptetőrendszer, hiszen az egyetemi erőforrásokat nem egyetemi polgárok is szeretik igénybe venni. Ezen felül olyan értékek is lehetnek ezek a gépek, melyeket pályázatokból beszerezve rongálás vagy lopás esetén nem könnyű újra beszerezni, így a fizikai védelmük fontos.

Az épület és a gépterem megfelelő védelme a villámvédelemtől a tűzoltó berendezésig terjed, és az adott gép konfigurációját védő eszközök is elérhetők. Ilyenek a kábelbilincsek, vagy a számítógépház megbontását jelző riasztók is. Az operációs rendszerek szintjén már többnyire logikai lehet a védelem, de token alapú (USB, intelligens kártya) megoldással is vegyíthető a védelem megszervezése.

7.5.2. Linux

A Linux rendszerek alapvető felhasználói hozzáférés-védelmi mechanizmusát a PAM (Pluggable Authentication Modules) alrendszer szolgáltatja, mely segítségével a jelszóadatbázisok széles skálájával azonosíthatók a rendszerbe belépni kívánó felhasználók. A PAM modulok alkalmazásfüggő konfigurációs állományai az `/etc/pam.d` könyvtárban találhatók, különösen érdemes figyelemmel lenni az `ssh` és `login` programok konfigurációjának tekintetében, mivel általános esetben ezek biztosítanak közvetlen hozzáférést a rendszerhez. Az `ftp`, és más, hálózaton keresztül nyílt jelszavakkal dolgozó szolgáltatások PAM konfigurációjában érdemes a shell elérésre is jogosult felhasználók azonos jelszóval történő belépését tiltani a jelszólehallgatás hatékonyságának csökkentése érdekében.

A rendszer hálózati védelmét a PAM-ra épülő alkalmazásokon kívül általános tényezőként kiegészíti a rendszer alapját alkotó LIBC rendszer által biztosított `TCP wrappers` szolgáltatás, mely a rendszerben futó, LIBC függvényeket, vagy a külső `TCPD` wrappert használó alkalmazások által nyitott szolgáltatási pontokra csatlakozó kliensek előszűrését teszi lehetővé a `/etc/hosts.allow` és `/etc/hosts.deny` konfigurációs állományok segítségével.

A legkülső és legáltalánosabb hálózati hozzáférés-védelmi réteg az iptables, vagyis a Linux kernel állapottartó csomagszűrő szolgáltatása, mely az `iptables` nevű parancs segítségével karbantartható, soronként feldolgozott táblázatok által megszabott elemi műveletek végrehajtását végzi a rendszerbe beérkező, az áthaladó és a kimenő IP adatforgalmon.

7.5.3. Windows

Az azonosítást[9] (authentication) követheti a hitelesítés avagy tanúsítás (certification), majd ezután következik a felruházás (authorization), amikor a jogosultságok (access rights) kerülnek kiosztásra a beállított jogosultság-kezelés vagy lista szerint (access control, access control list).

Az alapelveket a gyártó honlapján is elérhetjük (http://tinyurl.com/6c8gb), és bár a technikai részletek a Windows 2000 rendszerhez készültek, ezekből emeljük ki rövid összefoglalásban a fontosabb és más rendszerben is érvényes elveket, míg részletesebb XP-specifikus beállítások is elérhetők a gyártó lapjáról (http://tinyurl.com/brnad).

[9] A főbb technikákról röviden: http://www.cert.hu/ismert/3jelszo/azonositasok.html

1. Rendszergazdai azonosítót csak a következőkre használjunk, más feladatokra legyenek megfelelő jogosultsággal beállított felhasználó azonosítók a rendszerben:
   - rendszer telepítése és biztonsági frissítések telepítése, rendszer frissítése és javítása;
   - beállítások végzése, jogosultságok menedzselése;
2. A felhasználókkal és a munkájukkal szemben két fontos alapkövetelmény, hogy:
   - legyenek tagjai a felhasználók csoportjának;
   - a számukra szükséges alkalmazásokat tudják elérni/futtatni;
3. A lehetséges biztonsági beállításokat már a telepítéskor érdemes beállítani (pl. NTFS fájlrendszerre történő telepítés)
   - az egyes felhasználók saját könyvtárához (HKEY_Current_User) és profiljához (%UserProfile%) teljes jogot kell adni, míg a Dokumentumok vagy Temp mappákhoz legfeljebb létrehozási és olvasási jogokat;
   - alapjában véve nincs szükség a guest és a power user azonosítókra/csoportra (a guest belépési lehetősége letiltható), a Rendszergazdai és a felhasználói azonosító és csoport a legtöbb esetben elégséges;
4. Munka közben, ha szükség van a rendszergazdai jogkörre, lehetőség van a runas parancs segítségével végrehajtani a megbízható alkalmazást, így elkerüljük a ki-bejelentkezés kényelmetlenségét, és nem kell teljes jogkörű rendszergazdaként dolgoznunk a gépen, csak akkor, amikor erre szükség van:

```
RUNAS /u:gép_neve\Rendszergazda parancs
 RUNAS /u:computername\administrator cmd
```

5. Figyelemmel kell lenni az öröklésekre, így a hierarchiában a megfelelő szinten kell beállítani a jogokat és a rekurzivitást (minden mappában lévő elemre vonatkozzon-e vagy sem az adott beállítás). Ezt segíti az Windows XP Professional-ben lévő Effective Permissions ablak, ahol megtekinthetjük, hogy az adott felhasználónak melyek a tényleges jogai, és melyeket kapja az ebbe a csoportba sorolt felhasználó.
6. Időnként elemezni kell az Event Viewer (Eseménynapló, parancssorból: eventvwr) üzeneteit, melyek különböző szintű (és eszerint súlyosságú) üzenetekben tájékoztatnak a rendszerben történt eseményekről.
7. Használjuk a „Security Configuration and Analysis snap-in" eszközt, melynek használatához bevezető leírás található a http://tinyurl.com/6u2jp címen (indítása parancssorból: mmc /s), és az első lépések megtétele után az alkalmazásban megjelennek a további teendők és lehetőségek leírásai, melyek segítségével ellenőrizhetjük rendszerünk biztonsági beállításait (az előre elkészített minták alapján az ellenőrzés viszonylag sok időt és erőforrást igényel, így lehetőség szerint ezt ne munka közben végezzük). Az eredmény fájlba is menthető (a futás elején rákérdez a fájl nevére).

7.6. Megfigyelés és elemzés (monitoring)

A leggyakoribb hálózatmonitorozó eszközök az Ethereal, és a Nessus akár mindkét platformra is elérhető. Ezeken kívül a Windows rendszerekhez a Microsoft termékeinél is jobban használható Sysinternals segédprogramok ajánlhatók. A http://sysinternals.com lapon más hasznos segédprogramok is elérhetők többféle operációs rendszerhez az informatikus szakemberek életének könnyítésére.

7.6.1. Linux

A megfigyelés és elemzés elsődleges felülete a `syslog`, a rendszer naplózó processze. A legtöbb alkalmazás rendelkezik valamiféle `syslog` interfésszel, melyen keresztül a működését érintő alapvető eseményekről üzeneteket helyezhet el a rendszer központi naplógyűjtőjében, mely aztán kategóriák alapján szétbontva fájlokban tárolja, a konzolra nyomtatja és/vagy hálózaton tovább is küldheti egy központi gyűjtőállomás számára. A központosított, megbízható naplógyűjtésre a `syslog-ng` program felhasználását ajánljuk, azonban az egyszerű, helyi elemzésre szánt naplógyűjtésre megfelelnek az alapértelmezett syslogd és a klogd programok is.

A syslog és az egyéb, önálló naplógyűjtést végző programok fájljait a `/var/log` könyvtárban, vagy alkönyvtáraiban találhatjuk meg. A naplófájlok elemzésére a `logwatch` programot futtathatjuk, mely különböző alkalmazások számára előkészített összesítő modulok használatának segítségével napi jellemző összesítésekkel segíti a rendszer állapotának folyamatos megfigyelését.

A naplófájlok szabályalapú szűrésére a `logcheck` program használatát ajánljuk, mely az `iptables` logikájához hasonlóan a figyelmen kívül hagyást engedélyező szabályokat felsorakoztató táblázatok formájában tárolt konfigurációja alapján a rendszerbe érkező kritikus, vagy szokatlan naplóbejegyzések kiemelését végzi.

A rendszer működésének megfigyelése és elemzése a Linux rendszerben is, a műveleteket segítő alkalmazások használata mellett, az éber emberi közreműködést igényli. Ezen éberség megtartásának érdekében célszerű a rendszer automatikus szűrési mechanizmusait a lehetőségekhez mérten pontosan és szigorúan beállítani, hogy a megfigyelés során valóban az emberi intelligenciát igénylő feladatok elvégzésére fordulhasson a biztonságos működés fenntartására szánt idő.

7.6.2. Windows

A Sysinternals programoknak része több olyan *mon végű program, ami valamit monitoroz, így egy szakember a vírusok vagy más kártékony programok utáni nyomozáskor tudja követni, hogy az hova és mibe ír bele (pl. jelentéseket a program készítőjének az adott rendszerben talált hiányosságokról), miből veszi az adatait (pl. más támadásra kiszemelt rendszerek adatait, mint az IP címek), milyen portokon keresztül kommunikál stb.

Ezekkel az eszközökkel lehet védekezni is (processzek leállítása, fájlok átnevezése, portok lezárása stb.), amíg nem áll rendelkezésre egy irtó program, vagy a vírusirtónk nem végzi el az irtást, amint elérhető és telepítésre került az irtáshoz szükséges teljes programkód vagy az ismert vírusmintákat tartalmazó adatbázis frissítése.

A futó processzek listájában (Task Manager) láthatjuk a rendszerben működő alap processzeket és alkalmazásokat, de amennyiben elemezni szeretnénk az aktuális listát (ne frissüljön), vagy elmenteni egy fájlba, akkor a Sysinternals PSTools eszközcsomagját használhatjuk (pl. a `pslist > processzek.txt` parancs eredménye lesz egy `processzek.txt` fájl, benne a parancskiadáskor futó processzek listája.

Részletesebb elemzés esetén, amennyiben az is érdekes lehet, hogy mely processz vagy alkalmazás melyik másikkal működik együtt, vagy melyiket melyik hívta meg stb. ajánlatos a Sysinternals Process Explorer-t használni. Ebben hierarchikusan szerepelnek a futó processzek, egymáshoz való viszonyuk is látható, és többek között a prioritásuk is állítható. Mindez igényel egy felhasználóinál nagyobb szakértelmet az adott rendszerben.

7.7. Integritás-ellenőrzés

Természetesen infrastruktúránk felügyeletének alapjait annak építőköveinek, vagyis a benne található számítógépeknek a folyamatos kontrollja, megfigyelése tartozik. Feladatunk lényege az, hogy folyamatosan tisztában legyünk azzal, hogy mi történik az egyes hosztokon.

Ehhez a következő részterületeket kell folyamatosan figyelemmel kísérnünk:

- Integritás-ellenőrzés, vagyis a hoszton található fájlok változásának felügyelete. Természetesen nem elvárható hogy ne legyen változás, de a rendszer részét képező fájlok változását észre kell vennünk.
- A hoszton futó alkalmazások figyelemmel kísérése, ill. az ezek működése közbeni anomáliák észlelése.
- A biztonsági kockázatok időszakos felmérése.

Az alább ismertetett eszközök használatakor, ill. feltörés gyanús esetben figyelembe kell venni, hogy ha a támadó root jogosultságokat szerzett, akkor rosszindulatú kernelmodulok segítségével sajnos azt „hazudik", amit csak akar, így minden információt – illetve információ hiányát – fenntartásokkal kell kezelni.

7.7.1. Integritás-ellenőrzés Linux rendszeren

A fájlok változásának követésére linuxos környezetben több megoldás is létezik. Ezek egy része úgy működik, hogy egy adatbázist hoz létre a fájlrendszerről, amelyben eltárolja a különböző fájlok valamilyen hashelt értékeit, majd a későbbiekben ezekkel veti össze az aktuális fájlrendszert. Fontos, hogy mivel itt az ellenőrzés alapját az elkészült adatbázis képezi, ezért ha azt nem normál működés közben elérhetetlen helyen (például másik gépen, lemezen stb.) kerül tárolásra, akkor a gép kompromittálódása esetében ez az adatbázis módosítására ad lehetőséget. A külön adattárolón, vagy egyéb módon (pl. kulccsal védett) adatbázisok használata, viszont jelentős többletadminisztrációt eredményez, hiszen minden adatbázis frissítésnél kézzel kell beavatkozni. A tapasztalatok azt mutatják, hogy általában a támadások nagy része e nélkül is észrevehető, mondjuk óránkénti ellenőrzés mellett, és a befektetett pluszmunka nem éri meg az esetleges nyereséget, de természetesen magunknak kell mérlegelnünk, hogy a tárolt adatok/nyújtott szolgáltatások mellett megelégszünk-e a kevésbé biztos módszerrel. (Hozzátéve, hogy ezen ellenőrző programok is becsaphatóak, ha hamis információt kapnak a fájlról).

A mintarendszer az AIDE nevű integritás ellenőrző programot használja a fentiek megvalósítására.

A másik rendelkezésre álló lehetőség a rendszerünkön található fájlok összevetése a disztribúció csomagjaiban találhatóakkal. E módszer használatával is kiszűrhetőek a kicserélt állományok. Előnye, hogy nincs szükség adatbázis karbantartására, hiszen az a disztribúció csomagjai alapján elkészíthető, hátránya viszont, hogy csak az szerepel benne, ami része a terjesztésnek.

A mintarendszer a Debian csomagkezelőjére épül, az erre a célra használandó parancs ebben a csomagkezelőben közvetlenül nem található meg, e célra a debsums nevű program használható. Megjegyezzük, hogy a számos disztribúció alapját képező rpm csomagkezelőben a –verify kapcsoló biztosítja a fenti funkcionalitást.

7.7.2. Logelemzés

A hoszt működésének figyelemmel kísérése a minden Linux rendszeren nagy számban megtalálható naplófájlok nyomon követésén keresztül valósítható meg. Tekintve azonban, hogy a normális működés is számos bejegyzést eredményez, érdemes olyan logelemző programokat használni, amelyek figyelik az anomáliákat, esetleg összesítik az eredményeket.

Mintarendszerünkben erre a célra két csomag elérhető. Az egyik a logcheck, a másik a logwatch. Az előbbi egyszerűbb funkcionalitással bír, arra képes, hogy a logokat átvizsgálva bizonyos mintákat (melyek a normál működéshez tartoznak) kihagyjon, másokat pedig külön kiemeljen, mint a biztonsággal kapcsolatos „gyanús" sort, és az így szűrt és rendezett maradékot küldje el e-mailben.

A logcheck nagy előnye, hogy könnyű testre szabni. Az /etc/logcheck/ könyvtár alkönyvtáraiban található fájlokba kell beírni azokat a reguláris kifejezéseket, amelyek illeszkedése esetén az adott akciót hajtja végre a program.

A logwatch egy fokkal kifinomultabb eszköz, az a kapott eredményeket szolgáltatásonként képes elemezni, és összesíteni, ezzel könnyebbé téve a kapott adatok áttekintését, azonban működésének módosítása közel sem olyan egyszerű.

7.7.3. Biztonsági kockázatok felmérése

A rendszereink működésének figyelemmel kísérése mellett szükség van arra is, hogy tisztában legyünk azokkal az ismert kockázatokkal, amelyeknek rendszerünk ki van téve, és lehetőség szerint ezeket küszöböljük ki. Amellett, hogy érdemes a szolgáltatások nyújtására használt programok kockázatainak, biztonsági „előéletének" utána nézni, vannak ezen kockázatok felderítésére szolgáló automaták.

Alapvetően kétféle ellenőrzési lehetőségünk van, a belső, és a külső ellenőrzés. Az előbbire a Tiger nevű programot javasoljuk, mely különböző biztonsági szempontból hibás beállítások felderítésével, ill. időszakos ellenőrzésekkel segít minket. Megjegyzendő, hogy a Tiger képes pluginként használni mind az Aide-t, mind az Integrit-et és a Tripwire-t is.

A külső ellenőrzésre úgy kerülhet sor, hogy egy külső gépről vizsgáljuk meg a célpontot, és az így nyert információk alapján keresünk ismert biztonsági hiányosságokat. Ilyenkor figyelembe kell venni, hogy az ellenőrző gép ugyanazt látja-e mint valamely hálózaton kívüli gép. Ha nem, akkor – valószínűleg – több lehetőségünk van vizsgálódásra, ha viszont igen, akkor esetleg lényeges dolgokat nem fogunk észrevenni. Erre a célra a legelterjedtebb eszköz a Nessus, mely a vizsgálat után összefoglalja számunkra az ismert problémákat.

7.8. Vizsgálat, bizonyíték-gyűjtés, igazságügyi eljárás

Egyes esetekben az informatikai biztonsági esemény annyira súlyos, hogy a törvényi szabályok szerint igazságügyi szakaszba lép az ügy. Akkor is hasznos lehet az ilyen eljárások alkalmával követendő lépések ismertetése, ha egy intézmény házon belül akarja elrendezni a kellemetlen helyzetet (pl. fegyelmi eljáráshoz szükséges bizonyíték-gyűjtés).

7.8.1. Észlelés és lehetséges incidensek

A biztonsági problémára utató leggyakoribb jel a hálózati forgalom váratlan megnövekedése. Amennyiben ez a támadók miatt van, akkor mélyen bent vannak a rendszerben, és a legkülönfélébb dolgokat művelhetik: szerzői jogvédelem alá tartozó alkotásokat cserélnek, másokat támadnak (persze, a mi nevünkben), reklámlevelekkel árasztják el a hálózatot stb.

A másik jellemző tünet, amit erős behatolásérzékelő vagy tűzfal nélkül is észre lehet venni, az a váratlan forgalom megjelenése. A SANS ezt nevezi a „hazai pálya előnyének", vagyis, a saját rendszerben előforduló eseményeket ismerve hamar feltűnik, ha valami az eddigiektől eltérően működik. Amennyiben a forgalom mennyisége tér el, annak okát kell kideríteni, de amennyiben pl. számos NetBIOS kommunikáció észlelhető egy menedzsment interfészen, ami normális esetben kódolt kapcsolaton keresztül szokott megtörténni, akkor valószínűleg biztonsági problémánk van.

Annak megértéséhez, hogy egy esemény normális vagy sem, ismerni kell a normális működés tulajdonságait. Ennek megismeréséhez szabadon elérhető eszközök állnak rendelkezésre, melyek egy Linuxos laptopról futtathatók a legegyszerűbben. Mivel ez nem egy integrált vállalati rendszer, szükséges egy saját készlet összeállítása, melyben a hasznos alkalmazások megtalálhatók. Az Internetről letölthető többféle ilyen összeállítás, de idővel úgyis saját szükségletek szerint válogatjuk össze magunknak, mindazt ami bevált. Néhány méltán neves alapeszköz, amit a legtöbb készlet tartalmaz:

- *Ethereal*: Windows és Linux rendszerre szabadon elérhető grafikus felülettel rendelkező hálózati forgalom-elemző.

  http://www.ethereal.com/

- *EtherApe*: Jelenleg csak Linux rendszerre elérhető hálózati kommunikációs térképkészítő, kiváló segítség a normális hálózati kommunikáció meghatározásához.

  http://etherape.sourceforge.net/

- *tcpreplay*: elmentett hálózati kommunikáció „újrajátszásához" való eszköz, mellyel az újrajátszás sebessége is állítható, így segítve a tesztelés-elemzés menetét.

  http://tcpreplay.sourceforge.net/

- Dan Farmer és Wietse Venema gyűjteménye feltört UNIX-ok elemzésére.

  http://www.porcupine.org/forensics/tct.html

Az egyes eszközöket valamilyen kombinációban is lehet használni (pl. tcpreplay és EtherApe a hálózati forgalom anomáliák felderítésére). Mára olyan mértékű szolgáltatás-palettával rendelkeznek ezek az alkalmazások, hogy minden lehetőségük kihasználása csak idővel válik lehetségessé a megfelelő számú és mélységű használat után.

Nagyon fontos, hogy az adott hálózatban legyen legalább egy ember, aki használja ezeket az eszközöket, és figyeli a kiugró jelenségeket, hogy idejében nyomozni lehessen az okok után, és azonosítani lehessen a támadást.

7.8.1.1. A támadás azonosítása

Minden hálózatra kötött számítógép támadásnak lesz kitéve, hiszen manapság az automatikus pásztázó és sebezhetőség-kereső eszközök nem is foglalkoznak azzal, hogy kié a célgép. Felmérik egy-egy tartomány összetételét, felépítését, és amint megjelenik egy sebezhetőség, e korábbi felmérés eredményei alapján keresik ki a támadható gépet.

A károk csökkentésének kulcsa a támadás azonosítása és megértése, valamint a helyreállítás és a következő támadás elkerülésének lehetőségei. Tekintsük a Web-szerver szolgáltatás példáját, amikor a Web4sale.com cég hálózatán kimagasló forgalmat észlelünk. A cég egy központi pontból menedzseli a szolgáltatást, privát hálózatot használva (10.20.0.0/24). Ebből a C osztályú hálózati címtartományból várjuk el a hálózati kommunikációban résztvevők címeit. Ebből kiindulva térképezzük fel, hogy mi történhetett a cég hálózatán.

Az első lépés a gyanús forgalom okának felderítése. Az elvárt címtartományon belül is SSH-val kódolt kommunikációt tételezünk fel, így minden, ami nem ennek megfelelően zajlik, az gyanús. Egy Linuxos laptoppal és a tcpdump programmal felszerelve kiszemelünk egy hosztot, melyet megfigyelünk és csomagjait elkapjuk (sniff). A switch-elt hálózatban a router mirror portjára csatlakozva a számunkra érdekes forgalmat tükrözzük:

```
tcpdump -i eth0 -s 1500 host winbox.private.com
```

ahol eth0 a lehallgató felület és winbox.private.com a megfigyelt szerver menedzsment felülete.

A kapott adatokból két anomáliát észlelünk. Elsőként a nagymértékű NetBIOS forgalmat a privát felületen:

```
10/02/02  08:27:18  netbios.public_ip.com  137   ->   winbox.private.net  137
10/02/02  08:27:19  netbios.public_ip.com  137   ->   winbox.private.net  137
10/02/02  08:27:20  netbios.public_ip.com  137   ->   winbox.private.net  137
```

Második anomáliaként egy nyilvános IP-cím szerepel az adatokban (netbios.public_ip.com).

Első szabály: a hazai pálya előnyének kihasználása. Mivel e két anomáliának nem szabadna előfordulnia a hálózati forgalomban, biztosak lehetünk a jogosulatlan hozzáférésben. A következő lépés a Winbox gépen történt események feltárása, miszerint kívülről vagy belülről történik a jogosulatlan használat. Ehhez magát a hosztot kell vizsgálni.

7.8.1.2. Hoszt-alapú vizsgálat

Nagyon fontos tudatosítani, hogy egy vizsgálat alatt lévő hoszton semmiben sem szabad megbízni. Úgy kell tekinteni, hogy a hosztot feltörték, root-kit-et helyeztek el rajta, és figyelik azt is, hogy valaki vizsgálja-e a rendszert[10]. Elsőként el kell dönteni, hogy a gépet lekapcsoljuk a hálózatról vagy sem. Nincs egyértelmű válasz, mert ha úgy ítéljük meg, hogy a tevékenység további károkat okozhat, akkor kapcsoljuk le, míg ha a jogosulatlan felhasználó működését megfigyelve tudunk több információt szerezni és felderíteni tevékenységét és kilétét, akkor ne kapcsoljuk le a gépet a hálózatról.

Az igazságügyi vizsgálat a bűncselekménnyel kapcsolatos információ megszerzésének, rögzítésének és követésének a művészete annak érdekében, hogy egy lehetséges bírósági ügyben használni lehessen ezeket az adatokat. Ennek érdekében minden elővigyázatosságot meg kell tennünk azért, hogy az adatok pontosak, megbízhatóak, és nem módosultak az adatgyűjtés előrehaladtával. Az adatok gyűjtésének és feldolgozásának menetét rögzítve és az adatmozgatást követve (ki kezdeményezte, erre mi történt) támogatjuk a „felügyeleti lánc" megőrzését. Ez egy nem egyértelmű feladat, melyet a szakkönyvek is hosszasan taglalnak. A jó gyakorlat a bizonyítékok gyűjtésére a *megbízható eszközök használata*, *a mobil adathordozóra történő adatrögzítés* és *az adatok hitelességének biztosítása*.

Az első kihívás a megbízható eszközök használata. Alapjában véve minden szükséges eszköz megtalálható egy Windows 2000 rendszerben is, de mivel nem bízhatunk egy kompromittáltnak tekintett rendszerben, ezért össze kell állítanunk a magunk eszközkészletét. Ehhez szükségünk van egy (lehetőleg hálózatra nem kötött) operációs rendszer másolatra, melyet CD-re írunk. A CD tartalmazza a következő eszközöket is:

- at.exe – időzítetten futó alkalmazások
- cmd.exe – parancsablak
- dir.exe – könyvtártartalom listázása (win2k alatt nem külön fájl!)

[10] Klasszikus példa: A azt hiszi, hogy B-vel, B azt hiszi, hogy A-val kommunikál, és C azt hiszi, hogy csak ő egyedül ékelődött be és hallgat le mindent...

- ipconfig.exe – Internet-beállítások lekérdezése, módosítása
- nbtstat.exe – TCP/IP feletti NetBIOS kapcsolatok statisztikái
- net.exe – a lokális hálózat lekérdezései (net help parancs ad részletezést a lehetőségekről)
- netstat.exe – hálózati kapcsolatok részletei és statisztikája
- nslookup.exe – Name Server lekérdezése
- route.exe – a routing tábla lekérdezése, módosítása
- tracert.exe – az adott géptől a paraméterként megadott gépig vezető út lekérdezése

Az Interneten több hasznos, szabadon használható biztonsági eszköz is elérhető. Ilyen az integritás-ellenőrzéshez használható md5sum.exe. A program generálja vagy ellenőrzi egy fájl MD5 lenyomatát, így ellenőrizhetjük, hogy az általunk ismert (pl. honlapon közzétett) lenyomattal rendelkezik-e az adott fájl.

Forgalomfigyelés során a legjobb módszer a vizsgált forgalom lemezre mentése és MD5 lenyomat készítése. Fontos, hogy a lenyomat nem garantálja az adatok manipulálatlanságát a lenyomat készítése alatt, így ezen a ponton az eljárás kikezdhető. Ezért szükséges a feltört gépről egy másik gépre történő mentés, és a lenyomat ez utóbbin történő képzése, nem a feltört gépen végezve mindezt.

Windows rendszerekhez a legjobb eszközök a **Sysinternals.com** címén érhetők el[11]. Ezek közül néhány eszköz (az oldalon többféle operációs rendszerre sokkal több eszköz található), amely az egyes művelet monitorozására (innen a 'mon' végződések) alkalmazható:

- Diskmon – lemezműveletek
- Filemon – fájlrendszer műveleteinek
- PMon – futó eljárások és szálak
- Process Explorer – fájlok, registry kulcsok (a Regmon valós időben mutatja a bejegyzések változásait), objektumok, DLL-ek betöltése stb. Az egyes eljárások tulajdonosait is megmutatja.
- PsTools – Parancssori eszközök helyi vagy távoli gépen futó eljárások listázására, távoli eljárások futtatására, gép-újraindításra, naplófájlok lementésére stb.

A példabeli esetben azt kell meghatároznunk, hogy ki lépett be a rendszerbe, milyen erőforrások kerültek megosztásra és milyen eljárások futnak. Minden parancs CD-ről (pl. E:\) fut, és az eredmények lemezre (A:\) kerülnek:

```
E:\nbtstat -a winbox.private.com > a:\nbtstat-a_output.txt
E:\md5sum a:\nbtstat-a_output.txt > a:\nbtstat-a_output.md5
```

Ez az eljárás minden elkövetkező parancs esetén követendő, így később lehet elemezni az elmentett válaszokat. Az egyes eljárásokról hasznos könyvek[12] is elérhetők különböző eszközök felsorolásával, de ez az anyag inkább a technikákról szól, mint az elérhető eszközökről.

7.8.1.3. Mit keressünk?

A legelső feladat annak kiderítése, hogy mi történt, ki volt a vétkes és milyen hatása van az eseménynek. Sokan a megérzésre, tapasztalatra vagy éppen szerencsére hivatkoznak, de vannak mindenki által elfogadott célok és eszközök, melyeket alkalmazni lehet.

Cél	Eszköz / Módszer
Szokatlan eljárások azonosítása	pslist, psinfo, psfile

[11] Az eredeti cikkben a Foundstone.com szerepelt, de azóta felvásárolták a céget, termékei pénzesek.

[12] *Pl. Keith Jones: Anti-Hacker Toolkit, Kevin Mandia: Incident Response, cikkek a SecurityFocus-on.*

Szokatlan nyitott portok azonosítása	netstat, Fport, psservice
Szokatlan nyitott fájlok azonosítása	psfile, listdlls, Fport
Belépett felhasználók azonosítása	psloggedon, nbtstat
Eljárások tulajdonosainak azonosítása	psloggedon
Routing táblák vizsgálata	netstat, route
Időszakos fájlok vizsgálata	dir, type
Gyanús alkönyvtárak/mappák azonosítása	dir, Explorer

Rekonstruálni kell az eseményt. Ismerve a normális körülményeket, rövid idő alatt le tudjuk folytatni a helyszíni vizsgálatot és egyben az adatgyűjtést. A későbbi vizsgálatok már hosszabbak lehetnek. Ráadásul folyamatosan a nyomok után kell kutatnunk, hogy azonosítsuk a támadók tevékenységét. Az ideiglenes fájlokat olyan szektor-szerkesztővel (sector editor) kell vizsgálni, melyekkel a részben felfedezett nyomokat is észlelhetjük. A kötegelt fájlok (*.bat) esetén azt kell vizsgálni, hogy módosultak-e, vagy tartalmuk olyan-e, mely a támadó további lépéseit segíti. Végül, a Windows rendszerekben az eseménynapló (Event Log) és a biztonsági napló (Security Log) is vizsgálható és vizsgálandó is.

A **winbox.private.com** esetében a következő rendszereszközökre van szükség a vizsgálat során: netstat, route, nbstat, hostname, net, dir. Ezeken felül szükség van az Fport, pslist, psloggedon és a pssservice segédeszközökre, hogy azonosítsuk a felfedezett gyanús eljárások tulajdonosait. A következő rész egy kivonat az eljárás során kiadott parancsokból és gyűjtött adatokból:

E:\hostname
```
Winbox.private.com
```

E:\nbtstat –a winbox.private.com
```
NetBIOS Remote Machine Name Table
Name          Type   Status
---------------------------------------------
WINBOX              <00>   UNIQUE Registered
WINBOX              <02>   UNIQUE Registered
PROD          <00>   GROUP  Registered
PROD          <1E>   GROUP  Registered
..__MSBROUWS__      <01>   GROUP  Registered
ADMINISTRATOR       <03>   UNIQUE Registered
MAC ADDRESS = XX-XX-XX-XX-XX-XX
```

E:\net session
```
Computer  User name  Client Type     Opens   Idle time
-------------------------------------------------------------
\\TGT1  ADMINISTRATOR           0       00:00:27
\\TGT2  ADMINISTRATOR           0       00:00:15
\\TGT3  ADMINISTRATOR           0       00:00:23
\\TGT4  ADMINISTRATOR           0       00:00:05
```

E:\Fport.exe
```
Fport v2.0 - TCP/IP Process to Port Mapper
Copyright 2000 by Foundstone, Inc
http://www.foundstone.com

Pid  Process    Port  Proto      Path
420  svchost ->  135   TCP       C:\WINNT\system32\svchost.exe
8    System  ->  445   TCP
888  MSTask  -> 1025   TCP       C:\WINNT\system32\MSTask.exe
8    System  -> 1027   TCP
8    System  ->  445   UDP
430  svchost ->   80   TCP       C:\Program Files\Apache\httpd.exe
1625 servu   -> 3215   TCP       C:\Client_Data\Inetpub\_vti-bin\ \servu.exe
```

Elsőként a célgép nevét ellenőriztük le, majd azt néztük meg, hogy ki jelentkezett be a rendszerbe, milyen eljárások futnak és milyen szolgáltatások aktívak az eszközön. A kimenetek között a nyomozás szempontjából több figyelembe veendő adat is van. Az első a NetBIOS megosztás a célgép és több más gép között a szerverek belső címének használatával. Tudjuk, hogy ez nem természetes, mivel a belső hálózati címek csak menedzsment célokat szolgálnának, és az erőforrás-megosztás a Fájl és Nyomtató-megosztással tételesen tilos a szabályzat szerint.

Minden esetben láthatjuk a tiltott kapcsolódásokat a helyi rendszergazda (Administrator) jogköréhez, és ez aggasztó esemény. Ennél is aggasztóbb az Fport kimenete szerint az, hogy a célgép olyan FTP szervert futtat egy ideiglenes porton (1024 feletti), melynek szokatlan helyen van a forrása (C:\Client_Data\Inetpub_vti-bin\ \), amely egy rejtett alkönyvtárnak tűnik. Ez különösen gyanús, mivel a célgép Apache Web-szervert futtat, és az FTP-szerver könyvtárszerkezete egy olyan alkönyvtárban került elrejtésre, melynek struktúrája a Microsoft IIS szerver könyvtárstruktúrájára hasonlít.

Rekurzív módon listázva a rejtett alkönyvtárt megtaláltuk az FTP-szerver futtatható változatát, saját konfigurációs fájlját, több érdekes kötegelt fájlt, más konfigurációs fájlokat és eszközöket.

```
E:\dir /s /a C:\Client_Data\Inetpub\_vti-bin\" "\ /p
```

Különösen érdekes volt a VFS_MNT.BAT nevű fájl (részlet a fájlból):

```
net use F \\tgt1\c$\WINNT\system32\ \_vti-bin\ /user:Administrator AdminPass
net use G \\tgt2\c$\WINNT\system32\ \_vti-bin\ /user:Administrator AdminPass
net use H \\tgt3\c$\WINNT\system32\ \_vti-bin\ /user:Administrator AdminPass
net use I \\tgt4\c$\WINNT\system32\ \_vti-bin\ /user:Administrator AdminPass
```

Először a támadó mount-olt egy rejtett alkönyvtárt helyi rendszergazdai jogokkal minden célgépen, megengedve a virtuális fájlrendszer létrehozását a tiltott FTP-szervernek. Ilyen módon megosztani a rendszergazdai jelszót természetesen nagyon igénytelen megoldás, de ebben az esetben ez történt. Nagyobb gond, hogy a támadó ezek után egy vagy több gépen keresztül is továbbíthat adatokat a gépekre telepített rejtett FTP-szervereken keresztül.

7.8.1.4. Kapcsoljuk össze az eredményeket

Minden incidensnek megvannak a tanulságai és jellegzetességei: 1) a nagy cégek rendkívül nagy sávszélességű Internet-kapcsolattal rendelkeznek; 2) nagyméretű Windows NT és 2000 hálózatok működnek a cégeknél; 3) folyamatosan sérül a rendszergazdai és domain gazdai azonosítók bizalmassága; 4) olyan elosztott kétlépcsős FTP-szerverek széleskörű használata történik, ahol a szerver gyökérkönyvtára több gép megosztott meghajtói fölött létrehozott virtuális fájlrendszert alkot. Az esetünk is ezekkel a jegyekkel rendelkezik, és mint a legtöbb alkalommal, a támadók illegális tartalmat terjesztettek.

Az eljárásnak ebben az állapotában felfedésre került a kompromittálódás, a támadás lényege és megértettük, hogy mi történik a cég hálózatában, de még ki kell deríteni a támadók módszerét, amivel ezt elérték. Először néhány alapintézkedést kell elvégezni: a rendszergazda jelszavát meg kell változtatni, csomagszűrőkkel korlátozni kell a NetBIOS használatát a cégen belül működő switch-ekben. Ki kell derítenünk a támadási módszer lényegét vagy hatáskörét. Ennek érdekében a következő fázisban a hálózati forgalmat elemezzük.

7.8.2. A cél

Az illegális – pl. warez, pornó – tartalmak nagy forgalmat generálnak, így ennek költsége is nagy, ezért fontos a támadóknak a megfelelő erőforrásokhoz ingyen hozzájutni (gépek meg-

felelő számban, hálózati kapacitás megfelelő sávszélességgel). A megfelelő beállításokhoz a helyi és a cégszintű rendszergazda jogokat is megszerezték a támadók. Célunk kideríteni a következőket: 1.) mennyire terjedt ki a támadók hatásköre a helyi hálózaton belül? 2.) mi volt az eredendő kompromittáló eljárás? és 3.) ki(k) volt(ak) a támadó(k), ha ez kideríthető?

7.8.2.1. Eszközfejlesztés

Ideális esetben teljes mértékben felügyelt környezet és megfelelő személyzet áll rendelkezésre, akik figyelik az eseményeket, elemzik a naplókat, és megteszik a szükséges intézkedéseket. A legtöbb eset nem ideális: nem áll rendelkezésre hoszt vagy hálózati behatolásérzékelő, lépcsőzetes tűzfalrendszer, megfelelően erős azonosító és feljogosító rendszer, és többnyire a naplózás is hiányos. Nagyobb intézmények esetében a helyzet még rosszabb, mert a rendszer és naplóadatainak mennyisége nem teszi lehetővé az események valós idejű elemzését. Így az elkerülhetetlenül bekövetkező eseménykor szükség van egy gyors eszközre és eljárásra a hálózaton folyó események felméréséhez és a helyrehozatali intézkedésekhez.

Az ideális megoldás egy laptop lehet Windows (+ Windows 2000 Resource Kit) és Linux operációs rendszerekkel és olyan eszközökkel, melyek mindkét rendszerhez elérhetők. Ezek a következők:

Eszköz	Windows	Unix
Hálózati lehallgatók	Windump, Ethereal	Tcpdump, Ethereal, dsniff
Betörésészlelők	Snort	Snort
Elemzők	–	EtherApe, tcpreplay
Port pásztázók	Fscan, nmapwin	Nmap

16. Táblázat: Vizsgálati eszközök

Ezen kívül létezik még sok más eszköz is, de a fent említettekkel már el lehet kezdeni a munkát.

7.8.2.2. Első lépések, a munka kezdete

Mielőtt elindítunk egy lehallgató programot, meg kell határoznunk, hogy milyen adatokat keresünk a hálózati forgalomban. Egy 100 Mbps switch-elt hálózatban természetes, hogy elárasztható egy lehallgató, egy elemző hoszt vagy egy elemző. Elsőként az áldozattá vált gépet érdemes figyelni, olyat, amelyről tudható, hogy a támadó még használja.

Egy switch-elt hálózatban (a hálózati forgalom elemzése szempontjából a legrosszabb) legjobb a swicth és az áldozat közé a mirror portra csatlakozva lehallgatni a forgalmat. Nem kell a hálózati kapcsolatot megszakítva beékelődni, mert ezt a szakadást a támadó is észlelheti. Másik megoldás lehet az Ethernet aljzaton keresztüli csatlakozás. Az előbbi előnye, hogy megfelelő switch esetén gyorsan átállítható, hogy engedje megfigyelni a másik önálló hosztot vagy VLAN-t. Annak ellenére, hogy a tükrözés széles körben alkalmazott, biztosra vehető, hogy a VLAN csomagvesztést és ütközést fog okozni egyes switch-eken. Nem a legkívánatosabb megoldás, de használható, amit a következőkben mutatunk be.

7.8.2.3. Beépített Windows eszközök

A tcpdump (http://www.tcpdump.org) néven ismert Unix eszköz manapság elérhető Windows alatt is Windump (http://windump.polito.it) néven. A Windump a WinPcap (http://winpcap.polito.it) segítségével kommunikál a különböző hálózati eszközökkel. A Windump olvasni és írni is tudja a tcpdump bináris kimeneti formátumát, így az adatgyűjtés és elemzés történhet különböző rendszereken is.

A Windump telepítése után a switch mirror portjára csatlakozva elkezdődhet a forgalom lehallgatása. A forgalom megjelenítési formája lehet szöveges (ASCII), vagy hexa formátumú. A megfelelő módon alkalmazott szűrők segítségével a nagy mennyiségű forgalomból a számunkra érdekes részt tudjuk kiválasztani. Például a következő parancs az **áldozat** és **cél** közötti forgalmat szűrve menti a kimenetként adott fájlba:

```
C:\> Windump -i1 host aldozat and cel -w kimenet.txt
```

A valóságban az elején nem tudjuk, hogy mit keresünk, ezért *minden forgalmat* mentsünk el. Az Ethernet maximális átviteli egysége (MTU) 1500 byte, a snaplength változót (-s kapcsoló) is erre állítjuk. Végül a -n kapcsolóval megelőzzük, hogy a Windump konvertálja a hoszt címeket és port számokat, mert ez megkímél attól, hogy a DNS-nek küldjünk címfeloldási kéréseket, miközben a DNS is lehet, hogy már a támadók kezében van.

```
C:\> Windump -i1 -s1500 -n -w output
```

A Windump a kimenetet is el tudja olvasni, és különféle szűrőkön át értelmezhetjük a kapott eredményeket, bár sokkal hatásosabb egy grafikus elemzéssel is rendelkező eszközben vizsgálódni. E célra az egyik legjobb alkalmazás az Ethereal (http://www.ethereal.com).

Az Ethereal olyan grafikus felülettel rendelkezik, melyen kényelmes keresztugrások is végrehajthatók az elkapott csomagokon, azok fejlécén keresztül egészen a kommunikációban átvitt adatokig. A program alkalmas protokollelemzésre és TCP folyamok feltérképezésére is.

Az Ethereal egy sokoldalú szűrőnyelvet is tartalmaz, így sokkal könnyebb a begyűjtött adatokból azokat kiszűrni, amelyekre szükségünk van, de a megoldás arra is jó, hogy különböző szempontok szerinti szűréssel vizsgáljuk ugyanazt az adatmennyiséget.

7.8.2.4. Mit keressünk?

Annak ellenére, hogy minden támadásnak van valami sajátossága, vannak minden támadás esetén alkalmazható, vagyis ellenőrizendő technikák és jelenségek. Ezek a következők:

- Legnagyobb forgalmúak (kimenet)
- Legnagyobb forgalmúak (bemenet)
- Leginkább használt portok és protokollok
- Összehasonlítás ismert és valós forgalommal
- Koordináció / korreláció az igazságügyi vizsgálattal

Az első négy elem természetesnek tűnik, míg az utolsót gyakran el szokták felejteni. Emlékezzünk a példabeli esetre, amikor egy olyan fájlt találtunk az egyik feltört gépen, mely végrehajtásakor kapcsolatot akart létesíteni több más géppel is, mint rendszergazda jogú felhasználó. Teljes tartalomfigyeléssel és az Ethereal segítségével hozzáfoghatunk további anomáliák kereséséhez.

```
C:> Windump -i2 host TGT1 -s1500 -w output
```

Egy ideig gyűjtjük a forgalmi adatokat (jellemzően egy órai blokkot érdemes gyűjteni pár perces átfedésekkel), majd elemezzük őket. Az Etherealba töltött adatokra szűrőket alkalmazunk, hogy az SMB csomagokat (ezt alkalmazza a Windows Fájl Sharing) külön vizsgálhassuk, és megtaláljuk a TGT1-től és a TGT1-hez menő kapcsolatokat (mind a menedzsment interfészen, ami a windump -D szerint a 2. számú a mi esetünkben). Amint feltételeztük a TGT1 felé egy sor kapcsolódást találunk:

Nr.	Idő	Forrás	Cél	Protokoll	Infó
27	4.7277	winbox.victim.com	TGT1.victim.com	SMB	TREE CONNECT ANDX REQUEST, NTLMSSP AUTH
35	5.2552	winbox.victim.com	TGT1.victim.com	SMB	TREE CONNECT ANDX REQUEST, NTLMSSP AUTH
42	6.3521	winbox.victim.com	TGT1.victim.com	SMB	TREE CONNECT ANDX REQUEST, NTLMSSP AUTH

17. Táblázat: Vizsgálat során észlelt gyanús kapcsolódások

Ezek Windows fájlmegosztáshoz való hozzáférési kéréseknek tűnnek. Valójában gyors kapcsolódásokat látunk a winbox.victim.com felől a TGT1.victim.com felé, ami szintén gyanús, mert automatikus kapcsolódási kísérleteket jelent. Tovább szűrve az adatokat és az Ethereal natív protokoll dekódolóját használva a következő szekvenciákat tudjuk rekonstruálni:

```
FILTER:  (ip.addr eq 192.168.254.2 and ip.addr eq 192.168.254.205) and (tcp.port eq 1095
and tcp.port eq 445)

Resulting Stream (Excerpt1)

00000389  00 00 01 48 ff 53 4d 42  73 00 00 00 00 18 07 c8 ...H.SMB s.......
00000399  00 00 00 00 00 00 00 00  00 00 00 00 00 00 ff fe ........ ........
000003A9  01 08 50 00 0c ff 00 48  01 04 11 0a 00 01 00 00 ..P....H ........
000003B9  00 00 00 a6 00 00 00 00  00 d4 00 00 a0 0d 01 4e ........ .......N
000003C9  54 4c 4d 53 53 50 00 03  00 00 00 18 00 18 00 66 TLMSSP.. .......f
000003D9  00 00 00 18 00 18 00 7e  00 00 00 0c 00 0c 00 40 .......~ .......@
000003E9  00 00 00 0e 00 0e 00 4c  00 00 00 0c 00 0c 00 5a .......L .......Z
000003F9  00 00 00 10 00 10 00 96  00 00 00 15 82 88 e0 4c ........ .......W
00000409  00 41 00 50 00 54 00 4f  00 50 00 73 00 62 00 61 .I.N.B.O .X.A.d.m
00000419  00 72 00 69 00 73 00 68  00 4c 00 41 00 50 00 54 .i.n.i.s .t.r.a.t
00000429  00 4f 00 50 00 a0 22 69  06 b4 2d 12 7f 00 00 00 .o.r.."i ..-....
00000439  00 00 00 00 00 00 00 00  00 00 00 00 00 8b e9 d5 ........ ........
00000449  b8 26 a1 f2 01 06 6b 6c  e3 62 0d 7f fa 63 15 7f .&....kl .b..c.
00000459  7d d6 64 30 5e d0 ca 7d  5f 30 5f 13 a4 a7 c3 15 }.d0^..} _0_.....
00000469  d1 fb 87 33 8b 00 57 00  69 00 6e 00 64 00 6f 00 ...3..W. i.n.d.o.
00000479  77 00 73 00 20 00 32 00  30 00 30 00 32 00 20 00 w.s. .2. 0.0.2. .
00000489  32 00 36 00 30 00 30 00  20 00 53 00 65 00 72 00 2.6.0.0.  .S.e.r.
00000499  76 00 69 00 63 00 65 00  20 00 50 00 61 00 63 00 v.i.c.e.  .P.a.c.
000004A9  6b 00 20 00 31 00 00 00  57 00 69 00 6e 00 64 00 k. .1... W.i.n.d.
000004B9  6f 00 77 00 73 00 20 00  32 00 30 00 30 00 32 00 o.w.s. .  2.0.0.2.
000004C9  20 00 35 00 2e 00 31 00  00 00 00 00              .5...1. ....
```

Amikor az Ethereal dekódolja az SMB adatfolyamot, '.' jeleket illeszt be az egyes karakterek közé, mivel a legtöbb alacsony szintű naplózás és kommunikáció Unicode karaktereket használ a Windowsban. A Unicode-ban az 'US' karakterek felső byte-ján '00' karaktereket tartalmaznak. Az Ethereal minden nem ASCII karaktert '.' jelként dekódol. A „W.I.N.B.O.X.A.d.m.i.n.i.s.t.r.a.to.r.” sorozat a Winbox gép rendszergazdájának csatlakozási szándékát jelzi a TGT1-hez.

A rendszergazdai azonosítóval történő csatlakozási kísérleteken kívül egyéb azonosítókkal történő próbálkozásokat is lehet észlelni az adatokból, ilyenek az INET_GLOBAL, a Webszerver beállításaihoz használt megosztott azonosító, a HELPTECH1, a helpdesk egyik munkatársának azonosítója, és a USERCHUCK, mely a cég egyik magas szintű programozójának az azonosítója. Egyszerű lenne az utóbbi két azonosítóra gyanakodni, de ezen a ponton még nincs bizonyítékunk ellenük.

Annak ellenére, hogy sok esetben ez az elemző munka a segítség a támadók által végzett kommunikáció felderítésére, léteznek olyan eszközök, melyek még kényelmesebbé teszik az elemzést végző életét. Ilyen a dsniff (http://naughty.monkey.org/~dugsong/dsniff) is,

mely Windows rendszerre is elérhető (http://www.datanerds.net/~mike), és azonosító-jelszó dekódolásra képes a legtöbb internetes protokoll esetén (FTP, Telnet, HTTP, POP, NNTP, IMAP, SNMP, LDAP, Rlogin, NFS, SOCKS, X11, IRC, AIM, CVS, ICQ, Napster, Citrix ICA, Symantec pcAnywhere, NAI Sniffer, Microsoft SMB, és Oracle SQL).

7.8.2.5. Behatolás után kutatva

Látható, hogy a Windump és Ethereal sokat segít, de nem észleli a behatolást, csak a hálózati adatforgalom elemzésében használhatók egy adott gép esetén. Ha az egész hálózati forgalmat akarjuk elemezni, akkor az adatok mennyisége miatt ez nem egyszerű feladat. Ebből kifolyólag szükségünk van sokkal összetettebb eszközökre.

Snort (http://www.snort.org/)

Egy csomagszűrő alkalmazás és behatolásérzékelő rendszer. Elérhető Win32 platformra is, beleértve a naplózási képességet MySQL vagy MSSQL adatbázisba. Grafikus kezelőfelülettel (pl. ACID – http://www.andrew.cmu.edu/user/rdanyliw/snort/snortacid.html) támogatva azonnali statisztikák készíthetők (legnagyobb forgalom, protokollok stb. szerint). Az igazsághoz hozzátartozik, hogy a grafikus elemzés által használt adatbázis a forgalmi adatok alapján a háttérben készül, ami időt rabló feladat. Az adatbázis lekérdezések már gyorsak lehetnek. Ezen felül a snort szignatúra adatbázisát (előre megadott szabályok) saját szignatúrával bővítve meghatározhatjuk, hogy mit naplózzon, mit szűrjön, milyen eseményekre figyeljen. Például a következő szignatúrák minden NetBIOS kapcsolatot észlelni fognak a távoli gép F, G, H és I meghajtóin (ld. az esetünkben említett batch fájlban látottakat).

```
alert tcp any -> $HOME_NET 139 (msg:"NETBIOS SMB F$access"; flow:to_server,established; content:"\\F$|00
41 3a 00|";)
alert tcp any -> $HOME_NET 139 (msg:"NETBIOS SMB G$access"; flow:to_server,established; content:"\\G$|00
41 3a 00|";)
alert tcp any -> $HOME_NET 139 (msg:"NETBIOS SMB H$access"; flow:to_server,established; content:"\\H$|00
41 3a 00|";)
alert tcp any -> $HOME_NET 139 (msg:"NETBIOS SMB I$access"; flow:to_server,established; content:"\\I$|00
41 3a 00|";)
```

Ezek a szabályok riasztást váltanak ki, ha bármelyik hoszt megkísérli bemountolni az adott meghajtót a hálózaton keresztül a HOME_NET változót alkalmazva. A HOME_NET változót megváltoztatva, ahogy a forgalmat a mirror portra tereltük, folyamatosan kiterjeszthetjük keresésünk határait, mígnem az egész menedzsment VLAN-ban figyeljük a NetBIOS kísérleteket.

A szabályok megváltoztatásával vagy új szabályok megadásával több hálózati erőforrást, meghajtót stb. figyelve felépíthetjük a támadás mértékének térképét. Amióta tudjuk, hogy a helyi szabályok szerint NetBIOS (vagy bármilyen más kódolatlan) forgalom nem lehet a VLAN-on, minden riasztás támadó tevékenységnek feltételezhető.

Tanácsos kétféle monitorozó eszközt használni, a Snort valósidejű megfigyelést végez a számára megadott szignatúrák alapján, de szükség van egy teljes tartalom-elemzésre is, akár off-line módon is. Fontos az eljárásban, hogy minél tovább tart, annál többet fedezünk fel a támadók cselekményeiből. Az eredeti adathalmazt megtartva újraelemezhetjük az időközben hasznosnak tűnő új szabályokkal:

```
C:\SNORT> snort -r Windumpfile -c C:\SNORT\snort.conf
```

Mivel a NetBIOS forgalom nem fordulhatott volna elő a hálózaton, ezért ezek mindenképpen gyanúsak, sőt osztályozhatjuk súlyosabban is őket. Az Ethereal segítségével a Win-

dumpfile-ban a Snort által észlelt IP címeket vehetjük kézi vizsgálat alá, hogy felderítsük a feltört azonosítókat.

7.8.3. Megoldás

Elemzőként meg kell tudni különböztetni a normális forgalmat a gyanús és/vagy rosszindulatú forgalomtól. Ha ismerjük hálózatunkat, akkor nálunk a hazai pálya előnye, de sajnos ez nem mindig van így.

Sokáig nehéz volt a Windows rendszerek vizsgálata, de a Sysinternals-nak köszönhetően ma már számos eszköz rendelkezésünkre áll (pl. az előzőekben is említett PS* eszközök). Említésre méltó a Foundstone, az @Stake, a nyílt forráskódú szoftverek világából a WinPcap, Windump, Ethereal, Snort stb.

Léteznek adatforrások, melyek nem kerültek vizsgálat alá ebben az anyagban (pl. Windows esemény és biztonsági naplók, Web-szerver naplók stb.), pedig egy valós incidens során ezeket is teljes mértékben vizsgálni kell. A legtöbb esetben elég adat áll így is rendelkezésre az incidens kivizsgálásához, sokszor kardinális szabályok alapján egyértelműsíthetők a történtek.

Az elején említett batch fájl nem lehetett volna olyan sikeres, ha nem lett volna a Rendszergazda azonosító minden gépen ugyanaz (Global Domain Administrator = Local Administrator tovább csökkentve a teljes céges hálózat biztonságát). A második nagy probléma a rossz figyelő környezet, melyek strukturális változásokat és koncepcionális átalakításokat követeltek. Az említett eszközök segítségével és minimális költséggel lehetséges egy alapvető IDS képesség kiépítése, bár a felügyelet jelentős emberi erőforrást igényel majd.

A Windows biztonságáról nem nyitva vitát megjegyezzük, hogy minden széles körben elterjedt rendszer terjedésével arányosan több támadásnak van – és lesz – kitéve. Ennek köszönhetően a sebezhetőségei is gyakran és hatásosan kerülnek kihasználásra, így nagyméretű Windows rendszer esetén számítani kell a biztonsági eseményekre. Ezért nem árt felkészülni a következőképpen.

- Készítsünk egy eszközkészletet:
    - PS Utilities a Systinternals lapjáról.
    - A rendszerben lévő Windows verziókról egy megbízható verzió (alap-összeállítások image-ei is hasznosak lehetnek, mert egy újratelepítése sem kevés idő...).
    - Dual-boot-os Windows / Linux laptop Winpcap, Windump, Ethereal, és Snort+MySQL valamint Acid és ezek Linux változatai a Linuxos partíción.
- Windump/tcpdump és tcpreplay használata + EtherApe a hálózat felmérésére.
- IDS implementálása – legalább egy kezdetleges változatban.

Végül, de nem utolsó sorban szánjunk időt saját felkészülésünkre. Ma már elérhetők olyan könyvek vagy internetes források, melyekből ez önképzéssel is elvégezhető. A legfontosabb, hogy az eszközökkel leássunk a dolgok mélyére, ismerjük meg a tulajdonságaikat és gyakoroljuk használatukat.

Minden eljárás első lépése a felkészülés, mely legjobb esetben még azelőtt történik, hogy gond lenne. Ide kell érteni a rendszergazdák által megtehető megelőző lépéseket is.

7.8.3.1. Törvények

A számítógépes törvényszéki eljárások két fő témakörre helyezik a hangsúlyt: 1. bírósági bizonyító eljáráshoz szükséges és elfogadott módszerek, 2. az eljáró azon igénye, hogy elkerülje az ellene, vagy az általa képviselt szervezet ellen irányuló jogi lépések lehetőségét. Ezek az elemek egyúttal a gyanúsított személyiségi jogait is védik. Egy adott eljárásban mindig mérlegelni és konzultálni kell arról, hogy a törvényszéki eljárás és az adatvédelmi szabályok mely pontokon keresztezik egymást, és melyiknek van nagyobb prioritása illetve mire és milyen feltételekkel van törvényi szabályozás.

7.8.3.2. Szabályok és eljárások

Amennyiben léteznek adatvédelmi szabályok, ezek behatárolják az adatok vizsgálati mélységét és milyenségét is. Az ilyen szabályzat a munkaadó és a munkavállaló számára is hasznos, hogy a keretekben megegyezve, a szabályokat aláírva állapodjanak meg a részletekben. Ez azért is fontos az eljárás során, hogy lehessen tudni, ki mit tudott, hogy egy-egy cselekedet megengedett-e számára vagy sem.

Az incidens jelentésének szabályai azt biztosítják, hogy a jelentés a megfelelő helyre érkezik, és ha már észlelték, nem kallódik el a szervezeten belül.

7.8.3.3. Naplózás

Csak az vonható eljárás alá, ami elérhető, tehát a megfelelően felülírt, törölt vagy más okból elérhetetlen adatok nem sorolhatók ide. A nem naplózott eseményeket is nehéz kitalálni a végállapotból, de sokszor a naplózás beállításait nem ismerik és nem alkalmazzák a rendszergazdák.

A naplózásnál (mit és hogyan naplózzunk) fontos megtalálni az optimumot, hogy milyen előnyökkel jár a naplózás beállítása (milyen adatok lesznek meg), és milyen hátrányokat jelent (rendszer lassulása, lemezterület-foglaltság, kiegészítő erőforrások igénye stb.). A naplózás szabályai is rögzíthetők, így a naplók rögzítésének és őrzésének módja, ideje és hozzáférési jogosultságai is tisztázottak lehetnek.

Egyes rendszerek biztosítják, hogy előre beállított események riasztást vagy adott módú értesítést, netán reakciót váltsanak ki, így a naplózás átmehet behatolásérzékelésbe is. Az automatikus eszközök képességei még sokáig nem teszik feleslegessé az ember közreműködését, vagyis a naplókból kikövetkeztethető összefüggéseket – egy-egy tapasztalatra alapuló gyanút – emberi elemzéssel kell ellenőrizni.

7.8.3.4. Az eszköztár összeállítása

A választék elég széles, de csak idővel alakul ki a saját bevált (és jól megismert!) eszköztár, valamint az adott vizsgálandó rendszer esetén alkalmazandó eszközök tára. Jelen esetben a szabadon elérhető eszközökre helyezzük a hangsúlyt. Az előzőekben már felsoroltakon kívül a következőkre lesz még szükségünk.

Az eszközök közül a The Sleuth Kit (TSK, http://www.sleuthkit.org/sleuthkit/) és a hozzá tartozó Autopsy Forensic Browser (AFB, http://www.sleuthkit.org/autopsy/) emelendő ki, mely 2005 áprilisában is még frissített volt (több hasonló termék már nem érhető el régi címén vagy a készítők már nem frissítik). A TSK készlet biztosítja, hogy a 'dd'-vel lemásolt merevlemezen vizsgálatot végezzünk, míg az AFB egy HTML-alapú megjelenítést ad, hogy az eredményeket kezelhetőbb formában vizsgálhassuk.

Az eszközök használhatók a különböző Linux, MacOS és Windows fájlrendszereken történő vizsgálatokhoz is. Ezen kívül is léteznek szabadon elérhető eszközök, melyek hasznosak lehetnek. A legfontosabb szabály, hogy bármilyen eszközzel is dolgozunk, és bármit cselekszünk is, az eredeti bizonyítékokat ne változtassuk meg! Ne használjuk a feltételezhetően kompromittált rendszert önmaga vizsgálatára, hanem a kikapcsolt gépről mentsük le a merevlemezt (image készítés, teljes másolás), és csak ezt a másolatot használjuk a vizsgálódásra.

Igazságügyi eljárás esetében az eredeti adathordozóról készült másolat kerül tárolásra, és az eredeti eszköz kerül vizsgálatra!

7.8.3.5. Másolat (image) készítés

Az első lépés a teljes és hiteles másolat elkészítése, melyhez sokféle eszköz létezik[13]. A legtöbb nyomozó által használt eszköz a fizetős EnCase nevű alkalmazás (http://www.guidancesoftware.com/). A másik említésre érdemes az ugyancsak fizetős termék, a SafeBack (http://www.forensics-intl.com/safeback.html), de a cég honlapjáról elérhetők ingyenes termékek is.

Nem szükséges pénzt költeni a fizetős termékekre, ha technikai szempontból megfelelő megoldást keresünk, mert a data dumper (dd) a Linux rendszereken megtalálható. A man dd megmutatja a parancssoros eszközök használatát, de léteznek olyan paraméterei is, melyek kifejezetten az igazságügyi eljárás céljából történő munkát segítik.

A másolat megfelelőségét (ugyanaz, mint az eredeti?) és sértetlenségét (nem változott meg?) hash algoritmussal tudjuk biztosítani. Ilyen az MD5 és az SHA-1 algoritmus, de hamarosan áttérés várható ezeknél biztonságosabb változatokra (pl. SHA-256 és nagyobb számú névtársai). Az ilyen algoritmusok lényege, hogy akár 1 bit megváltozása esetén is más hash-t fognak eredményezni, így jelzik, hogy az adat megváltozott-e, avagy sem. Az eljárás alkalmazható egy-egy fájlra is. Az md5sum.exe szabadon elérhető, de legyünk körültekintőek, hogy megbízható forrásból szerezzük be.

7.8.4. Keresés a rendszerben

Az image elkészülte után indulhat a bizonyítékok utáni keresés, melyet többnyire az image készítő szoftverek is támogatnak. A következőkben nézzük meg egy Windows rendszer esetén hogyan érdemes a keresést végrehajtani.

Amikor egy felhasználó belép egy Windows XP vagy 2000 rendszerbe, először egy teljes könyvtárstruktúra jön létre az egyéni fájlok és beállítások tárolására (profile[14]). Ez a struktúra egy főkönyvtárból ágazik el, és ennek a neve megegyezik a felhasználó azonosítójával. Ebben a struktúrában található pl. az NTUSER.DAT fájl, melyben a felhasználóra vonatkozó beállítási információk találhatók. A fájl rejtett, így csak akkor látható, ha a rejtett fájlok megmutatását is beállítottuk a fájlböngészőnkben. Ez a fájl minden kilépéskor frissítésre kerül, így az utolsó írás időpontjából következtethetünk a felhasználó kilépési idejére.

A Cookies mappában a meglátogatott Internet oldalak tárolnak adatokat, ha a böngésző cookie tárolási szabályai ezt engedik (alapértelmezésben engedélyezett, de letiltható). Együtt az ideiglenes Internet fájlokkal (ld. később) jó képet nyerhetünk a felhasználó böngé-

[13] fontos, hogy a helyi hatóságok elfogadják-e az adott eszközzel történő másolatkészítést hiteles másolatnak!

[14] fontos, hogy 'roaming profile' esetén a cache ne a profile-ban legyen, mert egyrészt a profile mentése és betöltése hosszú időt vesz igénybe, másrészt a másrészt a cache-el növelt mérete miatt quota-s rendszerben meghaladva a quota-t komoly gondokat tud okozni a számítógép működésében, majd az elveszett profile pótlása kellemetlen.

szési tevékenységéről. A cookie-k kezelhető megjelenítését segíti több eszköz, közülük a CookieView-t említjük meg, mely szabadon letölthető a http://www.digital-detective.co.uk/ lapról a FreeTools alól. A böngészési eseményeket a NetAnalysis szoftverrel lehet jól elemezni, de ez már nem ingyenes terméke a cégnek.

Az operációs rendszer által különböző célokra létrehozott fájlok (*windows artefacts*) is hasznos támpontot jelentenek a nyomozásban. Ilyenek pl. az .lnk fájlok, melyek pl. a Munkaasztalon, a Küldés vagy a Start menüben jelennek meg. Ezek olyan hivatkozások, melyekkel a gyakran használt fájlok vagy alkalmazások könnyebben elérhetők.

Ezeknek a fájloknak a vizsgálata segíthet kideríteni egyes fájlok, mappák, alkalmazások vagy eszközök valamikori létezését, melyek már nem lelhetők fel a rendszerben. Ez akkor hasznos, amikor egyes fájlok már törlésre kerültek és nem állíthatók vissza, vagy hálózati meghajtókon voltak. Ilyen esetben, ha egy hivatkozás egy ZIP, JAZZ vagy USB meghajtóra mutat, hasznos útmutatást kapunk ezen médiák utáni keresésre majd megtalálásuk esetén tartalmuk elemzésére.

Hasznosak még a telepítés során vagy egy alkalmazás használata során keletkező ideiglenes fájlok. Ezek többnyire törlésre kerülnek, amikor a telepítés vagy az alkalmazás befejeződik vagy a számítógépet megfelelően kapcsolják ki (shutdown). Amennyiben egy alkalmazás lefagy, úgy ez a törlés nem megy végbe, így sok olyan bizonyíték maradhat, melyekről a felhasználó nem tudhat.

Azok, akik úgy nyomtatnak egy fájlt, hogy nem mentik el a gépre, szintén ellenőrizhetők, mert a nyomtatás során keletkezett .spl és .shd fájlok tartalmazzák a nyomtatott fájl nevét, tulajdonosát, a használt nyomtatót, és a nyomtatandó adatokat (vagy egy listát az ilyen adatokat tartalmazó fájlokról).

7.8.5. Mélyebbre ásás – törölt, rejtett, titkosított fájlok

A fájlok törlése után (a Kuka ürítése után is) a fájl adatai még elérhetők, csak a fájlt töröltként, elfoglalt területét pedig szabadként kezeli a rendszer, így a terület felülírásáig ez még elérhető marad. Egyes esetekben a felülírás után is elérhetők az adatok (ld. Kürt Rt. ez irányú munkássága).

A törölt fájlok behatárolására és teljes vagy legalább részleges visszaállítására elérhető eszközök közül a EasyRecovery Pro (http://www.ontrack.com/software) az egyik legjobb, de ez nem ingyenes. Interneten rákeresve a 'freeware file recovery' szavakra több találat közül is választhatunk, de egyet külön is megemlítünk: http://www.pcinspector.de.

A Kuka a felhasználó által törölt, de még visszaállítható fájlok tárháza, mely különösen nagy figyelmet kap a nyomozásokkor. Például a Windows98 rendszereken egy INFO vagy INFO2 fájl tartalmazta a Kukába került fájlok részleteit (többek között az eredeti helyüket, a törlés időpontját). A Kuka ürítésekor ez is törlésre került, de a már említett módon visszaállítható, ha még nem íródott felül. Megemlítendő, hogy a SHIFT billentyűvel támogatott törlés (SHIFT+Del) nem helyezi a Kukába a törölt adatot, hanem az valóban törlésre kerül (persze a fentiekben említett módon visszaállíthatók).

Amikor alkalmazás végzi a törlést (nem fájlkezelő), az alkalmazásbeli tárolástól és a visszaállítás támogatásától (ha van egyáltalán) függ a visszaállíthatóság. Ez történhet az alkalmazásból vagy olyan eszközökkel, melyeket az igazságügyi vizsgálatokat végzők írtak. Megtörténhet, hogy a nyomok eltüntetése érdekében egész partíciót törölnek, vagy formázzák a lemezt, de ebben az esetben sem tűnik el fizikailag az adat a lemezről.

Azok, akik el akarnak rejteni egy fájlt, annak jellemzőit módosítják (pl. átnevezik a kiterjesztését exe-ről txt-re), de a fájl jellemzőjét így is vizsgálhatjuk szignatúraelemzéssel. Ekkor a fájl fejléce és kiterjesztése között felfedhetők ellentmondások, melyek alapján tovább ellenőrizhetjük az adott fájlt. Ilyen elemző Linux alatt a file parancs, míg Windows alatt a „File for Windows" szabadon használható (ld. még a többi hasznos alkalmazást is a GNU-Win32 projekt lapján, http://gnuwin32.sourceforge.net/packages/file.htm).

Ha a visszaállított adat titkosított információ, akkor ez egy újfajta kihívást jelent a szakértőnek. Nincs egyértelmű, mindent feltörő alkalmazás, de lehetnek rosszul megválasztott jelszavak vagy tárolási módok, így az eljáró felfedezhet ilyen adatokat, melyekkel dekódolhatja a kódolt adatokat. Az is lehet, hogy az adott alkalmazás gyenge titkosítást használ, és a megfelelő megfejtő-programmal, elég erőforrással és idővel siker érhető el. Egyes esetekben a végigpróbálgatásos módszer is működhet, más esetben a háromszori rossz próbálkozás miatti védelem ezt nem engedi meg.

A jövőre nézve elmondható, hogy a folyamatos változásra kell felkészülni, hiszen az új és újabb lehetőségeket a támadók ki fogják használni, így sohasem késő megkezdeni a felkészülést az új és újabb védekezésekre és nyomozásokra az igazságügyi eljárás sikere, de egyben tévedhetetlensége érdekében.

7.8.6. Linux specifikus eszközök és eljárások

Bizonyos eszközök csak Linuxra érhetők el, ezért is ki kell térni a Linux alatti eljárásokra.

7.8.6.1. On-line forensics

Az élő rendszerek vizsgálata különös figyelmet követel, ugyanakkor ezzel a módszerrel tudjuk a legtöbb hasznos információt nyerni. A mentések során a rendszer különböző tárolóegységeit a rajtuk tárolt adatok elévülési, felülíródási idejét tekintve növekvő sorrendben célszerű sorra venni.

A leírásban felmerülő parancsok alapvetőek minden Unix és Linux környezetben, de néhány esetben felhasználjuk a „The Coroner's Toolkit", röviden TCT elnevezésű – kissé már elavult, de bizonyos szempontból még mindig egyedülálló – eszközkészlet egyes elemeit. (Elérhető itt: http://www.porcupine.org/forensics)

Beépülés

A futó rendszerbe való beépülés első lépése a rendszergazdai jogkör megszerzése. Szerencsés esetben a rendszer már rendelkezik futó rendszergazdai konzollal, azonban az is előfordulhat, hogy be kell jelentkezni. Ilyenkor felmerül az a probléma, hogy a rendszer állapota a belépési procedúra lezajlása miatt is valamelyest változik, különösen a gyorsan ürülő háttértárak (memória, swap) esetén, azonban fel kell készülni arra az esetre is, hogy a támadók a rendszergazdai belépés eseményére különböző takarító, nyomeltüntető műveleteket ütemeztek előre, ez esetben nincs mit tenni, a belépéssel mindenképp próbálkozzunk meg.

Az adminisztratív konzolhoz való hozzájutás után semmiképp ne adjunk ki további parancsokat, lehetőleg az automatikusan futó szkriptek futását is szakítsuk meg, hisz a rendszer akármelyik állománya kompromittálódhatott. Az egyetlen megoldás egy előkészített bináris programgyűjtemény használata, melyet egy külső, lehetőleg írásvédett adathordozó segítségével juttatunk a rendszerbe. A programgyűjtemény előkészítését a rendszer komponenseinek ismeretében egyedileg kell elvégezni, mindenképp ügyelve arra, hogy a gyűjteményben szereplő programok ne függjenek olyan dinamikusan betöltendő külső könyvtáraktól, melyeket a vizsgált rendszer szolgáltat, ezt megelőzendő célszerű lehet minden futtatható állo-

mányt statikusra fordítani, vagy pedig a dinamikus linker környezeti változóinak megfelelő beállításáról gondoskodni a futtatás során (lásd: ld.so dokumentáció, `man ld.so`)

Az előkészített programrendszer eléréséhez szükség van az azt tartalmazó adathordozó (kézenfekvő módon egy CD-ROM) csatlakoztatására. Ezt egyes rendszerekben az automounter bizonyos alapértelmezett könyvtárakban automatikusan végzi, azonban ez a rendszer állapotának jelentősebb változásával jár, ezért a `mount` parancs közvetlen használatát ajánljuk a következőképp:

```
mkdir /mnt/forensics
/bin/mount -t iso9660 -n /dev/scd0 /mnt/forensics
```

A fenti parancsok hatására létrejön az `/mnt/forensics` könyvtár, melybe a `mount` parancs segítségével becsatolásra kerül a megfelelő blokkeszközhöz (`/dev/scd0`) rendelt adathordozó. A `-n` paraméter tiltja a `mount` számára az `/etc/mtab` fájl frissítését.

A következőkben kiemelt parancsok használatakor külön nem kerül említésre, de minden esetben az előzőleg becsatolt eszközkészlet kizárólagos használatát ajánljuk. Ennek pontos megvalósítását az olvasóra bízzuk, így a megadott utasítások mindig az említett előkészített adathordozóról való futtatást feltételezik.

Távoli bizonyítékgyűjtés

Mivel a vizsgált rendszeren minimális változtatásokra törekszünk, szükségünk lesz egy hálózaton elérhető biztonságos adattárolási lehetőségre, illetve ennek hiányában egy szintén felcsatolandó írható adathordozóra (pl.: pendrive) is. Mivel a hálózati megoldás általában kézenfekvőbb, és nem igényli a vizsgált rendszer újabb módosításait, erre mutatunk egy lehetséges megoldást a `netcat` program segítségével.

Az üzeneteket fogadó számítógépen a következő egyszerű „fájlszervert“ futtathatjuk:

```
#!/bin/bash
while true; do
        tmpfile=`mktemp`
        nc -l -p 12345 > $tmpfile
        outfile=`date +forensics_%c`
        mv $tmpfile $outfile
        md5sum $outfile > $outfile.md5
done
```

Mint látható, ez a rövid shell script egy végtelen ciklusban az `12345`-ös TCP porton bejövő adatokra várakozik, melyek vétele után a fájlnévben eltárolja az időbélyeget, valamint a fájlhoz tartozó md5 ellenőrzőösszeget is elkészíti. Az md5 hash algoritmusban nem bízók természetesen más, eddig még fel nem tört kriptográfiai ellenőrzőösszeg számító algoritmust megvalósító programot használhatnak. Azért ajánljuk ezt a megoldást, mert a `netcat` segítségével könnyen alakíthatunk ki a helyi hálózati környezetnek megfelelő, nem standard, vagyis a támadók által előre nehezen kiszámítható kommunikációs csatornákat. A mini-fájlszerver természetesen további funkciókkal is tetszés szerint kiegészíthető.

Az ismertetett hálózati tároló állomás számára a vizsgált hosztról a következő parancs segítségével küldhetünk át adatokat:

```
nc -q0 1.2.3.4 12345
```

A fenti utasítás az `1.2.3.4` IP című számítógép `12345` TCP portjára küldi EOF jel vételéig az adatokat, majd zárja a kapcsolatot, és kilép. Látható, hogy az így kialakított távoli bizonyítékgyűjtő rendszerrel, a standard shell csővezeték szolgáltatást kihasználva bármely

parancs kimenetét egyszerűen naplózhatjuk. A továbbiakban a parancsok kimenetének a fenti netcat utasításba való átirányítását a „| . . .“ karaktersorozattal jelöljük.

Első lépésként a folyamatosan felülíródó átmeneti tárolók mentését végezzük el. Tevékenységünket a rendszeridő naplózásával kezdjük, és az audit trail teljességének érdekében ez legyen a gép áramtalanítása előtti utolsó művelet is:

```
date |...
```

További példáink előtt fontos megjegyezni, hogy mindenhol célszerű a numerikus értéktárolásra törekedni a szimbolikus megoldás helyett, részben azért, hogy az esetlegesen kompromittált leírófájlok, névadatbázisok ne befolyásolják az adatokat, részben pedig azért, hogy a névfeloldási műveletek az adatmentést ne lassítsák.

Cache területek mentése

A rendszergazda parancstörténetét az elterjedt `bash` parancsértelmezőt feltételezve a következő paranccsal menthetjük el:

```
history |...
```

A rendszer közelmúltban élő hálózati kapcsolatainak nyomát megtalálhatjuk a route tábla átmeneti tárolójában. A tároló állapotát a következő paranccsal menthetjük (az `n` kapcsoló tiltja a szimbolikus feloldást):

```
route -Cn |...
```

A helyi számítógép ARP cache táblája – a route tábla átmeneti tárolójához hasonlóan (azonban korlátozott hatótávval és más mechanizmusok alapján) – információkkal szolgálhat a vizsgált gép és a lokális hálózat közti, közelmúltban lezajlott kapcsolatfelvételekről (az `n` kapcsoló itt is a feloldást tiltja):

```
arp -an |...
```

Rendszerállapot mentése

Az átmeneti tárak begyűjtése után kezdjünk hozzá a rendszer állapotának vizsgálatához. A rendszer hálózati képének további rögzítéséhez érdemes a tűzfal állapotának mentése, célszerűen a forgalomstatisztikákkal együtt. Erre a célra a következő parancs szolgál (az `n` kapcsoló tiltja a névfeloldást):

```
iptables -L -nv -t filter |...
```

A csomagmódosító és címfordító táblák mentéséhez a fenti parancs ismétlése szükséges a `-t mangle` és `-t nat` opciók megadása mellett.

A rendszer összes route táblájának mentését szolgálja a következő parancs:

```
ip route list table all |...
```

A rendszerben futó processzek állapotát a `/proc` virtuális fájlrendszer követi, a lényeges adatok interpretációját a `ps` parancs segítségével végezhetjük. A rendelkezésre álló információk megértését a `ps` program man oldala segíti. A `/proc` fájlrendszerből még bővebb információkat is nyerhetünk, de egy meglehetősen átfogó képet kaphatunk például a következő `ps` paramétersorral is:

```
ps -e -o pid,spid,ppid,pgid,sid,tty,thcount,nice,rtprio,\
flags,label,%mem,sz,vsz,rss,eip,esp,stackp,\ euid,egid,fgid,fuid,ruid,rgid,suid,sgid,lstart,etime,cputime,\
blocked,psr,%cpu,cputime,c,caught,ignored,\
class,nwchan,sched,stat,fname,command,args |...
```

A futó processzek által nyitva tartott leírók listáját, mely magában foglalja a nyitott hálózati és processzek közti kapcsolatokat, a kapcsolatfelvételt váró és egyéb socketeket, valamint a megnyitott fájlokat is, a következő parancs segítségével tárolhatjuk el (az `-nPl` paraméterlista hatására numerikus és nem szimbolikus információk kerülnek tárolásra):

```
lsof -nPl |...
```

Előfordul, hogy a rendszert érő támadás során a futó rendszermag is kompromittálódik. A kernel állapotát a memória közvetlen manipulációjával is meg lehet oldani, azonban jóval elterjedtebb a betölthető kernel modulokon (LKM) keresztül történő behatolás, melynek nyomaira lelhetünk a betöltött modulok listájában:

```
cat /proc/modules |...
```

A modul-lista a kernelben egy láncolt memóriaszerkezetben helyezkedik el, melyből speciális megoldással bizonyos modulok kifűzhetik magukat. Ennek felderítésére léteznek speciális „nyomozó" kernelmodulok, de ezek működése nem teljesen megbízható, és az újabb kernelekhez lassan érkeznek a frissítések. Ezért, ha részletesebb vizsgálat válik szükségessé, célszerűbb lehet a lementett memóriakép és kernelparaméterek utólagos vizsgálata, például a `gdb` program segítségével.

A kernel által exportált szimbólumok mentését a 2.4-es sorozatú kernel esetén a következő paranccsal végezhetjük:

```
cat /proc/ksyms |...
```

A 2.6-os kernel esetében a leíró neve megváltozott, ezért ott a helyes parancs:

```
cat /proc/kallsyms |...
```

Ezeket a fájlokat a kernel binárisból utólagosan is kifejthető, de általában amúgy is tárolásra kerülő System.map statikus szimbólumfájlban található címekkel és szimbólumokkal összevetve nyomára akadhatunk a felülbírált kernelhívásoknak, vagy a szokatlan címeken elhelyezkedő, nem található modulhoz köthető szimbólumoknak.

A kernel állapotának utólagos gdb-vel történő elemzésére a teljes kernelmemória mentésére szükség lehet, melyet a következő paranccsal végezhetünk el:

```
dd if=/proc/kcore |...
```

Az eddig említett /proc fájlokon kívül az alábbiak szolgálhatnak még jellegzetesen tárolandó információkkal:

- `/proc/version` – Az operációs rendszer verziója.
- `/proc/sys/kernel/name` – A rendszer beállított hoszt neve.
- `/proc/sys/kernel/domainame` – A rendszer domain neve.
- `/proc/cpuinfo` – CPU információk.
- `/proc/swaps` – A használt swap partíciók leírása.
- `/proc/partitions` – Az összes helyi fájlrendszer leírása.
- `/proc/self/mounts` – Felcsatolt fájlrendszerek.
- `/proc/uptime` – A rendszer üzemideje.

A futó rendszer állapotának mentése után célszerű lehet még az aktuális fájlrendszer-állapot rögzítése is, melyet a következő, részletes információkat tároló find parancs-sorral tárolhatunk el:

```
find / -printf 'access:%Ac;create:%Cc;modify:%Tc;modes:%#m;blocks:%b;device:%d;fstype:%F;user:%U;goup:%
G;inode:%i;size:%s;use:%k;links:%n;type:%y;name:%p\n' |...
```

A tárolt létrehozási (`create`), módosítási (`modify`) és elérési (`access`) idők alapján a rendszer esetlegesen közelmúltban lecserélt állományainak listáját külön érdemes lehet megvizsgálni. Itt fontos megjegyezni, hogy a fenti információk közül egyedül a létrehozási idő mondható nehezen módosíthatónak, a többi időbélyeg utólagosan is könnyen manipulálható.

A POSIX attribútumokat támogató fájlrendszereken a `find` nem képes az attribútumok megjelenítésére, azonban egyes rootkitek a speciális jelzők módosításával próbálják meggátolni kritikus fájljaik felülírását. Az attribútumlistát a következő paranccsal tárolhatjuk:

```
find / -exec lsattr {} \; |...
```

A fejezet elején említett TCT egyik – bizonyos esetekben jól használható – eszköze, a `pcat`, mellyel egy gyanúsnak ítélt processz teljes memóriatérképét menthetjük. Ne feledkezzünk meg azonban arról, hogy ez a megoldás elég erőszakos, számítanunk kell tehát a rendszer összeomlására, ezért csak minden korábbi művelet végrehajtása után próbálkozzunk használatával.

A rendszer vizsgálata során esetlegesen felfedezett rendellenességek, nyitott kapcsolatok, gyanús szolgáltatást nyújtó, vagy sok erőforrást fogyasztó ismeretlen programok esetén kísértést jelenthet a rendellenes tevékenység azonnali korlátozása a processz leállításával, vagy a hálózati kapcsolat korlátozásával. Semmiképp se módosítsuk ilyen céllal a futó rendszer állapotán, mivel előfordulhat, hogy a támadók bizonyos csapda mechanizmusokat építenek be, melyek épp a hasonló események hatására takarító „önmegsemmisítő" műveleteket hajtanak végre. Hasonló okok miatt az élő rendszer mentése után nem szabad a rendszer szabványos leállását kezdeményezni, hanem a tápellátás megszakításával, összehangolt áramtalanítással, azonnali kikapcsolást kell előidézni.

Az on-line mentés egyes ötleteinek alapjául Mariusz Burdach „Forensic Analysis of a Live Linux System" című cikke szolgált. http://www.securityfocus.com/infocus/1769

7.8.6.2. Off-line forensincs

Az on-line állapotfeltárás és bizonyítékgyűjtés nem mindig oldható meg, és nem is lehet teljes az utólagos elemzési fázis nélkül. A ToReS rendszer bootolható CD kialakításának köszönhetően megfelelő alapot szolgáltat az off-line bizonyítékgyűjtés fázisainak elvégzéséhez, ha a rendszer CD-ről való indításakor a gazdagép merevlemezeinek felhasználását tiltjuk (`noswap` boot opció).

Mivel tapasztalatunk szerint a Linux rendszer a naplózó fájlrendszerek „csak olvasható" paraméterezéssel való felcsatolásakor sem őrzi meg a fájlrendszert tároló médium byte pontos állapotát, vizsgálatainkat a merevlemez, vagy az egyes partíciók kriptográfiai ellenőrzőösszeg számításával hitelesített duplikálásával kezdjük. Erre tökéletesen megfelel a `dd` parancs és valamelyik hash számító program, például az `md5sum` (bizalmatlanok más függvényeket is használhatnak). Egy partíció (példánkban a `/dev/hda1`) lementése az említett programok felhasználását és az on-line fejezetben leírt távoli adatmentési megoldást használva a következőképp történhet:

```
dd if=/dev/hda1 |...
md5sum -b /dev/hda1
```

Az `md5sum -b` kapcsolója a bináris feldolgozási módot kapcsolja be. Az ellenőrzőkódok egyezéséhez az on-line szekcióban említett mini-fájlszerverben megadott `md5sum` parancs hasonló paraméterezése szükséges.

Az így elkészített másolatot ezután tetszőlegesen duplikálhatjuk, felcsatolhatjuk, hisz az eredeti példányról újabb hiteles másolatokat készíthetünk a későbbiekben.

A swap partíciók, a kompromittált fájlrendszereken talált gyanús fájlok, valamint a törölt fájltöredékeket tartalmazó üres lemezterületek elemzésére szolgálhat a `strings` segédprogram, mellyel nyomtatható, szótöredéknek vélhető egybefüggő karaktersorozatok felkutatására van lehetőségünk.

A már említett TCT (The Coroner's Toolkit) sajnálatos módon főleg az ext2 fájlrendszer részletesebb elemzésére alkalmas eszközöket tartalmaz, a napjainkban elterjedt modern naplózó fájlrendszerekhez nem, vagy csak ritka esetben találhatunk hasonlóan erős eszközöket. Hasznossága miatt megemlítendő a `lazarus` nevű segédprogram, amely – a `strings` funkcióihoz hasonlíthatóan – jellegzetes szerkezetű fájltöredékek (például hálózati lehallgatási eredményeket tartalmazó naplófájlok) után való kutatásra alkalmas.

A fenti programokon kívül jellegzetesen felhasználhatók még az alapértelmezett Linux környezet által nyújtott segédprogramok, mint például a fejlett mintaillesztést lehetővé tevő `grep`, vagy a hosszú struktúrákban való navigációt segítő `less`.

Bizonyos esetekben további specializált visszafejtő vagy dekódoló segédprogramokra is szükség lehet, ezek ismertetése azonban már túlmutat jelen dokumentum keretein, de általánosan is elmondható, hogy mind az on-line, mind az off-line bizonyítékgyűjtés során felhalmozott adatmennyiség elemzése, értékelése és a következtetések levonása erősen intuitív, jelentős rendszer és programozási ismereteket, általános szakmai tudást és a körülmények részletes, pontos ismeretét igénylő feladat.

8. Kiegészítő biztonsági elemek

Ezek a kiegészítések már erősebb kényelmi sérelmeket vagy nagyobb költségeket okozhatnak, ezért óvatosan ajánljuk, de legalább érdemes tudni róluk, hogy alkalomadtán mi mindenre terjedhet még ki a figyelem.

E megoldások áttekintése különösen ajánlott azok számára, akik hajlamosak azt hinni, hogy a biztonság az valamilyen egy és mindenható szoftver telepítését és a központból konzolról elvégezhető beállítások tömegét jelenti.

8.1. Hardver és környezet

A Faraday-kalitka elve iskolai tanulmányokból is ismert, de a fizikai védelmi berendezések talán túlzónak hatnak egy kutatóintézet vagy egyetem környezetében. Ugyanakkor megfontolandó, hogy egy ipari kapcsolat esetében a kutatási eredmények milyen értéket jelenthetnek az ipari kémek számára, ezért érdemes-e olyan biztonsági megoldásokat alkalmazni, amelyek meggátolják az információ külső lehallgatását.

A védelem a teremtől (pl. fóliázott üveg, elektromágneses lehallgatás elleni védelem, mint a sugárzást tompító tapéta) a számítógépig terjedhet (árnyékolt monitor, kábelek), de néha érdemes a gép mögé is nézni, hogy nem telepítettek-e valakik olyan billentyűzetfigyelő eszközt, amellyel a billentyűzetleütések közül azokat figyelik, amelyek jelszavakat tartalmaznak.

Amit nagyobb költség nélkül meg lehet oldani, az legalább a teremkulcsok naplózott és jogosultság-ellenőrzéssel végzett hozzáférés-kezelése (pl. a terem kulcsához csak azok férhetnek hozzá, akik a portán elhelyezett kulcsos ládához rendelkeznek kulccsal, és szerepelnek a portára leadott listában, hogy ők felvehetik a kulcsot).

8.2. Smart card / Intelligens kártya

A tanulmány előző részében említettük az egyes technológiákat, így most csak annyit említünk meg, hogy a ToReS CD-n helyet kapott a PCSC csomag is, melynek segítségével a számítógépre kötött kártyaolvasó, és a bele helyezett kártya használható akár SSH kommunikációhoz, vagy éppen PAM-modulként a bejelentkezéshez is.

Manapság mind a Windows mind a Linux rendszerek támogatják az intelligens kártyák használatát, de sajnos még van tennivaló az egyes kártyák és rendszereik közötti kompatibilitás eléréséhez (pl. egyik Java Card fejlesztőkörnyezetben készült alkalmazás futtatható legyen másik gyártó Java Card rendszerében, azaz kártyájával).

A Windows rendszerekhez többnyire pénzért érhetők el a fejlesztőkörnyezetek, de a Linux rendszerrel dolgozók a MUSCLE program lapjáról egészíthetik ki disztribúciójuk kártyás képességeit (ld. http://www.linuxnet.com). Röviden tekintsük át a telepítés lépéseit[15].

8.2.1. Intelligens kártyákról

Az intelligens kártyákról annyit érdemes mindenképpen tudni, hogy alap felépítésükben két fajtájuk van: mikroprocesszoros és memóriakártya. A memóriakártyák csak adattárolásra használhatóak, általában van egy biztonsági részük is ami megakadályozza az illetéktelenek hozzáférését a kártya adataihoz. A mikroprocesszoros kártya tartalmaz mikroprocesszort és három fajta memóriát (ROM, EEPROM, RAM). A mikroprocesszor miatt ezek a kártyák végre tudnak hajtani utasításokat, és tárolni tudnak adatokat a megfelelő memóri-

[15] A lépések leírását és tesztelését Zemniczki Zoltán végezte.

ában (többször írható nem felejtő memóriában). Ezen okok miatt biztonságosabbak a memóriakártyáknál.

A mikroprocesszor kártyákkal kétfajta kommunikációs protokollon keresztül lehet kommunikálni: T=0 és T=1. T=0 esetén „egy utasítás – egy válasz" byte-alapú a kommunikáció folyamata, míg T=1 esetén blokkokban küldünk parancsokat és a válaszokat is blokkokban kapjuk. A kommunikáció/parancs egységeket APDU-nak hívjuk (Application Programming Data Unit), és ezeknek felépítésükben és lehetséges tartalmukban meghatározott formátuma van.

8.2.2. Kártyaolvasó telepítése

A mostani kártyaolvasók kapcsolódási felülete a géphez soros vagy USB porton keresztül történik. A soros olvasók esetében telepíteni kell egy driver-t ami az internetről letölthető ez nem mindig a gyártó honlapján található meg (általában a MUSCLE honlapján http://www.linuxnet.com címen a Software → Driver útvonalon megtalálható mindaz, amire szükség lehet).

A program telepítésének módja a README vagy az INSTALL fájlban megtalálható. Általában három parancs követi egymást: ./configure, make és make install. A ./configure-hoz lehet különböző kapcsolókat megadni, amellyel az olvasó beállításait lehet változtatni(ez le van írva az INSTALL fájlban, de olyan is lehetséges hogy csak a make és make install parancsokat kell kiadni). Ezen driver-ek tartalmazhatnak egy tesztelő programot is, amellyel ellenőrizhetjük az olvasó helyes működését.

Az USB-s olvasóknál nem kell az előző lépés. A kapcsolatot a gép és olvasó között egy middleware program a PCSC (Personal Computer Smart Card) valósítja meg. Általában a Linux már alapból feltelepíti a szükséges csomagokat, de érdemesebb a legújabbat letölteni ez is megtalálható a MUSCLE honlapján. A program telepítéséhez sok más program is szükséges ezért elsőre nem mindig sikerül. Ennek a programnak is megtalálható a telepítési útmutatója. Itt is a fentebb említett három parancsot kell kiadni egymás után. A ./configure parancs kiadásánál összegyűjti a rendszer információit, de nem mindig talál meg mindent a program (ezeket általában hibával jelzi is). A hiba leírásánál általában leírja, hogy milyen csomagot nem talált. Ezt grafikus felület estén általában gyorsan fel lehet telepíteni a csomagkezelővel (packager) amely majdnem minden Linux disztribúcióban megtalálható. A ./configure parancsnak vannak különböző kapcsolói:

- --prefix=/könyvtárnév megadhatjuk hogy hova telepítse a PCSC-t,
- -enable-FEATURE ezzel engedélyezünk különböző opciókat a PCSC futásához (pl. Dameon, USB).
- USB olvasóknál ./configure -enable-usb parancsot kell kiadni, hogy a PCSC-nek engedélyezzük az USB olvasók használatát.

Az USB olvasókhoz már csak a driver-ünket kell telepíteni. A driver szintén letölthető a MUSCLE honlapról általában a CCID driver-rel működik mindegyik olvasó. Ezt letöltve majd telepítve a pcsc/drivers könyvtárba az USB olvasó működésre kész.

A soros olvasónál létre kell hozni egy reader.conf fájlt, ezt a PCSC-vel is megtehetjük. A pcsc/bin könyvtárban található egy installifd program, amely először rákérdez az olvasó paramétereire majd létrehozza a reader.conf fájlt, és ezt bemásolva a /etc könyvtárunkba a soros olvasó is működésre kész.

8.2.3. A kommunikáció első lépései és formátuma

A PCSC elindításához a pcscd parancsot kell kiadni ilyenkor a háttéren elkezd futni a program, és máris lehet kommunikálni az olvasóba behelyezett kártyával. Ha szeretnénk látni, hogy éppen mi történik az olvasó és a kártya között, akkor érdemesebb a –f (a PCSC az ablakban fusson) és a –d stdout (az üzenetek megjelennek a standard outputon) kapcsolókat is használni. Ebben az esetben azt is kiírja, ha kártyát helyezünk vagy távolítunk el az olvasóból. Minden készen áll arra, hogy APDU üzenetekkel kommunikáljunk a kártyával.

Az APDU parancsoknak általános formátuma CLA INS P1 P2 Length Data, mindegyik 2 hexa számból áll. A data nem 2 hexa hanem attól függ, hogy mennyi adatot akarok küldeni a kártyának. A CLA kártyafüggő, pl.: a programozható kártyáknak 00, de a GSM kártyáknak 08. INS értéke attól függ, hogy milyen parancsot akarunk a kártyának küldeni, de mindezt általában a kártya dokumentációjában leírják. P1,P2 paraméterek a parancsokhoz pl. 1-es vagy 2-es PIN-kódot küldöm a kártyának. A Length padig az adat hossza hexaban.

8.2.4. Nyelvek és wrapper-ek

A kommunikáció a kártyával bármilyen programozási nyelven történhet. A PCSC-hez találhatóak különböző wrapperek a MUSCLE honlapon, amelyek segítségével bármilyen nyelven lehet parancsokat küldeni a kártyának. A parancsok kiadása történhet C program írásával is a pcsc/doc könyvtárban megtalálható az API hozzá. Mostanában egyre népszerűbb a magasszintű nyelveken való programozás. Erre hozták létre a wrappereket pl.: Java nyelven a jpcsc segítségével lehet „konvertálni" a kódunkat. A wrapperek telepítése sem sokban különbözik az előzőekben leírt telepítéstől (ld. fentebb a három parancsot). A ./configure helyett a főkönyvtár tartalmazhat egy konfigurációs fájlt. Amit mindkét esetben biztosan meg kell adni az a PCSC könyvtár helye. Miután feltelepítettük a wrappert jpcsc esetén be kell importálni a jpcsc.jar-t. Ezután megírhatjuk a programot, az API alapján. Az src/sample könyvtárban található az APDUTool.java fájl, ennek kipróbálásával majd végignézésével egyszerű megírni az első programunkat.

8.3. Biometria

Manapság nagyobb beszerzéseknél előfordul, hogy a gép mellé adnak – vagy laptop esetén már beépítve érkezik – az ujjlenyomat-olvasó. Mivel a tapasztaltak tudják, hogy a biztonság nem egy termék, hanem eljárás, ezért egy ujjlenyomat-olvasóval rendelkező rendszer még sérülékenyebb is lehet, mint egy e nélkül működő rendszer (pl. a felhasználó azt hiszi, hogy ha van ilyen eszköze, akkor nem kell erős jelszavakat használnia, és a beléptetőrendszer megengedi, hogy ha nincs bekötve az ujjlenyomat-olvasó, akkor a jelszavas belépés működjön...).

Az ilyen eszközök beszerzésénél ne a szórólapoknak higgyünk, hanem saját tesztjeinknek, hogy mennyire használható a rendszer az ujjlenyomatok rögzítésére (mennyire megfelelő és megbízható a mintavételezési eljárása), majd a hitelesítésre (pl. 1:N alapú, amikor a kapott mintát a tárolt adatbázissal hasonlítja össze – kerülendő, mert ekkor védeni kell az adatbázist is a lopás vagy hamisítás ellen, vagy 1:1 alapú, és a mintát egy intelligens kártya tárolja, amin nehezebb a minta kompromittálása – ez már a jobbak közé tartozik).

Fontos, hogy a 3T (tudás – PIN, tulajdon – kártya, tulajdonság – biometria) hármasból a megfelelő szintű biztonságot legalább két faktor használatával lehet elérni, így önmagában a biometria sem elegendő a megfelelő hitelesítéshez. Hasonló módon fontos ismerni, hogy a rendszer detektálja-e, vajon élő minta alapján végez-e mintaelemzést, létezik-e hátsó riasztás a rendszerben (pl. ujjlenyomat esetén fenyegetettség alatt használható a „riasztást ki-

váltó ujj"), és milyen módon befolyásolható a téves felismerés és a téves elfogadás aránya (ezek egymás kárára állíthatók!) az igényeknek megfelelően.

A biometriáról magyar nyelven bővebben olvashatunk és egy előadást is megnézhetünk a Biztostű portálon [Biztostű].

8.4. Honeypot

A honeypotok egyre népszerűbb hálózatvédelmi és a támadók szokásait elemző eszközök. Tulajdonképpen arról van szó, hogy „csali" számítógépeket, hálózatokat helyezünk el azért, hogy az ezeket ért támadásokból különböző következtetéseket vonhassunk le. Ez teljesen passzív védekezési módszer, vagyis önmagában nem alkalmas védekezésre, ennek ellenére felhasználási lehetősége meglehetősen kiterjedt:

- Csali számítógépek elhelyezése, vagyis olyan másra nem használt gépek, amelyekhez történő csatlakozási kísérletek nagy valószínűséggel illegális forgalmat jeleznek, így észrevehetünk támadási kísérleteket, de vírusokat is.
- Sok ilyen hamis géppel jelentősen lassítható a különböző szkennelések, ill. az ezeket használó vírusok, wormok terjedése.
- Hamis szolgáltatások nyújtása, amivel tanulmányozható a rosszindulatú próbálkozók stratégiája. Pl. egy hamis, open-relayként viselkedő levelező szerverrel hamar felismerhetjük, hogy azok az automaták, amikkel ilyen szervereket keresnek, szinte mindig küldenek egy próbalevelet egy létező címre.
- Akár igazi gépeket is kitehetünk, úgymond prédának.
- Automatikusan gyárthatunk mintákat a behatolás érzékelő program (pl. snort) számára.

A honeypotokat általában két kategóriába szokták sorolni. Az egyik a *low interaction honeypots*, amelyekbe az emulált szolgáltatásokat nyújtó megoldásokat soroljuk. Ezek csak úgy csinálnak, mintha bizonyos szolgáltatással „beszélgetne" a hozzá forduló, de valójában nem csinálnak semmit. Sőt gyakran teljes – sokszáz gépes – hálózatokat emulál egyetlen gép. A legelterjedtebb ilyen eszköz a honeyd (http://www.honeyd.org), amely jól parámeterezhető, könnyen kezelhető rendszer.

A másik kategória a *high interaction honeypots*, melyek valódi számítógépek, valódi szolgáltatásokkal. Ezek előnye, hogy működésük nagyon részletes napló-információkat szolgáltathat, azonban e naplók elemzése nagyfokú hozzáértést igényel, így ha célunk nem kifejezetten az elemzés, vagy biztonságtechnikai fejlesztés, úgy alkalmazásuk nem igazán szükséges.

9. Következtetések, zárszó

A második mérföldkőre kitűzött feladatot elvégeztük, és ennek eredményeképpen létrejött egyrészt egy az előzőhöz képest tömörebb, a szakemberek (rendszergazdák, biztonsági felelősök) számára készült tanulmány, másrészt egy olyan mintarendszer, mely ingyenesen elérhető eszközökkel támogatja a szakemberek munkáját az informatikai biztonság kezelésében. Az összeállított mintarendszer ingyenessége nem jelent minőségi alkut, sok esetben a piacon drága pénzért kapható eszközöket is felülmúlja egy-egy eleme.

A mintarendszer egészét tekintve és a tartalmát adó alkalmazások, programcsomagok és egyebek tekintetében is a rendszer szabadon felhasználható non-profit környezetben. Ezáltal a pénzügyi források alacsonyabb mértéke sem lehet akadály azok számára, akik a biztonságért tenni akarnak, de eddig nem volt olyan megfelelő segítségük, mellyel ezt megtehették volna. Ezáltal azok biztonsága is nőhet, akik megengedhették maguknak, hogy a megfelelő összegeket költség erre, hiszen a fenyegetettségek őket is kevésbé fenyegetik, ha a környezetük is jobban védett.

A harmadik és egyben utolsó mérföldkő ezennel el is indul, és 2006 tavaszára előáll egy olyan keretrendszer, melynek köszönhetően az incidenskezelésre létrehozandó szervezetek vagy kisebb egységek kapnak kézzelfogható segítséget és egyben módszertant munkájuk megtervezéséhez, megszervezéséhez és szervezeti-technikai fenntartásához.

A keretrendszer a megfelelő és széles körben elfogadott módszertanok, szabályok és szabványok vagy éppen ajánlások alapján adja meg az incidenskezelő egység kezdeti és folyamatos munkájához szükséges ismereteket, mintaszabályzatokat és működési módszertanokat. Ez a keretrendszer és az ilyen módon és alapokon történő felépítés biztosítja azt, hogy a több különböző egység térbeli helytől és megalakulási időtől függetlenül tudjon együttműködni, és így olyan informatikai biztonsági védőernyőt képezni, mely alatt dolgozók feladatukra tudnak koncentrálni, és a károkozások mértéke is csökkenthető.

Ezen keresztül térül meg a projektbe fektetett pénz, és ennek a kutatásnak az eredményei szolgálhatnak a jövőben is olyan célokat, melyek a szervezett és tudatos informatikai biztonsági védekezéseket tűzik maguk elé. Nem utolsó sorban a jövő szakemberei is haszonnal forgathatják az írott és az elektronikus anyagokat, temérdek időt takarítva meg a megfelelő források és információk, eszközök és segédanyagok felkutatásában.

Budapest, 2005. június

10. Mellékletek

10.1. ToReS CD indítása

A biztonsági mintarendszert támogató, ToReS (Tools Related to Security) kódnévre keresztelt, élő GNU/Linux rendszer futtatására minden, CD-ROM rendszerbetöltést támogató, korszerű számítógép képes, amely legalább 128 MByte memóriával és legalább VESA 2.0 kompatibilis videóvezérlővel rendelkezik. A rendszer indításakor a CD indítóképernyője tárul elénk:

10.1. Ábra: A beköszönő ToReS logo

Némi várakozás után a rendszer betöltése folytatódik, a felhasználónak azonban lehetősége van az automatikus indítási paraméterek felülbírálására a „`boot:`“ promptban. Az alapértelmezett kernel nevének megadása után a következő paraméterek felsorolásával módosítható az indítás folyamata:

paraméter	hatás
`noscsi`	a rendszerindulás során a számítógépre kapcsolt esetleges SCSI eszközök figyelmen kívül hagyása (például az indítás esetleges megakadásának elkerülése érdekében)
`nousb`	a `noscsi` kapcsolóhoz hasonlóan az USB eszközök figyelmen kívül hagyása
`nofirewire`	a firewire eszközök kihagyása
`nodma`	az IDE merevlemezek DMA alapú működésének tiltása
`usb2`	az USB2 alrendszer kizárólagos használata (USB1 tiltva)
`fromhd`	a rendszer indítása során a CD-ROM eszközök figyelmen kívül hagyása (merevlemezről történő indítás esetén lehet hasznos)
`fromhd=/dev/...`	a ToReS rendszert tartalmazó ISO9660, EXT2 vagy FAT fájlrendszerhez tartozó blokkeszközt megadva az automatikus keresés pontosabbá válik
`2`	a teljes rendszer indítása csak parancssor üzemmódban (runlevel 2), a parancssorból a grafikus felületet az `/etc/init.d/kdm start` vagy az `init 5` parancsok segítségével indíthatjuk el

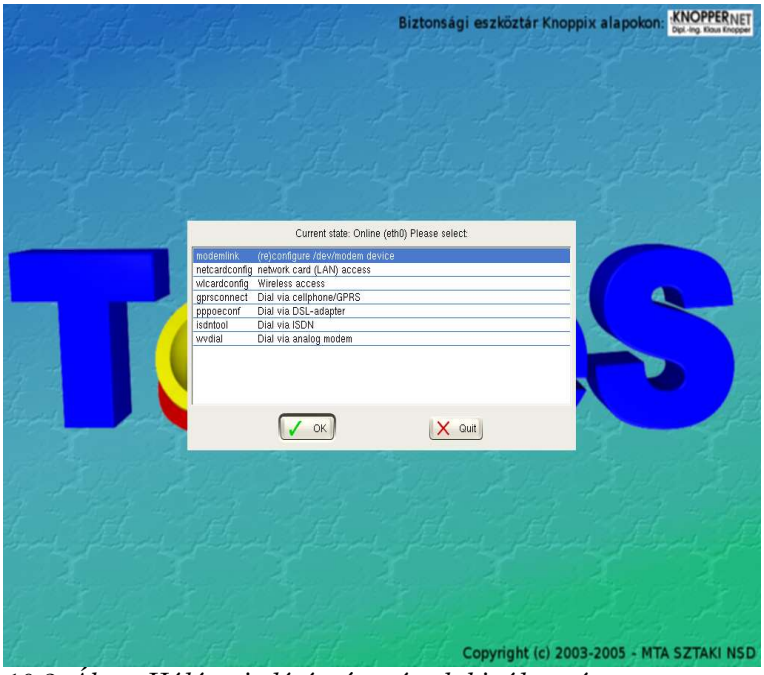
18. Táblázat: A ToReS indítás folyamata

Jelen dokumentum elkészültekor a kernel a `kr2611gr` néven érhető el, vagyis például a számítógép SCSI eszközeinek figyelmen kívül hagyásához és a rendszer parancssorban való indításához a „`kr2611gr noscsi 2`“ parancsot kell a `boot:` prompt után beírni.

A rendszer felsorolt normál indítási módjain kívül még két speciális paraméter áll rendelkezésünkre, az egyik a `memtest`, melynek hatására egy memória-ellenőrző segédprogram futása indul meg, a másik pedig a `shell`, mely egy vésztartalék és hibafejtési célokat szolgáló parancssori környezet indítását eredményezi, melyben a `busybox`, standard Unix parancsokat felsorakoztató, kompakt eszköztár, valamint a több hasonló célú program stílusát támogató `Joe` (`jmacs`, `jpico`, `jstar`) szövegszerkesztő parancsok segítségével kézzel végezhetjük el a rendszer indítását (például a vész-parancssori környezet főkönyvtárában található `linuxrc` script alapján), de más feladatok elvégzésére is alkalmas mini-Linux rendszerként felhasználva egyéb célokat is elérhetünk.

10.1.1. Hálózati konfiguráció

Az 10.3. ábrán látható módon a ToReS rendszer indítása során DHCP kéréseket bocsát ki hálózati interfészein, majd folytatja a rendszer indítását, hogy a kérések esetleges megválaszolása esetén a hálózati eszközök konfigurációjára legyen a Linux kernelnek és a DHCP kliensnek elegendő ideje (1).

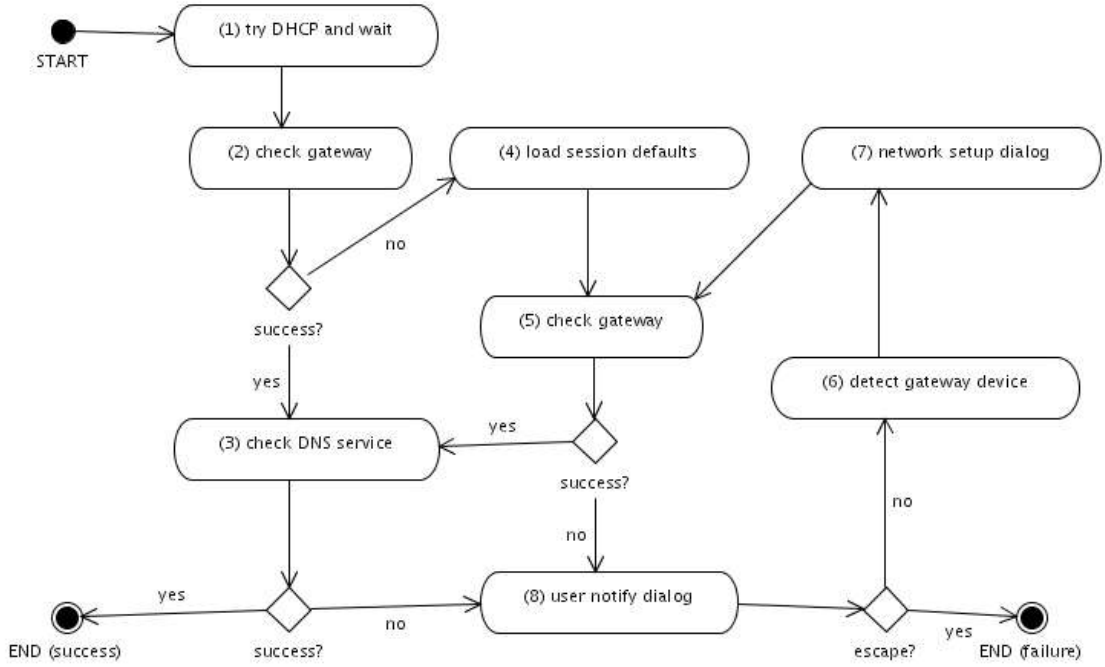


10.2. Ábra: Hálózati elérés típusának kiválasztása

A grafikus alrendszer indítása során egy hálózati detekciót végző rutin fut le, mely sorrendben ellenőrzi a rendszerben esetlegesen az előző sikeres DHCP folyamat által beállított hálózati paramétereket. Első lépésként az alapértelmezett átjáró ellenőrzése történik meg (2). Ez a folyamat két lépésből áll. Első lépésben a hálózati átjáró IP címének ICMP echo-request kérésekkel való ellenőrzése történik meg, majd ha ez sikeres, a BIX két főbb routerének echo-request üzenetekkel való elérését kísérli meg a rendszer.

Ha mindkét echo próbálkozás sikeres, a teszt is sikeresnek számít, azonban bármelyik hibája esetén a rendszer megpróbálja az előzőleg mentett hálózati konfigurációt visszaállítani (4), ez természetesen csak akkor lesz sikeres, ha a cowloop eszközvezérlő naplófájlját merevlemezről állította vissza a rendszerindítási folyamat.

A hálózati konfiguráció visszatöltése után a rendszer, a már ismertetett módon, ismét ellenőrzi a hálózati átjárót (5), majd a teszt sikeressége esetén a DNS szerverek ellenőrzése (2) történik meg a www.sztaki.hu domain lekérdezési tesztjével. Ha a lekérdezés sikertelen, a rendszer egy előre elkészített DNS szerver listából próbál szervereket választani, ha ezek lekérdezése is meghiúsult, a rendszer tájékoztatja a felhasználót arról, hogy a hálózat nem teljesen üzemképes, majd lehetőséget ad a hálózat nélküli munka megkezdéséhez (8), ellenkező esetben a hálózatot teljesen működőképesnek ítéljük (success).



10.3. Ábra: a ToReS hálózatdetektáló folyamata

Ugyanehhez a felhasználó számára tájékoztatást és kilépési pontot biztosító kérdéshez (8) jut a feldolgozás, az hálózati átjáró másodszori ellenőrzésének (5) sikertelensége esetén. Ha a felhasználó nem a kilépést (failure) választja, a rendszer megpróbál hálózati átjárót keresni a számítógépen detektált hálózati eszközökön más hálózatokba tartozó ARP kérések kibocsátásával (6).

Az esetlegesen detektált Internet elérésre alkalmas hálózati eszköz megnevezése mellett a vezérlés egy hálózati konfigurációs dialógusba (7) vezeti a felhasználót, ahol a rendszer által támogatott hálózati eszközök kézzel történő konfigurációjára kerülhet sor. A konfiguráció befejezése után a rendszer visszatér a hálózati átjáró ellenőrzéséhez (5).

A rendszer esetleges hibájának lehet felróni, hogy a DHCP kérésekre kapott működőképes hálózati konfigurációt (1) előnyben részesíti. Ez a viselkedés csak az elmentett cowloop naplófájllal rendelkező, speciális hálózati beállításokat igénylő felhasználók esetén lehet zavaró, számukra a DHCP kérések indítása a hálózatdetektáló szkript konfigurációjában kikapcsolható.

10.1.2. A rendszer általános használatáról

A ToReS rendszer az elterjedten használt, felhasználóbarát KDE grafikus felületet biztosítja a mintarendszer alapjául szolgáló alkalmazások és konfigurációs lehetőségek megismertetésére. A teljes értékű grafikus, asztali operációs rendszer alapfelhasználójának neve

`knoppix`, mely egyben rendszergazdai képességekkel is rendelkezik a `sudo` parancs használatának segítségével.

A rendszer számos alkalmazása csak különböző azonosítási lépések során engedélyezi használatát. Mivel az azonosítási rendszer alapjául szolgáló Linux szerkezet ezt nem teszi lehetővé, a ToReS nem rendelkezik egyetlen központi adminisztratív felhasználóval, azonban az azonosítási kérelmek egységesítésének céljából az alapértelmezett jelszó mindenütt „`tores`“. Ezt a jelszót kell megadni minden olyan esetben, ahol a rendszer azonosítási kérelemmel fordul a felhasználó felé, azonban a felhasználói név „`root`“, „`knoppix`“ vagy „`tores`“ lehet, mely a helyzettől és alkalmazástól függően könnyen, vagy nehezebben következtethető ki. Mivel a rendszer bármely részének, így az alapértelmezett felhasználóneveknek és jelszavaknak, megváltoztatására lehetőség van, a jelszavak pedig sok alkalmazás sokféle jelszóadatbázisának formátumában, általában egyirányú kódolással kerülnek tárolásra, nem készülhetett az egyes programok azonosítási igényeit tárgyaló dinamikus súgó.

A grafikus rendszer indítása után a ToReS rendszer logóját ábrázoló gyorsindító-ikon segítségével érhetők el a mintarendszer alapját képző, feladatkör szerint kategorizált alkalmazásindító linkek, melyek zárójelbe tett értelmező leírását megelőzően egy `M:` jelöli, ha az adott alkalmazás parancssori felhasználású, vagyis a link csak a dokumentációs rendszer indítását, és a megfelelő `man` oldal megjelenítését végzi el.

Az alkalmazások, ahol ez a lehetőség rendelkezésre állt, magyar nyelven működnek, azonban a grafikus felhasználói felületen kívül, főként a speciális hálózati programok működtetéséhez az angol nyelv ismeretén kívül néhány esetben az IP alapú protokollok és az általános operációs rendszerek ismerete is szükséges lehet. A felsorakoztatott, de a jelen dokumentumban leírt mintarendszer tekintetében részletesebben nem tárgyalt eszközök inkább a teljes funkcionalitás, nem pedig a triviális felhasználás igényével kerültek összeválogatásra.

10.4. Ábra: ToReS eszközök (menürészlet)

10.1.3. Állapotmegőrzés, remastering

A ToReS rendszer felhasználása során fontos lehet a felhasználók számára a CD könnyű módosítása. Ez a `cowloop` eszközkezelő által lehetővé tett, teljes írási támogatás miatt egy-

szerűen elvégezhetővé vált. A felhasználó a Debian rendszerekben szokásos `dpkg`, `apt-get` parancsokkal, valamint ezek frontend-jeivel, de akár egy egyszerű szövegszerkesztővel is átformálhatja a rendszert, eltávolíthatja a feleslegesnek ítélt csomagokat, vagy újabbakat telepíthet.

Ha a ToReS rendszer egymás utáni indításai között a bevezetett változásokat meg szeretnénk őrizni, nincs más dolgunk, mint egy 0 byte méretű, `tores.cow` elnevezésű fájlt létrehozni rendszerünk egy EXT2, EXT3 vagy VFAT rendszerű partíciójának `tores` nevű alkönyvtárában. A rendszerindítás során a ToReS CD észleli a létrehozott fájlt, és felhasználja a `cowloop` rendszer változáskövetési funkciói során a rendszer állapotánál tárolása céljából.

A rendszer módosítása után nincs más teendő, mint bekonfigurálni a ToReS rendszerhez készített, `/usr/src/remaster/remaster` nevű segédprogramot, majd elindítani azt, és várakozni az új, módosított ToReS verziót tartalmazó, bootolható CD ISO image létrejöttére. A remaster program szükség esetén képes az initrd image új kernelhez való előkészítésére is.

A remaster futásához minimális követelmény egy legalább 4 Gigabyte üres tárterülettel rendelkező, írható partíció, és 1 Gigabyte RAM. A RAM természetesen virtuális memóriával is pótolható, sőt, ha a remaster program érzékeli, hogy a rendszerben kevés a memória, a céllemezen plusz 1 Gigabyte terület árán el is végzi a virtuális memória lapozófájljának konfigurációját.

A remaster folyamat a következő lépésekből áll:

1. A rendelkezésre álló memória és lemezterület ellenőrzése.
2. Szükség esetén a virtuális memória lapozófájljának létrehozása.
3. A felhasználó konfigurációs beállításai alapján esetlegesen új initrd állomány előkészítése.
4. Üres, 0-s bájtokkal feltöltött sparse fájl létrehozása 4G tárterülettel. A sparse fájlok egyes fájlrendszerek esetében a csupa 0 blokkokat gazdaságosabban lyukakkal („hole”) jelölve tárolják.
5. A sparse fájl formázása EXT2 fájlrendszerre, az így létrejött fájlrendszer becsatolása egy átmeneti könyvtárba.
6. A gyökérfájlrendszer teljes tartalmának másolása az átmenetileg becsatolt könyvtárba.
7. Az sparse fájl kicsatolása az átmeneti könyvtárból.
8. A sparse fájl tömörítése a `cloop` eszközvezérlő által támogatott formátumban. Ehhez a művelethez szükséges a sok memória, mivel a tömörítőprogram az indexek elkészítéséhez az egész blokkstruktúrát a memóriában tárolja.
9. A sparse fájl törlése után a létrehozott, tömörített EXT2 gyökérfájlrendszert tartalmazó cloop fájl, valamint a szükséges segédfájlok, vagyis a kernel és initrd image és az `isolinux.bin` alapján új El-Torito formátumú ISO9660 CD image generálása.
10. A tömörített `cloop` fájl, és az átmeneti könyvtárak törlése.

A cloop fájlrendszer optimális helykihasználása érdekében a sparse fájlban létrehozott EXT2 fájlrendszer minimális változtatására kell törekedni, mivel a `cloop` blokkeszközön az üres hely tömörített blokkjai is helyet foglalnak, hisz az alkalmazott `gzip` tömörítési eljárás

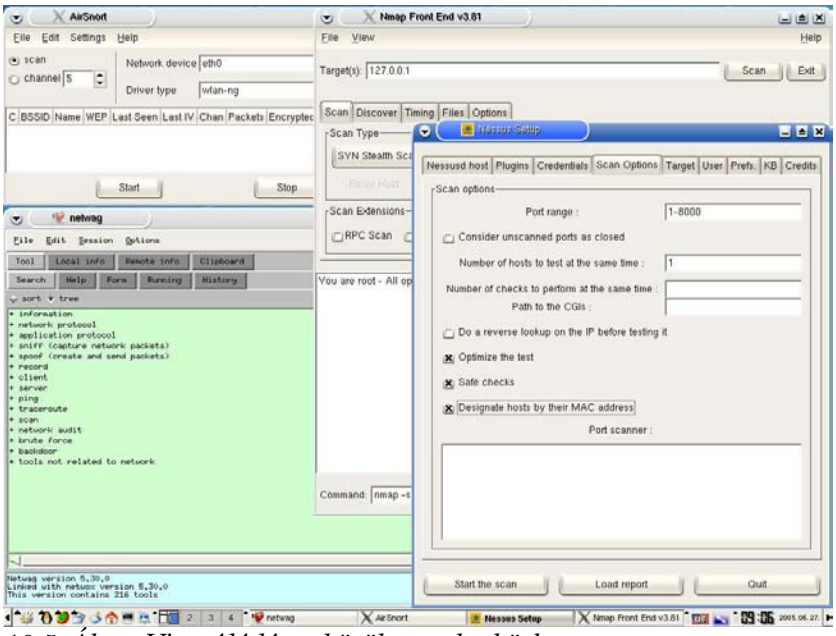
nem ismeri az üres blokk fogalmát. Ezért a gyökérfájlrendszer egy menetben történő másolására az `rsync` nevű fejlett szinkronizációs programot használjuk, melynek másolás során használt átmeneti állományainak helye konfigurálható.

10.1.4. Tipikus felhasználási lehetőségek

Az 5.2 ábrán látható, biztonsági szempontokat figyelembe vevő módon kiépített hálózat számos pontján használhatók fel a ToReS rendszer által nyújtott megoldások. Természetesen az alább felsorolt megoldások nem mindegyike tekinthető kötelezőnek, hisz a ToReS által nyújtott funkciók bármelyikére található más, számos esetben költséges, megoldás is. A felsorolás inkább áttekintésként szolgál a rendszer teljes felhasználási körét illetőleg:

- A kliens számítógépek („1”, „7”) esetében a ToReS rendszer nehezebben alkalmazható, de megfelelő módosításokkal, kiegészítésekkel a live CD jó alapot szolgáltathat egy képzettebb rendszergazda számára egy asztali munkaállomás kialakításához is. A rendszer által alapszolgáltatásként nyújtott, biztonságos bejelentkezést, VPN kapcsolatot, TLS/SSL titkosított kommunikációs csatornákat elérhetővé tevő kliens alkalmazások jól működnek együtt a ToReS által megvalósított kiszolgálókkal, tűzfalakkal, de más termékek hasonló szolgáltatásaival is.
- A kliens számítógépekhez hasonlóan a Linux operációs rendszert támogató groupware („3”), adatbáziskezelő („4”) és általános kiszolgáló („5”) megoldások számára is megfelelő alapot biztosíthat a ToReS által nyújtott RBAC szolgáltatással kialakítható védett futtatási környezet.
- A ToReS rendszer által nyújtott, biztonságos PROXY („8”) szolgáltatások jól felhasználhatók egyéb tűzfalrendszerek kiegészítőiként, és biztonsági beállíthatóságuk mellett a korlátozott sebességű Internet eléréssel rendelkező hálózatok esetében minőségi javulást is eredményezhetnek az Internet felhasználásában.
- A „9” jelzésű DMZ kiszolgálók és a „10” jelölésű cseréjekor, az előző fejezetben említett megjegyzések mellett, még ki kell kötni azt is, hogy egy biztonsági szempontokat figyelembe vevő rendszer telepítése és használata esetén a konfigurációt és felügyeletet végző rendszergazdán is sok múlik, vagyis a beállítások hibás, vagy túl lazára történő megváltoztatása esetén a ToReS rendszer által nyújtott szolgáltatások a lecserélt rendszernél kevéssé biztonságos helyzetet is eredményezhetnek.
- Mivel egy általános hálózatban (5.2 ábra) nem feltétlenül található a 5.3 ábrán található, „11” jelzésű VPN szerver, a ToReS rendszer kézenfekvő felhasználási lehetőségei közé tartozik ennek a rendszernek a megvalósítása. A ToReS által támogatott IPSec, MPPE és OpenVPN protokollok a VPN felhasználási lehetőségek nagy részét lefedik.
- A DIAL-IN szolgáltatás („12”) az esetek többségében már a ToReS rendszer bevezetése előtt is megtalálható, hiszen egy funkcionális igényforrásból származó, ráadásul kiveszőfélben levő, megoldásról van szó. Ennek ellenére a ToReS rendszer megfelelő szoftveres támogatást nyújt egy behívóközpont szolgáltatásainak ellátására. A speciális hardveres eszközök, modem-poolok vezérléséhez a rendszermag újrakonfigurálása, kiegészítése válhat szükségessé.
- A biztonsági szempontokat is figyelembe vevő rendszerek által alkalmazott tűzfal-megoldások („13”, „14”) esetében a ToReS rendszer magas szintű szolgáltatásokat, és kényelmes konfigurációs felületet nyújt. Képességeit tekintve a rendszer megelőzi a kereskedelmi forgalomban kapható kompakt tűzfalmegoldásokat, azzal a megjegyzéssel, hogy a ToReS rendszer nem csak tűzfalként, hanem egyben routerként és forgalomanalizátor-

ként is üzemeltethető, ami további költségmegtakarítás mellett az egyszerű, áttekinthető karbantarthatóságot is lehetővé teszi.



10.5. Ábra: Vizsgálódó eszközök munka közben...

10.2. TOP-listák

Elérhetők olyan időnként frissített toplisták, melyek részletesen leírják, hogy melyek az aktuális leggyakoribb fenyegetettségek, és ez alapján lehet a védekezési súlypontokat is tervezni a saját rendszerünkben.

10.2.1. SANS topten, what works?

A SANS folyamatosan közzé teszi az informatikai biztonsággal kapcsolatos támadás-toplistákat Windows és Unix rendszerekre 10-10 pontba összegyűjtve. 2005 nyarán a listák ilyen címszavakat tartalmaznak:

Windows elleni fenyegetettségek:	UNIX elleni fenyegetettségek:
#1 Web Servers & Services	#1 BIND Domain Name System
#2 Workstation Service	#2 Web Server
#3 Windows Remote Access Services	#3 Authentication
#4 Microsoft SQL Server (MSSQL)	#4 Version Control Systems
#5 Windows Authentication	#5 Mail Transport Service
#6 Web Browsers	#6 Simple Network Management Protocol (SNMP)
#7 File-Sharing Applications	#7 Open Secure Sockets Layer (SSL)
#8 LSAS Exposures	#8 Misconfiguration of Enterprise Services NIS/NFS
#9 Mail Client	#9 Databases
#10 Instant Messaging	#10 Kernel

A http://www.sans.org/top20/ címen mindegyik fenyegetettséghez elérhetők bővebb leírások, a felderítéshez szükséges lépések ismertetésével, és a megvalósítható védekezések is

említésre kerülnek. A szabadon elérhető eszközök legtöbbjét (a nevesebbeket és széles körben használhatókat mindenképpen) a ToReS CD is tartalmazza.

10.2.2. Egyéb listák, előrejelzések, várható trendek

Ha egy intézményben még arra is lehetősége van egy rendszergazdának, hogy tervezze a jövőbeli informatikai biztonsági koncepciót, akkor ezt a kivételes helyzetet ki kell használni, és olyan írásokat venni alapul, melyek a jövőt illető tévedési valószínűsége kisebb a mások által jósoltaknál.

Azoknak az informatikai biztonsági feladatokat ellátó szakembereknek, akik szemléletüket is formálni akarják, Bruce Schneier írásait ajánlhatjuk. A régebbi írások, cikkek, havi hírlevél, blog mind elérhetők a http://www.schneier.com címen.

Mások is készítenek elemzéseket és trend-leírásokat, pl. a CERT-ek a hozzájuk befutó jelentések alapján, a Symantec a Riptech felvásárlása óta a kihelyezett szenzorai alapján, vagy említhető még a SecurityFocus (Bugtraq listát is üzemeltető cég) hasonló szenzoros megoldása is. Ezekre mind elmondható, hogy minél több és jobb eloszlású a szenzorhálózat, annál több és pontosabb információ nyerhető a fenyegetettségek számáról, terjedéséről és később a forráshelyek számának szűkítésében is szerepet vállalhatnak a „térképeik" alapján. Ezekből vonják le következtetéseiket, melyeket rendszeresen közzétesznek.

10.3. Zsebsorozat

Sokszor szükség lehet egy rövid tömör leírásra, melyet akár zsebre is vághatunk, és remélhetőleg ritkán kell elővenni, de éppen ezért könnyen elfelejthetjük a teendőket. A kétoldalas lapok háromba hajthatók, és így kényelmesen elférnek a zsebben is.

A Windows és Linux rendszerekhez készült betörésészlelést segítő „kisokosok" a CD-hez mellékelve találhatók, de szabadon elérhetők a következő címeken is:

http://www.cert.hu/ismert/5incidens/ID_Linux_hun.pdf

http://www.cert.hu/ismert/7incidens/ID_Windows_hun.pdf

11. Irodalomjegyzék, ajánlott irodalom

A tanulmányban sok olyan hivatkozás van, ami szűkebben csak az adott résznél volt említésre méltó, így ezekre csak ott találhatók meg a hivatkozások. Az általánosabb, több területen is hivatkozott irodalmakat gyűjtöttük össze ebben a részben, így ez többnyire ajánlott irodalom, mintsem minden elemében meghivatkozott irodalomjegyzék.

[Adv_HB] Chris May et. al. CERT®/CC Training and Education Center Advanced Information Assurance Handbook, March 2004
http://www.sei.cmu.edu/pub/documents/04.reports/pdf/04hb001.pdf

[Artisjus] Szerzői jogi hivatal és a szerzői joggal kapcsolatos információk:
http://www.artisjus.hu/aszerzoijogrol/jogszabalyok.html

[AVT] Az Adatvédelmi törvényről és az Adatvédelmi Biztos Hivataláról szóló internetes források:
http://abiweb.obh.hu/abi/jogszabalyok/1992_LXIII.htm

[Attack_tree] Bruce Schneier publikációja alapján felépített keretrendszer, a taxonómiák rokona, de más formába öntött felépítése.
http://www.schneier.com/paper-attacktrees-ddj-ft.html

[Attack_treeP] Támadási fa készítésére szolgáló elmélet, mely az [Attack_tree] publikáció alapján készült kibővített elméleti alapokon nyugvó gyakorlati alkalmazás.
http://www.amenaza.com/

[Biometria] Előadás a biometriáról és az előadó honlapja:
http://www.biztostu.hu/multimedia/videok_es_hozzavalok.htm
http://home.mit.bme.hu/~orvos/oktatas/speci/

[Biztostű] ITEM K+F 350 nyilvántartási szám alatt készített Informatikai Biztonsági Oktató portál.
http://www.biztostu.hu/oktatas/biztonsag-szervezes.htm

[Brooks] Brooks törvénye: „Egy késésben lévő projekt bővítése újabb programozókkal további késést eredményez”. Általánosabban: egy projekt bonyolultsága és kommunikációs költsége a fejlesztők számával négyzetesen emelkedik, míg az elvégzett munka mennyisége csupán lineárisan nő (The Mythical Man-Month: Essays on Software Engineering, ISBN: 0201835959)
http://www.hwsw.hu/oldal.php3?cikkid=848&oldal=5

[BS7799] British Standard 7799, ld. [ISO17799]

[Bugtraq] A Securityfocus nevű cég (a kezdeti sikerek után alakult azzá) által üzemeltetett levelezési lista az informatikai biztonság széles spektrumát lefedő témák közlésére és megvitatására
http://www.securityfocus.com/archive/1

[CC] Common Criteria, ld. [ISO15408]

[CERT] Computer Emergency Response Team, számítógépes vészhelyzetre reagáló egység, melynek az alapító intézményen kívül számos országonkénti szervezete is létezik:
http://www.cert.org
http://www.cert.hu

[CobiT] Control Objectives for Information and related Technology (Irányelvek az információ-technológia irányításához, kontrolljához és ellenőrzéséhez)
http://www.isaca.org/cobit.htm

[CobiT_HU] A CobiT magyar fordítása
http://www.ihm.hu/kutatasok/tanulmanyok/tanulmanyok_20030623_1.html
http://tinyurl.com/2clld

[Dictionary] Egy szabadon használható és le is tölthető informatikai biztonsági szakszavakat tartalmazó szótár:
http://www.itsecurity.com/dictionary/dictionary.htm
http://www.itsecurity.com/BigDic.zip

[DoS_hist] Szolgáltatásmegtagadó támadások rövid története a Washington Post-ban is megjelent cikk alapján:
http://www.computercops.us/article3963.html

[EPIC] Electronic Privacy Information Center, az elektronikus adatvédelemmel foglalkozó szervezet, melynek honlapján sok érdekes információ és hasznos alkalmazás érhető el azok számára, akik a személyiségi jogaikra igényesek:
http://www.epic.org/privacy/tools.html

[FW_evol] Evolution of the Firewall Industry (a tűzfal-ipar fejlődése):
http://www.cisco.com/univercd/cc/td/doc/product/iaabu/centri4/user/scf4ch3.pdf
http://tinyurl.com/38how

[FW_tax] A Brief Taxonomy of Firewalls – Great Walls of Fire
http://www.giac.org/practical/gsec/Gary_Smith_GSEC.pdf

[Fogalomtár] KIKERES, Közigazgatási fogalomtár a Miniszterelnöki Hivatal szolgáltatása (illetve az IHM honlapjáról elérhető a Hunguard cég által összeállított anyag PDF-ben):
http://www.fogalomtar.hu/cstore3/index.fm
http://www.ihm.hu/kutatasok_tanulmanyok/b/it_biztonsagi_fogalomtar.pdf
http://tinyurl.com/2kcvk

[Hist_IDS] Guy Bruneau: The History and Evolution of Intrusion Detection
http://www.sans.org/rr/papers/30/344.pdf

[ICAT] ICAT Metabase Documentation, Your CVE Vulnerability Search Engine. ICAT is a fine-grained searchable index of standardized vulnerabilities that links users into publicly available vulnerability and patch information
http://icat.nist.gov/icat.cfm

[IDS_id] The Science of Intrusion Detection System Attack Identification
http://www.cisco.com/warp/public/cc/pd/sqsw/sqidsz/prodlit/idssa_wp.pdf
http://tinyurl.com/2t2x4

[Igügy] Alapok: http://www.securityfocus.com/infocus/1653 (Windows Forensics: A Case Study, Part One, December 31, 2002), és http://www.securityfocus.com/infocus/1672 (Windows Forensics: A Case Study, Part Two, March 5, 2003) by Stephen Barish

[Infosec] National Information Systems Security (INFOSEC) Glossary, NSTISSI No. 4009, June 5, 1992, (National Security Telecommunications and Information Systems Security Committee, NSA, Ft. Meade, MD 20755-6000). Elérhető PDF és HTML (online) változatban is:
http://www.nstissc.gov/Assets/pdf/4009.pdf
http://www.sciencedaily.com/encyclopedia/national_information_systems_security_glossary
http://tinyurl.com/2rr49

[ISO15408] Common criteria szabványa, mely több különböző ország biztonsági szabványaiból jött létre 1993 környékén. A teljes történet, a régebbi és az aktuális verziók is elérhetők:
http://csrc.nist.gov/cc/

[ISO17799] A BS7799 (British Standard, Information Technology – Code of practice for information security management) szabvány első részéből származó szabvány az informatikai biztonság menedzselésének leírására. Részletesebben ld. []-ben ismertetett tanulmány 2. kötetében.

[ITB] Informatikai Tárcaközi Bizottságok ajánlásai (biztonsággal közvetlenül kapcsolatosak: 8, 12, 16 számú ajánlások)
http://www.itb.hu/ajanlasok/

[Jelszavak] A jelszavak választásáról szóló összefoglaló leírás:
http://www.cert.hu/ismertetok/jelszo.html

[Koziol] Jack Koziol: Dissecting Snort
http://www.informit.com/articles/article.asp?p=101148

[Lawr_IDS] Lawrence R. Halme and R. Kenneth Bauer: AINT Misbehaving: A Taxonomy of Anti-Intrusion Techniques
http://www.sans.org/resources/idfaq/aint.php

[MBSA] Microsoft Baseline Security Analyzer, a Windows operációs rendszerek gyenge pontjait feltérképező alkalmazás, melyhez magyar nyelvű útmutató is elérhető a CERT.HU lapon:
http://www.cert.hu/ismertetok/mbsa/win2k_mbsa.html

[MITS] Magyar Információs Társadalom Stratégiák
http://193.6.108.12/anyagok/stea/Mits/

[MUSCLE] Movement for Using Smart Card in a Linux Environment (Linux környezetben történő intelligens kártya felhasználás irányába történő elmozdulás), projekt és letölthető segédletek honlapja:
http://www.linuxnet.com/

[NAT_RFC] Network Address Translator RFC:
http://www.ietf.org/rfc/rfc1631.txt

[NIDS_impl] How To Guide – Implementing a Network Based Intrusion Detection System
http://downloads.securityfocus.com/library/switched.pdf

[Paramédia] Ajánlások az elérhető médiáért. Mozgalmuk szervezeti hátterét a kiemelten közhasznú Látó-tér Alapítvány, a Vakok Szolgáltató Központja, illetve a W3C magyarországi irodája együttesen adja.
http://www.paramedia.hu/ajanlasaink.html

[PGP] Pretty Good Privacy, azaz egészen jó adatvédelem, ha szabadon fordítjuk. Digitális aláírásra, adat-, kommunikáció- és háttértár-titkosításra. Rövid ismertető leírás (v8.0-ról):
http://www.cert.hu/ismertetok/pgp_win.html

[RossABib] Ross Anderson “Security Engineering” könyvének hivatkozásjegyzéke és saját honlapja:
http://www.cl.cam.ac.uk/ftp/users/rja14/bib_anderson.pdf
http://www.cl.cam.ac.uk/ftp/users/rja14/

[SANS_pol] The SANS Security Policy Project. Mintaszabályzatok szabadon hozzáférhető gyűjteménye.
http://www.sans.org/resources/policies/

[SANS_list] Roadmap to Security Tools and Services Online. A biztonsági eszközök és rendszerek csoportosított listája, és a csoportok meghatározása:
http://www.sans.org/tools/roadmap.php

[SMC_threats] Bruce Schneier cikke az intelligens kártyákat érhető támadásokról (elérhető magyar fordításban is!)
http://www.schneier.com/paper-smart-card-threats.html
http://www.biztostu.hu/tovabbi_anyagok/hallgatoi_munkak/Pal_Lecz/sct.pdf
(http://tinyurl.com/yrqwb)

[Spam1] Brian Burton SpamProbe - Bayesian Spam Filtering Tweaks
http://spamprobe.sourceforge.net/paper.html

[Spam2] Paul Graham. “Better bayesian filtering” January 2003
http://www.paulgraham.com/better.html

[Spam3] John Graham-Cumming. The Spammers' Compendium. September 15, 2003.
http://popfile.sourceforge.net/SpamConference011703.pdf

[SOCKS] Socks protocol Version 5 és az ide kapcsolódó RFC-k:
ftp://ftp.rfc-editor.org/in-notes/rfc1928.txt
ftp://ftp.rfc-editor.org/in-notes/rfc1929.txt
ftp://ftp.rfc-editor.org/in-notes/rfc1961.txt

[Sourceforge] A legnagyobb nyílt forráskódú szoftverfejlesztői oldal több tízezer fejlesztés nyilvántartásával, melyek közül a biztonsággal kapcsolatos fejlesztések listája külön kiemelendő:
http://sourceforge.net/softwaremap/trove_list.php?form_cat=43

[Szerzői_jog] A szerzői jogi törvény informatikára vonatkozó egyes részeit taglaló és kifejtő oldal:
http://www.artisjus.hu/dokumentumok/jog.html

[Tan_1] Az informatikai hálózati infrastruktúra biztonsági kockázatai és kontrolljai (e tanulmány első része), MTA-SZTAKI – IHM, 2004.
http://www.cert.hu/ismert/?cat=00tanulmany

[Tiresias] Guidelines for the Design of Accessible Information and Communication Technology Systems
http://www.tiresias.org/guidelines/index.htm
http://www.tiresias.org/guidelines/access-ability/Access-Ability.pdf

[WindowE] Window of exposure, a „felfedés kerete”, vagyis a biztonsági rések nyilvánossá tételének időablakai és a lehetséges állapotok elemzése és hatásvizsgálata. HTML és PDF változat
http://www.counterpane.com/window.html
http://www.counterpane.com/window.pdf

* * *