Informatikai biztonság és kriptográfia

Jegyzet

Folláth János, Huszti Andrea és Pethő Attila

2010

A tananyag a TÁMOP-4.1.2-08/1/A-2009-0046 számú Kelet-magyarországi Informatika Tananyag Tárház projekt keretében készült. A tananyagfejlesztés az Európai Unió támogatásával és az Európai Szociális Alap társfinanszírozásával valósult meg.

Tartalomjegyzék

Bevezetés

A számítógépek elterjedése és különösen az, hogy az internet szinte minden munkahelyen és háztartásban jelen van lehetőség ad arra, hogy nagyon sok adathoz nagyon rövid idő alatt hozzájussunk. Az internetes világ kinyílt előttünk, de a számítógépeink is kinyíltak a világ felé. Ha az új technológiát nem használjuk tudatosan, akkor akár privát szféránkat is komoly veszély fenyegeti.

Jegyzetünkben az informatikai biztonság kiterjedt problémaköréről adunk áttekintést az alap- és mesterképzésben részt vevő informatikus hallgatóknak. Célunk az, hogy a hallgatók megismerjék az informatikai rendszerekre leselkedő veszélyeket és azokat az ügyviteli és technológiai eszközöket, amelyekkel eredményesen lehet kivédeni az esetleges támadásokat. Jegyzetünk nem biztonsági szakemberek képzésére készült, hanem olyanoknak, akik munkájuk során folyamatosan alkalmazzák az informatikai eszközöket és jövendő munkahelyeiken az informatikai kultúra képviselői lesznek. Nagyon fontosnak tartjuk, hogy ők alaposan az informatikai biztonság alapfogalmait. Tudatos fellépésükkel segíthetik munkahelyük biztonságos működését.

A jegyzet anyagát a Kossuth Lajos Tudományegyetemen 1997-től tartott Kriptográfia, majd a Debreceni Egyetemen 2006-ban bevezetett Informatikai biztonság alapjai előadások anyagainak felhasználásával készítettük. Ezek egy szemeszteres kurzusok, így nincs lehetőség a jegyzetben kifejtett anyag részletes tárgyalására. A szerzők azonban olyan tananyagot akartak készíteni, amely hosszabb távon is használható és a téma iránt mélyebben érdeklődő hallgatók számára is információkat tartalmaz.

Köszönettel tartozunk a jegyzet lektorainak: Dr. Bérczes Attilának, Erdősi Péter Mátéának, Dr. Rózsahegyi Zsoltnak és Tóth Istvánnak, akik szakmai és nyelvi tanácsaikkal jelentősen hozzájárultak a tananyag színvonalának emeléséhez.

Debrecen, 2011. március 27.

1 Az adatvédelem szükségessége és céljai.

1.1 Miért kell védeni az adatokat?

Az adatok kezelése, különösen akkor, ha nagy mennyiségben gyűjtik össze, dolgozzák fel és strukturáltan tárolják azokat, komoly figyelmet érdemel. Ennek szükségességét már a hagyományos, papír alapú adatkezelés során felismerték és gyakorolták. Például a népesség nyilvántartást, a személyi azonosítók használatát, a népszámlálás módját és a szerzői jogokat törvények szabályozták; a vállalatok a gyártási technológiákat vagy a termékek terveit szigorúan védték és védik. Az adatok ugyanis illetéktelen kezekbe jutva komoly veszteséget okozhatnak jogos tulajdonosaiknak, visszaélésekre adhatnak módot.

A számítástechnika széleskörű elterjedésével az adatfeldolgozás technológiája lényegesen megváltozott, tömegessé vált. Az adatok döntő többségét egységes formában, digitálisan gyűjtik, tárolják és továbbítják. Az adatokat koncentráltan adatbázisokban, adattárházakban tárolják, ami a feldolgozás hatékonyságát lényegesen növeli. Kialakultak olyan vállalkozások, amelyek adatok szolgáltatásával foglalkoznak. Az utazási irodák például az utazásra, szállásra és egyéb szolgáltatásokra vonatkozó adatokat adnak el az ügyfeleiknek. A biztosítótársaságok tőkéjének jelentős hányadát az ügyfeleikre vonatkozó adatok jelentik. A tájékozódásunkat jelentősen megkönnyítik a GPS eszközök, amelyek árát döntő részben a memóriájukban tárolt térképek jelentik. Az előző példák illusztrálják azt, hogy az adatoknak komoly értékük lehet, azokra iparágak épülhetnek. Ami értékes, azt pedig védeni kell!

Ma már kialakult és egyre bővül az adatok piaca. Szolgáltatók biztosítják olyan adatok másodlagos felhasználását, amelyek mások számára is értékesek. A rendelkezésünkre álló adattömeg speciális tulajdonsága, hogy mennyisége folyamatosan nő, így a feldolgozásra és a felhasználásra bővülő lehetőséget kínál. A számítógépek és a világháló kiterjeszti a világot is, virtuális terek jönnek létre és élik a maguk világát a reális világgal párhuzamosan. Jó példát jelentenek a játékszoftverek, virtuális múzeumok, képtárak, épületeket bemutató szoftverek. A virtuális tereket nagyon jól fel lehet használni bizonyos helyzetek szimulálására, így például gépkocsi-, mozdony vagy repülőgép vezetés tanulására és gyakorlására. Korábban a gépkocsik és repülőgépek terveiből makettet készítettek és tulajdonságaikat szélcsatornában tesztelték. A szimulációs szoftverek fejlődése lehetővé tette, hogy a makettekkel való drága teszteket a tervezés későbbi fázisában végezhessék el. A durvább hibákat a szimulátor szűri ki. A virtuális világoknak is kialakult az adatvédelmi követelményrendszere és ez - az alkalmazások bővülésével - egyre bonyolultabb lesz.

Személyes adatainkat - az államigazgatás szervezetei mellett - sok helyen tartják nyilván: egészségügyi és oktatási intézmények, bankok, biztosító társaságok, szolgáltatók, egyesületek, hogy csak néhány példát említsünk. Összekapcsolva ezeket életünkről olyan részletes elemzés készíthető, amely már a magánélet sérthetetlenségét is veszélyezteti. Bizonyos személyes adatok, pl. az egészségi állapotra, anyagi helyzetre, vallásra, párttagságra vonatkozó adatok kezelése engedélyhez kötött és különös gondosságot kíván.

Az infokommunikációs hálózatokon nagy távolságba nagy mennyiségű adatot lehet olcsón és gyorsan eljuttatni, illetve adatokhoz hozzáférni. Ez újabb lendületet adott az informatika fejlődésének és új alkalmazásokra adott lehetőséget. A hálózatok azonban nyilvános csatornának minősülnek, az adatcsomagokhoz illetéktelenek is hozzáférhetnek. Egyrészt a nemkívánatos hozzáférés lehetőségét minimálisra kell csökkenteni, másrészt olykor bizalmasan kezelendő adatokat kell rajtuk továbbítani. Szóval, a hálózatokon is védeni kell az adatainkat.

Az elektronikus adatfeldolgozás óriási előnye a hagyományos, papír alapúval szemben, hogy az adatokat egységes formában, digitálisan ábrázoljuk és tároljuk. Csak a megjelenítés algoritmusa dönti el, hogy szöveg, kép, hang vagy ezek kombinációja jelenik meg az ember számára. A digitális ábrázolás, az elektronikus tárolási mód és az egyre magasabb színvonalú szoftverek nagyon egyszerűvé teszik az adatok kiegészítését, másolását, módosítását, a hivatkozást a dokumentum valamely részére vagy másik dokumentumra vagy különböző típusú adatok egymással párhuzamos szerkesztését is. Például a honlapokon is jól megfigyelhetjük ezeket a lehetőségeket. A digitális ábrázolás lehetővé teszi az adatok nagy sűrűségű tárolását. Egy zsebben könnyen elférő smart kártyán vagy penndrive-on eltárolható a 22 kötetes Magyar Nagylexikon teljes anyaga.

Az adatfeldolgozás szempontjából előnyös tulajdonságok azonban az adatok védelme szempontjából hátrányosak. A fizikailag egységes tárolás egyszerűvé teszi az állományok illetéktelen kiegészítését, másolását, módosítását vagy hamisítását. A kis területen való tárolás pedig megkönnyíti az adatok ellopását. A hálózatok átviteli sebességének folyamatos növekedése is rejt magában biztonsági kockázatot; nagy mennyiségű adatot lehet gyorsan és észrevétlenül (az eredeti tárolási helyen – a művelet végrehajtása után nem érzékelhető - másolat készítésével) egy távoli felhasználóhoz eljuttatni.

Miután röviden áttekintettük, hogy miért kell az adatokat védeni, összefoglaljuk az illetéktelen beavatkozások tipikus eseteit. Az adatokat védeni kell különösen

- jogosulatlan hozzáférés (értékes, személyes),
- megváltoztatás (egyedi, pótolhatatlan),
- nyilvánosságra hozás (bizalmas)
- törlés, illetőleg sérülés vagy megsemmisülés ellen (sérülékeny).
- továbbítás során
- tároláskor (itt speciális problémát jelent a hosszú távú tárolás, azaz archiválás).

Egy természetes vagy jogi személy valamely rendszerben tárolt, nem publikus adatainak illetéktelen hozzáféréstől való védelmét valamint a fontos – pl. üzleti jellegű – információk folyamatos rendelkezésre állásának biztosítását *adatvédelemnek* nevezzük. A védelmi stratégiák és technológiák kidolgozásakor érdemes több biztonsági fokozatot kialakítani az adatok fontosságától függően. Így különböző biztonsági kategóriába sorolhatjuk a személyes, illetve pénzügyi adatokat, egy másikba a szolgálati titkokat, a nagy mennyiségű személyes adatokat, és egy újabb kategóriát képezhetnek az államtitkok valamint az emberek személyes adatait tartalmazó különféle adatbázisokhoz való hozzáférés.

Vállalati számítógépes rendszer esetén a számítógépek szinte mindig hálózatba kötve üzemelnek. A közös használatra szánt adatokat, dokumentumokat vagy akár az egyes

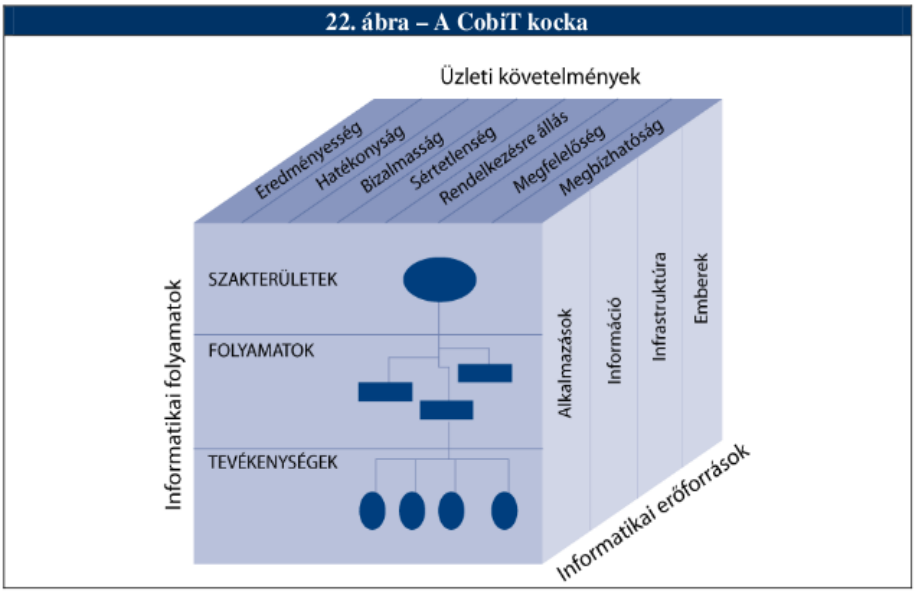
felhasználók saját adatait is egy vagy több központi számítógépen (szerveren) tárolják. Vállalati környezetben elengedhetetlen, hogy az egyes felhasználókat külön azonosítsuk és az adatokhoz való hozzáférési jogaikat személyre szólóan definiáljuk. A hordozható eszközök megjelenésével újabb lehetőségünk nyílik adataink tárolására. Ezen eszközök lehetővé teszik, hogy adatainkhoz könnyen és gyorsan hozzáférhessünk. Ilyen eszközök közé tartozik a laptop, a PDA, a smart kártya valamint a mobiltelefon is. Használatuk egyszerű, és az adatok szükség esetén gyorsan elérhetők. A kézi eszközök használatának hátrányaként kell azonban megemlítenünk, hogy elvesztésük esetén bizalmas adatainkhoz illetéktelenek hozzáférhetnek és adatainkkal visszaélhetnek. Ezen túl adatainkat végleg elveszítjük akkor is, ha kizárólagosan a kézi eszközökön tároltuk őket.

Az adatok gyűjtése, tárolása és felhasználása elválik egymástól. Mindegyik másféle védelmet igényel. Az elkülönülés kétféle lehet:

- Térbeli elkülönülés: az infokommunikációs hálózatok korában, például az ügyfélszolgálati rendszereknél vagy a banki ügyintézésnél a tárolás jellemzően központi szerveren történik, az adatgyűjtés és a felhasználás azonban akár több száz kilométer távolságban is történhet.
- Időbeli elkülönülés: bizonyos információkat csak jóval azok keletkezése után használják fel illetve kötelező azokat bizonyos ideig megőrizni. Például a személyi jövedelemadó bevallásánál használt adatokat 5 évig, az egészségi állapotra vonatkozó adatokat 50 évig kell megőrizni.

1.2 Az adatvédelem céljai és eszközei

Az előzőekből látható, hogy az adatvédelem egy nagyon bonyolult tevékenység, amelynek sokféle követelménynek kell eleget tennie és sokféle technológiát kell alkalmaznia. A bonyolult viszonyrendszer egy sematikus ábrázolása a háromoldalú CobiT® kocka.



1.1 ábra

Az adatvédelem célja az üzleti folyamatok hatékony támogatása, ezeket a követelményeket tekintjük át az alábbiakban. Az egyes pontoknál megadjuk azokat az algoritmikus technológiákat, amelyek a védelem fő komponensei.

1.2.1 Működőképesség

Működőképesség alatt a rendszernek és a rendszer elemeinek az elvárt és igényelt üzemelési állapotban való fennmaradását értjük. A működőképesség fogaloma sok esetben azonos az üzembiztonság fogalmával. Ezen állapot fenntartásának alapfeladatait a rendszeradminisztrátor (rendszer menedzser) látja el.

A rendszernek nem csak üzemképesnek kell lennie, de a működését bizonyos ésszerű hatékonysággal kell végeznie. Minden óvintézkedésnek ára van. Az ár lehet egyszeri, a biztonsági intézkedések foganatosításakor fellépő vagy az üzemeltetés során folyamatosan jelentkező költség. Érintheti magát a rendszert vagy a rendszer környezetét. Az ár nem csak pénzben jelentkezhet: lehet idő, kényelem, rugalmasság, vagy magánélet. Az óvintézkedések általában a rendszert támadó, az elvárttól eltérő, rosszindulatú szereplők ellen irányulnak, viszont az őszinte, előírásszerűen viselkedő szereplőket is érintik. Egy bizonyos fokon túli biztonság nem praktikus, mert egyszerűen túl nagy az ár. A biztonság kérdése bonyolult és nagy kihívást jelent, az egyszerű megoldások rendszerint súlyos hiányosságokat hordoznak. A hibás biztonsági intézkedések rendszerint rosszabbak, mintha egyáltalán nem lennének. Ugyanúgy időbe és pénzbe kerül a foganatosításuk, ugyanúgy (vagy még jobban) akadályozzák a rendszer működését és még mindemellett hamis biztonságérzetbe is ringatnak.

A jó biztonsági rendszer, a biztonsági intézkedések, a szoftverek és a berendezések megtervezése illetve hatásos alkalmazása bonyolult feladat. Sok különböző szempontot kell egyszerre értékelni és figyelembe venni. A feladatot megkönnyíti egy strukturált megközelítés, a folyamat lépésekre bontása. A tervezési fázis minden lépését egy kérdéssel ragadhatjuk meg (BS):

1) *Mik azok az eszközök illetve erőforrások, amiket meg akarunk védeni?*

Ez a kérdés elsőre triviálisnak tűnhet, mégis alapvető fontosságú, hogy a célt meghatározzuk és azután ne is tévesszük szem elől. Az egyik legalapvetőbb hiba ennek a kérdésnek az elhanyagolása, mégis sokan követik el. Ez a kérdés magában foglalja az egész biztonsági probléma megértését és definiálását.

2) *Milyen veszélyek fenyegetik az adott erőforrásokat?*

A biztonsági kérdések rendszerint magukban foglalják az egy vagy több potenciális támadó elleni védelmet is. Ennek a kérdésnek a megválaszolásakor kell megadnunk, hogy milyen erőforrásainkat akarjuk megvédeni, milyen következményekkel kell szembenéznünk, ha ez nem sikerül. Ennél a lépésnél kell végiggondolni azt is, hogy kik ellen akarjuk a védelmet megtervezni és hogy az egyes potenciális támadók milyen lehetőségekkel és

erőforrásokkal rendelkezhetnek illetve, hogy mennyit hajlandóak kockáztatni, mekkora áldozatot hajlandóak meghozni a támadás során.

3) *Milyen hatásfokkal kezeli ezeket a kockázatokat a választott biztonsági megoldás?*

Minden egyes alkalmazni kívánt biztonsági megoldás esetén mérlegelnünk kell, hogy az milyen hatásfokkal képes védekezni azon támadások ellen, amelyek ellen szántuk. Ez nem csak a biztonsági megoldás sikerességének a vizsgálatát jelenti, hanem annak a felmérését is, hogy hogyan hat a környezetére illetve milyen kölcsönhatásra lép azzal. Fontos annak a felmérése is, hogy a biztonsági megoldás milyen gyakorisággal és milyen következményekkel vallhat kudarcot.

4) *A választott megoldás milyen új biztonsági réseket okoz?*

Az erőforrások, amiket meg akarunk védeni rendszerint szintén összetett entitások, nagy bonyolultságú rendszerek, és mint ilyenekre egy alkalmazott módosításnak nem várt hatásai lehetnek. Gyakran a biztonsági rendszer okozta működésbeli módosítások dominószerűen hullámzanak végig az adott rendszeren. Minden ilyen közvetett hatást is figyelembe kell vennünk a biztonsági megoldás értékelésekor és mérlegelnünk kell, hogy az okozott problémák kisebbek-e mint, amit a megoldás kiküszöbölni hivatott.

5) *Megéri-e alkalmazni a megoldást?*

Minden egyes megoldásnak ára van. Mint ahogy az a korábbiakban már említésre került, ez lehet pénz, idő, az alkalmazás kényelmetlensége, az alkalmazottak személyes jogai, csökkenő teljesítmény. Mindezekkel a tervezés folyamán számolnunk kell.

Amikor biztonsági kérdésekről beszélünk, fontos megkülönböztetnünk a fenyegetést a kockázattól. A fenyegetés egy lehetséges módját jelenti a rendszer megtámadásának, amit egy támadó alkalmazhat. A kockázat ellenben azt jelenti, amikor ehhez hozzászámoljuk és mérlegeljük a siker valószínűségét, a szükséges erőforrásokat és áldozatokat, amit a támadónak meg kell hoznia a siker érdekében illetve, hogy az adott sikeres támadás milyen következményekkel jár a megvédeni kívánt erőforrásainkra nézve.

A költségesebb biztonsági megoldásokat rendszerint szervezetek, leggyakrabban üzleti társaságok alkalmazzák. Az üzleti szereplők számára egy biztonsági kockázat semmiben sem különbözik attól a számos egyéb kockázattól, amikkel egy vállalatnak szembe kell néznie az üzleti működés során. A kockázat kezelése elemi fontosságú számukra, ezért jól bevált módszereket és szakértőket alkalmaznak a probléma megoldására. Például, ha a vállalat alkalmazottai penndrive-okat, dvd-ket lopnak haza rendszeresen, akkor azt szemrebbenés nélkül hagyni fogják mindaddig, amíg az okozott kár kisebb, mint egy esetleges ellenintézkedés ára. Az alkalmazottak táskáiban meg zsebeiben dvd-k és tűzőgépek után kutató biztonsági őröknek például meglehetősen rossz hatása van mind a munkamorálra, mind a cég megítélésére.

Mivel ebből a szemszögből a biztonsági kockázatok is egyszerű pénzügyi kérdést jelentenek, egy drága ám hatékony biztonsági megoldás helyett sokkal kedvezőbb lehet egy olcsóbb megoldás egy megfelelő biztosítással kiegészítve.

A fent említett, ötlépéses módszer középpontjában a kockázatkezelés áll. Az ötödik lépésben, a végén történik a kockázatok kiértékelése és az elfogadható ellenintézkedések előnyeinek és hátrányainak mérlegelése.

A biztonsági kockázatok kezelését tovább nehezíti, hogy maga a biztonság kérdése, érzete illetve a kockázattűrő képesség is szubjektív. Emberről emberre és szervezetről szervezetre változik, hogy mekkora kockázatot tartanak még elviselhetőnek, "biztonságos" -nak.

Egy biztonsági fenyegetés kezelésére rendszerint rengeteg eszköz, megoldás áll rendelkezésre, és mindezek kiértékelésekor a szubjektív kockázattűrő képességen túl a megoldás hátulütőinek, hátrányainak a szubjektivitását is figyelembe kell venni. Az a kényelmetlenség, ami az egyik személy vagy szervezet számára semmilyen problémát nem jelent, teljességgel elfogadhatatlan lehet egy másik számára.

Mindezek a számítások még akkor is komoly nehézségeket jelentenek, ha elegendő információ áll a rendelkezésünkre. Ám a legtöbb esetben jelentős ismeretlen tényezőkkel kell számolnunk, és gyakran az ismert adataink is meglehetősen pontatlanok.

Szintén figyelembe kell vennünk, hogy a rendszer vagy erőforrás, amit meg akarunk védeni, legtöbbször nem áll önmagában, hanem folytonosan vagy épp csak időről időre interakcióba lép más környező rendszerekkel, amik többnyire más személyek, szervezetek irányítása alatt állnak. Gyakran az alkalmazott megoldások rájuk is hatással lehetnek, és az indítékaiktól és az erőviszonyoktól függően ezeknek a szereplőknek is lehet beleszólása az adott biztonsági döntés meghozásába.

1.2.2 Rendelkezésre állás (elérhetőség) → azonosítás

A rendelkezésre állás (elérhetőség) olyan tulajdonság, amely lehetővé teszi, hogy a feljogosított entitás által támasztott igény alapján az adott objektum elérhető és használható legyen. Az a tényleges állapot, amikor egy informatikai rendszer szolgáltatásai - amely szolgáltatások különbözők lehetnek - állandóan vagy egy meghatározott időben rendelkezésre állnak, és a rendszer működőképessége sem átmenetileg, sem pedig tartósan nincs akadályozva. Ebben az összefüggésben jelentősége van az információ vagy adatok rendelkezésre állásának, elérhetőségének is.

A szolgáltatás tehát

- minél hosszabb ideig (bármikor) és
- minél több helyről (bárhonnan) elérhető kell legyen és
- a végrehajtás folyamatának követhetőnek kell lennie.

A szolgáltatás általában annál értékesebb, minél többen, minél hosszabb ideig használják. A folyamatos üzem lehetővé teszi, hogy az ügyeinket számunkra kedvező időpontban intézzük. Ez lényeges változás a hagyományos ügyintézéshez képest, amikor a szolgáltató határozta meg a hozzáférési időt. A transzparencia, azaz a végrehajtási folyamat követhetősége, viszonylag új követelmény, egyelőre kevés helyen valósul meg, jelentősége azonban egyre nő.

1.2.3 Sértetlenség → digitális aláírás

A sértetlenséget általában az információkra, adatokra illetve a programokra értelmezik. Az információk sértetlensége alatt azt értjük, hogy az információkat csak az arra jogosultak változtathatják meg, egyébként véletlenül sem módosulhatnak. Ez az alap-veszélyforrás a programokat is érinti, mivel az adatok sértetlenségét csak rendeltetésszerű feldolgozás és átvitel esetén lehet biztosítani. A sértetlenség fogalma alatt gyakran értik a sértetlenségen túli teljességet továbbá az ellentmondás-mentességet és a korrektséget is, együttesen: az integritást. Az integritás ebben az összefüggésben azt jelenti, hogy az információ valamennyi része rendelkezésre áll, elérhető. Korrektek azok az információk, amelyek a valós dologi vagy (pl. modellezésnél) a feltételezett állapotot helyesen írják le.

- A dokumentumok készítése, feldolgozása és felhasználása időben és térben elválik. Minden egyes esetben rögzíteni kell, hogy ki mikor fért hozzá az adatokhoz, módosította-e azokat.
- Visszakereshetővé kell tenni, hogy mely adatok lettek módosítva, és mi volt az eredeti tartalmuk (javíthatóság, visszaállíthatóság).
- Biztosítani kell, hogy bármikor és bárhol azonosítani lehessen a készítő(ke)t és módosító(ka)t,
- A hibás vagy illegális másolatot, változatot fel kell ismerni,

Egy dokumentum eredetiségének és sértetlenségének bizonyítására használjuk a digitális aláírást, amelyről később részletesen írunk.

1.2.4 Hitelesség → digitális aláírás

A hitelesség a számítógépes adat, információ valódiságára vonatkozik. A hiteles adatot tehát tényleg az hozta létre, aki azt magáról állítja. Közvetlen információcsere esetén ezt úgy biztosíthatjuk, hogy vagy ismerjük az információ forrását vagy a forrás az azonosságát és jogosultságát valamilyen módon (például igazolvánnyal) tanúsítja. Digitális információcsere esetén a szereplők, mint arra korábban már rámutattunk, térben és időben is távol lehetnek egymástól. A hiteles információszolgáltatást azonban ilyen feltételek mellett is biztosítani kell, és ezt meg is tudjuk tenni.

Az adathalászok például eredetinek tűnő hamisítványokra irányítják a gyanútlan felhasználót, aki ott a szokásos módon azonosítja magát. Az azonosító adatok így az adathalászokhoz jutnak, akik ezek után, mint sajátjukat felhasználhatják azt (például megcsapolhatják a felhasználó bankszámláját). Digitális világunkban szaporodik az álhíreket vagy félrevezető információkat tartalmazó honlapok száma. Ezeket is a digitális aláírás alkalmazásával lehet kiszűrni.

Az 1.2 ábra egy adathalász honlapra mutat. Látható, hogy a hitelességet jelző kis lakat hiányzik a képről.

1.2 ábra

1.2.5 Bizalmasság → titkosítás

Az információk vagy adatok esetében a bizalmasság azt jelenti, hogy hozzájuk csak az arra feljogosítottak és csak az előírt módokon férhetnek hozzá. Nem fordulhat elő úgynevezett jogosulatlan információszerzés. Ez vonatkozhat programokra, mint szélesebb értelemben vett információkra is (például, ha valamely eljárás előírásait egy programmal írjuk le, és azt titokban kívánjuk tartani).

Mindennapi életünkben ritkán kell bizalmas üzenetet továbbítani. Ugyanakkor – mint azt korábban már hangsúlyoztuk - az információcsere és tárolás jellemzően nyílt csatornán és szabványos eszközökkel történik. Bizalmas információkat kell azonban néha ilyen esetekben is küldeni (jelszó, személyes-és vállalati titok, stb.), amit az üzenetek kódolásával, titkosításával lehet elérni. Amikor például banki tranzakciót hajtunk végre, akkor a banki és a saját számítógépünk a háttérben kicserél egy, csak kettejük által ismert, szimmetrikus titkosító kulcsot, és azt használva kerül sor az üzenetek bizalmas cseréjére.

Nyílt csatorna pl. az internet vagy a mobiltelefonos hálózat, ahonnan bárki, bármikor legális adatokhoz tud hozzáférni. Zárt csatorna pl. két intézmény közötti közvetlen kapcsolat. Szabványos eszközök pl. cd-k, dvd-k, memória kártyák, hidegháború idején „forró drót”.

Az informatikai biztonság legfontosabb feladatainak előbbi felsorolása egyben a jelenlegi ismereteink szerinti fontossági sorrendet is jelenti. A digitális információk védelmében meghatározó szerepet játszó algoritmikus eszközök egymásra épülnek. Az utóbbi hierarchia alapját a titkosító eljárások jelentik. Ezekből vezethetőek le az azonosítás és a digitális aláírás algoritmusai, melyek tehát elméletileg is legfeljebb olyan biztonságosak lehetnek, mint az alapul szolgáló titkosító eljárások. Az algoritmikus adatvédelem matematikai alapjaival és alapalgoritmusaival jegyzetünk második részében foglalkozunk.

1.3 A hagyományos és a modern szolgáltatás összehasonlítása

Miután megfogalmaztuk az adatvédelem okait és fő céljait, összehasonlítjuk a hagyományos és a modern adatgyűjtés és -szolgáltatás jellemzőit. Előbbit papír alapú adatkezelésnek is szokás nevezni. Legalább nagy vonalakban fontos ennek a jellemzőit is megismerni, mert nagyon régóta használják és közben sok, értékes tapasztalat gyűlt össze vele kapcsolatban, amelyeket használni lehet a digitális adatkezelés során is. A tárgyalt területen a szabályozás már viszonylag régen kialakult és szabályrendszere az évszázadok során ki is finomult. A modern módszereknek meg kell felelniük ezeknek a szabályoknak. A szerződések aláírásának jogintézménye például nagyon hatékonyan működik a papír alapú dokumentumoknál, ezért a digitális dokumentumokra is legalább ennyire hatékony és biztonságos eljárásokat kell kifejleszteni és elfogadtatni a társadalommal.

Az adatok biztonságos és strukturált tárolása és bizalmas kezelése is régi igény, melyre jól működő módszerek alakultak ki. Ezek ismerete és hatékony adaptálása is fontos feladat.

Végezetül nem felejthetjük el, hogy a hagyományos és a modern adatkezelési metódusok - egymást segítve (néha hátráltatva) és egymást kiegészítve - még hosszú ideig egymás mellett fognak élni. Évtizedek óta hallhatunk, olvashatunk a Gutenberg-Galaxis és a nyomtatott sajtó haláláról vagy, ezzel összefüggésben, a papírmentes irodáról, kórházról. A jóslatok ellenére a nyomtatott könyvek valamint a napi- és hetilapok továbbra is mindennapi életünk szerves részét képezik, bár információ tartalmuk jelentősen módosult. A papírmentes munkahely, a jelentős erőfeszítések dacára sem valósult meg, sőt a statisztikák szerint a fejlett társadalmak papírfogyasztása folyamatosan nő.

Az alábbiakban az adatvédelem szempontjából elemezzük a hagyományos és a modern adatszolgáltatás jellemzőit. Elemzésünkben nem törekszünk teljességre, mert nem ez a jegyzet fő témája.

1.3.1 Hagyományos adatszolgáltatás

A hagyományos adatszolgáltatás szembetűnő jellegzetessége, hogy *meghatározott helyen, helyeken vehető igénybe*. Az ügyfélszolgálati tér (iroda) a feladata ellátásának

megfelelően van kialakítva és felszerelve. Olyan épületben helyezik el, ahol az ügyfelek (felhasználók) széles köre könnyen elérheti.

A hagyományos adatszolgáltatás második jellegzetessége, hogy csak meghatározott időben (munkaidőben) vehető igénybe. A nyitvatartási idő hosszát és elhelyezését több tényező is befolyásolja: az ügyfelek érdeklődése, az ügyintézők munkaideje, az irodák takarítása, karbantartása, stb. Az ügyfelek igényeinek kielégítése a legfontosabb befolyásoló tényező, azaz, hogy mikor tudják és akarják felkeresni az irodát. Ha az iroda például olyan időszakban van nyitva, amikor tömegközlekedési eszközökkel nehezen érhető el, akkor kevesen fogják igénybe venni a szolgáltatásait. Az irodák nyitva tartását döntően befolyásoló másik tényező a dolgozók munkaidejének hossza és beosztása. Ezt rendszerint a dolgozóval kötött munkaszerződés vagy – nagyobb szervezetek esetén – a kollektív szerződés szabályozza. A törvényes munkaidőnél hosszabb ideig vagy szokatlan időben (például éjszaka) a dolgozó általában nem, vagy csak jelentős túlmunkadíj fejében kötelezhető munkára. Szombaton és vasárnap valamint ünnepnapokon az irodák rendszerint zárva tartanak.

Az irodák nyitva tartási idejének van még egy fontos korlátozó tényezője, mégpedig az, hogy a munkahelyeket rendszeresen takarítani kell, sőt időnként nagyobb felújítást (festés, bútorok, irodatechnika cseréje, stb.) is el kell végezni. Ezekben az időszakokban az ügyfelek fogadásának szünetelni kell.

Az ügyintézés szigorú térbeli és időbeli korlátjának lazítása természetes igény, különösen a piaci alapon működő szolgáltatásoknál, így az adatszolgáltatásnál is. A kommunikációs eszközök fejlődésével párhuzamosan lehetőség nyílott bizonyos ügyek levélben, telefonon vagy egyéb telekommunikációs eszközön keresztül történő intézésére is. Utóbbi megoldás átmenetet jelent a hagyományos és modern adatszolgáltatás között. Ezek a lehetőségek azonban csak offline üzemmódban működhetnek.

Az adatok tárolása és továbbítása alapvetően analóg adathordozókon történik. Elsősorban a papírra asszociálunk, de a hanganyagokat, fényképeket és filmeket más módon, például mágnes- vagy celluloid szalagon is őrzik. Ezek teljessége és változatlansága viszonylag könnyen ellenőrizhető. A hosszabb terjedelmű, papíron tárolt dokumentumok oldalait számozzák, bizonyos oldalszámok hiánya rögtön jelzi, hogy a dokumentum nem teljes. Gépelt vagy nyomtatott dokumentumokat csak nagyon nehezen lehet észrevétlenül megváltoztatni, hiszen utólag csak írógéppel vagy kézzel lehet szöveget beírni és a tördelés sem követhető. A fényképek retusálása a negatívon jól észrevehető, a celluloidszalagon tárolt filmekből bizonyos képek vagy jelenetek kivágása után a szalagot össze kellett ragasztani, ami árulkodik az illetéktelen beavatkozásról.

A dokumentumokra ráírják keletkezésének idejét és aláírással hitelesítik azt. A dátum azt jelenti, hogy a dokumentum annak időpontjában már létezett, az aláírás pedig tanúsítja, hogy az aláíró ismerte a dokumentumot, annak tartalmával egyetért és azt magára nézve kötelezőnek tartja. A szabályszerűen aláírt dokumentumok hitelesek, azoknak jogi hatályuk van. Bizonyos esetekben előfordul a dokumentumok előre vagy hátra datálása is, ez azonban a jogi következményeken nem változtat.

Az irodában az ügyintéző és az ügyfél személyesen találkozik. Az ügyfél tehát bizonyos lehet abban, hogy az ügyintéző annak a szolgáltatónak a nevében és felhatalmazásával jár el, amelynek az irodájában dolgozik. Az ügyfél személyesen vagy meghatalmazottja útján intézi az ügyét. Az előbbi tipikus a természetes személyek esetén, bár ők is sokszor vesznek igénybe meghatalmazottat például ügyvédet, adótanácsadót. Jogi személyekkel kapcsolatos ügyek intézése mindig a képviselőkön vagy a meghatalmazottakon keresztül történik. Az ügyfeleknek valamilyen igazolvánnyal azonosítaniuk kell magukat, azaz bizonyítaniuk kell, hogy valóban azok, akiknek állítják magukat. A képviselőknek és meghatalmazottaknak ezen felül még azt is bizonyítaniuk kell, hogy jogosultak eljárni ügyfelük érdekében, azaz rendelkeznek képviseleti joggal, illetve meghatalmazással. Az azonosítással részletesen a 6. fejezetben foglalkozunk. Itt csak azt jegyezzük meg, hogy a személyes találkozó alkalmával történő azonosításnak és jogosultság ellenőrzésnek évszázadok alatt kiforrott, leegyszerűsödött és nagyon biztonságos módszerei vannak.

A hagyományos adatszolgáltatás előnyei között elsőként azt emeljük ki, hogy nagyon régóta gyakorolják a működtetését, ezért kialakultak azok a formák és eljárások, amelyek a különböző korlátozó feltételeknek legjobban megfelelnek. A szolgáltatók és az ügyfelek is megtanulták működtetni és használni ezeket a szolgáltatásokat; folyamatosan finomították és modernizálták őket. A polgárok döntő része számára *megszokottá* vált a hagyományos módszer.

Egy szolgáltatás csak akkor lehet sikeres, ha sokan veszik igénybe. Az állami és önkormányzati intézmények eleve polgárok széles köre számára nyújtanak szolgáltatásokat. Ahhoz, hogy sok ügyfél igényét ki lehessen elégíteni a szolgáltatási eljárásnak *egyszerűnek* kell lenni. Ezt a hagyományos szolgáltatás biztosítja, vagy ha még nem, akkor az ügyfelek előbb vagy utóbb kikényszerítik azt. Az egyszerűséget az ügyintéző és az ügyfél személyes találkozása is biztosítja. Az *emberi kontaktus* lehetőséget ad arra, hogy az eljárás részleteiben felkészült ügyintéző tanácsokkal segítse az ügyfelet. Mai, atomokra hulló társadalmunkban jelentős igény van az olyan, akár felszínes emberi kontaktusokra is, amelyekre például az ügyfélszolgálati terek vagy az orvosi várószobák lehetőséget biztosítanak.

Biztonsági szempontból a hagyományos adatszolgáltatás több okból is kedvező. Az adatfeldolgozás folyamatát, legalább is bizonyos mennyiségi határig, *humán eszközökkel lehet ellenőrizni*. Amennyiben megfelelő képzettségű alkalmazottak végzik ezt a munkát, úgy nagy valószínűséggel ők kis hibaszázalékkal és megfelelően rugalmasan dolgoznak. Alulképzett vagy gyakorlatlan munkatársak lényegesen lassíthatják az eljárásokat. Az adatokat, adathordozókat olyan raktárakban helyezik el, amelyekben a fizikai védelmük könnyen megoldható. Sokszor elegendő csak a raktárak tűz-, víz- és elemi csapás elleni illetve betörésvédelméről gondoskodni. Az adatokat csak az adott tárolóhelyen lehet olvasni, csak a törlés veszélye fenyegeti azokat. Így a potenciális ellenségek köre szűk, csak azok jöhetnek szóba, akik a raktárhoz fizikailag hozzáférhetnek.

A hagyományos adatszolgáltatás *hátrányai közül első helyen a lassúságát* említjük. Az igény megjelenésétől az ügy elintézéséig az ügyfélnek (és az ügyintézőknek) nagyon sok lépést kell megtenniük, amelyek közül számos formális, az ügy szempontjából lényegtelen. A lassúság egyik oka a *sok* – fentebb részletezett – *időbeli és térbeli kötöttség*. A polgárnak az

ügyfélfogadási időben kell elmennie a megfelelő hivatalba. Ha késve érkezik vagy nem az adott ügyet intéző hivatalba megy, akkor másnap vagy másutt kell folytatnia az ügyintézést. Az emberi ügyintézésnek nemcsak előnyei, hanem hátrányai is vannak. Közülük a legfontosabb, hogy a rosszindulatú ügyintéző sokszor megkeseríti a felhasználó életét.

Az ügyintézést elsősorban emberek végzik, akiknek a munkabére valamint a munkafeltételeik megteremtése és fenntartása jelentős kiadást jelent. A szolgáltató szervezet tulajdonosainak fontos érdeke, hogy ezeket a kiadásokat minél alacsonyabb szintre csökkentsék, amit elsősorban az élő munkaerő mennyiségének csökkentésével érhetnek el. Az infokommunikációs eszközök és újabb munkaszervezési módszerek alkalmazása jelentős kiadáscsökkentéssel járhat.

A banki szolgáltatások fejlődése iránymutató ebből a szempontból. Az ügyfelek megszerzésének korszakában sűrű fiókhálózatot hoztak létre, ahol az ügyfelek személyes kontaktusba kerülhetnek a pénzintézettel és megszerezhették a készpénzkiváltó technikákat. Ezek után egyre több tranzakció végrehajtását tették lehetővé az informatikai hálózatokon keresztül. Miután a banki ügyfelek megszokták és megszeretették a számukra is sokkal kényelmesebb új technikát, a fiókhálózatot lényegesen csökkenteni, koncentrálni lehetett. A jegyzet egyik szerzője ezt a fejlődést személyesen is megélte a Saarvidéki Egyetemen. Amikor először töltött hosszabb időt az intézményben, az egyetemi campuson levő bankfiók extenzíven fejlődött, későbbi látogatásai során tapasztalhatta a bankfiók sorvadását, majd bezárását.

A hagyományos ügyintézés fő hátránya, hogy *nagy adathalmazra nem használható*. Az adatmennyiség növekedésével a feldolgozás kézi úton nem végezhető el és az adatok közötti összefüggések az emberek számára átláthatatlanná válnak. Az adatfeldolgozás automatizálási igénye a XIX század végén az USA-ban tartott népszámlálás során vetődött fel, amelyet a lyukkártyák bevezetésével Herman Hollerith német származású amerikai mérnök fejlesztett ki. Ma már nemcsak az adatfeldolgozást, hanem az adattovábbítást is automatikusan tudjuk végrehajtani, melynek a legfontosabb fizikai megvalósítója az informatikai hálózat.

1.3.2 Modern, digitális adatszolgáltatás

A hagyományos és a modern adatszolgáltatás közötti alapvető különbséget az adathordozók jellege és a munkafolyamat szervezése jelenti. A XX. század második felében ezen a területen terjedtek el először a digitális számítógépek, majd az informatikai hálózatok. Az adatfeldolgozás folyamatát algoritmizálták, melynek következtében alapvetően megváltozott a szereplők tevékenysége.

Természetesen a számítógépek megfelelő elhelyezéséről is gondoskodni kell, de ezek folyamatosan működhetnek, és hálózaton keresztül ma már lényegében *bárhonnan és bármikor elérhetőek*. Itt nemcsak az internetre kell gondolnunk, hanem a mobiltelefonos hálózatokra is, amelyeknek a jelentősége az adatszolgáltatás területén folyamatosan növekszik. Az adatokhoz való hozzáférhetőséget csak az egyes eszközök üzembiztonsága és karbantartási igénye korlátozza. A szolgáltató számítógépek duplikálásával és az adatbázisok

tükrözésével valamint a működtető környezet megfelelő kialakításával (redundáns betáp, szünetmentes áramforrás, generátor, klímatizálás, stb.) az üzembiztonság jelentősen növelhető. A tükrözés azt jelenti, hogy az egyik berendezés üzemzavara esetén egy másik automatikusan átveszi annak a funkcióit.

A *hálózati infrastruktúra üzembiztonsága* jelenti a folyamatos üzem második korlátját. Ha két hálózati végpontot csak egyetlen vezeték köt össze, akkor annak meghibásodása esetén lehetetlenné válik a kommunikáció. A hazai informatikai infrastruktúra kiépítésének időszakában, amikor a budapesti központból egy-egy gerincvezeték vezetett a vidéki nagyvárosokba, többször előfordult, hogy azt földmunkák közben elvágták és így az érintett körzetben a hiba elhárításáig megszűnt az internet szolgáltatás. Ma már rendelkezésre állnak alternatív útvonalak is, amelyek az üzembiztonságot lényegesen növelik.

Figyelemre méltó az informatikai infrastruktúra kiépülésének gyorsasága más hasonló méretű és bonyolultságú infrastruktúrákkal összehasonlítva. Az elektromos hálózatot például a XIX. század végén kezdték el kiépíteni hazánkban, de csak az 1960-as években kötöttek be minden települést az országos hálózatba. Abban az időben még kisvárosokban is gyakori volt a kényszerű áramszünet. Az országos hálózat kiépítése és stabil működésének biztosítása legalább 70 évet vett igénybe. Az első informatikai hálózatok a múlt század nyolcvanas éveinek végén kezdtek el működni hazánkban. (A Matematikai Épületben az első hálózati szegmenset 1988-ban építették ki a 3. emeleti tantermek és a Számítástudományi Tanszék irodái között. Az egyetemi optikai gerinchálózat 1992-ben készült el.) Húsz év alatt tehát az informatikai infrastruktúra közel olyan sűrűséget és üzembiztonságot ért el, mint az elektromos infrastruktúra 70 év alatt.

Manapság gyakori, hogy az adat tulajdonosa és szolgáltatója nem ugyanaz a szervezet, így szerződésben szabályozzák a jogaikat és kötelezettségeiket. A szerződésben szerepel, hogy a szolgáltató az év legalább hány százalékában biztosítja az adatok elérését. Attól függően, hogy a százalékot hány kilences írja le beszélünk 2, 3, 4, stb kilences üzembiztonságról, ami 99; 99,9; 99,99; stb. százalékot jelent. Könnyen kiszámítható, hogy egy évben 525600 perc van. A 99 %-os üzembiztonság azt jelenti, hogy a szolgáltatás egy évben összesen legfeljebb 3,65 napon keresztül szünetelhet; a 99,9 %-os üzembiztonság esetén 8,76 órán végül a 99,99 %-os azt, hogy legfeljebb 52,56 percig, tehát egy óránál rövidebb ideig szünetelhet.

A digitálisan tárolt, és továbbított adatok, mint azt az 1. fejezetben részletesen kifejtettük, könnyen módosíthatóak, így a változatlanság és hitelesség biztosítása sokkal nehezebb, mint az analóg információhordozóknál. Intenzív alapkutatások nyomán az informatika kifejlesztette azokat a technikákat, amelyek ma már ipari méretekben is megoldják ezt a problémát. A dokumentumok integritását elsősorban a digitális aláírás és a digitális vízjel alkalmazásával lehet biztosítani. Ezekről a technikákról a jegyzet 9. fejezetében írunk.

Jelentős különbség a hagyományos és a digitális információszolgáltatás között, hogy utóbbi esetben a szolgáltató képviselői és az ügyfelek nem találkoznak személyesen, hanem a felhasználó a számítógépén vagy mobiltelefonján keresztül a szolgáltató számítógépével cserél információt. Biztosítani kell tehát, hogy egyrészt az ügyfél megnyugtató módon

azonosíthassa a szolgáltató számítógépét, másrészt a szolgáltató gépe is egyértelműen azonosíthassa a felhasználót. Az azonosítás nagyon fontos probléma, hiszen utána a felhasználó már automatikusan hozzáférhet a hozzá rendelt erőforrásokhoz és adatokhoz. Ezzel a kiterjedt és bonyolult problémakörrel a 6 fejezetben részletesen foglalkozunk. A szolgáltatást általában sokan és sokféle szerepkörben veszik igénybe. Vannak olyan felhasználók is, akik többféle szerepkörben is hozzáférhetnek a szolgáltatáshoz. Egy ember például lehet alkalmazott és ügyfél is. A jogosultságok pontos definiálása és karbantartása is fontos követelmény, amellyel részletesen szintén a 6. fejezetben foglalkozunk.

A digitális adatszolgáltatás fő *előnye*, hogy a felhasználó a számára legkedvezőbb időben és helyen veheti igénybe a szolgáltatást. Nem kell a szolgáltató munkaidejéhez igazodnia, hanem a számára legkedvezőbb időben végezheti el a tranzakciókat. Az ügyintézés lényegesen gyorsabb, mert nem kell elmenni valamelyik irodába és sorba állni, hanem akár otthonról is intézkedhet. Hazánkban 2010-ben 800 olyan település volt, amelyekben nincs sem ATM sem bankfiók. A tanyák száma, amelyek távol fekszenek bankfiókoktól és egyéb helyhez kötött szolgáltatásoktól ennek sokszorosa. Az ilyen helyeken élők számára az internetes adatszolgáltatás óriási könnyebbséget jelent. Ugyanígy könnyebbséget jelent a betegeknek, mozgássérülteknek is.

Az ügyek elintézése néha bonyolult, hosszan tartó folyamat. A hagyományos feldolgozás folyamatáról, arról, hogy hol áll ügyünk intézése, csak az iroda rendszeres felkeresésével tájékozódhatunk, ha erre egyáltalán lehetőséget adnak. A digitális ügyintézés transzparenssé tehető, az ügyfél bármikor lekérdezheti, hogy milyen stádiumban van az ügyének intézése.

A folyamatok nyomon követésére jó példát jelent egyre több tudományos folyóirat gyakorlata. A szerző(k) elektronikusan nyújtják be dolgozatukat, amikor is hozzáférési jogosultságokat kapnak a művükkel kapcsolatos események nyomon követésére. Ellenőrizhetik, hogy dolgozatukat mikor adták ki lektoroknak, majd a beérkezett véleményeket (bejelentkezés után) elolvashatják és szükség esetén letölthetik. A módosítások végrehajtása után ismét feltöltik a dolgozatot, ezeket a lépéseket addig ismétlik, míg a végleges döntés meg nem születik. Negatív döntés esetén a dolgozat életciklusa a folyóiratnál befejeződik, pozitív döntéskor azonban tovább folytatódik a technikai szerkesztéssel és megjelenéssel. Kívánatos lenne hasonló transzparencia az államigazgatás és az önkormányzati ügyintézés keretében is.

A szolgáltatók szempontjából a digitális adatszolgáltatás döntő előnye az olcsósága. Sokkal kevesebb élő munkaerőt, elsősorban ügyintézőt kell alkalmazni, mint a hagyományos adatszolgáltatóknak. A folyamatokat részben a számítógépek, részben a felhasználók hajtják végre. Az ellenőrzés ugyanakkor lényegesen fontosabb és bonyolultabb.

A kulcserőforrások – szerverek, adattárolók és hálózatok - fizikai és technikai védelme is lényegesen komplexebb feladat, mint a hagyományos adatfeldolgozásnál. Nem elegendő ezeket jól védett helységben elhelyezni, mert - a szolgáltatás lényegének következtében - a hálózatra kötött eszközök bárki számára elérhetőek. A nyilvános hálózaton, sajnos, nagyon sok rosszindulatú felhasználó is szörfözik. Ezek minden eszközt felhasználnak a számukra értékes információk megszerzésére. Később részletesen foglalkozunk a védelmi technikákkal,

itt most csak annyit említünk, hogy a kulcserőforrások védelme komoly technikai apparátust és speciális felkészültségű szakembereket igényel. Mindez jelentős kiadással jár a szolgáltató szervezetnek.

Humán szempontból a digitális adatszolgáltatás hátránya, hogy személytelen, hiányzik belőle a felhasználó és az ügyintéző személyes kapcsolata. Megváltozik a felhasználók számára az ügyintézés folyamata is. Főleg az idősebb emberek nehezen kezelik a klaviatúrát és az egeret, ami pedig a modern ügyintézéshez nélkülözhetetlen. A gyakorlatlan felhasználók számára nagy problémát jelent a biztonságos jelszavak megválasztása és nyilvántartása (megjegyzése) is. Ezekkel a kérdésekkel is fogunk részletesen foglalkozni.

2 Az adatok osztályozása

Az előző fejezetben kifejtettük, hogy az adatoknak materiális vagy emocionális értéke van, ezért szükséges védeni azokat. Nagy különbség van azonban az egyes adatok értéke között, amit figyelembe kell vennünk, amikor a védelmüket megtervezzük. Könnyen érthető, alapvető szabály, hogy *100 Ft értéket nem érdemes 101 Ft költséggel védeni*. Költőien írja le ezt a szabályt Arany János, A bajusz című versének alábbi részlete: (1854)

Nincsen otthon,
Csak az asszony,
Hogy megfőzzön,
Vagy dagasszon;
Vagy ha néhol egy beteg
Szalmaágyon fentereg;
Vagy a seprű, házőrzőnek
Felállítva küszöbre;
De ha Isten meg nem őrzi,
Ott lehet az örökre.

Témánk, az adatvédelem, szempontjából elsősorban az utolsó négy sor mérvadó. A verset Arany János 1854-ben írta. Akkor a falusi házat nem zárták be a lakói, ha elmentek otthonról, hanem a seprűt tették ki házőrzőnek. A házban nem volt olyan érték, amely komolyabb védelmet igényelt. A seprűnek a házőrzés mellett fontosabb funkciója is volt: jelezte az esetleges látogatóknak, hogy a háziak nincsenek otthon.

A költői kitérő előtt megfogalmazott általános érvényű szabály természetesen az adatokra is igaz. Ismerni kell az értéküket, ha védeni akarjuk őket. Az adatok értékének mérésére nincsenek egzakt módszerek. Nem tudjuk megmondani, hogy a számítógépünkön tárolt információk hány forintot érnek. Ettől rosszabb, hogy az állam, az önkormányzatok és a vállaltok által fenntartott adatbázisok értékét sem tudjuk mérni. Az informatikai eszközök: szerverek, tárolók, aktív- és passzív hálózati elemek, de még az alkalmazói szoftverek is a piacról szerezhetőek be, kialakult áruk van. A szervezetek leltárában a beszerzési áron tartják nyilván és értéküket törvényben vagy szabályzatban meghatározott módon évente csökkentik. A szervezet által összegyűjtött és rendezett adatmennyiség ezzel szemben folyamatosan nő. Az adatbázisok értékét azonban senki, sehol nem tartja nyilván és így nem is foglalkozik értékük változásával. Igaz ez még a szervezetek szempontjából létfontosságú adatbázisokra is.

Egyetemünk tanulmányi rendszere néhány év óta központi szerepet játszik a hallgatók tanulmányai nyilvántartásában és szervezésében. Személyi adataikon kívül az ösztöndíjukat, a vizsgaeredményeiket, fizetési kötelezettségeiket és még sok egyéb információt tartalmaz ez a hatalmas adatbázis. A legtöbb adat ma még párhuzamosan, papír adathordozón is elérhető. Az adatbázis több példányban és archiválva is létezik, így kicsi az esélye annak, hogy egyszerre minden ma elérhető példány megsemmisül. Már az is igen nagy munkát adna a tanulmányi

ügyintézőknek, ha az aktív példányok semmisülnének meg és az archivált adatbázisban néhány hét elveszett adatát pótolni kellene. Különösen igaz ez a félév eleji és végi időszakra. A feldolgozó kapacitás növelése és modernizálása érdekében az üzemeltető pontos számokkal érvelhet; bemutathatja a felhasználói érdeklődés és az adatbázis növekedésének ütemét és prognosztizálni tudja a rendelkezésre álló kapacitás teljesítőképességének határát. A számok ismeretében, amelyek „kemény" érvek, a döntéshozó megítéli, hogy a javasolt fejlesztés milyen mértékben hajtható végre. Az adatbázis biztonságának növelése érdekében azonban csak „puha" érveket: tapasztalati tényeket, analógiákat, nemzetközi és hazai tapasztalatokat sorakoztathat fel az üzemeltető. Ezért nagymértékben a vezető szubjektív döntésén múlik az informatikai biztonsági eszközök beszerzése és fejlesztése.

Egzakt mérőszámokat nem tudunk tehát az adatokhoz rendelni, osztályozásukra azonban legalább szubjektív szempontok szerint szükség van. A következő két fejezetben ilyen osztályozást ismertetünk J.M.D. Hunter [15] könyve nyomán.

A fent megfogalmazott ökölszabály nemcsak az adat tulajdonosára vonatkozik, hanem az azt eltulajdonítani akaró támadóra is. Természetesen neki sem éri meg 100 forintot 101 forint befektetéssel megszerezni, bár ismerünk olyan példát is, amikor a támadót nem a közvetlen anyagi siker, hanem valamilyen presztízsszempont motiválja. A tulajdonos szempontjából az adatok *érzékenysége*, a támadó szempontjából pedig az adatok *fontossága* szerint osztályozzuk azokat. Egy szervezetben célszerű mindkét szempont szerint áttekinteni az adatokat. Lehetséges ugyanis, hogy a tulajdonos szempontjából értéktelennek ítélt adat egy hekker számára értékes lehet. Egy erkölcsileg amortizálódott szerver például értéktelen a tulajdonos számára. Az új szerver mellett a régi már csak ócskavas. Mégsem lehet egyszerűen eladni valakinek vagy átadni karitatív célból. Előtte alaposan le kell takarítani a merevlemezeit. Ha ugyanis ez nem történik meg és az adatok egy hekkerhez jutnak, akkor néhány hetes vagy hónapos adatok is fontos támpontokat jelenthetnek az új szerver elleni támadáshoz.

2.1 Az információ osztályozása érzékenysége szempontjából

Ez a szempont azt jelenti, hogy tulajdonosa számára mennyire fontos egy információ, milyen erkölcsi vagy anyagi következményeket prognosztizál arra az esetre, ha az adatok illetéktelen kezekbe kerülnek. Lényeges különbség van abból a szempontból is, hogy a tulajdonos természetes vagy jogi személy. Első esetben a kár elsősorban az érintett személyiségi jogainak megsértését jelenti, de érheti közvetlen anyagi kár is. Jogi személyeknél fordított a helyzet. Információk kiszivárgása, kilopása közvetlen anyagi kárral jár, de a szervezet jó hírnevén is komoly csorba keletkezhet, ha például korrupcióra vagy közelgő csődhelyzetre utaló információk kerülnek ki. Ennek megfelelően először a vállalati, majd a magánszféra adatainak osztályozásával foglalkozunk.

a.) Vállalatok szempontjából egy adatot *nyilvánosnak* tekintünk, ha az mindenki számára megismerhető. Vannak olyan nyilvános adatok, amelyek valamely törvény erejénél fogva azok, mások azért nyilvánosak, mert a vállalat ezt érdekének tartja. Az első esetre példa

a vállalat tulajdonosi szerkezet, éves mérlegbeszámolója; kereskedelmi vállalkozásoknál a végfelhasználói szerződések feltételei. Vállalatok közötti szerződések szövege általában titkos információnak számít, de az Adatvédelmi törvény – ld. 4.1 fejezet – erejénél fogva nyilvános adatnak számítanak a közbeszerzési eljárás során keletkezett szerződések.

A nyilvános adatok másik csoportját képezik azok, amelyeket a vállalat saját elhatározásából publikál. Ilyenre jelentenek klasszikus példát a cégtáblák. A reklámok és a vállalati honlapok a cégtáblák modern változatainak tekinthetők. A vállalat ezeket az adatokat legtöbbször igyekszik is eljuttatni minél több emberhez. Érdekelt abban, hogy minél nagyobb kör ismerje meg az adatokat, mert így juthat újabb partnerekhez. A nyilvános adatok felhasználását engedélyhez is köthetik, vannak olyanok, amelyekre szigorú szabályok, előírások vonatkoznak. A jegyzet 6.2 fejezetébe szerettem volna például egy kártya alakú személyi igazolvány mintát illusztrációként betenni. Találtam is ilyet egy honlapon, de a minta felhasználását előzetes engedélyhez kötötték. Ilyet nem akartam kérni. Német mintát használhattam volna. Nem szükséges különösebb védelem, de arra vigyázni kell, hogy az adatokat ne módosítsák és hamisítsák.

Személyesnek nevezünk egy információt, ha az nem tartozik a nyilvánosságra, de ha kitudódik, akkor nem okoz nagy problémát a vállalat működésében vagy gazdálkodásában. Ilyen adatnak tekinthető például egy vállalat belső felépítése, dolgozóinak a létszáma, belső telefonszáma és e-mail címe, a középvezetők neve és minden olyan információ, amelyet a vállalat vezetése érdemesnek tart arra, hogy a munkatársak tudomására hozzon. Ezeket az információkat meghatározott körben, ma tipikusan intraneten keresztül, terjesztenek. Az intranet használatát jogosultsághoz kötik, amely lehet azon terminálok IP címével korlátozni, ahonnan az intranet elérhető. Ezen kívül szükséges lehet valamilyen – tipikusan jelszavas – azonosítás is.

Bizalmasnak tekintjük azokat az információkat, melynek kitudódása a vállalat működése vagy gazdálkodása szempontjából komoly problémát okozhat, vagy pedig a versenytársaknak gazdasági előnyt jelenthet. Ilyen lehet például, egy tenderfelhívásra készített árajánlat, vagy a bérek nagysága. Az archivált információk is ebbe a csoportba tartoznak, például a beszállítókkal kötött, de már lejárt szerződések. Az informatikai infrastruktúra meghatározó elemeinek – szerverek, tárolók, hálózati elemek - elhelyezése is bizalmas adat. A bizalmas információkat szervezési és technikai eszközökkel is hatékonyan kell védeni, amelyek meghatározzák a hozzáférési jogosultságokat és eljárásokat.

*Titkos*ak az olyan információk, amelyek jelentős értéket képviselnek, nyilvánosságra kerülésük komoly (pénzügyi) veszteséget, bizalomvesztést okozhat. Ha ilyen adat illetéktelenekhez jut, akkor a vállalat versenyhelyzetét jelentősen rontja. Ebbe a csoportba tartoznak a vállalat vezetésének stratégiai döntései, a termékek készítésének technológiai leírásai, például a coca-cola receptje. A szerződések, kivéve a közbeszerzési eljárás győzteseként kötött szerződéseket, is titkos adatnak számítanak.

Ma már titkos adatnak tekintjük a felhasználók bejelentkezési és jogosultsági adatait tartalmazó jelszó állományokat is (ld. 6.3 fejezet). A jelszavak maguk nem tartoznak ebbe a kategóriába, mert azokat jól konfigurált rendszerek esetén csak a tulajdonosuk ismerheti, így csak ő gondoskodhat védelmükről.

A titkos adatok nagyon fontosak a vállalatnak, így védelmükre kiemelt figyelmet kell fordítani. Azok körét, akik titkos adatokhoz hozzáférhetnek személyre és nem beosztásra szólóan kell a tulajdonosnak megállapítania. A hozzáférési eljárást szigorúan kell szabályozni és a hozzáférés tényét naplózni kell.

b.) *Természetes személyek* adatainak védelméről az Adatvédelmi törvény (ld. 4.1 fejezet) rendelkezik. Itt nem a törvény előírásait ismételjük meg, hanem a bevezetőben megfogalmazott elveknek megfelelően szubjektív megállapításokat teszünk a személyes adatok értékéről. Nyilvános adatok a természetes azonosítók, mint nem, haj- vagy szemszín, magasság, stb.. A név is nyilvános adat, de a születési adatok, a telefonszám, és az e-mail cím csak akkor, ha az érintett hozzájárul. Általában minden olyan személyes adat nyilvános, amelyet a tulajdonosa nyilvánosságra hoz, például a honlapján szerepel. A közösségi oldalak elterjedésével egyre nagyobb problémát jelent a meggondolatlanul nyilvánosságra hozott adatok tömege. Az internetes kereső eszközökkel lehetőség van ugyanis arra, hogy részletes személyi profilt állítsanak össze a különböző oldalakon nyilvánosságra hozott információkból.

Vannak olyan személyes adatok is, amelyek valamely törvény erejénél fogva nyilvánosak. Ilyenek például a vállalati (rész)tulajdon, egyesületi tagság, stb..

A személyes adatok közül *bizalmasak* azok, amelyeket nem hoznak nyilvánosságra, de meghatározott célból közölnünk kell adatkezelőkkel. A jövedelmet például közölni, sőt igazolni kell, ha pénzintézettől kölcsönt veszünk fel. Ha biztosítást kötünk, akkor fel kell sorolni az érintett értéktárgyakat. Adóbevalláson, vállalkozási szerződésen szerepeltetni kell az adószámot. Ha pénzt akarunk bankszámlánkra utaltatni vagy arról akarunk fizetni, akkor meg kell adni a számlaszámot stb.. Az adatkezelőnek az ilyen adatokat bizalmasan kell kezelnie és csak a megállapodásban szereplő célra használhatja fel.

A jövedelem és a tulajdon általában személyes adat, de bizonyos közfeladatot ellátó személyeknek – például parlamenti és önkormányzati képviselők, a Kormány tagjai, stb. – évente nyilvánosságra kell hoznia az anyagi helyzetükre vonatkozó adatokat.

Titkos minden olyan személyes adat, amelyet a tulajdonosa nem hoz nyilvánosságra. Az egészségi állapotra, a párttagságra, az etnikai hovatartozásra és a vallásra vonatkozó adatok a törvény szerint különleges adatok, kezelésük különös gondosságot igényel. Az informatikai biztonság szempontjából nagyon fontos titkos adatok az elektronikus, személyes azonosítók, mint például a PIN kód, jelszó, stb.. Ezeket nem szabad más személlyel közölni, másnak átadni, mert aki ezekkel rendelkezik, helyettünk tud eljárni fontos ügyekben. Ez akkor okoz problémát, ha akaratunk ellenére cselekszik a minket megszemélyesítő személy.

2.2 Az információ osztályozása fontosság szempontjából

Az előző fejezetben az információ értékét a tulajdonosa szempontjából osztályoztuk. A tulajdonos attól függően hajlandó költeni az információ védelmére, hogy mennyire értékeli annak értékét. Ez azonban egyoldalú szemlélet, ami téves következtetésekre és döntésekre vezethet. Az információkat ugyanis nemcsak a tulajdonos, hanem a potenciális hekkerek

szempontjából is érdemes értékelni. Számba kell venni a támadó rendelkezésére álló eszközöket és azok költségét. A hekker ugyanis nem költ többet az információ megszerzésére, mint amennyi nyereséget annak megszerzésétől várhat. Itt persze még nagyobb a bizonytalanság, mint a saját érték becslésében. Csak azokat a hekker-eszközöket ismerjük ugyanis, amelyeket már legalább egyszer használtak. A hekkerek folyamatosan találnak és próbálnak ki olyan új módszereket, amelyekre nem lehet előre felkészülni.

A támadó szempontjából *lényegtelen* egy információ, ha nem vezet közelebb a cél eléréséhez. Ha például a célja az, hogy egy vállalat informatikai rendszerébe betörjön, akkor a többi vállalatra vonatkozó adatok lényegtelenek. Adott pillanatban lényegtelennek ítélt adat azonban felértékelődhet, ha megváltozik a támadás célpontja vagy technológiája. A hekker tehát eldönti, hogy milyen adatokra koncentrál és azokat - függetlenül azok aktuális értékétől – folyamatosan gyűjti és rendszerezi. Egy e-mail cím általában lényegtelen információ. Ha azonban egy hekker sok e-mail címet gyűjt össze és a tulajdonosaik számítógépeiből egy zombi hálózatot (ld. 3.3.3 fejezet) épít fel, akkor nagyon hatásos támadásokat tud indítani.

Fontos egy információ a támadó számára, ha az ugyan közvetlenül nem használható fel, de lényegesen közelebb visz a cél eléréséhez. Ha például a hekker megszerzi egy felhasználó jelszavát, akkor csak korlátozott erőforrásokhoz fér hozzá, de elemezve a környezetet és a megszemélyesített felhasználó lehetőségeit, a támadás eszkalálásához hasznos információkhoz juthat. Fontos információ lehet a felsővezetők vagy a rendszergazdák személyi profiljának ismerete, ebből következtetni lehet a felhasználói nevekre vagy jelszavakra. Sikeres hekkertámadások általában nem direkt módon, hanem fontos információk helyes kiértékelésén és felhasználásán vezetnek sikerre.

A *lényeges* információk közvetlenül vezetnek eredményre. A hekkernek nagyon sok adatot fel kell dolgoznia, amíg lényeges információt talál. Kevin Mitnick [21] leír egy esetet, amikor talált egy számítógépet, amit használaton kívül helyeztek, de még mindig benne volt a megtámadni kívánt vállalat hálózatában. A gépről kiindulva fel tudta térképezni a vállalat hálózatát a rajta levő erőforrásokkal. A számítógépen talált adatokból következtetni lehetett a rendszergazda felhasználói nevére és jelszavára. Az összegyűjtött információk végül sikeres támadást eredményeztek.

3 Kockázati tényezők és védelmi intézkedések

Csak akkor tudjuk hatékonyan és gazdaságosan védeni adatainkat, ha számba vesszük a lehetséges veszélyforrásokat. Ezek egy része általános jellegű, mindenféle érték védelme során figyelni kell rájuk. Mások az informatikai rendszerek jellegzetességéből adódnak, azok specifikus veszélyforrásai. Konkrét esetben természetesen nem minden, az alábbiakban részletezett, kockázati tényező jelentkezik és a súlyuk is különböző lehet. Egy banki ügyintéző munkaidőn kívül, az otthonából nem léphet be ebben a szerepkörben munkahelyének rendszerébe. Az egyetemi oktatók ezzel szemben bármikor elérik munkahelyük erőforrásait. Folyamatos üzemben működő adatszolgáltatónak fel kell készülnie az áramkimaradásra és védekezni kell ellene, egy péküzem raktári informatikai rendszerében, ezzel szemben, nem keletkezik áramkimaradás miatt óriási kár.

Informatikai rendszerek üzemeltetőinek tehát fel kell mérni a releváns kockázati tényezőket, elemezni kell azok hatását és meg kell tervezni az ellenük való védelmet. A szervezetben bekövetkező változások és a gyors technikai fejlődés következtében újabb veszélyforrások jelenhetnek meg, a korábbiak súlya csökkenhet, sőt elhanyagolhatóvá is válhat. Ezért rendszeresen meg kell ismételni a kockázatelemzést és szükség esetén megfelelő intézkedésekkel reagálni kell az új helyzetre.

3.1 Fizikai veszélyforrások.

A terminálok, a szerverek és a hálózatot működtető aktív elemek nagy értékű berendezések, amelyek elektromos árammal működnek, így azokra alkalmazni kell az általános tűz- és vagyonvédelmi szabályokat. Ezeket nem tárgyaljuk részletesen, de néhány kiemelten fontos szempontot felsorolunk. A számítógép, mint minden elektromos árammal működő berendezés, tűzveszélyes, ezért a közvetlen közelébe ne tartsunk könnye gyúló anyagokat. A számítástechnika hőskorában a számítógépek áramfelvétele és terjedelme lényegesen nagyobb, míg teljesítménye csak töredéke volt a mai gépekének. A PC-k teljesítményének növekedésével azonban a hőtermelésük is megnőtt. Ma már külön ventillátorral kell gondoskodni a processzorok hűtéséről. A nagy teljesítményű szerverek pedig olyan jelentős hőt termelnek, hogy klímatizált helységben kell elhelyezni őket.

Váratlan áramkimaradás jelentős veszteséget okozhat az el nem mentett adatállományok elvesztésével, hosszabb ideig tartó munkafolyamat félbeszakításával és az érzékeny szoftverrendszerek hirtelen leállásával. Olyan munkahelyeken tehát, ahol a váratlan, akár csak néhány másodpercig tartó, áramkimaradás károkat okozhat, szünetmentes áramforrást (UPS, Uninterrupted Power Supply) kell beszerelni. A szünetmentes áramforrásokban akkumulátorok vannak. Áramkimaradás esetén ezekből kapnak áramot a számítástechnikai berendezések. Ezen kívül az eszköz figyelmezteti a rákötött eszközöket,

hogy készüljenek fel a leállásra. Sokkal drágább beruházást igényel, de ennek megfelelően sokkal nagyobb biztonságot nyújt saját áramfejlesztő generátor telepítése.

Az informatikai rendszerek legértékesebb elemeit, amelyeken az adatbázisok találhatóak és az alkalmazások futnak, tehát a szervereket és a tárolókat lehetőleg olyan helyiségbe kell elhelyezni, amelyik egy nagyobb épület központjában található, így nincs a külvilággal közvetlenül érintkező fala. Szigorúan szabályozni és naplózni kell az ilyen helységekbe való belépést is. Ezekkel az intézkedésekkel a fizikai betörés lehetőségét nehezítjük meg. Elektromos berendezések egyik fő ellensége a nedves környezet. A központi erőforrásokat ne telepítsék olyan helyiségbe, amely fölött vagy mellett vizes helyiségek, például mosdók, konyha, található. Lehetőleg vízvezeték se vezessen keresztül ilyen helységen.

Számítástechnikai berendezések érzékenyek a mágneses térre és az elektromágneses sugárzásra. A hatás kétféle irányú; egyrészt az erős mágneses tér vagy sugárzás károsíthatja a tárolt adatokat, másrészt a berendezések maguk is sugárforrások és a sugárzásukból következtetni lehet a berendezés által végzett munkára. Először a külső sugárzás hatásaival foglalkozunk. Korábbi berendezések mágnesszalagokon és mágneslemezeken tárolták az adatokat és ezeken még ma is megtalálhatóak az adatok. Ezen tárolókon nagyon sok elemei mágnes található, amelyeknek polaritása a biteket reprezentálja. Erős mágneses térbe helyezve az elemi mágnesek polaritása megváltozik, a bitek törlődnek. Sok mágneslemezről tűntek el adatok villamoson való utazás közben.

A ma használatos adathordozók: CD, DVD lemezek, flash memóriák már nem mágneses elven tárolják az adatokat, így ez a veszély nem fenyegeti őket. A CD és DVD lemezekre lézerrel írják fel az információt reprezentáló bitsorozatokat. Ha a lézer egy nagyon kis lyukat éget a lemez felületébe, akkor az jelöli az 1-et, ahová nem váj lyukat, ott 0 van. A biteket igen sűrűn helyezik el a lemezen, így lehetséges sok adat tárolása. Ez a tárolási mód igen biztonságos, nagyon kicsi a valószínűsége, hogy egy bit megváltozzon. A nagy sűrűség miatt azonban a kozmikus sugárzás következményében is néhány bit óhatatlanul megváltozik. Ennek kivédésére hibajavító kódot alkalmaznak ezeken az adathordozókon és az olvasók online dekódolják az adatokat.

A számítástechnikai berendezések közül különösen a processzorok és a régebbi, katódsugárcsöves, monitorok bocsátanak ki elektromágneses sugárzást. Ezek erőssége kicsi, az egészségre ártalmatlan, de megfelelő berendezéssel néhány méter távolságból is felfogható. Az ilyen illetéktelen megfigyelés különösen veszélyes, hiszen arról az adat tulajdonosa egyáltalán nem szerez tudomást, így csak megelőző intézkedésekkel védekezhet ellene. A monitor lényegében a rajta levő képet sugározza, amelyből a támadó megismerheti a képernyőn megjelenő adatokat és az aktuális munkafolyamatot. Fontos dokumentumokról, esetleg belépési kódokról is információt szerezhet. Hasonlóan ahhoz, amikor a terminált úgy helyezik el egy irodában, hogy azt az ügyfelek is nézhessék.

A processzorok által kibocsátott sugárzás az éppen folyó számításokról nyújt információt. Ezt a sugárzást már egyszerű eszközökkel is hanghullámokká lehet alakítani, ami hallhatóvá teszi a processzorok „zenéjét”. A processzorok, mint tudjuk, csak néhány egyszerű műveletet tudnak végrehajtani, de ezek időigénye különböző. A zene finomabb elemzéséből

következtetni lehet arra, hogy a processzor éppen milyen műveletsort hajt végre. A megfigyelés eredménye általában nem túl érdekes, de nem megfelelő körültekintéssel implementált titkosító algoritmusok megfigyelésével akár a titkos kulcsot is ki lehet nyerni. A szakirodalomban *másodlagos csatorna támadásnak* (side chanel attack) nevezett támadási lehetőség egyik legnevezetesebb fajtáját, a timing attackot az RSA algoritmusról szóló részben írunk részletesen.

Különösen érzékeny adatok kezelése esetén (például banki rendszerek, hitelesítő szervezetek, stb.) biztosítani kell az elektromágneses sugárzást leárnyékoló berendezéseket a szerverek és a szenzitív adatokat megjelenítő terminálok számára is. Manapság a legjobb azonosító és digitális aláírást végző eszköznek a smart kártyákat tartják. Egyszerűek, könnyen kezelhetőek és olcsóak. Ugyanakkor ezek processzora is bocsát ki elektromágneses sugárzást, amelynek megfigyelésével akár a tulajdonos privát kulcsa is az ellenséghez kerülhet. Ezek és az azonosításban ugyancsak nagy jövő előtt álló mobiltelefonok árnyékolására még nem ismeretes megnyugtató eljárás.

3.2 Emberi veszélyforrások

Az informatikai rendszerek azon a pontjai leginkább veszélyeztetettek, amelyeknél ember-gép interakció történik. Nagy különbség van aközött, hogy az interakcióban részt vevő személy milyen gyakorlattal rendelkezik és mennyire tudatosan és figyelmesen végzi a feladatát. Attól függően, hogy a felhasználó mekkora hatáskörrel rendelkezik a rendszer működése során és milyen akciókat hajthat végre beszélhetünk ügyintézőről, üzemeltetőről, mérnökről és programozóról. Ez persze egy nagyon durva csoportosítás, de jelen célunknak megfelel és szükség esetén tovább lehet finomítani.

Az *ügyintéző* szűk jogkörrel rendelkezik, elsősorban adatbevitelt végez és csak a hozzárendelt, korlátozott erőforrásokat használhatja. Egy jól implementált rendszernél még rosszakarattal is legfeljebb lokálisan okozhat kárt. Ennek a kategóriának a tagjai nagy szórást mutatnak abból a szempontból, hogy milyen gyakorlattal rendelkeznek a számítástechnika felhasználásában. A gyakorlatlan felhasználók körében a legnagyobb problémát az jelenti, hogy nem érzik át az azonosítás fontosságát. Gyakran egyszerűen kitalálható jelszót választanak, nem változtatják meg rendszeresen a jelszavaikat vagy a jelszót, PIN kódot, stb. a monitorra írják fel, ahol illetéktelenek is láthatják azt. Smart kártyák használatakor tipikus probléma, hogy a PIN kódot a kártyával egy helyen tartják. A jelszómenedzsmenttel később részletesen fogunk foglalkozni.

Annak ellenére, hogy az ügyintézők korlátozott jogkörrel rendelkeznek, jogosultságaik megszerzése és felhasználása ugródeszkát jelenthet egy támadás számára. A támadók minden eszközt, például megtévesztést és zsarolást is, felhasználnak információk gyűjtésére. Keresik a leggyengébb láncszeme(ke)t, amit sokszor az ügyintézők között találnak meg. Számos, tanulságos példát olvashat az érdeklődő Mitnick és Simon [21] könyvében.

Az *üzemeltető* széleskörű ismerettel és jogosítvánnyal rendelkezik a rendszer felhasználásával kapcsolatban. Rendszerint munkaviszonyban áll az adattulajdonos szervezettel, de tevékenységét egyre gyakrabban kiszervezik. A szervezettel kapcsolatban

fontos információkkal rendelkezik, így titoktartási kötelezettsége van. Rendkívül fontos tehát, hogy ezen emberek ne csak szakmailag legyenek felkészültek, hanem erkölcsi szempontból is feddhetetlenek legyenek. A rendszer üzemszerű működéséért, paraméterezéséért és a hibaelhárítás megszervezéséért felelős.

Feladatai közé tartozik a felhasználók nyilvántartása és jogosultságaik beállítása. Nyomon kell követnie a felhasználók életciklusát a munkahelyre való belépéstől kezdve kilépésükig. Munkába álláskor a felhasználó megkapja a rendszer felhasználásához szükséges információkat és kódokat. Ezen kívül az üzemeltető beállítja a hozzáférési jogosultságaikat is. Sok szervezetnél az üzemeltető dönti el azt is, hogy a felhasználók milyen erőforrásokhoz juthassanak hozzá, milyen jogosítványokkal rendelkezzenek. Ezt a gyakorlatot, különösen nagyobb szervezeteknél nem tartjuk helyesnek, mert az erőforrások és jogosítványok kiosztása felelős döntés, amelyet nem célszerű az üzemeltetőre hárítani. Sokkal jobb, ha ezeket a jogosultságokat a munkakör és beosztás függvényében szabályzatokban rögzítik és az üzemeltető csak a szabályzat végrehajtásával foglalkozik. Ez nemcsak a rendszer biztonságát védi, hanem az üzemeltető felelősségét is csökkenti.

A jogosultság nem statikus, hanem vannak periódikusan változó, illetve egyszeri elemei is. Napi periodicitással történik például a munkába állás, amikor a dolgozó hozzáférhet a munkahelyén hozzá rendelt erőforrásokhoz, majd a munkaidő lejárta után a napi kijelentkezés, ami után a hozzáférését letiltják. Ilyen események kezelésére a biztonsági rendszert fel kell készíteni, nem szabad azt az üzemeltetőre bízni. Bizonyos munkahelyeken és munkakörökben a távoli és munkaidőn kívüli hozzáférés is megengedett. Előfordul, hogy a felhasználó elfelejti a jelszavát, a jelszava kompromittálódik, esetleg elveszíti vagy megrongálódik az azonosításra szolgáló eszköze. Ilyenkor az üzemeltetőnek, az előírásoknak megfelelően naplózva, cserélni kell az azonosítót. Megváltoznak a felhasználó jogosultságai akkor, ha új munkakörbe kerül. Ezek ritkán előforduló események, az üzemeltető felel a változások átvezetéséért. A felhasználó rendszerrel kapcsolatos életciklusa befejeződik, amikor kilép a szervezetből. Ekkor a hozzáférési jogosultságait törölni kell.

A *mérnök* magas szintű informatikai képzettséggel rendelkező személy, aki nagy informatikai rendszereket implementál. Általában nem áll azzal a szervezettel közvetlen alkalmazásban, ahol a tevékenységét végzi. Munkájának eredményes elvégzéséhez szükséges, hogy a szervezetet és az automatizálandó munkafolyamatokat jól megismerje. Sok bizalmas információt ismer meg, így titoktartási kötelezettsége van, amelyet szerződésben is biztosítani kell. Az általa készített dokumentumokat, implementációs terveket bizalmasan kell kezelni, illetéktelen kézbe jutva ugyanis hasznos információkat nyújthatnak egy esetleges támadónak.

A *programozó* készíti azokat a programokat, amelyek informatikai rendszereinken működnek. Az informatika hőskorában a számítógépes „guruk” egyedül vagy kis csoportokban dolgoztak, így felkészültségük és tudásuk meghatározó volt az elkészített rendszerek biztonsága szempontjából is. Beépíthettek olyan funkciókat is, amelyek kárt okozhattak. A számítógépes folklórból ismert történet, amikor egy programozót egy banki rendszer elkészítésével bíztak meg. Észrevette, hogy a kamat kiszámításánál rendszeresen kell kerekíteni, ritkán jön ki pontos eredmény. A programot úgy készítette el, hogy az mindig lefelé kerekített és a kerekítési hiba összegét a saját számlájára tette át. Az egyedi hiba csak

néhány fillér, fel sem tűnik a tulajdonosnak, összesítve azonban nagy tétel. A programozó ezzel az ötletével, a szabályozás hiányosságát kihasználva jelentős bevételre tett szert.

A nagy programrendszereket ma már jól szervezett vállalkozások készítik, a kódolást sokszor olcsó munkaerővel készíttetik. Az elkészített termékek minőségét komoly minőségbiztosítási rendszer felügyeli. Az egyéni hibák és rosszindulatú beavatkozások káros következményeit ezzel jelentősen lehet csökkenteni. A hibák kiszűrésének másik módja a nyílt forráskódú programok alkalmazása. Ebben az esetben a minőségbiztosítást az a közösség végzi, amelyik a programot fejleszti. Minden felhasználó más egyéniség, így sokféleképpen használják a programot, amelynek a hiányosságaira gyorsan fény derül.

A programozók és a rendszert üzemeltető mérnökök jól ismerik az azonosítással kapcsolatos problémákat. Ha azonban ők maguk vagy jelszavuk korrumpálódik, akkor az nagy veszélyt jelent a rendszerre, hiszen eltulajdoníthatják a programok és adatbázisok fontos elemeit vagy a betolakodó megváltoztathatja a számítógép beállításait és a programokat. Azonosításukat ezért nagyon biztonságosan kell elvégezni. Egyszerű, de fontos szabály, hogy *az üzemeltető csak akkor lépjen be a rendszerbe üzemeltetői szerepkörben, ha arra feltétlenül szükség van*. Különben dolgozzon felhasználói szerepkörben.

Informatikai rendszerek üzemeltetői egybehangzóan állítják, hogy a legtöbb kárt a social engineering alapú támadások okozzák. Azokat a támadási módszereket, amelyek az emberi viselkedés gyenge pontjait használják ki, nevezzük social engineeringnek. Magyar fordítása alkalmazott szociológia lehetne. Ezek a gyenge pontok lehetnek például a velünk született jóindulat, kíváncsiság; hiányos ismeretek és a memóriánk korlátai. A social engineering segítségével a bűnözők hozzáférést szerezhetnek a számítógéphez. A manipuláció célja többnyire az, hogy a számítógépkalózok titokban kémprogramot vagy más kártékony programot telepítsenek a számítógépre, vagy rávegyék a felhasználót, hogy kiadja jelszavait vagy más bizalmas pénzügyi és személyes adatait. Ezek a technikák régóta ismertek, a kémek és a detektívek évszázadok óta használják. Agatha Christie regényeinek egyik főszereplője, Miss Marple például ügyesen használja ki, hogy vele, a kedves, idős hölggyel szívesen csevegnek az emberek. Nem tételezik fel róla, hogy az információkat egy bonyolult bűnügy felderítésére használja. A történeteket a megtévesztéses támadás tipikus példáinak tekinthetjük. A számítástechnika fejlődésével a social engineering korábban nem ismert módszerei is kialakultak. Itt csak az adathalászattal foglalkozunk, sok más tanulságos példát találhat az olvasó Mitnick és Simon [21] könyvében.

Az *adathalászok* célpontjai elsősorban a honlappal rendelkező pénzintézetek, de a támadás lényegében minden online szolgáltató ellen irányulhat. Olyan honlapot készítenek, amelyik nagyon hasonlít valamely pénzintézet honlapjára. Ilyen példát mutattunk be az 1.2.4 fejezetben. Nagyon sok, véletlenszerűen kiválasztott e-mail címre küldenek levelet, amelyben kérik a címzettet, hogy nyissa meg a hamis honlapot. Azon személyazonosítás céljából kérik a gyanútlan felhasználó adatait; felhasználói nevét és jelszavát. Ha a célszemély megadja a kért adatokat, akkor azok nem a pénzintézethez, hanem a csalókhoz kerülnek, akik ha gyorsan cselekednek, akkor kiüríthetik a felhasználó számláját.

A felhasználók leveleinek tartalma is érdekli az adathalászokat. Az elektronikus levélforgalomból feltérképezhetik a felhasználó kapcsolatrendszerét, szokásait és érdeklődési

körét. Ezeket az információkat általában nem lehet közvetlenül hasznosítani, de támpontot jelenthetnek a támadásokhoz. A 3.1 ábrán bemutatott e-mailt a szerző 2010.10.09-én kapta és arra a webmasterservice@info.al címre kellett volna válaszolnia. Természetesen nem tette és azt tanácsolja, hogy senki se válaszoljon hasonló tartalmú levelekre.

```
Ez a teljes fiókja ellenőrzési folyamat
elmúlt évben a karbantartásáról a Webmail fiók. Ön
van szükség ahhoz, hogy ezt az üzenetet és adja meg azonosítóját
és JELSZÓ tér (*******). Meg kell csinálni, így mielőtt a
következő 48 órában kézhezvételétől e-mailt, vagy a számla
inaktiválódik, és töröljük adatbázisunkból.

Teljes neve:
Webmail felhasználói név:
Webmail Jelszó:
Jelszó megerősítése:
Születési idő:

Az Ön számlája is
ellenőrizni;https://mail.unideb.hu/squirrelmail/src/login.php

© Minden jog fenntartva. 2010 DEBRECENI EGYETEM H-4032 Debrecen, Egyetem tér 1.
```

3.1 ábra

A felsorolt cselekedetek közvetlenül a felhasználók számára okoznak kárt, csökkentik azonban a szolgáltató szervezetbe és általában a technikai civilizációnkba vetett bizalmat. Az adathalászat ellen különben egyrészt felvilágosítással, másrészt pedig aszimmetrikus titkosításon alapuló vagy biometrikus azonosítók alkalmazásával lehet eredményesen harcolni. Tudatosítani kell az informatikát használókban, hogy a kéretlen maileket hagyják figyelmen kívül és csak akkor fogadjanak el interneten érkező ajánlatokat, ha körültekintően megbizonyosodnak azok komolyságáról.

3.3 Technikai veszélyforrások

A fizikai és emberi veszélyforrások természete és szerepe időben lassan változik. Az áramkimaradás, a nedvesség vagy a számítástechnikai berendezések elektromágneses sugárzása olyan adottságok, amelyekkel az informatikai rendszerek üzemeltetőinek mindig számolni kell. Hasonló a helyzet az emberek fáradékonyságával, figyelmetlenségével vagy éppen hiszékenységével. Természetesen vannak olyan fizikai veszélyforrások, amelyek a technika fejlődésével vesztenek jelentőségükből vagy éppen teljesen meg is szűnnek. Gondoljunk például a mágneses adathordozókra.

Az ügyintézők felkészültsége is sokat változott az elmúlt évtizedekben. Tíz-húsz évvel ezelőtt középkorú embereknek kellett megtanulni egy számukra teljesen új eszköz kezelését. Sokuknak már a klaviatúra és az egér használata is gondot jelentett. Időközben felnövekedett egy olyan nemzedék, amely már gyerekkorában játékszerként találkozott a számítógéppel, de ha otthon nem is volt lehetősége számítógép használatára, akkor az iskolában sajátította el az

informatika alapjait. Bár a felhasználók tudása jelentősen nőtt és tudatossága javult, a social engineering típusú támadásoknak továbbra is ki vannak téve.

Lényegesen más a helyzet a technikai veszélyforrásokkal kapcsolatban. Az eszközök és a szoftverek területén viharos fejlődés történt. A számítógépek sebessége exponenciálisan nőt és ugyanez igaz tárolókapacitásukra is. Ezzel párhuzamosan üzembiztonságuk is sokkal nagyobb lett.

A fejlődés jellemzésére szolgál a következő néhány példa. A KLTE Matematikai Épületének 3. emeletén 1989-ben épült meg az első informatikai hálózat, amelyre a Számítástudományi Tanszék oktatóinak irodái és egy tanterem csatlakozott. Utóbbiban 20 PC-n dolgozhattak a programozó, majd a programtervező matematikus hallgatók. Az egyik szerző 1992-ben vásárolta meg az első PC-jét és büszke volt a 20 MB-os winchesterére. A Debreceni Universitas Egyesület tagintézményei[1] közötti nagy sebességű, üvegszálas informatikai hálózatot 1993-ban helyezték üzembe. Sebessége 100 Mbit/sec volt. A kapacitások szűkösségét jól jellemzi a következő, 1999-ből származó idézet: „Az Internet nem oktatási ill. kutatási célokra történő túlzott használata nem csak olyan kirívó példákat eredményezett, mint amikor például egyes játékprogramok felhasználása és letöltése akkora kapacitást foglalt le egyik egyetemünk informatikai rendszeréből, hogy szinte lebénult a hálózat.” ([31])

Indokolt tehát, hogy a technikai veszélyforrásokról szóló fejezetet több részre bontsuk. Először azokat a veszélyforrásokat ismertetjük, amelyek a számítógépekkel és a hálózatokkal kapcsolatosak. A második részben a kártékony programokkal: vírusokkal, férgekkel és trójaiakkal foglalkozunk. A következő rész témája a kéretlen levél, amely nemcsak napi bosszúságforrás, hanem komoly károkat is okozhat. A klasszikus technikai veszélyforrások ismertetését a leterheléses támadás ismertetésével zárjuk, amelyet a felhasználók kevésbé, de a rendszergazdák nagyon jól ismernek. A XXI. század első évtizedében viharosan terjedtek el a mobil eszközök. Újabban ezek is ki vannak téve támadásnak és velük is lehet informatikai rendszereket támadni. Erről szól a befejező rész.

3.3.1 A számítógépek és hálózatok, mint veszélyforrások

Egy-két évtizeddel ezelőtt a számítógépek és a hálózatok üzembiztonsága lényegesen kisebb volt, mint a mai berendezéseké. A gépek gyakran meghibásodtak és a csatlakozók is könnyen kimozdultak a helyükről. A munkaidő utáni vagy előtti takarítás következtében az elektromos csatlakozók kiestek és ezt a „hibát” az ügyintézők sokszor csak a rendszergazda közreműködésével tudták kijavítani. Az informatikai hálózatok passzív elemeit gyakran elvágták, néha rágcsálók rongálták meg, lehetetlenné téve így a távoli hozzáférést. Komoly problémát jelentett az adathordozók, tipikusan a floppylemezek sérülékenysége is. Nagyon költséges volt és ezért kevés vállalat engedhette meg magának az adatbázisok tükrözését. A

[1] Debreceni Agrártudományi Egyetem, Debreceni orvostudományi Egyetem, Kossuth Lajos Tudományegyetem, Debreceni Református Hittudományi Egyetem, MTA Atommagkutató Intézet.

technika sérülékenysége párosulva az üzemeltetők fegyelmezetlenségével többször vezetett évek alatt felépített nagy adatbázisok megsemmisüléséhez. Adatbiztonság szempontjából tehát maguk a számítógépek és a hálózatok fontos technikai veszélyforrások voltak. Különösen vonatkozik ez a megállapítás az adatok elérhetőségére. Gál Zoltán 1993-ban a Kossuth Lajos Tudományegyetem elektronikus levelező rendszerét ismertetve írja: ”Ezt a kommunikációt az X.25 bizonytalan működése nagymértékben nehézkessé teszi, […] Nagyon gyakran előfordul, hogy a forrás és a cél gépek nincsenek egyidőben bekapcsolva vagy közöttük nem lehet élő kapcsolatot felépíteni. Éppen ezért a küldő és a fogadó levelező rendszerek közötti „megegyezés” nem interaktívan, hanem kötegelt módon történik.” ([**12**])

A szoftverek sem működtek olyan biztonságosan, mint azt ma természetesnek tartjuk. A személyi számítógépek és hálózatok operációs rendszerei (DOS, DRDOS, Windows, Apple, MacIntosh, Novell, stb.) sok hibát tartalmaztak, aminek következtében gyakran összeomlottak, újra kellett installálni őket. Bár a UNIX-nál kezdetektől fogva az egyirányú függvényt alkalmazó azonosítási technikát alkalmazták, ez a többi, csoportmunkát támogató, operációs rendszerek körében csak az 1990-es évek közepén vált általánossá. A felhasználó azonosítás nagyon gyenge volt. Hálózatba kapcsolt számítógépek között a hálózati operációs rendszerek lehetővé tették az adatcserét. Ez lényegesen megkönnyítette a sokat utazó felhasználók dolgát, hiszen nem kellett magukkal cipelni a munkájukhoz szükséges állományaikat. Ugyanakkor az adatok, beleértve a jelszót és a bizalmas adatokat is, kódolatlanul vándoroltak a nyilvános hálózaton. Megjegyezzük, hogy a bizalmas adatcseréhez szükséges technológia rendelkezésre állt, de az informatikai hálózatot még az 1990-es évek közepén is elsősorban az akadémiai szféra használta. A tervezők ennek megfelelően abból indultak ki, hogy a hálózaton alapvetően nyilvános tartalmakat forgalmaznak. Az Internet üzleti felhasználásának és a felhasználók számának növekedésével a hálózati morál lényegesen megváltozott. Nyilvánosan küldött bejelentkezési adatok a hekkerek kezébe kerülve lehetővé teszik számukra bizalmas adatbázisokhoz való hozzáférést, amivel gyakran élnek is. Az 1995 februárjában, a Netscape által kifejlesztett SSL protokoll megjelenése oldotta meg ezt a problémát.

Ma a számítástechnikai berendezések már sokkal biztonságosabban működnek. Az infrastruktúra aktív és passzív elemei ritkán hibásodnak meg; különösen akkor igaz ez a megállapítás, ha figyelembe vesszük életciklusuk hosszát. Három-négy évenként az aktív elemek erkölcsileg elavulnak, az újabb programokat, programverziókat már nem lehet rajtuk hatékonyan futtatni. Az Informatikai Kar hallgatói laboratóriumai szinte folyamatos üzemben működnek és a hallgatók ritkán bánnak velük kesztyűs kézzel. A terminálok döntő többségét azonban elegendő három évenként cserélni.

Az informatikai hálózatok sűrűsége és sebessége is jelentősen nőtt. Ma már hazánk legtöbb települését eléri a szélessávú hálózat. Jelentős változást jelentett ebből a szempontból a vezetékes és mobil telefonhálózatok elterjedése és ezzel párhuzamosan az informatika és kommunikáció konvergenciája. Mobil eszközökkel az Internet ma már hazánk szinte minden pontjáról elérhető. Lényegesen javult a hálózatos infrastruktúra üzembiztonsága is. Mindezek miatt az aktív és passzív eszközök hibái a korábbinál jóval kisebb szerepet játszanak az adatbiztonság szempontjából.

Ugyanakkor a mai informatikai rendszerek igen bonyolultak, nagy méretűek, így óhatatlanul tartalmaznak hibákat. Komoly biztonsági kockázatot jelent az eszközök heterogenitása is. Ugyanarra a lokális hálózatra nagyon sok, különböző korú és teljesítményű berendezés van felfűzve. Ezeknek a berendezéseknek az operációs rendszerei is különböznek, összehangolásuk közben komoly hibákat lehet véteni. Kockázatot jelentenek a rutin munkából ideiglenesen vagy véglegesen kivont azon berendezések, amelyeket nem kapcsoltak le a hálózatról. Ezeken csak alkalomszerűen vagy sohasem frissítik a szoftvereket, a régebbi verziók ismert hibáin keresztül pedig rés nyílhat a támadók számára, amit azok adandó alkalommal ki is használnak. Célszerű tehát a használaton kívül helyezett berendezéseket rögtön lekapcsolni az élő hálózatról.

Itt kell szólnunk az adattárolókkal kapcsolatos problémákról is. Ezek kapacitása is drámaian nőtt, a tárolást hosszú időtartamra, biztonságosan megoldják ugyanakkor méretük lényegesen csökkent. A hekkerek által, interneten keresztül végrehajtott akciók jelentik az adatlopás látványos és nagy médiavisszhangot kiváltó módját. A szervezeteknek azonban összességében sokkal nagyobb kárt okoznak a belső munkatársak által eltulajdonított adatok. Nagy Britanniában, 2008-ban például 277 jelentős adatlopási eset történt, amelyeknek átlagos értéke 1,73 millió font volt. Ezek oka főként az volt, hogy a szervezetek nem szabályozzák megfelelően az USB tárolók munkahelyi használatát, pontosabban nem tiltják meg az ügyintézőknek ilyen eszközök használatát. Márpedig ezeken a nagy kapacitású, de kis helyen elférő eszközökön nagyon sok, fontos adatot el lehet tulajdonítani.

A központi adattárolóknak is megvan a maguk életciklusa, bizonyos idő után nagyobbra és korszerűbbre kell cserélni azokat. Ilyenkor nem elegendő a rajtuk levő adatok logikai törlése, hanem gondoskodni kell az adattárolók biztonságos megsemmisítéséről. Üzemen kívül helyezett vagy lecserélt, de még használható számítógépeket a vállalatok szívesen ajándékozzák iskoláknak vagy szociális intézményeknek. Ilyenkor is gondoskodni kell a gépen levő adatok visszafordíthatatlan törléséről.

Informatikai biztonság szempontjából természetesen az operációs rendszerek és alkalmazói szoftverek is nagyon fontosak. A nagyon változatos funkciókat ellátó szoftverek biztonságáról általánosságban nem lehet véleményt mondani, azokat egyedi esetben kell megvizsgálni. Az elmúlt évtizedekben azonban kialakultak azok a szoftverfejlesztési és tesztelési technikák, amelyek garanciát adnak a durva hibák elkerülésére. A szoftvergyártók rendszeresen figyelik a termékükkel kapcsolatos tapasztalatokat és az esetleges hibákat a regisztrált felhasználóknál frissítő programokkal javítják. Fontos, hogy a frissítéseket rögtön telepítsük, mert ezzel sok bosszúságot lehet megtakarítani. Előfordul ugyanis, hogy egy program régebbi verziójában olyan biztonsági rést találtak, amelyen keresztül be tudnak hatolni a rendszerünkbe. Ezek az információk a világhálón gyorsan elterjednek és a hekkerek igyekeznek megtámadni a hasonló programmal dolgozó számítógépeket. A frissített programmal működőknél már nem érnek célt, hiszen a rést betömték, a régi verzióval dolgozókhoz azonban be tudnak hatolni.

3.3.2 Kártékony programok

A sokszor bizonytalanul működő hardverelemek és infrastruktúra mellett a rosszindulatú programok (malware) is gyakran előforduló károkozók az informatikai rendszerekben. Ennek a programfajtának tipikus képviselői a vírusok, a férgek és a trójaiak.

3.3.2.1 Vírusok

Bár a vírusok ma is eminens veszélyforrások, történetük visszanyúlik az 1970-es évekre. A *vírusok (és a férgek)* legfontosabb jellemzője, hogy reprodukcióra képes számítógépes programok. Biológiai névrokonaikhoz hasonlóan felépítésük nagyon egyszerű, méretük pedig kicsi. Elsősorban bosszúságot és időveszteséget okoznak, nemcsak a lokális erőforrásokat terhelik, de az Internet forgalmának is jelentős hányadát lefoglalják. A vírusok közvetlen kárt is okozhatnak, adatokat törölhetnek és módosíthatnak a számítógépeken.

Rendszerint az operációs rendszerek vagy olyan alkalmazói programok hiányosságait használják ki, amelyek futtatható állományokat tartalmazhatnak. Első példányaik az 1970-es években jelentek meg (pl. Brain, Jerusalem). A „computer virus" elnevezést Fred Cohen egy 1983-as dolgozatában vezette be. Az elmúlt évtizedekben folyamatos volt a versenyfutás a vírusfejlesztők és a vírusirtók között. Ma már több tízezer számítógépes vírust ismerünk, amelyek azonban néhány alapfajta változatai mutációi. A kártékony programok, közöttük a vírusok, korai történetéről nagyon alapos elemzés található Denning [**8**], az újabb vírusokról pedig Hunter [15235] könyvében, illetve az [16235] tanulmányban.

A vírusok három részből állnak, úgymint kereső, reprodukáló és akciót végrehajtó részből. Bizonyos esetekben a harmadik rész hiányozhat. A kereső rész figyeli a számítógép működését és a reprodukálásra kedvező helyzetet érzékelve rögtön akcióba lép. Kedvező helyzetet jelent az, ha megfertőzhető állományt, állományokat talál az elérhető memóriaterületen. A kereső rész ügyessége határozza meg a vírus fertőzőképességét, azaz terjedési sebességét. A kereső rész által felkutatott állományokba a reprodukáló rész másolja be a vírus kódját, esetleg a kódnak valamilyen variánsát. A másolás csak akkor történik meg, ha az állomány még nem fertőzött. Többször ugyanis nem érdemes megfertőzni egy állományt, mert a vírusok viharosan terjednek és a többszörös infekciót sokkal könnyebb észrevenni. Az akciót végrehajtó rész, ha egyáltalán tartozik ilyen a vírushoz, valamilyen egyszerű műveletet indít el. Ez lehet egy üzenet megjelenítése a képernyőn, állományok törlése vagy átnevezése. Az 1980-as évek végén nevezetes volt a „potyogtatós" vírus. Az adatokat abban az időben nem grafikusan, hanem alfanumerikusan, azaz karakterenként jelenítették meg. Ha egy „potyogtatós" vírus volt a számítógépen, akkor a képernyőn a sorok elkezdtek összekuszálódni, majd a karakterek sorra lehullottak. Végül a képernyő teljesen üres volt. Sokan megijedtek a látvány következtében, pedig csak egy ártalmatlan csínytevés áldozatai voltak, a vírus kiirtása után a számítógép rendben működött tovább.

Kezdetben a *fájlvírusok* és a *boot szektort fertőző* (BSI) vírusok terjedtek el a legjobban. Az előbbiek azt használták ki, hogy a WINDOWS operációs rendszerben a .exe illetve a .com kiterjesztésű fájlok közvetlenül futtathatóak. Ha a vírus valamilyen módon, leginkább egy fertőzött fájllal, bekerül egy számítógépbe és a fertőzött fájlt elindítják, akkor a vezérlés átkerül a vírusra. A kereső modul a memóriában egy még nem fertőzött, futtatható

fájl után kutat. Ha talál ilyet, akkor annak a végéhez hozzáfűzi saját másolatát és az érintett fájl belépési pontját úgy módosítja, hogy a vezérlés indításakor a vírusra kerüljön, majd a vírusban foglalt utasítások végrehajtása után visszakerüljön az eredeti belépési pontra. Ezzel megtörténik egy újabb állomány fertőzése, a vírus esetleg végrehajt még egy akciót, majd visszaadja a vezérlést az eredetileg elindított programnak. Az egész eljárás nagyon gyorsan lejátszódik, így a felhasználó nem veszi észre, hogy a gépe fertőzést kapott. Egy idő után azonban lelassul a gépének a működése és, kártékony vírussal történő fertőzés esetén, adatvesztés is történhet.

A BSI vírusok más terjedési mechanizmust alkalmaznak. A számítógépre kerülve megkeresik valamely lemez boot szektorának ritkán használt szektorát és ide másolják magukat. Amikor a számítógép a fertőzött lemezről olvas, akkor azt a boot szektorral kezdi. A vírus úgy intézi, hogy átmásolhassa magát egy még nem fertőzött lemezre, majd visszaadja a vezérlést a fájlkeresőnek. A BSI vírusok főként fertőzött floppy lemezekkel terjedtek, így azok jelentőségének csökkenése miatt a BSI vírusok lényegében eltűntek.

Napjainkban a *makró* és az *e-mail* vírusok okozzák a legtöbb problémát. A Microsoft Office alkalmazások: World, Excel, PowerPoint és sok Java alkalmazásban is el lehet helyezni olyan végrehajtható programokat, közismert néven makrókat, amelyek a sokszor végrehajtandó lépések végrehajtását automatizálják. A makró vírusokat ugyanazon a nyelven írják, mint a hasznos makrókat, majd megfertőzve velük egy fájlt eljuttatják azt egy felhasználónak. Kinyitva a fertőzött fájlt a vírus keres egy hasonló alkalmazást és megfertőzi azt. A legtöbb makró vírus World állományokat fertőz, utána következnek az Excel állományok, ami az alkalmazások gyakorisága miatt természetes.

Az utóbbi időben az e-mail vírusok vették át a főszerepet. Elektronikus levelek mellékleteként terjednek. Amikor a felhasználó kinyitja a mail a fertőzött mellékletet, akkor aktivizálódik, ami vagy valamely rezidens fájl vagy az áldozat által küldött mailek fertőzését jelenti.

3.3.2.2 Férgek

A *férgek* (worm) is reprodukcióra képes programok ellentétben azonban a vírusokkal terjedésükhöz nincs szükségük hordozó állományokra. Bár az első féreg programot már 1982-ben elkészítette John Shoch és John Hupp eleinte kevés követőjük akadt, mert csak informatikai hálózatokon terjednek. Káros tevékenységük ma már kiterjed a mobiltelefon hálózatokra is. A férgek leggyakrabban e-maillel jutnak el újabb felhasználókhoz. Tipikus forgatókönyv az, hogy a felhasználó kap egy elektronikus levelet, amely tartalmaz egy mellékletet valamilyen hasznosnak tűnő információval. Ilyen lehet egy program regisztrációját feltörő kód vagy pornográf oldalak hozzáférési adatai. A linkre kattintva általában nem a hasznos adatot kapja meg, hanem aktivizálódik a féreg.

Ma már olyan férgek is terjednek a világhálón, amelyek aktivizálásához nem kell megnyitni a mellékletet, elég a levelet elolvasni. A Bubbleboy és a KAKWorm voltak az első ilyen típusú férgek. Ezek azt használták ki, hogy az Outlook és az Outlook Express levelező programok HTML formátumú leveleket is meg tudnak jeleníteni. A HTML-ben írt levelekbe

azonban el lehet rejteni VBScript betéteket és ActiveX vezérlőket is. Ha ezek futtatását a felhasználó engedélyezi, akkor rögtön aktivizálódik a féreg.

3.3.2.3 Trójaiak

A *trójaiak* az internetes korszak termékei, csak hálózatba kapcsolt számítógépeken tudják kifejteni tevékenységüket. Ezek is kicsi programok, de a vírusoktól és férgektől eltérően reprodukcióra képtelenek, így maguktól nem is terjednek. Általában valamilyen hasznosnak látszó programban rejtik el őket. Innen származik az elnevezésük is, amelyik a Homérosz híres eposzában, az Iliászban, szereplő trójai falóra utal[2].

Az e-mail vírusokhoz és férgekhez hasonlóan elektronikus levél mellékleteként kaphatjuk meg őket. Terjedhetnek weboldalakról letöltött, fertőzött állományokkal is. Feladatuk, hogy telepedjenek rá számítógépekre és azokról információkat juttassanak el gazdájuknak. Általában a háttérben, csendben meghúzódva gyűjtik vagy küldik gazdájuknak az adatokat.

A *jelszólopó* trójaiak a billentyűzetet figyelik és közben összegyűjtik a felhasználó által használt felhasználói neveket és jelszavakat. Egy másik típusuk a felhasználó jelszavait tartalmazó kódolt vagy kódolatlan állományokat keresi a számítógép memóriájában. Feladatukat teljesítve elküldik a begyűjtött adatokat egy meghatározott e-mail címre.

A *backdoor* programok igyekeznek felderíteni a nem eléggé védett és így kinyitható kommunikációs portokat. Ha találnak ilyet, akkor kinyitják azt, értesítik a gazdájukat a sikeres akcióról, aki a nyitott porton keresztül bejuthat a számítógépbe, ahonnan már hatékonyan tud további támadásokat indítani.

A *letöltő* trójaiak egy számítógépbe kerülve további programokat töltenek le a gépre. Ezek persze további, bonyolultabb akció végrehajtására képes programok is lehetnek.

A *kémprogramok* (spyware), a jelszólopókhoz hasonlóan települnek a gépekre, de nem jelszavakat, hanem a felhasználó személyes adatait, esetleg számítógép használati szokásait gyűjtik össze és továbbítják.

A 3.2 ábra[3] a 2010. március havi vírusstatisztika első 20 helyezettjét mutatja. Sok, különböző rosszindulatú program képviseli magát ezen a listán. Hasonló összegzés havi és évi felbontásban rendszeresen készül, így meg lehet figyelni ezeknek a veszélyes programoknak az életciklusát is.

---

[2] Az eposz szerint a görögök hosszú ideig, sikertelenül ostromolták Trója várát Ekkor Odüsszeusz tanácsára egy hatalmas falovat építettek, amelyet a város kapujához vontattak, majd színleg visszavonultak. A trójaiak megörültek az ellenség visszavonulásának és bevontatták falaikon belülre a falovat. Éjszaka, amikor a védők aludtak, a faló belsejéből görög katonák másztak ki, majd kinyitották a város kapuit. A támadók időközben visszalopakodtak a város közelébe és a nyitott kapun keresztül bevonultak Trójába, majd elfoglalták azt.

[3] Forrás: Kártevő TOP 20 – 2010 március, http://www.viruspajzs.hu/kartevo-top-20-2010-marcius/

Helyezés	Előző hónap	Kártevő	Fertőzések száma
1	1	Net-Worm.Win32.Kido.ir	332833
2	2	Virus.Win32.Sality.aa	211229
3	3	Net-Worm.Win32.Kido.ih	186685
4	4	Net-Worm.Win32.Kido.iq	181825
5	5	Worm.Win32.FlyStudio.cu	121027
6	6	Trojan-Downloader.Win32.VB.eql	68580
7	Új	Trojan.Win32.AutoRun.abj	66331
8	9	Virus.Win32.Virut.ce	61003
9	10	Packed.Win32.Krap.l	55823
10	8	Worm.Win32.AutoIt.tc	55065
11	15	Worm.Win32.Mabezat.b	49521
12	7	Exploit.JS.Aurora.a	43776
13	Új	Packed.Win32.Krap.as	40912
14	Új	Trojan.Win32.AutoRun.aay	40754
15	18	Trojan-Dropper.Win32.Flystud.yo	40190
16	12	Virus.Win32.Induc.a	38683
17	13	not-a-virus:AdWare.Win32.RK.aw	38547
18	Új	Trojan.Win32.AutoRun.abd	37037
19	14	not-a-virus:AdWare.Win32.Boran.z	36996
20	20	not-a-virus:AdWare.Win32.FunWeb.q	34177

3.2. ábra

A vírusok, férgek és trójaiak ellen védekezhetünk úgy, hogy nem fogadunk el fájlokat ellenőrizetlen forrásból, nem nyitunk ki ismeretlen küldőtől érkező, mellékletet tartalmazó mailt és nem töltünk le állományokat bármilyen weboldalról. Rosszindulatú programokat gyakran kaphatunk erotikus vagy pornográf tartalmú weboldalakról. Sajnos ezekkel az intézkedésekkel csak rövid ideig lehet tisztán tartani a számítógépet. Ma már elengedhetetlen egy vírusirtó, sőt a komplexebb védelmet biztosító tűzfal használata.

A vírusirtók nemcsak a vírusok, hanem a többi rosszindulatú program ellen is védelmet nyújtanak. Számolni kell azzal is, hogy a vírusirtó működéséhez is erőforrásra van szükség, amellyel csökken a hasznos tevékenységre fordítható teljesítmény. Érdemes tehát hatékonyan működő vírusirtót beszerezni. Vírusirtó kiválasztásakor a hatékonyság mellett, a folyamatos frissítés is nagyon fontos szempont. Folyamatosan jelennek meg ugyanis új rosszindulatú programok, amelyekkel a korábbi verziók nem tudnak mit kezdeni. A vírusirtó programok ugyanis tartalmazzák az ismert rosszindulatú programok lenyomatait és ezekkel megegyező mintákat keresnek a védendő számítógépen tárolt állományokban, pontosabban az állományok azon részében, ahol a rosszindulatú programok tapasztalatok szerint elő szoktak fordulni. Új rosszindulatú program lenyomata természetesen nem lehet benne egy korábbi vírusirtó adatbázisában, így azt fel sem tudja ismerni. A frissítések tehát számítógépeink biztonságát szolgálják!

3.3.3 Kéretlen levelek

A *kéretlen levelek* (spam mail) mindennapjaink kellemetlen és, mint arra az előző fejezetben rámutattunk, sokszor kártékony tartalmat hordozó velejárói. Az egyre hatékonyabb szűrők ellenére elektronikus levelesládánkba számtalan ilyen üzenet érkezik. Nem mindig volt

ez így. Az elektronikus levelezés hazánkban 1989-ben indult el a Magyar Tudományos Akadémia kutatóintézetei, néhány egyetem és országos gyűjtőkörű könyvtár között. Az USA-ban és a nyugat Európai országokban már néhány évvel korábban elindult az Internet és azon a levelezés, de a technológiát a szocialista országok nem vásárolhatták meg. A szükséges szoftvert ezért az MTA Számítástechnikai és Automatizálási Kutató Intézete fejlesztette ki és ELLA-nak nevezték el. A hazai rendszer bekapcsolódott a nemzetközi forgalomba is, így külföldi partnerekkel is lehetet e-mailt váltani. A polgári célokat szolgáló hálózatot – hazánkhoz hasonlóan – külföldön is szinte kizárólag az egyetemek és kutatóintézetek használták. Az internetes etikett, azaz Netikett tiltotta üzleti reklámok, félrevezető vagy obszcén tartalmú mailek küldését.

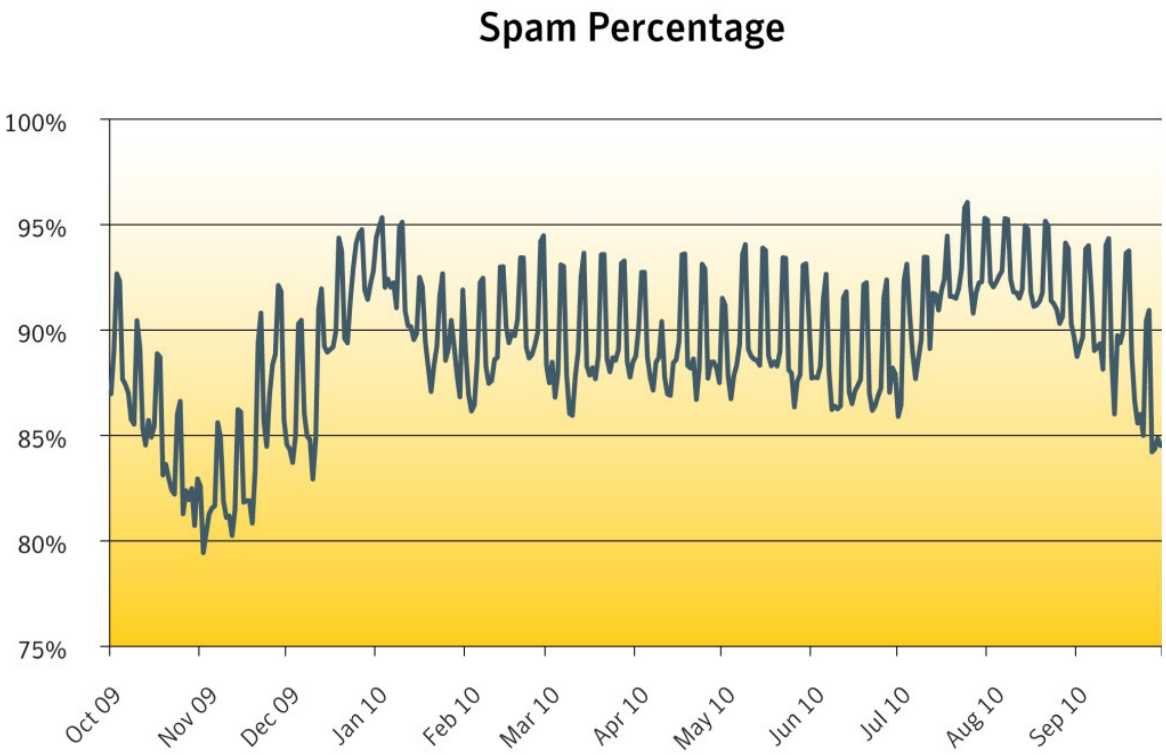
Az internet penetráció növekedésével, az alkalmazások egyszerűsödésével és bővülésével az üzleti szféra is egyre nagyobb érdeklődést mutatott az internet iránt. Az elektronikus levelezés előnye a hagyományossal szemben a gyorsaság mellett az is, hogy nagyon egyszerű körleveleket küldeni. Összeállítva e-mail címek egy listáját egyetlen gombnyomásra lehet ugyanazt a levelet a listán szereplő címekre továbbítani. Ezt a kényelmes funkciót a kutatók gyorsan felismerték és rendszeresen használták is tudományos konferenciák szervezésére vagy új kutatási eredmények széles körben való elterjesztésére. Az internet üzleti célú használatának kezdetekor kézenfekvő gondolat volt, hogy ezt a technikát reklámok továbbítására is fel lehet használni.

Ettől kezdve folyamatosan bővült a kéretlen mailek tartalmának választéka. Küldenek reklámokat, üzleti ajánlatokat, nagy haszonnal kecsegtető befektetési ajánlatokat, politikai reklámokat; felhasználják az e-mailt kártékony programok terjesztésére és adathalászatra. Kereső robotokkal egyszerűen összegyűjthetőek a weblapokra kirakott e-mail címek, ezekből pedig könnyen lehet akár több millió címet tartalmazó listát is összeállítani. Néhány évvel ezelőtt az egyik szerző kéretlen levélben kapott ajánlatot ilyen lista megvásárlására.

Az e-mail címek gyűjtésének megnehezítésére többféle technikát alkalmaznak: nem szövegesen, hanem képként teszik ki a weblapra a címet, A @ jel helyett at-et írnak, a . helyett pedig dot-ot. A szerző címe ebben az átírásban tehát így néz ki: Petho dot Attila at inf dot unideb dot hu. A címeket és regisztrációs kódokat elrejthetik olyan képekben is, amelyek különbséget tudnak tenni az ember és a robotok intelligenciája között. Ha egy szóban a karaktereket deformáljuk és különböző betűtípussal valamint mérettel írjuk és még hullámzik is a szöveg, akkor azt egy robot ma még nem tudja értelmezni, egy ember azonban olvasni és reprodukálni képes. A 3.3 ábra egy ilyen képbe elrejtett kódot mutat.

3.3 ábra

A kéretlen levelek nemcsak személyes bosszúságot és fölösleges munkát jelentenek, hanem az internet forgalmának is jelentős részét képezik. A 3.4 ábrán[4] a 100 %-ot jelző felső vonal az internet teljes e-mail forgalmát jelöli, a fekete hullámvonal pedig a spamek arányát a teljes forgalomból. A statisztika 2009 októberétől 2010 szeptemberéig mutatja a helyzetet napi bontásban. Látható, hogy a spamek aránya általában 90 % körül van, néha eléri a 95 %-ot is, nagyon ritkán megy azonban 80 % alá. Ez azt jelenti, hogy minden 100 e-mail közül 90 spam.



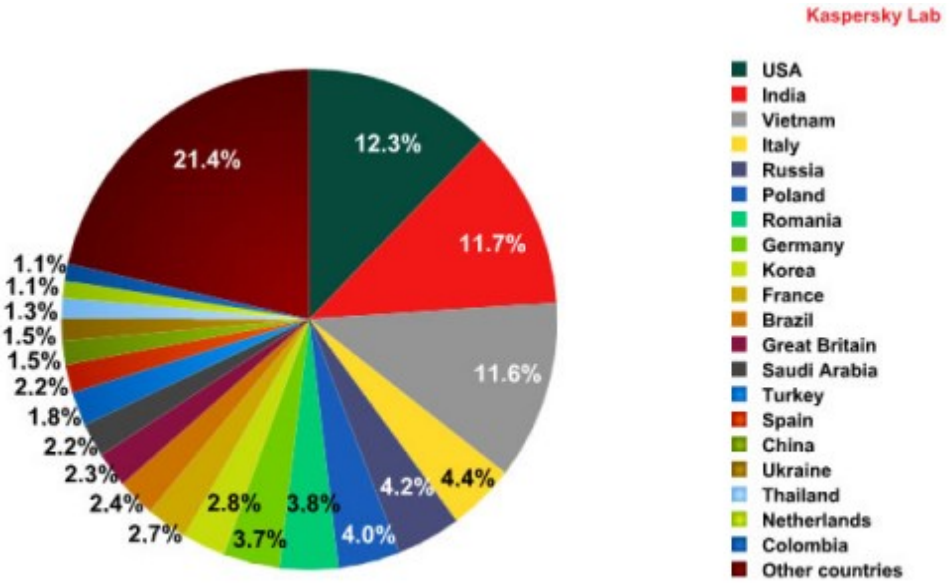
3.4 ábra

A felhasználói fiókokba is bőven kerül kéretlen levél, ott azonban az arányuk sokkal alacsonyabb, tapasztalatom szerint a 10 %-ot sem éri el. Ha ugyanis a spamek aránya 90 % lenne, akkor a legtöbb felhasználó bezárná a levelesládáját. A levelező rendszerek üzemeltetői is tudják ezt és komoly erőfeszítéseket tesznek a kéretlen levelek korai kiszűrésére. Az egyik leghatásosabb eszköz erre az internet forgalmának monitorozása. Ha egy levelező gépről spamek küldését észlelik, akkor azt a gépet tiltó listára teszik. Néhány évvel ezelőtt ez történt az Informatikai Kar levelező szerverével is. A lefolytatott vizsgálat során kiderült, hogy egy hallgató üzleti reklámokat tartalmazó spameket küldött az egyik számítógépről. Akcióját

[4] Forrás: State of Spam & Phishing, A Monthly Report, symantec, October 2010-11-02 http://www.symantec.com/content/en/us/enterprise/other_resources/b-state_of_spam_and_phishing_report_10-2010.en-us.pdf

néhány perc alatt észlelték és le is tiltották a levelező szerverünket. A hallgató beismerte tettét, így az egyetem saját hatáskörében intézkedett és fél évre felfüggesztette a hallgatói jogviszonyát.

Az internetet monitorozó szervezetek azt is összesítik, hogy mely országokból indul el a spamek áradata. A 3.5 ábrán[5] a 2010 áprilisi Kaspersky spam jelentést mutatjuk be. A diagramból látszik, hogy a vizsgált hónapban a legtöbb spam az USA-ból indult, amit kis különbséggel követ India és Vietnam. Megjegyzendő, hogy a számítógépek tulajdonosai vagy felhasználói – a mi hallgatónkkal ellentétben – sokszor egyáltalán nincsenek tudatában, hogy gépük kéretlen leveleket küld. Vannak ugyanis olyan technikák, amelyekkel úgynevezett zombihálózatokat (botnet) lehet létre hozni. Egy zombi hálózatnak több tízezer tagja is lehet. 2010 júliusában őrizetbe vettek egy szlovén férfit, akit egy 12 millió számítógépből álló zombihálózat megszervezésével és üzemeltetésével gyanúsítanak[6]. Ezek erősen centralizált hálózatok, amelynek a tagjai nem is tudnak a tagságukról, a központtól és egymástól is nagy távolságban lehetnek. Egyébként normálisan működő számítógépek, amelyek időnként végrehajtják a központi számítógéptől érkező utasításokat. Zombihálózatokat rendszerint spam mailek küldésére és túlterheléses támadás végrehajtására hoznak létre.



3.5 ábra

A monitorozás hatásos megelőző eszköz, de az elküldött kéretlen levelek ellen is kell védekezni. Kéretlen leveleket olyan nyelvi elemző programokkal lehet kiszűrni, amelyek a küldő szokatlan címe, a tárgyban vagy a levél törzsében szereplő szavak alapján teszik karanténba a gyanús küldeményeket. A leveleket szűrni lehet abból a szempontból is, hogy

---

[5] Kaspersky spam jelentés: 2010 április, http://www.viruspajzs.hu/kaspersky-spam-jelentes-2010-aprilis/
[6] Forrás: http://www.msnbc.msn.com/id/38439213/ns/technology_and_science-security

tartalmaznak-e csatolmányként bizonyos típusú képfájlokat. Szűrőprogramokat már a tűzfalakban is alkalmaznak, amelyek az elsődleges, durva rostálást végzik. Másodlagos, finomabb hangolású szűrést végeznek a levelező szoftverek, amelyeknek ezt a funkcióját magunk is beállíthatjuk. Előfordulhat, hogy a levelező rendszer olyan levelet is spamnek minősít, amely pedig hasznos, sőt fontos információt tartalmaz. Ilyenkor célszerű a szűrési paraméterek újrakonfigurálása. A szűrőprogramok fontosságát mutatja, hogy például a Debreceni Egyetemre érkező mailek 90 %-át nem engedi tovább a címzetteknek.

3.3.4 Túlterheléses támadás

A túlterheléses támadás, amelynek a közismert angol rövidítése DoS (denial of service) jelenti az egyik legnagyobb kihívást a hálózatos szerverek üzemeltetői számára. Jól tudjuk, hogy minél több szolgáltatás kapcsolódik egy rendszerhez és azt minél többen veszik igénybe, annál nagyobb a kiszolgáló egységek leterheltsége, ami a felhasználók számára a válaszidők hosszának növekedéseként jelenik meg. Ha a rövid időn belül beérkező igények száma egy határértéket túllép, akkor a szerver terhelése olyan nagy lehet, hogy már nem tud új kéréseket fogadni.

Hallgatók nagyon jól tudják, hogy a félévek illetve a vizsgaidőszak kezdetekor a tanulmányi rendszer nagyon leterhelt, sok türelem kell ahhoz, hogy egy közkedvelt tárgyat felvegyenek vagy kedvezőnek látszó vizsgaidőpontra bejelentkezzenek. Nemcsak az egyetemen történhet ilyen esemény.

A 2010/11-es tanévre 2010. február 15-én 24 óráig lehetett jelentkezni a felsőoktatási intézményekbe. Az utolsó pillanatban jelentkezők nagy száma miatt az online felvételi regisztrációs rendszer azonban lelassult, sőt időnként le is állt. Mivel sok jelentkező maradt hoppon, így a jelentkezési határidőt 22 órával meghosszabbították.

2008. november 20-án kezdte meg működését az Európai Unió internetes könyvtára, az Europeana. Az informatikai hátteret a Koninklijke Bibliotheek, a holland nemzeti könyvtár biztosítja, amelyik óránként ötmillió látogatót még képes kiszolgálni. Az első napon azonban, a jól sikerült nemzetközi promóciónak köszönhetően, olyan nagy volt az érdeklődés, hogy óránként 20 millióan próbálták betölteni az Europeana oldalát. A rendszer ezt a terhelést nem bírta és két nappal később összeomlott. Szervezeti változtatásokat és technikai fejlesztéseket hajtottak végre az újraindítás előtt, amelyek hatásosnak bizonyultak, mert hasonló probléma később már nem fordult elő.

Támadási céllal is elő lehet állítani ilyen helyzetet. Egy vagy néhány számítógépről nagyon sok kérést küldenek a célgépnek. A kérések generálása sokkal rövidebb időt vesz igénybe, mint azok kiszolgálása, így elő lehet állítani olyan helyzetet, amikor a célba vett szerver túlterhelése olyan nagy lesz, hogy a legális kéréseket már nem tudja kiszolgálni, a rendszer esetleg össze is omlik. Ha rosszul van konfigurálva a szerver, akkor a támadó akár be is juthat a rendszerbe. Néhány évvel ezelőtt ilyen támadási technikával jutott be egy hallgató

egyetemünk akkori tanulmányi rendszerébe, méghozzá rendszergazdai jogosultságokkal. Ezeket a jogokat csak arra használta, hogy körülnézett az állományokban.

Az ilyen egyszerű támadások mechanizmusát ma már jól ismerjük és az újabb rendszerek fel vannak készítve azok elhárítására. Egy IP címről érkező tömeges igényt például nem veszik figyelembe, vagy egy korlátot elérve további kérések elől lezárják a kommunikációs portokat. A támadók ezért új technikát fejlesztettek ki, az úgynevezett szétosztott túlterheléses támadást, amit DDoS-nak (distributed denial of service) neveztek el. Ennek a lényege az, hogy sok számítógép összehangoltan intéz túlterheléses támadást egy szerver ellen. Az előző fejezetben említett zombihálózatok egyik fő alkalmazási területe a DDoS támadások végrehajtása.

Nyomatékosan felhívjuk a figyelmet arra, hogy a számítógépek elleni támadások nem diákcsínyek, hanem bűncselekmények. A Büntetőtörvénykönyv 300/C paragrafusa szerint ugyanis: „Aki adat bevitelével, továbbításával, megváltoztatásával, törlésével, illetőleg egyéb művelet végzésével a számítástechnikai rendszer működését jogosulatlanul akadályozza, vétséget követ el, és két évig terjedő szabadságvesztéssel büntetendő.”

3.3.5 Mobil eszközök veszélyeztetettsége

Századunk első éveinek legjelentősebb változása az informatika területén, a hálózatok széleskörű elterjedése mellett, a mobil eszközök, így mobil telefonok, PDA-k, notebookok, netbookok, stb. számának robbanásszerű növekedése. E mellett tanúi vagyunk az informatika és a kommunikációs technológia konvergenciájának is. Számítógépről telefonálhatunk, azon hallgathatunk rádiót, nézhetünk televíziót és olvashatjuk a legújabb híreket vagy éppen könyveket. A másik oldalon a mobiltelefonokba egyre erősebb processzorokat és nagyobb memóriát építenek be és szoftvereik is egyre bonyolultabbak és hatékonyabbak. Így válik lehetővé, hogy mobilunkon olvashatjuk e-maileinket, követhetjük kedvenc hírportálunk újdonságait és elterjedőben van a mobil fizetés is.

A konvergencia mindkét oldalán találhatók veszélyforrások. Az utóbbi időben megjelentek és gyorsan terjednek a mobiltelefonokat célba vevő káros programok: vírusok, trójaiak és kéretlen sms-ek. A káros programok leginkább kéretlen sms-el vagy fertőzött szoftverekkel terjednek. Leginkább anyagi kárt okoznak a készülék tulajdonosainak. Találtak például olyan férget, amely rendszeresen emelt díjas hívást kezdeményezett külföldi számra. 2010. november 12-én az Index számolt be egy Kínai mobilvírusról, amely éppen vírusirtónak álcázta magát. „A vírus a megfertőzött telefon SIM-kártyájáról és névjegyzékéből elküldi a neveket és telefonszámokat a hekkerek szerverére, majd a rendszer a begyűjtött számokra kétféle spamüzenetet küld. Az egyik fajta sms-ben egy link van, amely magát a vírust tölti le a telefonra. A másik típusú üzenet pedig egy online szoftverboltra visz el, ahol az automatikusan levont pénzért cserébe semmi sem kap a látogató.”

Természetesen a kicsi, de nagy számítási és tároló kapacitással, hangfelvételre és fényképezésre alkalmas eszközökkel bizalmas adatokat lehet alig észrevehetően gyűjteni és

kicsempészni kevéssé felkészült szervezetektől. Fontos adatokat kezelő szervezeti egységek dolgozóinak ezért célszerű megtiltani az okos telefonok használatát. Sok vállalatnál a látogatóktól is elkérik belépéskor a mobil eszközöket.

A mobil eszközök példája mutatja, hogy folyamatosan figyelni kell a technika fejlődését, mert azzal újabb veszélyforrások jelenhetnek meg.

3.4 A veszélyeztetettséget befolyásoló tényezők

A korábbi fejezetekben áttekintést adtunk az adatokra leselkedő veszélyekről. A bőséges, de korántsem teljes felsorolás ijesztőnek tűnik. Már első átolvasásra is világos azonban, hogy a veszélyeztető tényezők nem egyforma súllyal jelentkeznek a különböző egyéneknél és szervezeteknél. Most azokat a fő szempontokat foglaljuk össze, amelyeket elsősorban kell figyelembe venni a veszélyeztetés mértékénél. Az elemzés azért fontos, mert csak ez után lehet a szükséges védelmi intézkedésekről dönteni.

A veszélyeztetettség mértéke szempontjából a legfontosabb szempont a *felhasználó vagy a szervezet tevékenysége* és – ami ezzel szorosan összefügg – az adataik értéke. A támadók különösen kedvelt célpontjai a pénzintézetek és az állam- vagy szolgálati titkokat kezelő szervezetek. Olyan adatokkal foglalkoznak ezek, amelyek a támadók birtokába jutva könnyen felhasználhatóak saját céljaikra vagy értékesíthetőek harmadik félnek. Nevezett szervezettípusok fenyegetettsége nem az információs társadalom terméke, mindig is a rablók és kémek fő célpontjai voltak. Az erősen veszélyeztetett kategóriába tartoznak a digitális információ előállításával és terjesztésével foglalkozó szervezetek is. Évtizedek óta folyik a harc például a könnyűzenei és a filmipar, valamint az alkotásaikat illetéktelenül másolók és sokszorosítók között.

A különleges adatokat kezelő szervezetek, például kórházak, pártok, egyházi közösségek stb., nem az anyagi nyerészkedés miatt védendőek, hanem mert az adataik a pácienseikről vagy tagjaikról senki másra nem tartozó, személyes információkat hordoznak.

Kevésbé veszélyeztetett kategóriába tartoznak a publikus adatokat tároló szervezetek, például a könyvtárak és archívumok, valamint az oktatási intézmények. Azok a felsőoktatási intézmények is különleges figyelmet érdemelnek, ahol informatikusokat is oktatnak. Ezek a hallgatók ugyanis nagyon jól ismerik az informatikai rendszereket és tanulmányaik vagy önképzésük során megismerik azok gyengeségeit is. Kíváncsiságból vagy virtusból ki is próbálják, hogy a megszerzett ismeretek működnek-e a gyakorlatban. Ha ezek a kísérletek szabályozott körülmények között folynak, akkor nagyon hasznosak és a rendszerek hibáinak kijavításához vezetnek. Gyengébb erkölcsű fiatalok a hekkerek táborát szaporíthatják.

A támadás várható mértéke függ a szervezet *méretétől* és *ismertségétől*. Egy nagyobb, országos hatáskörű szervezet, például egy nagy pénzintézet központja vagy egy minisztérium, sokkal több és értékesebb információt kezel, mint a pénzintézet fiókja vagy egy kis település önkormányzata. A médiában rendszeresen szereplő szervezetek és személyek nemcsak nagyobb forgalmat várhatnak ismertségüktől, hanem veszélyeztetettségük is nő. A potenciális támadók ugyanis tőlük nagyobb és értékesebb zsákmányra számítanak.

Az otthoni számítógépek sem tartalmaznak általában olyan információkat, amelyek miatt a támadóknak érdemes lenne komoly erőfeszítéseket tenni azok feltörésére. Ellenük a sok kicsi sokra megy elvet érvényesítve nem is egyedi, hanem automatizált, tömeges támadást intéznek. Egyszerű, de könnyen terjeszthető eszközökkel, tipikusan trójaiakkal és kéretlen levelekkel, árasztják el őket arra számítva, hogy néhány hiszékeny vagy figyelmetlen felhasználó áldozatul esik a támadásnak. Siker esetén kiürítik az áldozat bankszámláját vagy bekényszerítik a számítógépüket egy zombihálózatba.

A szervezet *elhelyezése* is fontos veszélyeztetettségi faktor. Nem mindegy, hogy milyen településen és azon belül más, hasonló szervezet közelében vagy távol van a telephely. Régi tapasztalat, hogy a hasonló tevékenységet folytató vállalkozások és intézmények jobban járnak, ha egymás közelébe, sokszor egy utcába települnek. Így alakultak ki a nagyvárosok adminisztratív, kereskedelmi és pénzügyi központjai. Ez egyszerűbbé és olcsóbbá teszi a szükséges infrastruktúra kialakítását és a telephelyek védelmét is.

A telephelyen belül nagyon fontos a koncentrált információ feldolgozó és tároló kapacitás, tehát a szerverek és az adattárolók megfelelő elhelyezése. A fizikai behatolás megnehezítése miatt ezeket célszerű egy központi épület magjában elhelyezni. Olyan helyre tehát, ahol például falbontással nem lehet hozzájuk közvetlenül hozzáférni. Ügyelni kell arra is, hogy más fizikai behatástól is a lehető legjobban védve legyenek.

A hálózatra nem kapcsolt számítógépeken tárolt adatokhoz távolról nem lehet hozzáférni, így lényegesen védettebbek, mint az internetre csatlakozóak. Ezzel azonban elveszítjük az internet használatának előnyeit is, ami napjainkban általában megengedhetetlen. A minősített digitális aláírás létrehozásához szükséges hitelesítés szolgáltatók tárolóinak egy része nincs hálózatra kötve, hanem más adathordozókon keresztül történik adataik frissítése és aktualizálása.

Az emberi veszélyforrásokra az előzőekben különös figyelmet szenteltünk. Arra is felhívtuk a figyelmet, hogy a *dolgozók képzettsége* és folyamatos tréningje csökkenti tévedéseik számát. Nagyon fontos, hogy a dolgozók tudatában legyenek az általuk kezelt adatok értékével és a munkahelyük adatvédelmi követelményeivel. Tudatos felhasználók legyenek, akiknek a módosítási javaslatait meghallgatják és figyelembe is veszik. Legyenek felkészítve a váratlan helyzetekre.

A jelszavas azonosítás ma már általánosan elterjedt, azonban még mindig gyakori a gyenge, könnyen kitalálható jelszavak használata, illetve a könnyelmű jelszókezelés. Utóbbi alatt azt értjük, hogy a felhasználó a jelszavát mások számára is hozzáférhető helyre, például a monitorára írja.

A dolgozók, beosztásuknak megfelelő szinten, legyenek tudatában a social engineering típusú támadások jellegével és az ilyenek elleni védekezés módszereivel. Ne adjanak ki még ártalmatlannak tűnő információt sem a szervezet működéséről. A központi erőforrásokat kezelő személyzetnek feddhetetlennek kell lenni.

A vállalkozás illetve intézmény *szervezeti felépítése* is lényegesen befolyásolja a kockázati tényezők súlyát, támadás esetén pedig a reakció gyorsaságát és hatékonyságát. Ez a faktor természetesen csak nagyobb szervezetek esetén játszik lényeges szerepet. Sok telephellyel rendelkező, széttagolt szervezetek biztonságát sokkal nehezebb garantálni, mint a

kompakt szervezetekét. Az előbbi esetben például az egységek közötti információforgalom lebonyolítására elsősorban a nyilvános hálózat jöhet szóba, mert saját hálózat kiépítése és üzemeltetése igen költséges. Ez technikailag lehetséges, de a nagy adattömeg kódolása időigényes feladat és a szükséges mennyiségű kód menedzselése komoly veszélyforrás. Nagy adattömeg mozgatása is időigényes feladat. Az információfeldolgozást és az adatvédelmi tevékenységet tehát célszerű decentralizálni.

Végezetül, de nem utolsó sorban a kockázati tényezők súlya függ a vállalat illetve intézmény *szervezettségétől* is. Természetesen ez a faktor is csak nagy szervezetek esetén játszik értékelendő szerepet. Hasonló méretű szervezetek veszélyeztetettsége is különböző lehet attól függően, hogy a szervezet mennyire áttekinthető; a jogosultságok és felelősségek mennyire pontosan és körültekintően vannak meghatározva. Még jól szervezett vállalatoknál is bonyolult a felhasználók szerepköreinek és a rendszerhez való hozzáférési jogosultságainak kiosztása. Kusza szervezetek esetén ez súlyos következménnyel járó, hibás döntésekhez vezethet.

Az informatikai rendszerek bevezetése nagyobb szervezeteknél általában szigetszerűen történt. Az egyes rendszerek egymással párhuzamosan működtek és papíron cseréltek információt. A fejlesztések az egységek vezetőinek hatáskörébe és érdekeltségébe tartozott. Ott történt komoly fejlesztés, ahol a vezető azt fontosnak találta. Egy szervezeten belül is sokféle, egymással nem kompatibilis rendszer jött létre. A technikai fejlődés lehetővé, a racionális gazdálkodás pedig szükségessé tette egyre több autonóm rendszer működésének összehangolását. Mindez a vállalati szervezetre is hatást gyakorolt, az informatikával és azzal összefüggésben az informatikai biztonsággal, kapcsolatos döntések egyre nagyobb léptékűek lettek, egyre magasabb szinten hozták meg azokat. Modern nagyvállalatoknál az informatikai biztonságért felelős vezető a hierarchia magas szintjén van, sokszor az elsőszámú vezető közvetlen beosztottja. Ez megfelelő súlyt ad döntéseihez, amire szükség is van, hiszen azok a szervezet egészének működésére vannak kihatással.

Természetesen napjainkban is bőségesen találkozhatunk informatikai szempontból kevésbé fejlett szervezetekkel. Vannak olyanok, amelyek fejlődésük során eljutottak egy bizonyos szintre, de ott meg is álltak. Mások még nem jutottak el az informatika alkalmazásában addig a nagyságrendig, amikor már központi kezelésbe kerül ez a terület. A felsőoktatásra általában még a szigetszerű fejlesztések jellemzőek. A gazdálkodási, az igazgatási, a tanulmányi és a könyvtári rendszerek fejlesztése egymástól függetlenül történik, alig van információcsere a rendszerek között. Hasonló a helyzet az államigazgatásban és az önkormányzatoknál. Utóbbi esetben tovább bonyolítja a helyzetet, hogy a kis településeken működő önkormányzatoknál nem is gazdaságos önálló informatikai rendszert és adatbázisokat kiépíteni és üzemeltetni. Több település által közösen működtetett informatikai rendszert alig találhatunk.

3.5 Vezeték nélküli hálózatok

A vezeték nélküli személyi hálózatok tipikusan kis hatótávval rendelkeznek, működésük többnyire egy asztali számítógép környezetére (rendszerhálózatok) vagy egyetlen személy mozgásterére korlátozódik. Tipikus alkalmazásai a számítógép perifériákkal való vezeték nélküli összeköttetése illetve hordozható eszközök asztali számítógéppel való szinkronizálása. Mind a perifériák mind a hordozható eszközök korlátozott erőforrásokkal rendelkeznek, ez pedig erősen behatárolja az alkalmazott adóvevő hatótávját és átviteli sebességét is. Vezeték nélküli személyi hálózatokat megvalósító szabványos technológia a Bluetooth. Az átviteli közeg általános hozzáférhetőségének ellensúlyozására erős szimmetrikus titkosítást használ. A kis hatótávból adódóan ezen rendszerek védelme elsősorban a social engineering és a fizikai biztonság tárgykörébe tartozik.

A vezeték nélküli lokális hálózatok (WLAN - Wireless Local Area Network) mind felhasználás, hatótáv és összetevők tekintetében megfelel a hagyományos vezetékes Ethernet hálózatoknak.

Minden vezeték nélküli hálózat esetében több olyan fenyegetés is jelen van, amivel a vezetékes hálózatok esetén nem kell számolni. Gyakorlatilag bárki, aki rendelkezik vevőkészülékkel a hatókörzeten belül, belehallgathat az adásba. Továbbá ha a támadó rendelkezik a hatósugáron belül egy megfelelően beállított rádióadóval, írhat a csatornára, ezáltal módosíthatja, illetve újra elküldheti a már azonosított résztvevőktől származó kereteket.

Ez és a hálózat dinamikusan változó jellege (a hálózat összetétele folyamatosan változik és a hálózat elemei is változtathatják helyzetüket) együttesen kivételesen kedvező környezetet teremt egy potenciális támadó számára.

A vezeték nélküli lokális hálózatok megvalósítására több megoldás született, kezdetben bármiféle védelem nélkül, később gyenge biztonsági megoldásokkal. A vezeték nélküli lokális hálózatok esetén felmerülnek mindazok a biztonsági kérdések, amik a vezeték nélküli hálózatok esetében tipikusnak mondhatók, továbbá a szabvány tervezési hibáit kihasználó programok sokasága áll rendelkezésre (többségük ingyenesen letölthető az internetről). A WPA2 (Work Progress Administration) megfelelő üzemmódban üzemeltetve kielégítő biztonságot nyújthat.

A vezeték nélküli hálózati problémák miatt sok rendszerben egyszerűen kikapcsolják a vezeték nélküli hálózat védelmét, és a vezeték nélküli rendszerből csak akkor engedélyeznek hozzáférést bármilyen erőforráshoz, ha az egy VPN (Virtual Private Network - virtuális magánhálózat) alagúton keresztül történik. A gyakorlatban azonban a hordozható állomások jellegénél fogva nem részesülnek a rendszeres verziófrissítésekben, így nagyobb valószínűséggel rendelkeznek szoftvereik ismert sebezhetőséggel. A támadó a kliensgépeket támadva megszerezheti azok tanúsítványait és nyerhet jogosulatlan hozzáférést a rendszerhez.

A vezeték nélküli városi hálózatok (WMAN - Wireless Metropolitan Area Network) szorosan kapcsolódnak az úgynevezett „utolsó kilométer" problémához: a távbeszélő rendszerek liberalizációja után egy új piaci szereplő számára megfizethetetlenül drága volna háztartások millióihoz vezetékes rendszert kiépíteni. Sokkal olcsóbb és egyszerűbb a város közelében felállított antenna felé irányuló antennákat felszerelni az egyes háztartásokban.

A vezeték nélküli városi hálózat erre kínál lehetőséget, továbbá a nagy hatótávolság és átviteli sebesség egymástól viszonylag távol elhelyezkedő telephelyek lokális hálózatainak közvetlen összekötésére is alkalmassá teszi.

A fenti feladatok többnyire épületek hosszabb távú összeköttetéseinek megvalósítását jelentik. Lényeges különbség a lokális hálózatokkal szemben a hálózat statikus felépítése illetve az irányított antennák alkalmazása (ennek megfelelően a kapcsolat típusa is különböző: a WLAN adatszórást, míg a WMAN pont-pont kapcsolatot használ). A hálózat kiterjedése miatt a lokális hálózatokhoz hasonlóan kényelmes célpontot nyújt a potenciális támadók számára. A vezeték nélküli városi hálózatok megvalósításánál is kritikus fontosságú a megfelelő biztonsági intézkedések foganatosítása a jogosulatlan hozzáférések elkerülése végett.

Nagykiterjedésű hálózatoknak nevezzük az egész országokra vagy földrészekre kiterjedő hálózatok. Ilyen nagy méretekben a lefedettség megoldása, az architektúra kiépítése és üzemeltetése, karbantartása meglehetősen nagy költségekkel jár. Ezeknél a rendszereknél a költségek tipikusan nagy számú felhasználó között oszlanak meg. Tipikus példát jelentenek erre a mobiltelefon rendszerek. Ezen hálózatok esetében is fennállnak a továbbító közeg általános hozzáférhetőségéből adódó fenyegetések. Csakúgy, mint a kisebb méretű rendszereknél, itt is megpróbálkoztak titkosítással kizárni az illetéktelen résztvevőket a hálózatból, de csakúgy, mint a többi esetben, itt is kudarcot vallottak. A legelterjedtebb mobil-hálózat, a GSM esetében például szintén a tipikus problémák lépnek fel: a kölcsönös azonosítás hiánya lehetővé teszi a közbeékelődéses támadást, a titkosító algoritmusa pedig gyenge: az A5 három különböző, különböző időzítésű LFBR (linear feedback register) kimenetét kizáró vagy művelettel egyesítve állítja elő a kulcsfolyamot. Az első két műveletet kimerítő kulcstámadással támadva és a harmadik műveletet az első kettőből számolva a titkosító algoritmus feltörhető. További fenyegetések lépnek fel hálózatközi kommunikáció (roaming) és a klienseszköz (mobiltelefon) kis mérete következtében. A GSM szabvány az elsőt teljeséggel figyelmen kívül hagyja, a második ellen pedig sikertelenül próbál védekezni: a SIM kártyán található szimmetrikus kulcsot védő A3 és A8 algoritmusok által nyújtott védelem szintén kijátszható. A 3G az említett gyengeségek nagy részét orvosolja. További aggodalomra ad okot a számlázási adatok kérdése. A hálózat tulajdonosa számlázási adatok nyilvántartása címén nagy mennyiségű kényes, privát információt tárol a felhasználókról.

3.6 Tűzfalak

A tűzfal eredetileg az egyes épületekben esetlegesen kitörő tüzeket korlátozó falakat jelentette ám később a mérnöki nyelvben az egyes gépezetek illetve járművek hasonló feladatokat ellátó elemeit is annak nevezték.

A tűzfalak olyan szoftver vagy hardverelemek, amelyek a hálózat egy részének, jellemzően egy intranetnek a külső támadók elleni védelmére hivatottak. Mint legtöbb biztonsági intézkedés illetve eszköz, a tűzfal is erőforrásokat igényel: az egyes tűzfalak

típusuktól függően kisebb-nagyobb mértékben késleltetik a külső hálózat és a tűzfal által védett hálózatrész közötti kommunikációt.

A tűzfalak az általuk védett rendszer és a külvilág között állnak. Minden kimenő és bejövő kommunikáció rajtuk keresztül halad át. A tűzfal a rajta áthaladó csomagok közül csak azokat engedi át, amelyeket a beletáplált szabályoknak megfelelően veszélytelennek ítél. Ezt nevezzük csomagszűrésnek. Veszélyesnek ítélt csomagok esetén minden figyelmeztetés vagy visszajelzés nélkül kiszűri, elnyeli az adott csomagot.

A legegyszerűbb esetben ez annyit jelent, hogy megadhatjuk, hogy mely portokra, milyen típusú forgalmat engedélyezünk (bemenő, kimenő, TCP, UDP). Ez a típusú csomagszűrés jellemzően csak a hálózati protokollverem alsó három rétegét érinti. Az egyes alkalmazások vagy protokollok tiltása vagy engedélyezése az alkalmazás vagy a protokoll által használt standard portok szűrésével történik. Ha például az adott alkalmazás mindkét oldalon az alapértelmezettől eltérő portokat használ, akkor azzal meg tudja kerülni a tűzfal korlátozó szabályozásait.

Az úgynevezett alkalmazás tűzfalak, a hálózati protokollverem alkalmazás rétegének szintjén tevékenykednek. Jellegüknél fogva hozzáférnek az adott alkalmazás teljes forgalmához és betekinthetnek az adott protokoll által közvetített információtömegbe is. Ez nem csak az átmenő tartalom megszűrését teszi lehetővé (például egy adott projektre, személyre vagy szerződésre vonatkozó bármilyen információ kivitelét), de az ismert alkalmazás specifikus hibákat kihasználó kódrészletek kiszűrésére is módot ad. Az alkalmazás tűzfal lehet hálózatalapú vagy operációs rendszer szintű. Az operációs rendszer szintű alkalmazástűzfal nem a hálózati protokollverem alkalmazás rétegében operál, hanem az adott operációs rendszeren rendszerhívásokkal kommunikál a védett alkalmazással.

Az alkalmazás tűzfalak használata, mint ahogy nevük is mutatja, gyakran egy adott alkalmazáshoz vagy szolgáltatáshoz és nem a rendszer egészéhez kapcsolódik. Az adott szolgáltatás vagy alkalmazás megvédésére, a hozzáférés sokkal pontosabb szabályozására ad lehetőséget.

Az alkalmazás tűzfalak legnagyobb hátránya, hogy a bonyolult, magas szintű vizsgálatok, amik lehetővé teszik a pontosabb szabályozást és az alaposabb szűrést nagyobb erőforrásokat igényelnek és a kommunikációt is jobban késleltetik.

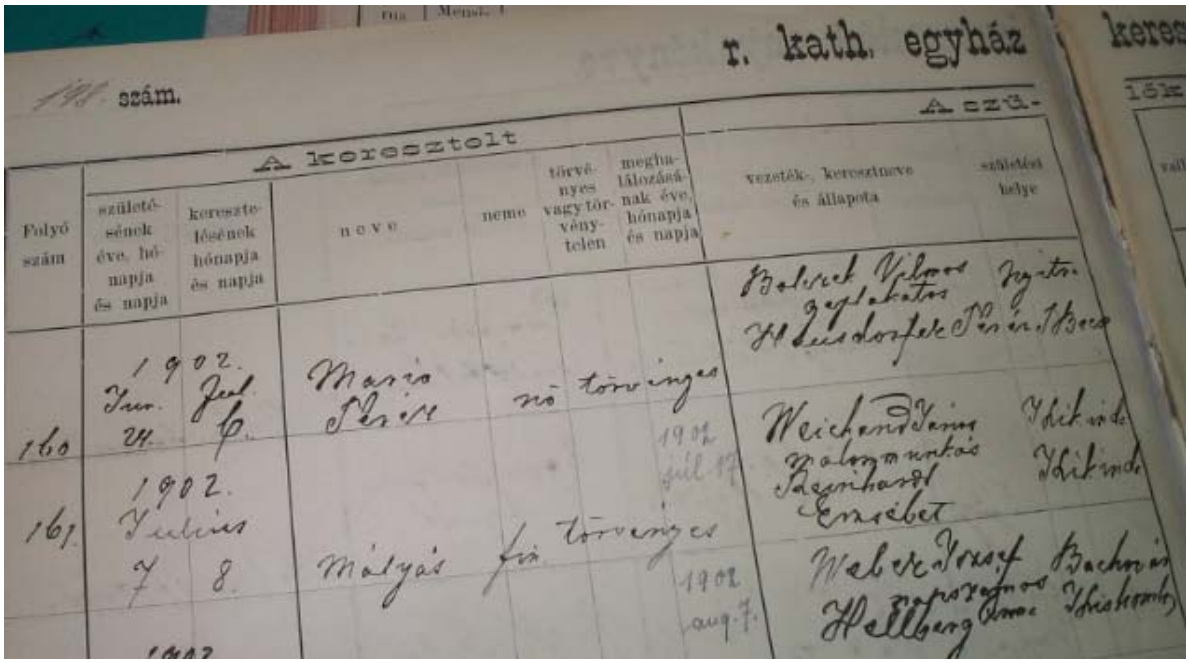
Azok a tűzfalak, amik csak az OSI modell három alsó rétegében tevékenykednek, sokkal kisebb erőforrásokat igényelnek, és kisebb késleltetéssel járnak. Ezt a hatékonyságot ötvözik a nagyobb fokú biztonsággal, az állapottal rendelkező csomagszűrő tűzfalak. Továbbra is csak az alsó három rétegben működnek, de nyilvántartják az egyes kapcsolatok aktuális állapotát és az adott fázisban érvénytelen vagy értelmetlen csomagokat kiszűrik. Mindehhez az összes éppen aktív kapcsolatot illetve azok állapotát nyilván kell tartaniuk. Hogy a belső kapcsolattáblájuk betelését megelőzzék, a túl hosszú ideig inaktív kapcsolatokat megszakítják. A kapcsolatokat nyilvántartó táblázat szándékos telítésével válaszképtelen állapotban lehet tartani a teljes védett rendszert, ez az alapja az ilyen tűzfalak ellen irányuló DoS támadásoknak.

A tűzfalak egy része NAT (Network Address Translation - Hálózati Cím Fordítás) funkcióval is rendelkezik. Ezt az eljárást elsősorban az ipv4 címterének szűkössége miatt

dolgozták ki, de a védelem szempontjából is jelentős hatással bír. A támadók csak a rendszer azon részei ellen léphetnek, amit "látnak", azaz meg tudnak célozni, címezni. A NAT elrejti a külvilág elől a rendszer elemeinek címeit így jelentősen megnehezíti a támadás első fázisát, a hálózati felderítést.

4 Adatbiztonság szabályozása, magyar törvények

Polgári társadalmakban a természetes és jogi személyek kötelességeit, tevékenységeik korlátait és egymással való együttműködéseik keretét, legmagasabb szinten, törvényekben fogalmazzák meg. A kézzel és géppel írott valamint nyomtatott dokumentumok kezelése és hitelességének biztosítása régen bekerült a törvénnyel szabályozandó témák közé. Életünk során nagyon sok, fontos dokumentum készül rólunk. Identitásunkat végső soron az anyakönyv bizonyítja, erről készül a születési anyakönyvi kivonat. A 4.1 ábrán egy 1902-ben készült anyakönyvi bejegyzést látunk. Tanulmányaink eredményét a képző helyek által kiállított bizonyítványok, nyelvtudásunkat a nyelvvizsga bizonyítvány igazolja, hogy csak néhány példát említsünk.



4.1 ábra

A jogi személyek identitását az alapító okirat bizonyítja. Ez rögzíti a tulajdonosok és a képviselők személyét. Működésük során folyamatosan keletkeznek hozzájuk kapcsolódó szerződések, számlák és még számtalan különböző dokumentum. Témánk szempontjából különösen fontosak a szerződések, hiszen konkrét esetben azok határozzák meg a felek jogait és kötelezettségeit. Vitás esetben csak hiteles szerződés alapján dönthet harmadik fél, általában a bíróság a felek között.

Az informatika és a szórakoztató elektronika fejlődése következtében az 1970-as években terjedt el adatok tömeges elektronikus rögzítése. A fejlődés motorja a szórakoztató elektronika volt és főként a mágnesszalagokon tárolt hangfelvételeket és filmeket jelentette.

Ezek döntően analóg formában tárolták a jeleket. A számítógépek ezzel szemben digitális technikával dolgoztak. A két terület fejlődése alapvetően változtatta meg a dokumentumok fizikai megjelenését. A megfogható, lapozható, aláírható, lepecsételhető papírlapokról nagyon gyorsan átkerültek a számítógép memóriájába. Virtuális alakjuk persze materializálódhat is, de csak külön kívánságra. Jelen tananyag is virtuális formában készül. Talán nyomtatnak belőle néhány példányt, de az olvasókhoz főleg digitális formában jut majd el.

Az új technológia sokkal egyszerűbbé és olcsóbbá tette például a hangfelvételek és a szoftverek tömeggyártását. Az elektronikusan rögzített – analóg és digitális - adatokat azonban könnyű másolni is. Erre persze a felhasználók is rájöttek és megindult az adathordozók magáncélú másolása, amely sokszor ipari méreteket is öltött. A szerzői jog klasszikus szabályozási mechanizmusait alaposan át kellett dolgozni ahhoz, hogy az új helyzetben is hatékonyan védje a szerzők és kiadók jogos érdekeit, de ne sértse a termékeket legálisan megszerzők érdekeit sem.

Nagyjából ugyanebben az időben kezdték felismerni a polgárok, hogy egyre több adatot gyűjtenek róluk állami szervezetek és magánvállalkozások. Követhetetlen volt az adatok sorsa, egyre több emberben tudatosodott, hogy a kialakuló nagy adatbázisok lehetővé teszik bármelyik állampolgár kapcsolatrendszerének, egészségi állapotának, szokásainak és anyagi helyzetének feltérképezését. Erősödött az igény, hogy a személyes adatok gyűjtését és felhasználását törvény szabályozza. Nem az adatok védelme jelentette tehát az újdonságot, azt már régen gyakorolták, néhány példával rögtön szolgálunk. Új jelenség a személyes adatok felértékelődése volt.

Székely Iván társadalmi informatikus a következőképpen foglalja össze a rendszerváltás után, az adatvédelem szempontjából leginkább meghatározó tényezőket: „Egyfelől […] alapvetően átalakult az állam információkezelő rendszere. Másfelől […] egy új információkezelő szektor nőtt fel az állami mellé: a nagy magáncégeké, a bankoké, a biztosítóké és egyéb vállalatoké. Úgy is mondhatnám, hogy a Nagy Testvérnek hirtelen nőtt egy második feje. Végül, a harmadik változás: igen nagy sebességgel megtörtént egy nagyszabású technikai modernizáció.”[7]

Bizonyos adatok védelme több évezredes múlttal rendelkezik. Hadvezérek például az utasításaikat tikosítva és futárral juttatták el az alvezéreknek. Julius Caesarnak tulajdonítják például az egyik egyszerű szimmetrikus titkosítást. A XVIII. század végén Thomas Jefferson (1743. április 13. – 1826. július 4.) az USA harmadik elnöke készített mechanikus titkosító eszközt, a Jefferson-kereket. A sort hosszan folytathatnánk, de akkor eltérnénk a fejezet témájától az adatvédelem szabályozásától.

Az állam és vállalati titkok bizalmas kezelésének szabályait régen kidolgozták és a változó környezetben aktualizálták. Fontos döntések előkészítésének dokumentumai, az erőszakszervezetek működése, ellátottsága, de akár a stratégiai szempontból fontos infrastruktúra elhelyezkedése is államtitkot jelenthettek. A vállalatok is védik bizalmas

[7] Talyigás Judit, E-világi beszélgetések.hu, Pesto Kiadó, 2003. Székely Iván, A történelemben lesz egy lyuk, 20.

adataikat. Ebbe a körbe a kutatási eredmények, új termékek készítésének technológiája, terve, stb. tartozik.

A legfontosabb informatikával kapcsolatos *büntetőjogi* rendelkezéseket három csoportba sorolhatjuk. Az első csoportban azok a bűncselekmények vannak, melyek eszköze maga az informatika. A másodikban az informatika nem az eszköz, hanem a bűncselekmény tárgya. Ide tartozik a Btk. 300/C. § (1) Aki számítástechnikai rendszerbe a számítástechnikai rendszer védelmét szolgáló intézkedés megsértésével vagy kijátszásával jogosulatlanul belép, vagy a belépési jogosultsága kereteit túllépve, illetőleg azt megsértve bent marad, vétséget követ el, és egy évig terjedő szabadságvesztéssel, közérdekű munkával vagy pénzbüntetéssel büntetendő.

A Büntető Törvénykönyv ezen, paragrafusa tartalmazza a jogosulatlan hozzáférés mellett a számítógépes rendszerek adatainak jogosulatlan bevitelével, módosításával, törlésével kapcsolatos eljárást.

A Btk. 300/E. § a rendszerbe való belépéshez szükséges jelszavak, kódok előállításával, megszerzésével és kereskedésével kapcsolatos bűncselekmények is bővebb kifejtésre kerülnek, amelyek lényegében a cracker tevékenységet fogalmazzák meg. A harmadik csoportba a szellemi tulajdon tárgyát képező információ ellen irányuló bűncselekmények tartoznak.

Az adatvédelem és az elektronikus szolgáltatások legmagasabb szintű, jogi szabályozása a törvényekben található. Nagyon szerteágazó problémakörről van szó, ami mutatja, hogy az elektronikus szolgáltatások lehetősége alaposan átformálja az állam- és a közigazgatást. Az alábbiakban felsoroljuk a legfontosabb releváns törvényeket és az elektronikus aláírást részletesen szabályozó két Kormányrendeletet.

- 1992. évi LXIII. tv. a személyes adatok védelméről és közérdekű adatok nyilvánosságra hozataláról
- 1995. évi LXV. tv. az államtitokról és a szolgálati titokról
- 1995. évi CXXV. tv. a Nemzetbiztonsági szolgálatokról
- 1996. évi LVII. tv. a tisztességtelen piaci magatartás és versenykorlátozás tilalmáról (üzleti titok védelme)
- 1997. évi CXLV. törvény és az ezt módosító 2003. évi LXXXI. Törvény az elektronikus cégeljárásról és a cégiratok elektronikus úton történő megismeréséről
- 1998. LXXXV. tv. A Nemzeti Biztonsági Felügyeletekről
- 2001. évi XXXV. tv. Az elektronikus aláírásról
- 1996. évi CXII. tv. a hitelintézetekről és pénzügyi vállalkozásokról (banktitok, az értékpapírtitok stb. védelme)
- 2001. évi XXXV. tv. az elektronikus aláírás (digitális aláírás, Hitelesítési Hatóság CA) jogi szabályozása
- 2001. évi CVIII. tv. az elektronikus szolgáltatások
- 78/2010. sz. Kormány rendelet az elektronikus aláírás közigazgatási használatához kapcsolódó követelményekről és az elektronikus kapcsolattartás egyes szabályairól

4.1 Az adatvédelmi törvény

Az 1992. évi LXIII. tv. a személyes adatok védelméről és közérdekű adatok nyilvánosságra hozataláról című törvényt nevezi a köznyelv röviden adatvédelmi törvénynek. Fontosságát és elterjedtségét már az is mutatja, hogy van köznyelvi elnevezése. A törvény célja annak biztosítása, hogy - néhány kivételtől eltekintve - személyes adataival mindenki maga rendelkezzen és a közérdekű adatokat mindenki megismerhesse. Hatálya csak a természetes személyek adataira terjed ki. Informatikai biztonsági szempontból a személyes adatok védelméről szóló passzusok érdekesek, ezért a közérdekű adatokkal foglalkozókkal nem foglalkozunk.

A törvény definiálja a

1.1 személyes adat
1.2 különleges adat
1.3 bűnügyi személyes adat
1.4 közérdekű adat és
1.5 közérdekből nyilvános adat

fogalmát. A különleges adatokat tételesen is felsorolja, ezek: a faji eredetre, a nemzeti és etnikai kisebbséghez tartozásra, a politikai véleményre vagy pártállásra, a vallásos vagy más világnézeti meggyőződésre, az érdekképviseleti tagságra, az egészségi állapotra, a kóros szenvedélyre, a szexuális életre vonatkozó adat, valamint a bűnügyi személyes adat. A többi esetben csak körülírja a megfelelő fogalmat. A jövedelem és a tulajdon általában személyes adat, de közszereplők, például parlamenti, önkormányzati képviselők, magas rangú állami tisztviselők, esetén más törvény a közérdekből nyilvános adat körébe utalja.

Adatkezelésnek számít az adatokon végrehajtott mindenféle művelet, beleértve azok védelmét is. Személyes vagy különleges adat csak akkor kezelhető, ha ahhoz az érintett hozzájárul vagy törvény, illetve önkormányzati rendelet írja elő. Adatkezelés csak meghatározott célból és csak az elengedhetetlenül szükséges mértékben végezhető. Ezen kívül meg lehet határozni az adatkezelés időtartamát is. Ez lehet határozott időtartam vagy a cél megvalósulásától függő is. Az adóbevallásokkal kapcsolatos adatokat például 5 évig kell megőrizni, a munkavállaló adatait addig, amíg a munkaviszonya fennáll.

Az adatkezelés célját és a rögzített adatok körét közölni kell az érintettel. Tájékoztatni kell az érintettet arról is, hogy mely adatok szolgáltatása kötelező és melyeket lehet önkéntesen megadni. Amennyiben valaki nem adja meg a kötelezően előírt adatokat, akkor meghiúsul a közte és az adatkezelő közötti kapcsolat. Opcionális adatok közlése következmény nélkül megtagadható. Ha például valaki munkát vállal, akkor közölni kell vele, hogy a munkaadó milyen adatokat rögzít róla. Amennyiben az adatok körét túlságosan bőnek ítéli, elállhat a szerződéstől. Egészségügyi intézményben és élelmiszer kereskedésben megkövetelik a dolgozók egészségi állapotára vonatkozó adatok tárolását és folyamatos frissítését. Hasonló igény informatikai vállalkozásoknál nem jogos. Számlanyitáskor kérhetik az aláírás mintát és a személyi igazolvány másolatát. Utóbbihoz nem szükséges hozzájárulni.

Az *adattovábbítás* is része az adatkezelésnek, így az általános rendelkezések erre a műveletre is vonatkoznak. Bizonyos esetben azonban szükség lehet arra, hogy adatokat

külföldre továbbítsanak. Külföldön is köthetünk házasságot, vállalhatunk munkát, lehetünk betegek vagy okozhatunk balesetet, hogy csak néhány példát említsünk. A törvény szempontjából az EGT országaira ugyanolyan megítélés vonatkozik, mintha a továbbítás hazánk területén belül történt volna. EGT-államok jelenleg az Európai Unió tagjai valamint Izland, Liechtenstein és Norvégia. Az EGT-n kívüli országokba adat csak akkor továbbítható, ha ahhoz az érintett hozzájárul vagy ott biztosított az átadott adatok megfelelő szintű védelme.

Az állampolgár bármikor tájékoztatást kérhet a személyes adatai kezeléséről. Ellenőrizheti a kezelt adatai pontosságát és szükség esetén azok helyesbítését kérheti. Tájékoztatást kell kapnia arról, hogy hová és milyen céllal továbbították az adatait. Ha az adatkezelést nem törvény vagy önkormányzati rendelet írja elő, akkor kérheti adatainak törlését is. Annak következményét, például az adatkezelővel kötött szerződése felmondását, viselnie kell.

Amennyiben az adatkezelés határideje elérkezik vagy - határozatlan idejű adatkezelés esetén - a célja megvalósul az adatokat meg kell semmisíteni.

Az adatvédelmi törvény hozta létre az *adatvédelmi biztos* intézményét és rendelkezett az *adatvédelmi nyilvántartás* szabályairól. Az adatvédelmi biztos feladata az adatvédelmi törvény és más adatkezeléssel kapcsolatos jogszabályok megtartásának ellenőrzése. Figyelemmel követi a terület fejlődését és szükség esetén a törvények végrehajtására, esetleg azok módosítására ajánlásokat fogad el. Ő kezeli az adatvédelmi nyilvántartást, amelybe a személyes adatokat kezelő köteles az adatkezelés megkezdése előtt bejelenteni az adatkezelés legfontosabb adatait. A biztos az adatkezelés megkezdése előtt ellenőrizheti az adatkezelés jogalapjának és biztonságos végrehajtása feltételeinek meglétét.

Végezetül a törvény rendelkezik arról, hogy az országos hatósági, munkaügyi vagy bűnügyi adatállományt kezelő, illetőleg feldolgozó adatkezelőnél és adatfeldolgozónál; a pénzügyi szervezetnél és a távközlési és közüzemi szolgáltatónál megfelelő végzettséggel rendelkező *belső adatvédelmi felelőst* kell kinevezni illetve megbízni. Ezen felül a felsorolt szervezeteknél és egyéb állami és önkormányzati adatkezelőknek *adatvédelmi és adatbiztonsági szabályzatot* kell készíteni.

4.2 Az elektronikus aláírásról szóló törvény

Dokumentumok hitelesítése nagyon fontos aktus a polgári társadalmakban. A társadalom szereplőinek egymáshoz és a közigazgatáshoz való konkrét viszonyát ugyanis nyilatkozatokban és szerződésekben fogalmazzák meg. Amennyiben vita támad a felek között a vállalt kötelezettségek teljesítése miatt, akkor független bírósághoz fordulhatnak. A bíróságok azonban csak olyan dokumentumokat használhatnak fel döntéseikben, amelyek hitelesek. A papír alakú dokumentumokat pecséttel és/vagy aláírással hitelesítették. Leckekönyvükben a vizsgajegy csak akkor érvényes, ha az oktató aláírta. Az orvos aláírása és személyes pecsétje nélkül a recept érvénytelen. Példáink sorát folytathatnánk, de már ennyi is eléggé illusztrálja az aláírás fontosságát és azt, hogy a társadalmi érintkezés sok területén szükséges az alkalmazása.

A múlt század 80-as éveinek végétől kezdődően a dokumentumok egyre nagyobb hányadát elektronikus úton állították elő. Hitelesítéskor azonban ki kellett nyomtatni azokat és úgy bánni velük, mintha a korábbi technológiával készültek volna. Természetes módon vetődött fel az igény digitális dokumentumok digitális hitelesítésére. A technológiai feltétel 1976 óta rendelkezésre állt és egyre jobban finomult. Ezek részletezésére később térünk vissza. A jogi szabályozás kidolgozása azonban hosszabb időt vett igénybe.

Közel egy évtizedes előkészítő munka után 2001. május 29-én fogadta el az Országgyűlés az elektronikus aláírást szabályozó törvényt. Ezzel hazánk elsők között lépett be azon országok sorába, ahol a törvény erejénél fogva is lehet elektronikus dokumentumokat hitelesíteni. Napjainkban lényegében a világ minden országban van hasonló jogszabály.

Teljes nevén 2001. évi XXXV. (V.26) törvény az elektronikus aláírásról preambuluma világosan megfogalmazza témájának fontosságát, ezért szó szerint idézzük: „Az Országgyűlés – felismerve és követve az egyetemes fejlődésnek az információs társadalom felé mutató irányát, az új évezred egyik legfontosabb kihívásának eleget téve – törvényt alkot az elektronikus aláírásról annak érdekében, hogy megteremtse a hiteles elektronikus nyilatkozattétel, illetőleg adattovábbítás jogszabályi feltételeit az üzleti életben, a közigazgatásban és az információs társadalom által érintett más életviszonyokban.”

Ebben a fejezetben a törvény jogi tartalmára koncentrálunk, a technikai részleteket az 9.2 fejezetben fejtjük ki. A törvény szerint „az elektronikus aláírás: elektronikusan aláírt elektronikus dokumentumhoz azonosítás céljából logikailag hozzárendelt vagy azzal elválaszthatatlanul összekapcsolt elektronikus adat.” A fokozott biztonságú elektronikus aláírás ezen kívül

a) „alkalmas az aláíró azonosítására,
b) egyedülállóan az aláíróhoz köthető,
c) olyan eszközökkel hozták létre, amelyek kizárólag az aláíró befolyása alatt állnak
d) és a dokumentum tartalmához olyan módon kapcsolódik, hogy minden – az aláírás elhelyezését követően a dokumentumban tett – módosítás érzékelhető.”

Végezetül a minősített elektronikus aláírás: „olyan - fokozott biztonságú - elektronikus aláírás, amelyet az aláíró biztonságos aláírás-létrehozó eszközzel hozott létre, és amelynek hitelesítése céljából minősített tanúsítványt bocsátottak ki.”

A három elektronikus aláírás csak a bizonyító erejükben és az abból következő alkalmazási körükben különbözik. Bírósági és közigazgatási hatósági eljárásokban az elektronikusan aláírt dokumentumok ugyanúgy használhatóak és ugyanolyan bizonyító erejűek, mint a hagyományosak. Kivételt képeznek az alapdokumentumnak számító anyakönyvi kivonatok. Ugyanakkor az új típusú aláírás nem tehető kötelezővé. Itt is van egy kivétel, mégpedig az adóbevallások, amelyeket jogi személyeknek elektronikusan kell benyújtani. A hagyományos aláírásnak is vannak fokozatai; „fokozott biztonságú”, ha tanúk előtt készül és „minősített”, ha közjegyző hitelesíti.

Jelenlegi tudásunk szerint biztonságos elektronikus aláírás csak aszimmetrikus titkosító algoritmusokkal készíthető, de a törvény nyitva hagyja annak a lehetőségét, hogy más típusú elektronikus aláírást is felfedezhetnek. Ez kiderül abból, hogy a törvény aláírás-ellenőrző adatról és aláírás-létrehozó adatról szól nem pedig a kriptográfiai szakirodalomban

szokásos nyilvános-titkos kulcspárról. A szükséges kulcspár nyilvános felét a hitelesítés szolgáltatónál kell elhelyezni, a másik felét pedig az aláírónál. A hitelesítés szolgáltató lényegében azt igazolja, hogy az aláírás ellenőrző kulcs egy bizonyos jogi vagy természetes személy tulajdona. Az aláírás minősítése a hitelesítés szolgáltató minősítésétől jelentősen függ. A törvény lehetőséget ad arra a gyakorlatban szokásos megoldásra, hogy a kulcsokat magunk vagy a munkáltatónk hitelesítse. Így azonban nem kaphatunk fokozott biztonságú vagy minősített elektronikus aláírást. A hitelesítés szolgáltató jogot szerezhet további szolgáltatásokra is, úgymint időbélyegzés, elektronikus archiválás valamint aláíró kulcspár generálása és a privát kulcs elhelyezése az aláírást létrehozó eszközön.

Az EGT valamely tagországában bejegyzett székhelyű hitelesítés szolgáltató által kibocsátott tanúsítvány egyenértékű a hazai kibocsátású tanúsítvánnyal.

A törvény részletesen szabályozza a hitelesítés szolgáltatási tevékenység indításának és működésének feltételeit. A szolgáltatás megkezdését be kell jelenteni a felügyelő hatóságnak. A bejelentéshez csatolni kell a szolgáltatási szabályzatot valamint az általános szerződési feltételeket. Hiteles okirat másolatával kell bizonyítani a kérelmező és alkalmazottai büntetlen előéletét és szakképzettségét. Felelősségbiztosítással és megfelelő pénzügyi háttérrel kell rendelkeznie. A hitelesítés szolgáltatás tehát szakképesítéshez kötött, bizalmi tevékenység, amelyet csak megfelelő financiális erőforrással rendelkező szervezet folytathat. Ezt az állapotot működése során mindvégig fenn kell tartania. Érthető a szigorú szabályozás! Az elektronikus aláírás ugyanis – mint arra korábban rámutattunk – ugyanolyan bizonyító erejű, mint a hagyományos. Komoly értékekről folyó vitában lehet döntő jelentősége és a vitában esetleg kiderülhet, hogy az aláírás nem eléggé biztonságos. Ilyenkor a hitelesítés szolgáltatónak kell fizetnie.

Lássunk néhány példát! A digitális aláírás alkalmazható nagy értékű szerződések aláírására. Természetes személyek benyújthatják az adóbevallásaikat elektronikus úton, digitálisan aláírva, jogi személyeknek ez kötelező. A példákat még hosszan sorolhatnánk. Ha a szerződő felek vagy az APEH és az adóalany között vita támad a kötelezettségek teljesítéséről, akkor a vita eldöntésében a digitálisan aláírt dokumentumnak alapvető szerepe van. Ilyenkor persze alaposan megvizsgálják azt is, hogy a digitális aláírás az előírásoknak megfelelően biztonságos-e. Belép tehát az eljárásba harmadik félként a hitelesítés szolgáltató és annak a terméke is. Ha bebizonyítható, hogy az alkalmazott aláírás nem elég biztonságos és a hibát a szolgáltató követte el, akkor neki kell viselni az anyagi felelősség megfelelő részét. Ilyenkor fontos a szolgáltató megfelelő anyagi háttere és felelősségbiztosítása.

A szolgáltató a szerződést megelőzően köteles tájékoztatni az igénybe vevőt a szolgáltatás felhasználási módjáról, biztonsági fokáról és a szerződés feltételeiről. Minősített tanúsítvány szolgáltatása esetén meghatározhatja a felhasználás földrajzi korlátait és az egy aláírással vállalható kötelezettség felső határát. Utóbbi korlátozás a szolgáltatót védi attól a kockázattól, amelyről az előző bekezdésben írtunk.

A hitelesítés szolgáltatás hosszú távú tevékenység. Az aláírt dokumentumok egy részére ugyanis jóval az aláírás után is szükség lehet. Gondoljanak például érettségi bizonyítványukra vagy jövendő diplomájukra, amelyekre életük során bármikor szükség lehet. A digitális aláírás ugyanakkor csak a tanúsítvánnyal együtt érvényes. A szolgáltatónak tehát

biztosítani kell a tanúsítvány archiválását. A törvény úgy rendelkezik, hogy a szolgáltató „a tanúsítványokkal kapcsolatos elektronikus információkat – beleértve az azok előállításával összefüggőeket is – és az ahhoz kapcsolódó személyes adatokat legalább a tanúsítvány érvényességének lejártától számított tíz évig, illetőleg az elektronikus aláírással, illetve az azzal aláírt elektronikus dokumentummal kapcsolatban felmerült jogvita jogerős lezárásáig megőrzi.” A tárolás történhet hitelesített archiválási szolgáltató igénybevételével is. A kötelezettség akkor is fennáll, ha a kapcsolat az ügyfél és a szolgáltató között megszűnik, például az ügyfél más szolgáltatót vesz igénybe vagy a szolgáltató befejezi tevékenységét. Utóbbira még visszatérünk.

A hitelesítés szolgáltató a törvény erejénél fogva jogosult a szolgáltatást igénybe vevő releváns adatait kezelni. Ellenőriznie kell az ügyfél személyazonosságát és az azonosítókat nyilván kell tartania. A személyes adatokat más célra természetesen nem használhatja fel és harmadik félnek az ügyfél beleegyezése nélkül nem adhatja tovább. Kivételt képez az az eset, amikor az elektronikus aláírással kapcsolatos bűncselekmény felderítése vagy megelőzése céljából az arra jogosított szervezetek kérik az adatokat.

Minden szervezetnek - így a hitelesítés szolgáltatóknak is - van életciklusa, mely az alapítással kezdődik, és a megszűnéssel fejeződik be. Korábban rámutattunk, hogy az általa kiadott tanúsítványoknak tovább kell élni. A tevékenység befejezésének szándékáról legalább hatvan nappal a megszűnés előtt értesítést kell küldeni a vele szerződésben álló ügyfeleknek valamint a felügyelő hatóságnak is. A bejelentés után új tanúsítványt már nem bocsáthat ki. A tervezett megszűnés előtt húsz nappal vissza kell vonnia az összes még érvényes tanúsítványt. A szolgáltatónak gondoskodnia kell arról, hogy a nyilvántartásait valamint az archivált tanúsítványokat más, vele azonos besorolású hitelesítés szolgáltató átvegye. Az új szolgáltatóról a felügyelő hatóságot értesíteni kell. A törvény rendelkezik arról is, hogy mi a teendő, ha a megszűnéssel kapcsolatos kötelezettségeinek a szolgáltató nem vagy csak részben tesz eleget. A szabályozás arra törekszik, hogy a megszűnés következtében az ügyfeleknek a lehető legkevesebb kára keletkezzen.

Biztonságos aszimmetrikus kulcspár létrehozása számelméleti algoritmusokkal történik. Ezeket részletesen tárgyaljuk a 9.2. fejezetben. Csak akkor lehetünk biztosak abban, hogy a kulcspárt más nem ismeri, ha azt magunk hoztuk létre. A szerzők véleménye az volt, hogy a kulcspárt a tulajdonosnak kell létrehoznia. Be kellett látniuk azonban, hogy a felhasználók nagy többsége csak könnyen kezelhető szoftverrel képes ilyen kulcspár létrehozására. Egy ilyen szoftver használata semmivel sem eredményezne biztonságosabb aláíró kulcspárt, mintha azt egy korrekt hitelesítés szolgáltató generálja. Másrészt az alkalmazások döntő többségénél az utóbbi megoldás megfelelő biztonságot nyújt a felhasználónak. A minősített hitelesítés szolgáltatás gyakorlatában az terjedt el, hogy a kulcspárt a szolgáltató generálja nem pedig az ügyfél, sőt nem is fogadják el az ügyfél által generált kulcsokat. A nem minősített szolgáltatások egy részénél a titkos kulcsokat az ügyfél a saját gépén generálja.

A törvény a kialakult gyakorlatot a szabályozással is alátámasztja. Hitelesítés szolgáltató elhelyezheti az aláírás létrehozó eszközön az aláírást létrehozó adatot. Ekkor gondoskodnia kell az aláírást létrehozó kulcs titkosságáról és az aláírást ellenőrző adat

sértetlenségéről. A titkos kulcsot a szolgáltatás során olyan módon kell kezelnie, hogy ne legyen visszafejthető, utána pedig meg kell semmisítenie. Tehát a szolgáltatónál sem maradhat meg az aláíró kulcs. Ha az ügyfél aláíró kulcsa elvész vagy megsemmisül, akkor új kulcspárt kell kérnie.

A törvény szabályozza az időbélyegzés és az archiválás szolgáltatást is valamint a felügyelő hatóság feladatait és jogosítványait. Ezekre a jegyzetben nem térünk ki. A közigazgatásban nem elég az elektronikus aláírás általános jogi szabályozása, a technikai részleteket is specifikálni kell. Ez „A közigazgatási hatósági eljárásokban felhasznált elektronikus aláírásokra és az azokhoz tartozó tanúsítványokra, valamint a tanúsítványokat kibocsátó hitelesítés szolgáltatókra vonatkozó követelményekről szóló 78/2010. sz. Kormány rendelet”-ben található. Tekintettel arra, hogy a közigazgatás a digitális aláírás egyik legnagyobb felhasználója, a szabályozás az élet egyéb területein is iránymutató.

5 Ügyviteli védelem.

Az előző fejezetekből már kiderül, hogy az adatok védelme nagyon szerteágazó probléma. Otthoni számítógépeinket is óvni kell a különböző veszélyforrásoktól; nem visszük vizes helységbe, úgy helyezzük el, hogy az áramellátás és a hálózati kapcsolat folyamatos legyen; jogtiszta szoftvereket használunk, és hatékony eszközöket telepítünk a kártékony programok ellen. Óvjuk az általunk készített dokumentumokat az illetéktelen hozzáféréstől.

A vállalatoknak, az állami és önkormányzati szervezeteknek is egyre növekvő feladatot jelent az adatok védelme. Kis szervezeteknél elegendő a szabályok szóban történő ismertetése. Nagy, sok dolgozót alkalmazó szervezeteknél ez ma már nem elegendő. Ilyen szervezeteknél írásban kell megfogalmazni az adatok védelmével kapcsolatos szabályokat. Az ügyviteli védelem az informatikai rendszert üzemeltető szervezet ügymenetébe épített védelmi intézkedések, biztonsági szabályok, tevékenységi formák együttese. A védelemnek ezt a formáját adminisztratív védelemnek is szokták nevezni. A szabályozás alapját a vonatkozó törvények és jogszabályok jelentik, de ezeket a konkrét tevékenységnek megfelelően pontosítani kell.

Az ügyviteli védelemnek két szintjét különböztethetjük meg, mégpedig a stratégiai, tervezési szintet és a mindennapi gyakorlatot érintő és szabályozó szintet. Előbbit az Informatikai Biztonsági Koncepció (IBK), utóbbit az Informatikai Biztonság Szabályzat (IBSz) tartalmazza. A szabályozás két formája szorosan összefügg egymással és a szervezet más szabályzataival. Fentebb már rámutattunk, hogy kisebb szervezeteknél nem is foglalják írásba a követendő szabályokat. Nagyobb szervezeteknél is összemosódhat a két szabályozás, esetleg az IBSz impliciten vagy expliciten tartalmazza a stratégiai elképzeléseket. Ez a megoldás nagy szervezeteknél azért nem célszerű, mert az IBSz-t sokkal gyakrabban kell a napi igényeknek megfelelően módosítani, mint az IBK-t.

A szabályozás - jellegénél fogva – uniformizál, csökkenti az egyéni kreativitást, kisebb-nagyobb kényelmetlenséget okoz. Előírhatják például, hogy kik jogosultak a hardver- vagy szoftverhibák elhárítására. Ilyenkor az ügyintézőnek meg kell várnia az illetékes kolléga intézkedését, ami esetleg hosszabb időt is igénybe vehet, és addig nem tudja végezni a saját munkáját. Még akkor is így kell tennie, ha tudja, hogyan háríthatja el a hibát. A szabályozás a kényelmetlenségek mellett biztonságot is nyújt a dolgozóknak. A szabályok betartása esetén nem lehet őket felelősségre vonni. Ha nem szabályozzák megfelelően az ügyviteli védelmi megoldásokat, akkor hiba esetén a rendszergazdákra hárul a felelősség.

A sorok írójával, aki akkor a DE Informatikai Intézetének igazgatója volt, tanulságos eset történt néhány évvel ezelőtt. Egy verőfényes tavaszi délutánon az irodámban ültem és már készültem a napi munka befejezésére, amikor felhívott az ügyeletes rektorhelyettes és közölte, hogy nem hagyhatom el az irodát és senkivel sem beszélhetek, amíg a rendőrség meg nem érkezik hozzám. A „vendégek” érkezéséig eltelt néhány percben lázasan gondolkodtam, hogy milyen törvénytelenséget követtem el, de semmi sem jutott eszembe. A nyomozók a számítógépeink elhelyezéséről és az általunk használt IP címekről érdeklődtek. Többszöri

kérdésükre sem tudtam kielégítő választ adni, hiszen az IP címek adminisztrálásával én nem foglalkoztam. Miután sikerült ezt megértetnem velük, engedélyezték a Rendszergazda Csoport vezetőjének bevonását a kihallgatásba. Ő már érdemi válaszokat tudott adni és átadta a keresett számítógépet is. A nyomozás nálunk jegyzőkönyv felvételével befejeződött, de az egyetemen még késő éjszakáig folytatódott.

Kiderült, hogy a nyomozók – nemzetközi akció keretében – egy fájlcserélő hálózatra csaptak le. Egyetemünkön a nagy sávszélességet kihasználva néhány tárolót helyeztek el. A hálózatnak ebben segítséget nyújtott néhány kolléga, akik sajátjukként helyezték el a számítógépeket a szerverszobában és felügyelték azok működését. A ***szabályozás*** hiányosságát kihasználva csak a rendszergazdáktól kértek baráti segítséget a gépek elhelyezésére. Az esetből gyorsan levontuk a tanulságokat és szigorúan szabályoztuk a berendezések be- és kihozatalát az épületből, valamint azt, hogy milyen eszközöket és milyen engedélyekkel lehet a szerverszobában elhelyezni.

Végezetül felhívjuk a figyelmet arra, hogy a legkörültekintőbb szabályozás sem lehet eredményes a dolgozók együttműködése nélkül. Csak a jól felkészült és a szabályokat tudatosan alkalmazó dolgozók képesek az elveket a gyakorlatba átültetni. Az informatikai biztonsági követelmények a technológia és szervezet fejlődése miatt folyamatosan változnak, amelyekre a munkatársakat nemcsak a betanulási időszakban, hanem rendszeres képzéssel kell felkészíteni.

5.1 Informatikai Biztonsági Koncepció

Az Informatikai Biztonsági Koncepció a szervezet felső vezetésének informatikai biztonsággal kapcsolatos stratégiai elképzeléseit foglalja össze. A koncepció tartalmazza a szervezet informatikai biztonságának követelményeit, az informatikai biztonság megteremtése érdekében szükséges hosszú távú intézkedéseket, ezek kölcsönhatásait és következményeit.

A koncepció tartalma függ a szervezet tevékenységétől és nagyságától, de fontosabb tartalmi összetevőit meg tudjuk határozni:

- a védelmi igény leírása: jelenlegi állapot, fenyegetettségek, fennálló kockázatok,
- az intézkedések fő irányai: a kockázatok menedzselése,
- a feladatok és felelősségek meghatározása és felosztása a védelmi intézkedésekben,
- idő- és költségterv a megvalósításra és időterv az IBK felülvizsgálatára.

A koncepció több lépésben készül, amelynek főbb szakaszai az alábbiak:

1. *Védelmi igény feltárása*: Ebben a szakaszban választjuk ki a szervezet működése szempontjából lényeges informatikai rendszereket, informatikai-alkalmazásokat, amelyek értékük alapján érdemesek a védelemre. Ide tartoznak a vállalat szerverei, és tárolóegységei, a lokális hálózata aktív és passzív elemei. Meg kell vizsgálni a vállalat tágabb környezetét: a világhálón való megjelenését, a beszállítókkal és alvállalkozókkal szembeni követelményeket. Át kell gondolni az aktuális, a közép és esetleg a hosszú távú technológiai és szervezeti fejlesztéseket és változtatásokat.

2. *Fenyegetettség elemzés*: Feltárjuk azokat a biztonságot fenyegető tényezőket, amelyek az első szakaszban kiválasztott alkalmazásokra veszélyesek lehetnek, kárt

okozhatnak. A lehetséges veszélyforrásokat a 3. fejezetben foglaltuk össze. Itt kell vizsgálni az informatikai rendszer gyenge pontjait is, ahol a rendszer a külvilággal érintkezik.

Egyre több tanulmány hívja fel a figyelmet arra, hogy az informatikai rendszerek biztonságát elsősorban nem külső támadók, hanem a belső munkatársak fenyegetik. Meg kell tehát határozni az eszközökhöz és az adatokhoz való hozzáférés irányelveit. Mely munkahelyeken lehet saját tárolókapacitással és kimeneti lehetőséggel – pl. USB port - rendelkező számítógépeket elhelyezni. Kik és milyen mértékben férhetnek hozzá a világhálóhoz, illetve milyen jogosultsággal és módon lehet hozzáférni külső számítógépről a vállalat erőforrásaihoz. A hozzáférések naplózásának módját is meg kell határozni.

3. *Kockázatelemzés*: Ebben a szakaszban azt értékeljük, milyen káros hatása lehet a fenyegető tényezőknek az informatikai rendszerre. Meghatározzuk a lehetséges károk bekövetkezésének gyakoriságát, valamint a kárértéket és ennek függvényében a szükséges védelem technológiáját és mértékét.

4. *Kockázat menedzselés*: Kiválasztjuk a fenyegető tényezők elleni intézkedéseket és azok hatását értékeljük. Megvizsgáljuk, hogy az egyes intézkedések mennyire hasznosak és milyen költséggel járnak. Ide tartozik a veszélyforrások elleni védekezés módjainak meghatározása, intézkedési tervek. Nemcsak az előre jelezhető veszélyekre és kockázatokra kell felkészülni, hanem a váratlan helyzetekben teendő lépéseket is meg kell határozni. Nagyon fontos a felelősök kijelölése és felelősségi körük definiálása. Időtervet kell készíteni az intézkedések bevezetésére és az intézkedések hatása felülvizsgálatának ütemezésére. Egyetlen stratégia sem él örökké, különösen igaz ez az informatikával összefüggő stratégiára. Előre meg kell tehát határozni az IBSz felülvizsgálatának az ütemezését.

5.2 Informatikai Biztonsági Szabályzat

Az IBSz célja a technológiai megoldások részletezése nélkül, általánosan meghatározni az informatikai erőforrások biztonságos működéséhez szükséges feltételeket, a feladat- és felelősségi köröket. Az informatikai vezető és az informatikai biztonsági ellenőr készíti el, és a vállalat vezetője adja ki. A szabályozás az általános élethelyzetekre vonatkozik, speciális esetekben a szabályozás szellemének megfelelően kell eljárni.

Elkészítése során figyelembe venni a szervezet más szabályzataival, úgymint Munkavédelmi előírás, Tűz- és vagyonvédelmi szabályzat, való összefüggéseket, valamint követni kell a magasabb szintű szabályozásokat.

Személyi hatálya kiterjed a vállalat informatikai szolgáltatásaiban részt vevő munkatársaira, akik az informatikai alkalmazás folyamatában, mint szolgáltató vagy felhasználó, részt vesznek. Különösen fontosak a rendszer- valamint az adatgazdák jogait és kötelességeit meghatározó fejezetei.

Tárgyi hatálya alá tartoznak a vállalat tulajdonában lévő, illetve az általa használt számítástechnikai berendezések, beleértve a szoftvereket is, a feldolgozás alatt lévő, tárolt és a feldolgozás során létrejött adatok, adathordozók. Részét képezik a következő típusú eszközök: passzív adatátviteli vonalak (típusuk szerint Ethernet, Token Ring, FDDI, ATM szegmensek, optikai és hagyományos összeköttetések), csatlakozók. Hatálya kiterjed a hálózati aktív

elemekre (repeaterek, bridge-ek, switchek, routerek, transceiverek, modemek. terminál-szerverek), továbbá minden hálózatra kötött számítógépes munkahelyre (PC, workstation, terminál, hálózati nyomtató) és szerverre függetlenül attól, hogy az mely egység használatában van. Az IBSz előírja miden érintett dolgozó felelősségét a gondjaira bízott nagy értékű eszköz vagyonvédelmével kapcsolatban.

Az informatikai rendszerbe állított ügyviteli szoftverek esetében tesztelési folyamatot kell lefolytatni. Más, nem a vállalat, által fejlesztett szoftver esetén csak jogtiszta forrásból származós szoftver telepítése végezhető el. A szoftver telepítését ezekre a gépekre csak az informatikai felelős végezhet, vagy ha ez nem lehetséges, akkor a telepítés csak az informatikai felelős tudtával és írásbeli beleegyezésével történhet. Nem jogtiszta, vagy vírusos programok tudatos feltelepítéséből származó károkért az azt okozó dolgozó felelősséggel tartozik.

Az informatikai rendszer adatbázis elemeit, vagyis a felhasználói programrendszerek által létrehozott illetve felhasznált adatokat két üzembiztonsági zavar veszélyezteti. Egyrészt a hardver meghibásodása révén sérülhet fizikailag az adathordozó, ez adatvesztést jelenthet. Másrészt a szoftver (operációs rendszer vagy felhasználói program) meghibásodása miatt logikailag sérülhet az adatbázis, ez jelenthet adatvesztést vagy adat meghibásodást.

5.2.1 Biztonsági fokozat

Az IBSz fontos feladata az informatikai berendezések és adatok biztonsági osztályba sorolása. A klasszifikáció a munkahelyekre és munkahelycsoportokra vonatkozik, a vállalat méretétől és szervezetének bonyolultságától függően az egyes egységek besorolása lényegesen eltérhet. Természetes például, hogy a központi szerverek és tárolók esetén sokkal magasabb biztonsági fokozatot kell alkalmazni, mint az adatrögzítői munkahelyeken. Az adatok minősítése, a hozzáférésre jogosultak körének és a hozzáférés módjának meghatározása az adatgazda joga és felelőssége. Az eszközök valamint a rajtuk tárolt adatok értékének függvényében az alábbi biztonsági fokozatokat szokás megkülönböztetni:

- alapbiztonság: általános informatikai feldolgozás (nyilvános és személyes adatok),
- fokozott biztonság: szolgálati titok, átlagos mennyiségű különleges adat informatikai feldolgozása (bizalmas adatok),
- kiemelt biztonság: államtitok, nagy mennyiségű különleges adat informatikai feldolgozása (titkos adatok).

Szükség esetén természetesen ettől finomabb osztályozás is készíthető. Általános elv azonban, hogy minél magasabb a biztonsági szint annál részletesebben és pontosabban kell meghatározni a biztonsági előírásokat. Ki és milyen feltételekkel módosíthatja a hardverkonfigurációt és telepíthet új szoftvert. Meg kell határozni a hozzáférési jogosultságokat és eljárásokat, a naplózás módját és időbeli hatályát.

5.2.2 Védelmi intézkedések

A továbbiakban áttekintjük, hogy milyen védelmi intézkedésekről rendelkezik az IBSz. Ezek átfogják az informatikai rendszer minden elemét az infrastruktúrától kezdve, a

felhasználókon keresztül a hálózati szolgáltatásokig. A védelmi intézkedések munkahelytől, pontosabban a munkahely biztonsági fokozatba sorolásától függő.

5.2.2.1 Infrastruktúra

Elsőként az informatikai infrastruktúra elhelyezésével, fizikai veszélyforrásokkal szembeni védelmével foglalkozunk. Vagyonvédelmi szempontból is fontos, hogy azok az irodák, ahol az informatikai eszközök találhatóak, csak az épület belseje felől legyenek megközelíthetőek. Az ajtókat kulccsal vagy naplózott belépést biztosító beengedő rendszerrel zárják. A földszinti ablakok ráccsal vagy biztonsági üveggel való ellátása erősen ajánlott/kötelező védelmi elem..

Az épületbe, illetve onnan ki csak és kizárólag szállítólevélben szereplő informatikai eszköz vagy alkatrész szállítható. Az informatikai eszköz kiszállítása csak és kizárólag az eszköz leltározásáért felelős vagy az általa előzetesen megbízott munkatárs írásos hozzájárulásával történhet. Ez az eljárás kötelezően kiterjed a meghibásodott alkatrész csere után történő elszállítására is!

A nem a vállalat tulajdonában álló – bérelt vagy vásárolt szolgáltatást nyújtó – informatikai eszközök elhelyezésekor jelölni kell az eszközön, hogy az harmadik fél tulajdona, feltüntetve a tulajdonos nevét, elérhetőségét. Ezeket az eszközöket csak és kizárólag a harmadik fél jogosult be-, illetve kiszállítani, de a kiszállítás tényét szállítólevélen igazolnia kell, és azt át kell adnia a kijelölt munkatársnak. Rendelkezni kell arról, hogy a munkatársak bevihetik-e és ha igen akkor milyen feltétellel a munkahelyükre a tulajdonukban vagy a náluk tartós használatukban levő berendezéseket. A szabályozás a mindent megengedéstől a teljes tiltásig sokféle lehet, valamint függhet a munkahelytől és beosztástól is. A döntésnél azt kell mérlegelni, hogy a vállalat informatikai rendszere mennyire védett az adatok illetéktelen letöltésével szemben.

Szabályozni kell a hibaelhárítás felelőseit és módját. Meg kell tehát határozni, hogy meghibásodás esetén ki illetékes intézkedni. Ebben a vonatkozásban is nagyon széles a skála. Sem anyagi, sem adatvédelmi szempontból nem mindegy ugyanis, hogy egy terminál, valamelyik központi vagy egy fontos hálózati egység romlik el.

Az erkölcsileg vagy fizikailag amortizálódott berendezések selejtezésének módját is szabályozni kell. A leselejtezett berendezéseket megsemmisíthetik, eladhatják a dolgozóknak, más szervezetnek esetleg felajánlhatják iskoláknak vagy karitatív célra. Minden esetben biztosítani kell azonban a berendezéseken tárolt adatok végleges törlését. **Azokat az adathordozókat, amelyeken érzékeny adatokat tároltak nem ajánlatos törlés után tovább adni, hanem ellenőrzött módon meg kell semmisíteni.**

5.2.2.2 Felhasználói jogok kezelése

A felhasználói életciklus kezelése fontos része az IBSz-nek. Ajánlatos különválasztani a jogok kiosztásáért felelős és azt technikailag végrehajtó személyeket. Egyszerűbben

kifejezve ne a rendszergazda döntse el egy felhasználó jogosultságait, hanem az adatgazda kezdeményezésére tegye meg azt. Célszerű minden akciót visszakereshető módon és stílszerű módon elektronikus formában naplózni.

Felhasználó felvétele

Új felhasználó hozzáférési igényét, az illetékes szervezet vezetője kezdeményezi az informatikai szolgáltatásért felelős egység felé, amely végrehajtja a szükséges lépéseket. A kezdeményezésre célszerű formanyomtatványt tervezni, amely lehet papír alapú, de jobb az elektronikus alapú. A kezdeményező kötelessége, hogy a felhasználó munkaköréhez kapcsolódó szerepeket meghatározza úgy, hogy az egyes hozzáférési rendszerek kikényszeríttet hozzáférési megoldást biztosítsanak, azaz a felhasználó csak azokhoz az adatokhoz férhessen hozzá, melyekhez az adatgazda számára hozzáférést engedélyezett. A felhasználónak ne legyen módja az adatokhoz kapcsolódó hozzáférési szabályok módosítására.

A felhasználót tájékoztatni kell az informatikai rendszer használatának szabályairól. Későbbi viták elkerülése végett célszerű a tájékoztatás megtörténtét és azt, hogy a felhasználó tudomásul veszi a rá vonatkozó szabályokat, írásban is rögzíteni. Különösen fontos ez abban az esetben, ha a felhasználónak rendszeresen módosítania kell az azonosítóját, vagy ha a munkakör biometrikus azonosító alkalmazásához kötött. Tudatosítani kell a dolgozókban, hogy a munkatársakkal közös, illetve mások által ismert jelszavak, azonosítók használata a személyes felelősség átvállalásával egyenlő.

Felhasználói jogok módosítása

A munkavállaló új beosztása kerülhet, új projekt végrehajtására kaphat megbízást vagy a vállalat új alkalmazásokat vezet be. Mindezek, de sok hasonló esemény is szükségessé teheti felhasználói jogok és szerepek módosítását. Ezt általában a felhasználási jogot biztosító eljárási szabályok megtartása mellett, az érdekelt adatgazda igényelheti az informatikai szervezettől.

Előfordul, hogy a munkavállaló hosszabb-rövidebb ideig távol van, például szabadságra megy, vagy hosszabb továbbképzésen vesz részt. Ilyenkor szükség lehet a jogai ideiglenes felfüggesztésére, illetve arra, hogy a szerepét, pontosabban az általa vitt ügyeket, helyettes vegye át. Egy kórházban például egy beteg kórlapjához csak a kezelőorvosa és annak felettesei férhetnek hozzá. Amikor a kezelőorvos hosszabb ideig távol van, más orvosnak kell átvenni a beteg kezelését. Ilyenkor hozzá kell férnie a beteg kórlapjához és azt saját nevében tovább kell vezetnie.

Gyakori probléma, hogy egy felhasználó elfelejti a jelszavát. Ilyenkor nem célszerű az új felhasználó belépésekor alkalmazott eljárás megismétlése, hanem biztosítani kell, hogy az azonosságának bizonyítása után az informatikai részlegtől új jelszót kapjon a korábbi jogosítványainak megtartása mellett.

Megtörténhet az is, hogy a felhasználó elveszíti az azonosítóját, esetleg megsemmisül vagy kompromittálódik az. Sok azonosító semmisülhetett meg például a 2010-es árvizek vagy a vörösiszap katasztrófa következtében. Ilyenkor haladéktalanul le kell tiltani az azonosító használatát és a felhasználónak újat kell kiadni.

Felhasználó törlése

Felhasználó törlését minden esetben a felhasználási jogot biztosító eljárási szabályok megtartása mellett az adott szervezet munkaügyi és személyi ügyeit intéző, nyilvántartásokat vezető szervezetének értesítése mellett az adatgazda kezdeményezheti. A felhasználó törlése lehetséges egy alkalmazásból, vagy kilépés esetén minden alkalmazásból. Az adatgazdának rendelkeznie kell a felhasználó feladatköréhez rendelt adatok kezeléséről. Az adatokat át lehet helyezni más felhasználó kezelésébe, a helyettesítésre kijelölt felhasználó jogosultságainak módosításával. Mód van az adatok archiválására vagy akár törlésére is. A követendő eljárásról minden esetben az adatgazda köteles dönteni.

Tilos a felhasználó törlése addig, míg kezelésében adatok vannak. Munkahelyről való kilépés esetén a dolgozó adatainak tárolására vonatkozó ok megszűnik, így az adatvédelmi törvény szerint a rá vonatkozó személyes adatokat, beleértve a felhasználói életciklusára vonatkozó személyes adatokat is, törölni kell. Ez természetesen **nem** jelenti az általa készített dokumentumok vagy a munkavégzése során keletkezett naplóbejegyzések törlését.

Hozzáférési rendszer naplózási követelményei

Jól szervezett informatikai rendszeren a felhasználói akciókat (bejelentkezéseket, kijelentkezéseket és szerepkör váltást; a felhasználói azonosító és jogosultság módosításával, cseréjével kapcsolatos tevékenységeket) naplózni kell. Tilos a felhasználói jelszavak naplózása.

Fokozott védelmet igénylő rendszerek esetén naplózni kell a felhasználó nevében végrehajtott állomány hozzáféréseket, és tranzakciókat is.

Osztály	Meghatározás	Hozzáférési feltételek
Nyilvános	Nyilvános információ.	Mindenki hozzáférhet.
Személyes	Védett információ, amelyeknek illetéktelenekhez jutása esetleg veszélyt okozhat.	A szervezet minden dolgozója hozzáférhet. Az információ külsők számára tiltott, de partnerek hozzáférést kaphatnak. Az IBSz-ben nem kell jogosultsági szabályokat megadni.
Bizalmas	Jelentős üzleti értékű védett információ. Illetéktelenekhez jutása lényeges gazdasági kárt okozhat.	Csak meghatározott felhasználói csoport jogosult a hozzáféréshez. A jogosultak csoportját az információ tulajdonosa (pl. az IBSz-ben) határozza meg.
Titkos	Nagyon jelentős üzleti értékű védett információ. Illetéktelenekhez jutása megsemmisítő gazdasági kárt okozhat.	Csak egy megnevezett felhasználói csoport jogosult a hozzáféréshez. A jogosultak csoportját az információ tulajdonosa (pl. az IBSz-ben) határozza meg.

Osztály	Biztonsági szint	
	Hozzáférési szabályok	**Azonosítás/jogosultságkezelés**
Nyilvános	Nincsenek követelmények	Nincsenek követelmények
Személyes	Jogosultsági szabályok csak az „egyszerű” felhasználókra vonatkoznak. Vannak kivételezett felhasználók (rendszergazdák), az információ tulajdonosa nem ellenőrzi a hozzáférési szabályokat.	Egyszerű azonosítási eljárások, p.l. eszköz birtoklása. Elegendő hozzáférési joggal rendelkező felhasználók meghatározhatják a jogosultságokat.
Bizalmas	Jogosultsági szabályok minden felhasználóra vonatkoznak. Lehetnek kivételezett felhasználók, de akcióikat korlátozottak és nyomon követhetőek.	Explicit azonosítás szükséges, a digitális azonosítót hozzá kell rendelni a személyekhez, csoportokhoz, stb. Világos jogosultsági szabályrendszer. Csak az információ tulajdonosa által kinevezett csoport határozhatja meg a jogosultsági szabályokat.
Titkos	Jogosultsági szabályok minden felhasználóra vonatkoznak. Nincsenek kivételezett felhasználók. Minden hozzáférési döntés nyomon követhető.	Explicit azonosítás szükséges, digitális azonosítót személyhez kell hozzárendelni. Világos jogosultsági szabályrendszer. Csak az információ tulajdonosa határozhatja meg a jogosultsági szabályokat.

5.2.2.3 Szoftver

A vállalatok által használt szoftverek beszerzéséről és telepítéséről általában az informatikáért felelős egység gondoskodik. A számítógépre történő szoftvertelepítést csak az adott alkalmazás rendszergazdája, vezetői utasításra végezheti el. Felhasználó a munkahelyi számítógépére más szoftvert sose tegyen fel. Szoftverek másolása egyik gépről a másikra, illetve bármilyen más adathordozóra tilos, ha azt nem a munkafolyamatok követelik meg. Ilyen feladatok végrehajtásánál mindig meg kell győződni a szoftverek jogvédelmének érvényre jutásáról. A vállalatnál használatos minden programról biztonsági másolatot, illetve munkamásolatot kell készíteni. A munkamásolatokat a biztonsági másolatoktól elkülönült helyen - lehetőleg egy másik tűzszakaszban - kell tárolni.

A vállalat valamennyi munkavállalója, aki munkavégzése során számítógépet használ vagy üzemeltet, felelős azért, valamint köteles meggyőződni arról, hogy a számítógépen csak jogtiszta szoftver legyen található. A vállalat tulajdonában lévő számítógépeken használt szoftverek is a vállalat tulajdonát képezik. Eltulajdonításuk, még a saját számítógépre való telepítés is, törvénybe ütközik.

Bizonyos munkahelyeken szükséges lehet internetről letöltött, szabad szoftver használata. Ebben az esetben a műveletet elvégző dolgozó felelőssége, hogy a letöltött szoftverrel együtt ne kerüljön kártékony program a vállalat informatikai rendszerébe. Ezen kívül azt is biztosítani kell, hogy az új szoftver ne akadályozza a többi működését.

5.2.2.4 Adathordozó

A vállalat tulajdonában lévő adathordozókat nem szabad személyes célokra használni, és saját adathordozót sem ajánlatos a munkahelyi számítógéppel kapcsolatba hozni, a vírusok esetleges fertőzései miatt. Adathordozókon lévő adatok csak vezetői engedéllyel másolhatók. A dolgozók felelőssége a gondjaikra bízott adathordozók fizikai védelme. Óvni kell ezeket a mágneses környezettől, a nedvességtől, karcolódástól, bárminemű folyékony anyagtól (italok, virágok öntözésére szolgáló víz, vegyszerek) és fokozottan védeni a porszennyeződésektől.

Az ügykezelés szabályai szerint minősített adatokat csak nyilvántartott adathordozóra szabad felvinni. A nyilvántartott adathordozókat külön azonosítóval kell ellátni, ezeknek az azonosítóknak a feldolgozási, felhasználási folyamat végéig egyértelműen azonosíthatóknak kell lenniük és az ügyviteli előírások szerint kell őket kezelni. Minősített adatokat tároló hordozókat tilos felügyelet nélkül nyílt helyen tárolni, vagy akár rövidebb időre ott hagyni. Az adatokról másolatok csak a munkafolyamatra vonatkozó előírásokkal összhangban készíthetők.

Sérült vagy hibás adathordozókat, amelyekre az operációs rendszer hibát jelez, újra felhasználni szigorúan tilos. Olyan adathordozókon, amelyekről csak adatokat olvasunk be, és

nem akarjuk az adatokat módosítani, illetve menteni, minden esetben kapcsoljuk be az írásvédelmet.

Az adathordozókat használaton kívül minden esetben zárjuk el egy biztonságos szekrénybe, illetve adjuk vissza a raktárnak vagy archívumnak. A leselejtezett adathordozókat fizikailag meg kell semmisíteni. Ezen adathordozók mind munkahelyi, mind otthoni használata tilos. A minősített adatokat tartalmazó selejt adathordozókat és számítógépes listákat zúzással kell megsemmisíteni.

5.2.2.5 Dokumentum

A számítógépen lévő dokumentumok védelmére a hardver és szoftver eszközökre vonatkozó eljárások vonatkoznak. A dokumentumok adatok is, melyekre adatként is kell vigyázni. A papíron lévő dokumentumokat a levéltárban, azaz biztonsági rendszerrel őrzött szobában tároljuk.

Az informatikai rendszer biztonságának meghatározó tényezője az események visszakövethetősége, rekonstruálhatósága a felelősségek megállapíthatósága. Ennek megfelelően fontos, hogy a szükséges dokumentálási feladatokat folyamatosan ellássuk, illetve a rendszerek saját naplóállományait megfelelően kezeljük. Minden olyan eseményt, amelyik eltér a megszokott üzemviteltől, pontosan naplózni kell. A naplónak tartalmaznia kell az esemény pontos leírását.

5.2.2.6 Adatok

Hozzáférés-védelem: A számítógépeken levő adatok védelme érdekében, a számítógépes védelmi rendszerek megfelelő alkalmazása minden felhasználó alapvető kötelessége. Az informatikai eszközök a feldolgozott, illetve tárolt adatok védelmi kategóriája által meghatározott szintű hozzáférés védelmet biztosító védelmi eszközökkel vannak ellátva. Ennek megfelelően jelszavakat, illetve jogosultságigazoló eszközöket kell kezelni. Az adatokhoz csak érvényes, személyre szóló, azonosítható jogosultsággal - legalább felhasználói névvel és jelszóval - lehet hozzáférni. Hálózati erőforrásokhoz csak érvényes felhasználói névvel és jelszóval lehet hozzáférni. A jelszavak cseréjéről rendszeresen gondoskodni kell. A jelszó és a jogosultságazonosító eszköz szigorúan személyhez kötött!

A felhasználóknak saját adataik védelméről a rendelkezésre álló eszközök megfelelő használatával kell gondoskodnia. A számítógépekbe való bejelentkezéshez használt jelszavaikat (password) úgy kell megválasztaniuk és kezelniük, hogy ahhoz más ne juthasson hozzá. Különösen ügyelniük kell az otthoni hozzáféréssel rendelkező felhasználóknak arra, hogy illetéktelenek ne jelentkezhessenek be a telephelyi hálózatba az ő felhasználónevük és jelszavuk felhasználásával.

Fokozott, illetve kiemelt védelmi kategóriába sorolt munkahelyeken speciális azonosító-, jogosultságigazoló eszközöket alkalmazzunk. Ezek használatára mindig egyedi

szabályzatok vonatkoznak, melyek pontos megismerése, és precíz betartása minden felhasználó fontos kötelessége.

Ezen eszközök kezelésének általános követelményei:

- Az azonosítóeszköz át nem ruházható, csak tulajdonosa használhatja.
- Az azonosítókat tilos felügyelet nélkül hagyni, a számítógépes munkahely rövid idejű elhagyása esetén is!
- Az azonosítókkal kapcsolatos PIN-kódok kezelésében is érvényesíteni kell a jelszavak használatára vonatkozó normákat.
- Tartós távollét esetén (szabadság, kiküldetés) a helyi szabályzásnak megfelelően gondoskodni kell a védelmi eszközök őrzéséről. Ez általában az azonosító zárt borítékban, az e feladattal megbízott személy által felügyelt lemez-, illetve páncélszekrényében történő elhelyezését jelenti.
- Az azonosító elvesztését azonnal jelezni kell az alkalmazás rendszergazdája felé.

Az adatok és programok védelméről azok rendszeres *mentésével* kell gondoskodni. A mentések gyakoriságát úgy kell megállapítani, hogy az adatok esetleges műszaki hiba, vagy más ok miatt való sérülése, megsemmisülése esetén azok lehetőleg minimális veszteséggel visszaállíthatók legyenek. Különös gondossággal, és megfelelően sűrűn kell menteni a több dolgozó által használt, illetve több szervezet dolgozói által is használt központi szerver gépek programjait és adatállományait.

A számítógépen, illetve hálózaton tárolt személyes adatok biztonsága érdekében, különösen az alábbi intézkedéseket kell foganatosítani:

Tükrözés: A hálózati kiszolgáló gép (a továbbiakban- szerver) a személyes adatok elvesztésének elkerülésére folyamatos tükrözéssel biztosítható egy tőle fizikailag különböző adathordozón.

Biztonsági mentés: A személyes adatokat tartalmazó adatbázisok aktív adataiból rendszeresen – például a bér- és munkaügyi nyilvántartás, valamint a személyzeti nyilvántartás anyagából havonta - kell külön adathordozóra biztonsági mentést készíteni. A biztonsági mentést tartalmazó adathordozót tűzbiztos fémkazettában kell őrizni.

Az adatok védelmének következő módja a hibatűrő adattároló eszközök használata. Az ilyen típusú redundáns eszközöket RAID szintek szerint szokás csoportosítani (RAID, Redundant Array of Inexpensive Disks, alacsony költségű lemezek redundáns tömbje).

Az adatállományok védelme. Az adatbázisokról az üzemeltetési dokumentációkban meghatározott időközönként és módon másolatot (mentést) kell készíteni és az előírások szerinti időtartamig megőrizni. Az egyes feldolgozások adatainak mentési rendszeréről és azok megőrzésének határidejéről a rendszer használatbavétele alkalmával a rendszer kifejlesztője dönt, és ezt az üzemeltetési dokumentációnak kell tartalmaznia.

Adatvesztés vagy adatsérülést esetén a legutolsó biztonsági másolat visszatöltése után az információk újbóli beviteléről gondoskodni kell. Az ehhez szükséges emberi és gépi kapacitást biztosításáért az informatikai egység vezetője felelős.

Az input adatok helyességének biztosítása. Az információs rendszer bemenő (input) adatai helyességének biztosítása és annak ellenőrzése az adott felhasználó feladata. Ettől

függetlenül a szervezés során fel kell tárni az input adatok belső logikai összefüggéseit, adatállományokkal való kapcsolatukat, amelyek segítségével a hibás adatok kiszűrhetőek, illetve bekerülési valószínűsége minimálisra csökkenthető és ezeket az adatbeviteli programokban ellenőrzési, beazonosítási funkcióként be kell építeni.

A feldolgozás helyességének védelme. Az egyes programok működésének helyességét a programozók programozói tesztanyag összeállításával és a futáseredmények helyességének vizsgálatával kötelesek ellenőrizni. A teljes rendszer működését szervezői tesztanyag összeállításával és futása utáni eredmény-helyesség vizsgálattal kell megállapítani. A programozói és szervező teszt alapján helyesnek bizonyult programrendszer működését „felhasználói" tesztelésnek kell alávetni. Ilyen lehet például a párhuzamos adatfeldolgozás, a felhasználó által összeállított minta input feldolgozás stb. Az üzemszerű feldolgozást csak a tesztanyagok futtatása által hibátlannak talált programrendszer esetén lehet megkezdeni. A rendszer hibátlan működését a felhasználó az output adatok logikai összefüggéseinek ellenőrzése után folyamatosan köteles figyelni és hiba észlelése esetén tájékoztatnia kell a fejlesztőt. A felhasználó ezen túl folyamatosan ellenőrzi a feldolgozás teljességét is. A felhasználókat a rendszer használatával, működésével kapcsolatosan oktatásban kell részesíteni. Ezért a rendszer kifejlesztője és üzembe állítója a felelős az érvényben lévő szerződés szerint.

Archiválás: A személyes adatokat tartalmazó adatbázisok passzív hányadát - a további kezelést már nem igénylő, változatlanul maradó adatokat - el kell választani az aktív résztől, majd a passzívált adatokat időtálló adathordozón kell rögzíteni. Az adatkezelések archiválását rendszeresen el kell végezni. Az archivált adatokat tartalmazó adathordozót tűzbiztos fémkazettában kell őrizni. Az adatok archiválását mindig az adott folyamatra vonatkozó előírásoknak megfelelően kell elvégezni. A személyi számítógépeken lévő adatok archiválásáért a felhasználó a felelős. A hálózaton keresztüli archiválást minden esetben egy erre a munkára kijelölt munkatársnak kell elvégeznie.

Az adatok megőrzésének idejét a minősítésük határozza meg. A minősítéshez tartozó megőrzési időket az érvényes utasítások alapján határozzuk meg. A megőrzés idejét egyértelműen fel kell tüntetni az adathordozó külső, illetve belső címkéjén is. A megőrzés idejének nyilvántartása annak a személynek a feladata, aki az archívumot kezeli.

Az adatok archiválását mindenkor vírusztesztelésnek és vírusmentesítésnek kell megelőznie. Mind a vírusteszteléshez, mind pedig az adatok mentéséhez a szervezet legfrissebb, hivatalosan támogatott szoftverét kell alkalmazni.

5.2.2.7 Hálózati védelem

A szervezet belső hálózatának és központi levelező szerverének üzemeltetése valamint a világhálóra való csatlakoztatás a központi informatikai egység feladata. Ennek az egységnek kell megoldani az interneten keresztül érkező káros programok és kéretlen levelek szűrését is. A hálózati rendszeradminisztrátor az informatikai egység vezetője által megbízott személy, aki a hálózat mindenkori kiépítettségét, azaz a hálózatra kötött minden eszköz naprakész nyilvántartását vezeti. Hozzájárulása nélkül a hálózatra eszköz nem köthető, a szervereken olyan szolgáltatás nem indítható, amely bármilyen mértékű, több egységet érintő

adatforgalommal jár. Jogkörét részben átadhatja valamely egység rendszeradminisztrátorának az átadott jogok és kötelezettségek pontos körülhatárolásával.

Az IBSz rögzíti a felhasználó kötelességeit, az általa elvégezhető és tiltott tevékenységeket, a számonkérés formáját, valamint a biztonsági események jelentésével kapcsolatos kötelezettségeket. Az Internet szolgáltatásait – böngészést, elektronikus levelezést – a munkavégzés céljából lehet igénybe venni. Tilos olyan adat továbbítása (küldése, letöltése), amely alkalmas kártékony kódnak a vállalat informatikai rendszerébe juttatására, valamint tilos minden olyan tevékenység, amely szerzői - és társjogok megsértését vonja maga után

5.2.3 A belső elektronikus levelezés szabályai

Az elektronikus levelezés az egyik leghasznosabb és leggyakrabban használt informatikai szolgáltatás. Növeli a vállalat belső és külső információ forgalmának sebességét és hatékonyságát. A dolgozók közötti együttműködést egyszerűbbé teszi. A pozitív hatások mellett azonban negatívok is jelentkeznek. A kéretlen mailek veszélyeiről a 3.3.3 fejezetben írtunk. Ezekkel szemben az informatikai egység által üzemeltetett tűzfal és levelező szerver konfigurálásával kell védekezni.

Szabályozási szempontból sokkal fontosabb az a kérdés, hogy a dolgozók milyen célra használhatják a munkahelyi e-mailjüket. Ha a dolgozók korlátlanul használhatják személyes célra is, akkor lehetőségük van arra, hogy a vállalatra vonatkozó bizalmas adatokat harmadik félnek továbbítsanak. Ezt nyilvánvalóan nem szabad megengedni. A túl szigorú szabályozás pedig az e-mail előnyeinek kihasználásától fosztja meg a vállalatot.

A szabályozás során abból kell kiindulni, hogy az elektronikus levelezési cím és a hozzá tartozó elektronikus levelesláda a vállalat tulajdona, azzal korlátlanul rendelkezik. Az erre a címre érkező és innen küldött levelek nem személyes adatok, azokat iktatni kell és tartalmukat a vállalat által megbízott személyek ellenőrizhetik. Ilyen mélységű ellenőrzés általában a dolgozónak sem érdeke.

Nagyobb szervezeteknél ma már megoldott az elektronikus levelek iktatása és nyilvántartása is. Ilyen esetben a dolgozó kiválása esetén könnyű a folyamatban levő feladatok átadása. A munkavégzés során keletkezett dokumentumokat sok esetben még ma is egy-egy dolgozó személyes elektronikus postaládájában tárolják. Gondoskodni kell tehát arról, hogy a dolgozó kilépése, vagy hosszabb idejű kiesése esetén a postaládához más, kijelölt felhasználó hozzáférhessen.

Megbízással vagy projekttel kapcsolatos leveleket célszerű külön könyvtárban gyűjteni és azt a lezárás után archiválni, illetve átadni a megbízást átvevő személynek. A szerző hat évig volt az Informatikai kar dékánja. A hivatali e-mail címére érkezett és ezen funkciójával összefüggő leveleket a megbízása lejárta után archiválta és másolatban átadta az új dékánnak.

Az informatikáért felelős szervezetnek rendszeresen mentést kell készítenie a felhasználók elektronikus postaládájáról.

5.2.4 Felelősség és ellenőrzés

Nem elegendő az IBSz-ben leírni a védelmi intézkedéseket, hanem a végrehajtásukhoz felelősöket is meg kell nevezni. A felelősség lehet konkrét feladathoz kötött, amilyenekről a korábbi fejezetekben szóltunk. Lehet azonban általános utasítási jogkör is, amit előre nem látható esetekben – például katasztrófa helyzet - gyakorolhat a kijelölt személy.

A felelős joga és kötelessége, hogy az IBSz-ben foglalt hatáskörben rendszeresen ellenőrizze az előírások betartását. Figyelmeztesse a felhasználókat a védelmi intézkedések betartására, szükség esetén pedig fegyelmi felelősségre vonást is kezdeményezhet.

A felelős joga és kötelessége továbbá, hogy az általa észlelt vagy tudomására jutott veszélyhelyzetnek a méretétől, várható következményeitől függően a megfelelő ideiglenes védekező lépéseket meghozza: az adott gép(ek) külső elérhetőségének a letiltása, az adott gép(ek)nek a hálózatról való eltávolítása, vagy más, az esetnek megfelelő mértékű intézkedés. A biztonságot érintő incidensekről összefoglaló beszámolót kell készíteni és azt az illetékes vezetőkhöz el kell juttatni. Minden felhasználónak kötelessége, hogy ha betörésgyanús esetet észlel, ezt azonnal jelentse a vállalat biztonsági csoportjának, és a szükséges mértékben működjön együtt a csoporttal a károk elhárítása érdekében.

Az informatikai rendszerek folyamatosan fejlődnek, régi eszközöket vonnak ki és alkalmazások szűnnek meg, illetve új eszközöket és alkalmazásokat vezetnek be. Rendszeresen jelennek meg új veszélyforrások. Ezért az IBSz-t rendszeresen felül kell vizsgálni és az aktuális helyzetnek megfelelően módosítani kell.

6 Azonosítás és jogosultságkezelés

Személyek, tárgyak, jellegzetes tereptárgyak, irodalmi alkotások, stb. megnevezése, majd nevük utáni azonosítása az emberek ősidők óta meglévő igénye. Azonosítani lehet valamit a leírása, földrajzi koordinátái, címe stb. alapján. Történeti fejlődés során kialakultak az emberek ma használatos természetes azonosítói.

Az ókorban csak a szabad polgároknak volt személyes azonosítójuk, a rabszolgákat a gazdáik után ismerték. A középkorban sem volt személyi igazolványa az embereknek. A személyi azonosítás csak néhány száz éve vált általánossá, de a viszonylag rövid idő ellenére azonosíthatóság igénye mélyen beívódott a modern emberekbe. Robert Merle, Madrapur című könyvének főhőse Mr. Vladimir Sergius a következőket gondolja, amikor elveszik az útlevelét: „És főképp mivel magyarázom azt a kínzó és a lelkem mélyén erős nyomot hagyó érzést, hogy az útlevelemmel együtt a személyazonosságomat is elvesztettem? Nem tudom megfejteni ezt a lelkiállapotot. Csak körülírni. És ha jól meggondolom, nem is olyan képtelenség, mert *aki nem tudja igazolni a többi ember előtt, hogy ki, rögtön semmivé válik, elmerül a sokmilliós egyforma tömegben*."

Az informatikai rendszerek veszélyforrásairól szóló fejezetben foglalkoztunk az emberi veszélyforrással (**Hiba! A hivatkozási forrás nem található.** fejezet). Ezen belül különösen fontos a felhasználók biztonságos azonosítása és jogosultságaik megfelelő kezelése. Egy felhasználó azonosítása annyit jelent, hogy megbizonyosodunk arról, hogy a felhasználó valóban az, akinek állítja magát. Informatikai rendszereknél az azonosítás egy ember-gép vagy gép-gép interakció. A számítógépek számára is értelmezhető, automatizálható megoldásokat kell tehát találni, amelyek lényegesen különböznek a emberek közötti azonosítás folyamatától.

Nagyobb informatikai rendszerben sokféle erőforrást és alkalmazást közösen menedzselnek. Ugyanakkor nem minden felhasználónak van szüksége arra, hogy mindent elérjen, sőt komoly kárt is okozhat, ha valaki olyan alkalmazáshoz férhet hozzá, amelyet nem tud használni vagy számára tiltott a használata. Ha például az elektronikus tanulmányi rendszerben egy hallgató hozzáférhet az érdemjegyek beírását lehetővé tevő alkalmazáshoz, akkor igénye szerint adhatna magának jegyet. Komplex rendszereknél tehát nagyon fontos a jogosultságkezelés is, ami azt jelenti, hogy a felhasználók csak meghatározott erőforrásokhoz és alkalmazásokhoz férhetnek hozzá.

Megjegyezzük, hogy a mai bonyolult informatikai rendszerekben egyre terjed a szolgáltatás alapú együttműködés. Ilyen architektúrában több autonóm rendszer dolgozik együtt, pontosabban, ha az egyiknek szüksége van egy bizonyos adatra, akkor nem maga határozza azt meg, hanem elkéri azt egy partner alkalmazástól. Például, ha szüksége van az EURÓ aktuális árfolyamára, akkor lekérdezi azt egy árfolyamot szolgáltató szerverről. Ahhoz, hogy ezt megtegye, azonosítania kell magát, a szerver pedig ellenőrzi, hogy jogosult-e a szolgáltatás igénybe vételére. Szóval ma már nemcsak a személyek, hanem eljárások azonosítására és jogosultságaik kezelésére is szükség van.

6.1 Azonosítási helyzetek

Sokféle élethelyzetben van szükség személyazonosságunk bizonyítására. Az alábbiakban felsoroljuk az azonosítás néhány fontos alkalmazási területét:

a) Közösséghez való tartozás bizonyítása. A kapcsolat lehet tőlünk független, tartós, akár életünk végéig tartó, mint az állampolgárság, amelyet a személyi igazolvány és az útlevél igazol. Lehet azonban tőlünk függő is, mint a pártok, szakmai társaságok és érdekérvényesítő szervezetek vagy klubok, stb. által kiadott azonosítók.
b) Képesség bizonyítása. Például gépjármű vezetői engedély, érettségi bizonyítvány, diploma, nyelvvizsga bizonyítvány, stb. Bizonyítványokat rendszerint rövidebb-hosszabb tanulási folyamatot lezáró képességfelmérő vizsga után adnak ki.
c) Szolgáltatás igénybevétele: bank-, hitel- és városkártyák, bérletek, stb.
d) Védett térbe való belépés. Sok alkalmazottat foglalkoztató szervezeteknek biztosítani kell azt, hogy csak azok léphessenek be az épületbe vagy munkahelyre, akik ott dolgoznak vagy engedélyt kaptak erre. Az ilyen típusú igazolványhoz már jogosultságok is tartoznak, így átmenetet képeznek a hagyományos igazolványok és az informatikai rendszerek azonosítói között.
e) Informatikai rendszerekhez való hozzáférés. Tulajdonképpen ez is egy védett tér, azonban nem valós, hanem virtuális tér. A virtuális tér egyre bővül és struktúrája is egyre bonyolultabbá válik. Ma már nemcsak az intézmények és vállalkozások védik a virtuális terüket, hanem az internetes játékok, a chatszobák és közösségi portálok – facebook, iwiw, myspace, stb – is.

A társadalom fejlődésével és differenciálódásával valamint a szolgáltatások számának növekedésével párhuzamosan folyamatosan nő azon helyzetek száma, amikor igazolni kell magunkat. Ennek megfelelően nő igazolványaink száma. Igazoló irataink is folyamatosan változnak, például a kis, könyv alakú személyi igazolványt és útlevelet felváltotta a sokkal könnyebben használható és géppel is olvasható kártya alakú. A kényelmesebb és hatékonyabb kezelés mellett növekedett az igazolványok előállításának és a rajtuk található azonosítóknak a bonyolultsága is. Az igazolványok ugyanis mindig kedvelt célpontjai voltak a hamisítóknak és más bűnözőknek. Hamisíthatatlan igazolványt lehetetlen készíteni, hiszen a bűnözők előbb vagy utóbb hozzájuthatnak azokhoz az eszközökhöz, amelyeken a valódi okiratok készülnek, megismerhetik az azonosító jegyeket és még az előállítás részleteit is megismerhetik. A cél tehát csak az lehet, hogy mindezeket az ismereteket csak nagy késéssel gyűjthessék össze, lehetőleg csak akkor, amikor a technológiai váltás megtörténik. Ezen a területen tehát a technológiai fejlődés fő hajtóereje a hamisítás egyre nehezebbé tétele.

6.2 Az azonosítók fajtái

A hagyományos azonosítók általában *birtoklás alapúak*. Olyan tárggyal vagy eszközzel igazoljuk magunkat, amely a birtokunkban van, szóval ez egy materializált

azonosító. Ilyenek az igazolványok, a középkorban elterjedt, de még ma is használatos pecsét vagy egy szervezethez való tartozást demonstráló jelvény. Egy lakáskulcs is birtoklás alapú azonosítónak tekinthető.

Az *igazolvány* tartalmazza azokat az adatokat, amelyekkel az igazolvány tulajdonosát azonosítani lehet. Tartalmazhatja a tulajdonos jogosultságait is. Az igazolványnak nehezen hamisítható anyagból kell készülni. Ugyanakkor az igazolványnak tartósnak és a fizikai módosításokkal szemben ellenállónak kell lennie. Tehát nehezen lehessen széttépni, szétvágni vagy kimosni a benne foglalt adatokat. Korábban erre a célra kidolgozott, speciális papírból készült, ma azonban egyre jobban terjed a műanyag kártya alapú igazolvány. Kisebb és strapabíróbb, mint a papír igazolvány és digitális formában nagy mennyiségű adatot is lehet rajta tárolni.

A kétségkívül legfontosabb és legelterjedtebb azonosító a *személyi igazolvány*. Bródy János, 1980-ban meg is énekelte a személyi igazolvány lényegét:

„Személyi igazolvány, van, tehát én létezem
Személyi igazolvány, az egyetlen igazolvány
Mellyel hitelt érdemlően
Igazolhatom, hogy azonos vagyok velem”

A dal írásának idejében a személyi igazolványnak kis könyv alakja volt és kemény fedele. Tartalmazta a tulajdonos fényképét, aláírását és a személyi adatait: név, leánykori név, születési idő és hely, valamint a speciális magyar azonosítót, a tulajdonos édesanyjának nevét. Az alapadatok mellett bejegyezték az állandó és ideiglenes lakcímet, később a személyi számot is. Ezeket ma is azonosító adatoknak tekintjük. A régi személyi igazolványomban azonban benne volt a foglalkozásom, a munkahelyem adatai, házasságom után a feleségem és gyermekeim adatai is. Utóbbiak ma már nem kerülnek be a személyi igazolványba. Az okmánynak magának is volt és ma is van azonosítója, mégpedig a sorszáma. A 6.1 ábrán bemutatjuk egy múlt századi személyi igazolvány első oldalát.

6.1 ábra

Az azonosító adatok közül kettőt külön kiemelünk: a fényképet és az aláírást. Ezek egymástól függetlenek és mindkettő nagyon jól alkalmas személyek azonosítására. Richard Dawkins biológus professzor jól fogalmazza meg, hogy miért használható a fénykép azonosításra: „Az embereknek rendkívüli képessége van az arcok megkülönböztetésére. Mint azt egy másik összefüggésben látni fogjuk, minden jel szerint külön erre a célra kifejlesztettünk az agyunkban egy részt, és bizonyos típusú agykárosodás megbénítja az arcfelismerő képességünket, ugyanakkor épen hagyja a látás többi részét. Mindenesetre az arcok, változatosságuk révén, alkalmasak a felismerésre.[8]” Az aláírással a viselkedés alapú azonosítók között fogunk foglalkozni.

A papír és kártyaalapú igazolványok mellett számos más birtoklás alapú azonosítási technikát is kidolgoztak. Ilyenek lehetnek: USB eszköz, token, RFID, NFC, stb.

Hagyományos azonosítónak tekinthetjük a *viselkedés* alapú azonosítókat, úgymint aláírás, kézírás, beszédhang, gépelési ritmus, járási mód, szóhasználat, testbeszéd, arcmimika. Ezek közül témánk szempontjából az aláírás a legfontosabb. Egyszerűen és lényegében bárhol, bármilyen élethelyzetben kivitelezhető. Kétféle viszonylatban is egyedi. Egyrészt minden embernek másféle az aláírása, másrészt ugyanaz a személy sem képes kétszer

[8] Richard Dawkins, Szivárványbontás; Tudomány, szemfényvesztés és a csoda igézete, Vince Kiadó, 2001. 95 .old.

egyforma aláírást produkálni. Másféle írószerszámmal hajtja végre a műveletet, változhat az írásának a sebessége. Ha valaki utánozni próbálja korábbi aláírását vagy más ember aláírását, akkor a figyelme megoszlik az automatikus cselekvés és a minta követése között. A betűk alakja első ránézésre egyforma, de ha jobban megvizsgáljuk, akkor finom különbségeket észlelhetünk. Grafológusok még sokáig folytathatnák azoknak a jegyeknek a felsorolását, amelyeket figyelnek egy aláírás azonosításakor.

A többi viselkedés alapú azonosító is érdekes lehet speciális alkalmazásokban, de adatvédelmi szempontból a jelentőségük marginális. A hanggal való azonosítás sok esetben nagyon hasznos lenne és próbálkoztak is ezzel, de egyelőre nem sok sikerrel.

A biológiai jellemzőkre épülő csoporthoz, amelyet *biometrikus* azonosítóknak nevezünk, tartoznak: az ujjlenyomat, kézlenyomat, kézgeometria, az arc (arckép, fénykép), a termogramm, szem (írisz, retina), illat, DNS. A fentiek közül ma már klasszikusnak számít az *ujjlenyomat*. Ennek egyediségét már a XVIII. században felismerték, de csak a XIX. sz. vége óta használják a kriminológiában. A bűntett helyszínén a tettesek hátra hagynak nyomokat. Ezek közül gyakori az ujjlenyomat, hiszen a kezünk a legügyesebb testrészünk. A nyomok detektálása és összehasonlítása a rendőrség adatbankjában felhalmozott mintákkal vagy a gyanúsított ujjlenyomatával sok tettes elfogásához vezetett. A kriminológiai azonosítás során a tettes – természetesen – nem működik együtt a nyomozókkal. Ritkán hagynak a helyszínen teljes lenyomatot, azok rendszerint csak ujjlenyomat töredékek vagy elmosódottak. Ha a nyilvántartásban szerepel is az ujjlenyomat, azt akár több millió minta közül kell kiválasztani.

Hosszú kutató-fejlesztő munka eredményeként ma már hatékony eszközök állnak rendelkezésünkre ujjlenyomatok digitalizálására és automatikus azonosítására.[9] Néhány országban – például USA, Japán - a beutazni szándékozóktól ujjlenyomatot vesznek. Az Európai Unió országai azt tervezik, hogy az ujjlenyomat a személyi igazolványra is rákerül. A digitális ujjlenyomat felismerése más elveken nyugszik, mint a bűnüldözésben használté. Ebben az esetben tiszta lenyomatokra számíthatunk, hiszen az érintettről feltételezhető, hogy együttműködik a hatósággal.

A fényképről, mint azonosítóról fentebb már írtunk. Digitális azonosítóként azonban nem használatos, mert egyelőre nincsenek eszközeink digitális fényképek azonosítására. Amíg az ujjlenyomat felnőtteknél nem vagy alig változik, így élethosszig tartó azonosító, addig az arc komoly változáson megy keresztül. Függ az arc képe a szemüvegtől, a hajviselettől, a férfiak bajuszt vagy szakállat növeszthetnek, változtathatják annak formáját.

---

[9] A KLTE-n Buzási Károly vezetésével az 1980-as évek közepén kidolgoztunk ilyen módszert. Bár az eljárásunk működőképes volt, a külső megbízó nem finanszírozta tovább a fejlesztést, így a munkát nem folytattuk.

Betegségek és az életkor is nyomott hagy az arcunkon. Ezeket a hatásokat a digitális azonosítási technika még nem tudja kezelni.

A kriminológiában ma a legpontosabb azonosítónak az ember DNS-ét tekintik. Ez természeténél fogva digitális azonosító, de meghatározása ma még elég bonyolult, így alkalmazási köre meglehetősen behatárolt.

Az azonosítók utolsó nagy csoportja *tudás alapú*. Ide tartoznak a jelszavak, a PIN kódok és az aszimmetrikus kulcsok. Ezek felelnek meg leginkább az informatikai rendszerek sajátosságainak. Közülük a jelszó klasszikus képződmény, a katonaságnál valószínűleg ősidők óta használják. Eredetileg egy csoporthoz való tartozás bizonyítására használták. E sorok írója is előfelvételes sorkatonaként találkozott vele. A laktanya őrségét naponta vezényelték. Az ügyeletes tiszt a délutáni eligazítás végén átadta az őrparancsnoknak az aktuális jelszót, amelyet az őrség minden tagjával közöltek. Az őrt álló katona az őt megközelítő személyt felszólította, hogy tisztes távolságban álljon meg és mondja érthetően a jelszót. Ha az tudta az érvényes jelszót, akkor joga volt az őrt megközelíteni. Ez az aktus az őrszolgálat során többször is megismétlődött, amikor leváltották a posztoló katonát. Előfordult az is, hogy az ügyeletes tiszt vagy az őrparancsnok ellenőrizte a katonák éberségét. Olyan esetre nem emlékszem, hogy az őrt ellenséges szándékkal közelítették meg. A parancs minden esetre úgy szólt, hogy ha az őrt megközelítő személy nem tudja a jelszót, akkor akár fegyver használatával is meg kell akadályozni, hogy behatoljon a laktanyába.

Az informatikai rendszerek védelmében használt jelszó hasonló funkciót lát el. Később részletesen foglalkozunk vele.

A PIN-kód a Personal Identification Number, azaz személyi azonosító szám négy, esetleg öt vagy hat számjegyből álló azonosító. Ennek megfelelően összesen tíz- vagy százezer, esetleg egymillió különböző lehetőség van a megválasztására, így gyenge azonosító. Próbálgatással könnyen kitalálható, ezért csak olyan esetekben alkalmazható, ahol a többszöri próbálkozásra nincs lehetőség. Ugyanakkor könnyen megjegyezhető, ezért a pénzkiadó automatáknál nagyon elterjedt és erősebb azonosítókkal kombinálva első szűrőnek is megfelel.

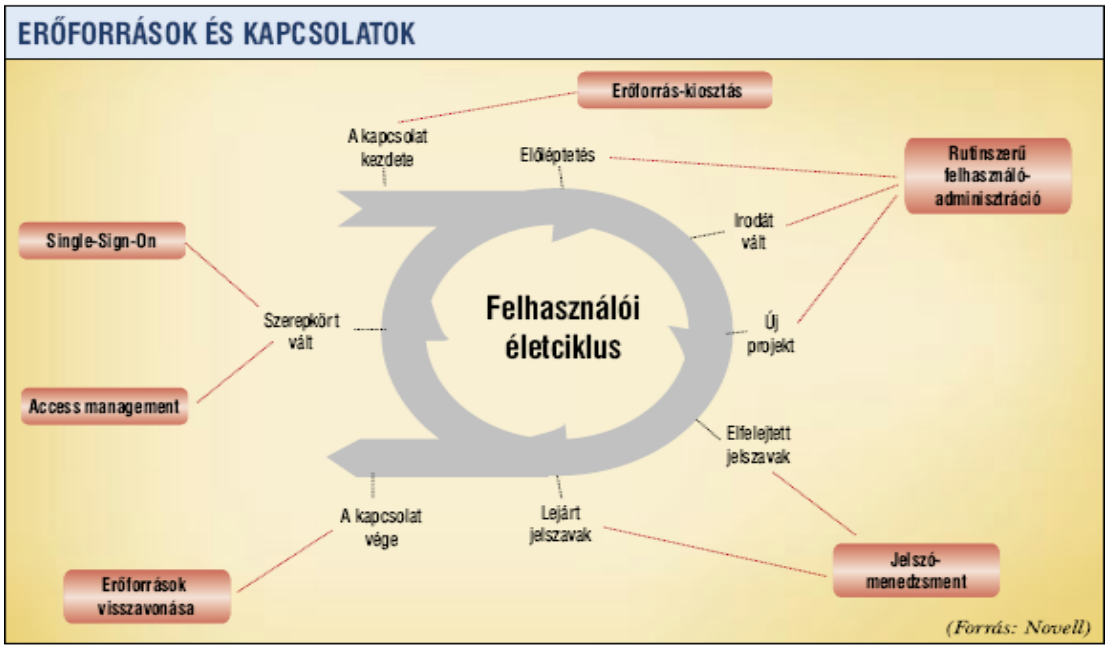
Az aszimmetrikus kulcsokat is ebbe a csoportba soroltuk, bár olyan nagy – több száz számjegyből álló - számok, amelyeket közönséges képességű emberek nem képesek megjegyezni. A kulcsot ezért valamilyen eszközön kell tárolni és szükség esetén onnan beolvasni a számítógépbe. Ezzel az azonosítóval is részletesen fogunk foglalkozni az aszimmetrikus titkosítással és a digitális aláírással foglalkozó fejezetekben.

Végezetül megjegyezzük, hogy egy azonosító lehet *egyszerű* és *összetett*. Utóbbit több szintűnek vagy több csatornásnak is nevezik. Az egyszerű azonosító egyetlen ellenőrzésre alkalmas jegyet tartalmaz. A fejezetben felsoroltak mindegyikét használhatjuk egyszerű azonosítóként. Az összetett azonosítók több, egymástól független ellenőrzésre alkalmas jegyet tartalmaznak. A személyi szám és a születés dátuma nem független azonosítók, hiszen az elsőből egyértelműen és könnyen származtatható a második. Hasonlóképpen testvéreknél nem független adat az édesanyjuk neve. Az azonosítás biztonságát nem veszélyezteti függő adatok használata, de fölösleges redundanciát jelent.

Jól megválasztott, független azonosítók jelentősen erősíthetik egymást. Ezért kombinálják a névvel a fényképet, az ujjlenyomatot vagy aláírást; a bankszámlaszámmal a jelszót, hogy csak néhány példát említsünk. Még biztonságosabb az azonosítás, ha ellenőrzéskor több, független csatornát is felhasználunk. Erre a legjobb példa sok internetes bank gyakorlata. A számla alapadataihoz már a bankszámlaszám és a jelszó ismeretében is hozzáférhetünk, tranzakció végrehajtását azonban meg kell erősíteni egy egyszer használatos kóddal, amelyet mobiltelefonunkra küldenek. Az internet és a mobilhálózat két egymástól független csatorna, így az esetleges támadónak mindkét csatornát ellenőriznie kell ahhoz, hogy a tranzakciót meghamisíthassák. Ma már lehetőség van mobiltelefonon keresztül végezni az internetes bankolást, amikor is a két azonosító függővé válik. A példa mutatja, hogy a technika változása következtében független azonosítók függővé válhatnak.

6.3 Az azonosítók életciklusa

Az előző fejezetekben bemutattuk a tipikus azonosítási élethelyzeteket és a gyakran használt azonosítókat. Most azt vizsgáljuk meg, hogy milyen események történnek egy azonosítóval annak létezésének idején. Elsősorban az általánosan jellemző eseményekre vagy olyanokra koncentrálunk, amely azonosítók széles osztályára érvényes. A fejezetben azt a szervezetet vagy alkalmazást, amely szolgáltatásai igénybe vételéhez az azonosítás szükséges röviden szolgáltatónak fogjuk nevezni.



6.2 ábra

A 6.2 ábra bemutatja a felhasználó életciklusát egy szervezeten belül. Az életciklus fázisait le kell képezni az azonosítók és a jogosultságok menedzselésében is. Ezeket az eseményeket követjük végig a fejezetben.

6.3.1 Regisztráció

Ha egy természetes vagy jogi személy, esetleg ezek egy csoportja – a továbbiakban ügyfél – elhatározza, hogy egy szolgáltatást igénybe vesz vagy kötelezik a szolgáltatás igénybe vételére, akkor kezdeményezi a szolgáltatónál a kapcsolatfelvételt. Ezt a szolgáltató el is utasíthatja, amikor a folyamat befejeződik. Pozitív válasz esetén kezdődhet az azonosító elkészítése. A kapcsolatfelvételt és az azonosító készítését *regisztrációnak* nevezzük és a szolgáltató részéről egy önálló szervezeti egység vagy – programok esetén – önálló modul végzi. A regisztráció során meg kell adni az előírt természetes azonosítókat, amelyeket a szolgáltató ellenőriz, és ha mindent rendben talál, akkor készülhet el az azonosító. Ha egy hivatalban történik a regisztráció, akkor az is készíti el az azonosítót, különben rendszerint az ügyfél adja meg azt. A regisztráló szervezet naplózza az aktust és szükség esetén tárolja az azonosító másolatát az ellenőrző szervezet adatbázisában. Az adatbázisban egyéb információt is tárolhatnak, közülük a leggyakoribbak az azonosító érvényességi ideje, a felhasználó szerepe vagy jogosultságainak listája. Egy felhasználóhoz tartozó rekord standard kereső kulcsát felhasználó névnek nevezzük.

Itt meg kell állnunk egy pillanatra. Az azonosítók másolatának tárolása a nem digitális azonosítók esetén természetes követelmény, mert lényegesen megkönnyíti az elveszett vagy megrongálódott azonosító pótlását. Kezdetben a digitális azonosítókat, például a jelszót is, kódolás nélkül tárolták. Arra figyeltek csak, hogy az adatbázist a memória nehezen hozzáférhető területén tárolják. Ez a megoldás azonban komoly veszélyforrás. Ha ugyanis egy hekker hozzáfér a szerverhez és megtalálja a jelszavakat tartalmazó adatbázist, akkor minden felhasználó azonosítója kompromittálódik, azaz minden felhasználónak új belépési kódot kell készíteni. Sokfelhasználós rendszereknél ez óriási költséggel és munkával jár.

A 90-es évektől kezdve a jelszavakat nem nyíltan, hanem egy egyirányú függvénnyel kódolva tárolják. Egy függvényt *egyirányúnak* nevezünk, ha a helyettesítési értékét gyorsan ki lehet számítani, de a helyettesítési értékből az argumentum értékét nagyon nehéz kiszámítani. Ilyen egyirányú függvénynek tekinthető a nyomtatott telefonkönyv, amelyben a tulajdonos ismeretében nagyon könnyű a telefonszámát kikeresni, de a telefonszámhoz nehéz annak tulajdonosát beazonosítani. Egyirányú függvényként például a DES-t vagy az ElGamal függvényt alkalmazzák, de a szolgáltatók saját fejlesztésű függvényt is alkalmazhatnak. Az azonosítók kódolásánál alkalmazott egyirányú függvényt *hash* függvénynek nevezzük.

Tagja vagyok az egyik legnagyobb magyar egészségpénztárnak, amely a portálján keresztül naprakész információt ad a tagok számlájának állásáról. Hosszabb ideig nem látogattam a portált és ahogy lenni szokott elfelejtettem a jelszavamat. E-mailen keresztül kértem, hogy új jelszót generálhassak. Az ügyintéző nagyon kedvesen, e-mail fordultával elküldte az elfelejtett jelszavamat. Szóval, az egyik legtöbb tagot kiszolgáló egészségpénztár 2009-ben kódolás nélkül tárolta a jelszavakat!

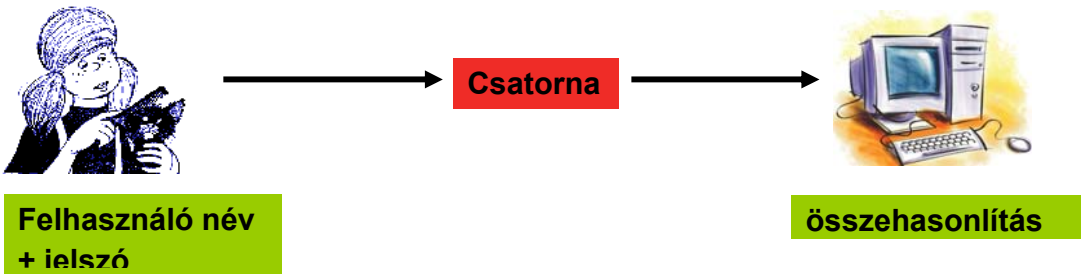
6.3.2 Ellenőrzés

Az azonosítót azért készítik, hogy igénybe vegyünk vele valamilyen szolgáltatást. Ez az azonosító érvényességi idején belül sokszor előfordul. Ilyenkor a szolgáltató *ellenőrző*

funkciója működik. Az ügyfél azzal kezdeményezi a szolgáltatás igénybe vételét, hogy megadja a felhasználói nevét. Az azonosító bevitele annak típusától függ. A tudás alapú azonosítót például a személy írja be a számítógépbe, az eszköz alapút behelyezzük egy alkalmas olvasó berendezésbe. A viselkedés alapú azonosítás esetén pedig valamilyen érzékelő eszköz figyeli a személyt és alakítja át az észlelt jeleket a számítógép által feldolgozható formára. Szigorúan véve az elektronikus azonosítás kétszintű folyamat; a felhasználói név *azonosítja* a felhasználót, majd a jelszó vagy más azonosító bevitele *hitelesíti*, hogy valóban arról a felhasználóról van szó, akinek az azonosítást kezdeményező állítja magát. A mindennapi szóhasználatban azonban nem foglalkozunk ezzel, így jegyzetünkben a hitelesítést az állományokra, illetve a nyilvános kulcs infrastruktúra egyik elemére tartjuk fenn.

Az azonosítók digitális mérete nagyon változó a jelszavak 40-64 bites méretétől, az 512-2048 bites aszimmetrikus kulcsokon keresztül, a több kilobájtos biometrikus azonosítókig terjed. Hasonlóan széles palettán mozog az ellenőrző eljárások bonyolultsága és ennek következtében sebességük is. Közös jellemzőjük azonban az, hogy a beolvasott azonosítót vagy azonosítókat az adatbázisban tárolt etalonnal hasonlítják össze. Nem kell azonban minden etalont végignézni, hanem csak azt vagy azokat, amelyek a felhasználó névhez tartozó rekordban vannak.

Jegyzetünk keretei nem teszik lehetővé, hogy minden azonosító típusra elemezzük az ellenőrzés folyamatát. Itt csak a jelszavas azonosításra koncentrálunk, mert ez a leggyakoribb elektronikus azonosítási módszer. Látni fogjuk, hogy már itt is sok problémába ütközünk.

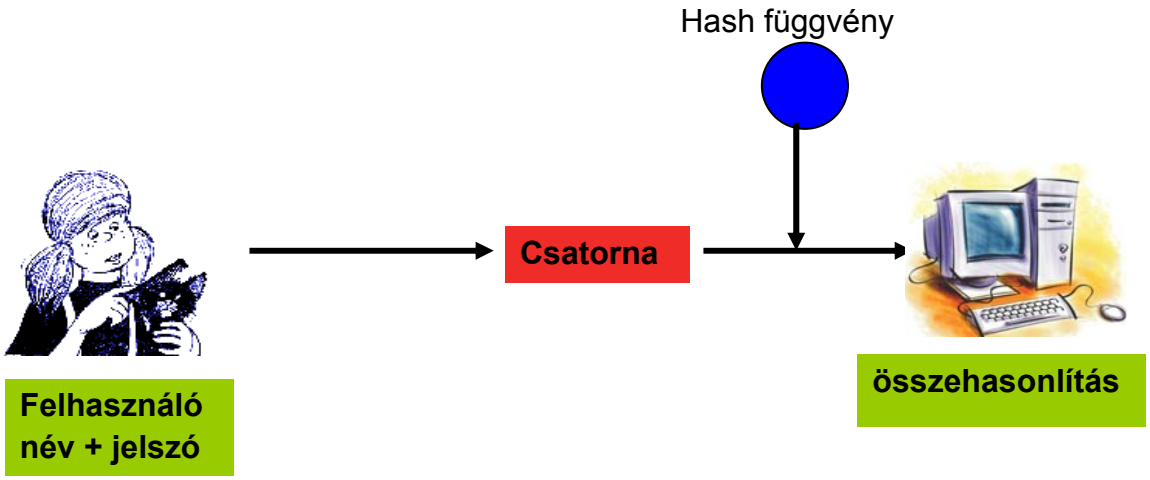


6.3 ábra

A 6.3 ábrán bemutatjuk a jelszavas azonosítás alapesetét. Kriszta akarja magát azonosítani számítógépnek. Kriszta megadja a felhasználói nevét és a jelszavát. Mindkét adat kódolatlanul megy keresztül a csatornán! A számítógép megnézi, hogy a felhasználói név érvényes-e. Ha nem, akkor elutasítja a kérést. Különben megnézi, hogy a felhasználói névhez tartozó rekord jelszó mezőjében található adat megegyezik-e a Kriszta által megadott jelszóval. Ha nem, akkor ismét elutasítja a kérést, különben elfogadja, hogy Kriszta valóban felhasználó és használhatja a jogosultsági listájában megadott erőforrásokat.

Amint a regisztrációról szóló részben megjegyeztük ennek az eljárásnak nagy hibája, hogy a számítógép adatbázisában a jelszavakat kódolatlanul tárolják. Azt is leírtuk, hogy ha a regisztrációnál egy hash függvényt alkalmazunk, akkor ez a probléma kiküszöbölhető. Ha

azonban a regisztrációkor nem a jelszavat, hanem annak kódolt változatát tároljuk, akkor ellenőrzéskor a Kriszta által megadott jelszóra ugyanezt a hash függvényt kell alkalmazni, mint amelyet a regisztrációkor használtak. A 6.4 ábra mutatja a módosított bejelentkezési eljárást.



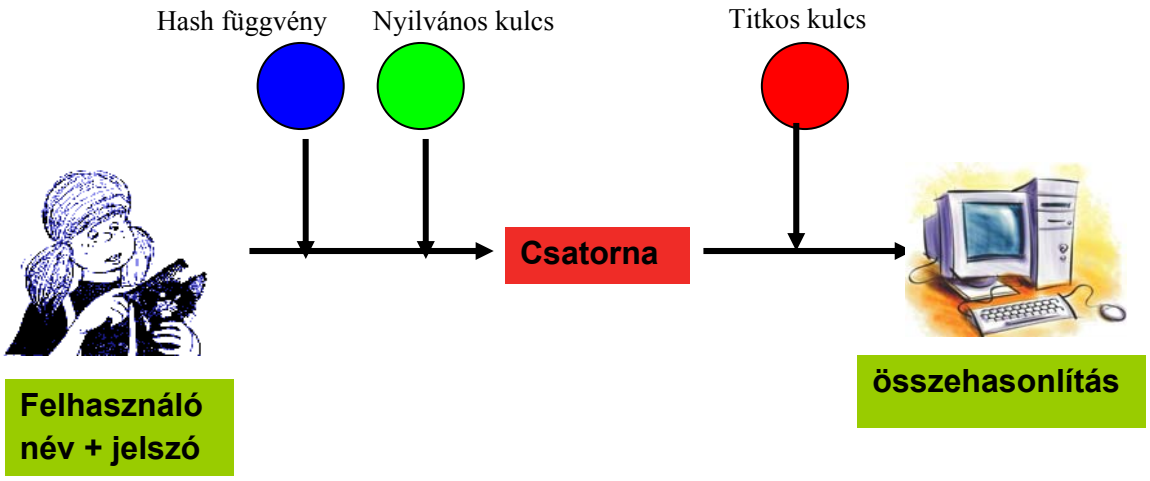
6.4 ábra

Az azonosítás folyamata majdnem ugyanaz, mint az első esetben, különbség csak annyi, hogy a számítógép a Kriszta által megadott jelszónak a hash értékét hasonlítja össze az adatbázisában található adattal. A hash érték kiszámításához némi erőforrásra van szükség, de ez bőven megtérül azzal, hogy a jelszavak tárolása lényegesen biztonságosabb lett.

Ez a mechanizmus nagyon egyszerű és kielégíti a mindennapi élethez szükséges biztonsági követelményeket. További finomítás nélkül azonban már egy távoli számítógéphez való belépéskor vagy internetes banki tranzakciókor sem biztonságos. A hagyományos IP protokollok (ftp, http) ugyanis az adatokat eredeti formájukban továbbítják, így – esetünkben - a jelszót csak a fogadó számítógép kódolja. A jelszó tehát kódolatlanul utazik a nyilvános hálózaton. Javít a helyzeten, ha már Kriszta számítógépe kiszámítja a jelszó hash értékét és csak az kerül a nyilvános csatornára. Ha ugyanis valaki hozzáfér a csatornán küldött üzenethez, akkor mindegy, hogy Kriszta jelszavát vagy csak annak hash-elt változatát szerzi be, utóbbival ugyanúgy meg tudja személyesíteni Krisztát, mint a jelszóval.

A múlt század végén kidolgozott és egyre jobban terjedő, biztonságos IP protokollokkal (ssl, https) úgy oldható meg ez a probléma, hogy a jelszót aszimmetrikus titkosítással kódolva küldik át az ellenőrző számítógépnek. Az aszimmetrikus titkosítással a 8.7 fejezetben fogunk foglalkozni. A számítógép a felhasználói név elfogadása után elküldi a tanúsítványát Krisztának. Kriszta ellenőrzi a tanúsítványt, és ha rendben találja, akkor kinyeri belőle a számítógép nyilvános kulcsát. A jelszavára alkalmazza a hash függvényt, majd a számítógép nyilvános kulcsával kódolja a hash értéket. Az így kiszámolt adatot elküldi nyilvános csatornán a számítógépnek, amelyik a titkos kulcsával dekódolja az üzenetet, majd a kapott adatot, amelynek meg kell egyeznie Kriszta jelszavának hash értékével,

összehasonlítja az adatbázisában tárolt adattal. Innentől kezdve a folyamat megegyezik az alapesetben leírtakkal. Az eljárást a 6.5 ábra mutatja.



6.5 ábra

A jelszavak biztonságos továbbításáért elég nagy árat kell fizetni. Az aszimmetrikus titkosítás ugyanis speciális szoftvert és komoly számítási erőt igényel. Később látni fogjuk, hogy a titkosítás és a visszafejtés külön-külön is körülbelül ezerszer lassúbb, mint a hash érték kiszámítása. A harmadik módszert tehát csak akkor érdemes bevetni, ha magas szintű biztonságot akarunk elérni és ez ki is fizetődik.

6.3.3 Azonosítók pótlása és visszavonása

A birtoklás alapú azonosítót el lehet veszíteni, ellophatják vagy annyira károsodhat, például kimoshatják vagy tűzbe esik, hogy használhatatlanná válik. Ilyenkor szükségessé válik annak visszavonása és új azonosítóval való pótlása. Az azonosító érvényességi idejének lejártával ugyanerre az aktusra kerül sor. A visszavonáshoz igazolni kell, hogy az azonosító valóban a miénk volt. Ha ez megtörténik, akkor az illetékes szervezet naplózza a visszavonás tényét az adatbázisába és általában új azonosítót állít ki.

A visszavonás végleges is lehet, ha a tulajdonos valamilyen ok miatt nem gyakorolhatja azokat a jogokat, amelyekre az azonosító felhatalmazta. Ilyen ok lehet, ha valaki meghal, olyan károsodást szenved, hogy nem vezethet járművet, kizárják egy szervezetből, stb. Ilyen esetekben tovább kell tárolni a visszavont azonosító másolatát, mert vissza lehet élni megszűnt vagy megszüntetett azonosítóval is. Nyikolaj Vasziljevics Gogol, Holt lelkek című regényének témája például az, hogy a főhős, Csicsikov, felvásárolja földesuraktól elhunyt jobbágyaikat. A regény végén kiderül, hogy mindezt azért teszi, hogy egy banktól nagy kölcsönt tudjon felvenni. Mire a bank kideríti, hogy a jobbágyok mind halottak, addig több év is eltelhet és a pénznek Csicsikovval együtt nyoma vész. A regény a XIX. században játszódik, amikor a nyilvántartások még nem voltak olyan hatékonyak, mint ma, de hasonló ötletekkel napjainkban is élhetnek a csalók.

Visszavonás után persze általában új azonosítót állítanak ki, amelynek folyamata azonos az eredeti azonosító elkészítésével.

Tudás alapú azonosítók érvényességi ideje is lejárhat, a munkahelyről kilépve megszűnhet a jogosultságunk, az azonosítót elfelejthetjük vagy kompromittálódhat, azaz tudomást szerezhet róla arra illetéktelen személy. Ilyenkor válhat szükségessé az azonosító pótlása, cseréje vagy törlése.

Az elfelejtett azonosítók pótlásának eljárási rendje attól függ, hogy mennyire értékes az általa elérhető adattömeg. Nyilvános vagy személyes adat esetén elegendő az e-mail címünket megadni és arra elküldenek egy olyan ideiglenes azonosítót, amelyet az első bejelentkezéskor meg kell változtatni. Bizalmas és titkos adatok esetén a regisztrációnál alkalmazott eljárást kell megismételni.

Bizonyos alkalmazások megkövetelik a jelszavak rendszeres módosítását, sőt azt is, hogy az új jelszó nem lehet azonos a legutóbb használt néhány jelszóval. Ez az előírás növeli, de csökkenti is a biztonsági szintet. A munkahelyi rendszerbe való belépéshez használt jelszavaknál, amelyet naponta kell beírni a számítógépbe a megoldás nagyon hasznos, mert a néhány hétig vagy hónapig használt jelszó kisebb eséllyel kompromittálódik. Jelszavaink nagy részét azonban ritkán használjuk, így nincs alkalmunk a memorizálásra. Valamilyen módon mégis rögzítik a jelszavakat, például a mobiltelefonban tárolják, vagy egy noteszbe írják be, rossz esetben felírják egy öntapadó cédulára és azt felragasztják a monitorra. Egyik megoldás sem tökéletes, de a legutóbb említettől mindenkit eltanácsolunk.

Az adathalászok gyakran jelszavak frissítésére vagy ellenőrzésére kérik fel a gyanútlan felhasználót. Ilyen példákat láttunk az 1.2 és a 3.1 ábrákon. Az 1.2 ábrán mutatott laphoz egy kéretlen mailen keresztül jutottam el, amelyben az adathalászok megkértek a bank nevében, hogy a jelszavam ellenőrzése céljából lépjek be a bank rendszerébe. Számos hasonló példát lehetne még bemutatni. Ismételten felhívjuk az olvasó figyelmét, hogy ilyen kérésre ne reagáljanak, a szolgáltatók nem szokták e-mailben felszólítani az ügyfeleiket jelszavaik frissítésére.

Az elektronikus aláírás törvényről szóló 4.2 fejezetben rámutattunk arra, hogy az ellenőrző kulcsot a hitelesítés szolgáltatónak, a kulcs érvényességi idejének lejárta után még legalább öt évig meg kell őrizni. Hasonló a helyzet a bizalmas dokumentumok archiválásához használt titkos kulccsal. Ezeket addig kell megőrizni, amíg szükséges lehet a dokumentum dekódolására. A titkos kulcs és a titkosításához használt eszköz valamelyikének hiányában ugyanis ezeket a dokumentumokat nem lehet visszaállítani, ami azt jelenti, hogy a rendelkezésre állásuk elveszett.

6.4 Azonosítók kompromittálása

Az azonosítók fizikailag sérülhetnek vagy megsemmisülhetnek, esetleg ellophatják őket. A tudás alapú azonosítók, különösen a jelszavak azonban más módon is illetéktelen kezekbe juthatnak. A **Hiba! A hivatkozási forrás nem található.** fejezetben foglalkoztunk olyan trójaiakkal, amelyek a számítógépre települve figyelik a billentyűzetet, és ha azt tapasztalják, hogy a felhasználó egy jelszót gépel be, akkor azt megjegyzik és továbbítják a trójai üzemeltetőjének.

Jelszavakat azonban másképp is ki lehet találni. A 6.3.1 fejezetben rámutattunk, hogy ma általánosan elterjedt, hogy a szolgáltató adatbázisában a jelszavakat egyirányú függvénnyel kódolva tárolják. Az azonosítás a szerveren történő egyirányú kódolással kiegészítve biztonságos akkor, ha az azonosító elegendően hosszú bináris jelsorozat, például biometrikus azonosító, de a jelszavas eljárás még így sem felel meg a nagy biztonságot követelő rendszereknél.

Jól ismert ugyanis, hogy ellene a *szótáras támadás* eredményes lehet. A kódolásnál használt egyirányú függvény ugyanis egy adott alkalmazásnál szabványos és leírása a dokumentációban megtalálható. A támadó tehát ismeri, és könnyen le tudja kódolni a hash függvényt. Ezek után elkészít a feltételezett jelszavakból egy bőséges listát, amelynek minden elemére kiszámítja a függvény értékét. A lista elkészítésénél igénybe veszi a social engineering felismeréseit. A felhasználó érdeke, hogy megjegyezze a jelszavát, ezért olyan szót választ, amelyet könnyen fel tud idézni. Természetes módon adódnak a családi vagy tulajdonnevek, állatnevek, autó vagy kozmetikai márkák, népszerű együttesek neve, stb. Érdemes felhasználni a célba vett nyelv szavainak minél bővebb szótárát, valamint egy nagy lexikon szócikkeit. Az internetről ma már nem kunszt bőséges szótárt összeállítani. A szótárt ki kell egészíteni értelmetlen szavakkal is. Ezek egy része a nyelv ABC-jének betűivel leírható összes legfeljebb öt-hat karakterből álló jelsorozat, és az alapszótár szavainak módosításával előálló karaktersorozatok. Módosítás lehet például, ha kis betűk helyett nagy betűket vagy számokat írunk.

Az angol ABC-ben 26 betű, a magyarban 35 betű van a kettős betűket nem számítva. A kis- és nagybetűket a számítógép eltérő módon tárolja, ha tehát még ezeket is figyelembe vesszük, akkor 52 betűvel számolhatunk angol, 70-nel magyar szövegek esetén. A 0-9 számjegyeket is megengedve további tízzel nő a különböző karakterek száma. A 6.1 táblázatban bemutatja, hogy ezekből hányféle értelmes és értelmetlen öt - nyolc betűből álló szót lehet alkotni. A táblázat adatait könnyű kiszámítani, hiszen jól tudjuk, hogy n betűből pontosan n^k darab különböző k hosszúságú karaktersorozat képezhető.

	Kisbetű	kisbetű + szám	Kis és nagybetű	Kis- és nagybetű és szám
Angol/5	$1{,}2\ 10^7$	$6{,}05\ 10^7$	$3{,}8\ 10^8$	$9{,}16\ 10^8$
Angol/6	$3{,}1\ 10^8$	$2{,}18\ 10^9$	$1{,}98\ 10^{10}$	$5{,}68\ 10^{10}$
Angol/7	$8{,}032\ 10^9$	$7{,}84\ 10^{10}$	$1{,}03\ 10^{12}$	$3{,}52\ 10^{12}$
Angol/8	$2{,}09\ 10^{11}$	$2{,}82\ 10^{12}$	$5{,}35\ 10^{13}$	$2{,}18\ 10^{14}$
Magyar/5	$5{,}25\ 10^7$	$1{,}85\ 10^8$	$1{,}68\ 10^9$	$3{,}28\ 10^9$
Magyar/6	$1{,}84\ 10^9$	$8{,}31\ 10^9$	$1{,}18\ 10^{11}$	$2{,}62\ 10^{11}$
Magyar/7	$6{,}44\ 10^{10}$	$3{,}74\ 10^{11}$	$8{,}24\ 10^{12}$	$2{,}1\ 10^{13}$
Magyar/8	$2{,}25\ 10^{12}$	$1{,}68\ 10^{13}$	$5{,}76\ 10^{14}$	$1{,}68\ 10^{15}$

6.1.táblázat

A szótár elkészítése után fordítsuk le a benne található szavakat a hash értékükre, azaz minden szóra alkalmazzuk az ismert hash függvényt. Így rendelkezésére áll a feltételezett

jelszavakból és azok „fordításából” álló szótár. Annak elkészítése a technológia mai színvonalán legfeljebb néhány napot vesz igénybe.

Ezek után, ha a hekker valamilyen módon hozzáfér az éles rendszer felhasználóinak adatbázisához vagy egy felhasználó hashelt jelszavához, akkor a kódolt jelszavakat megkeresi a saját szótárában, és ha egyezést talál, akkor már rendelkezésére is áll a kódolatlan jelszó is. Ehhez a felhasználói nevet már a felhasználói adatbázisban keresi vissza. A módszer nagyon hatékony, saját kísérleteinkben a felhasználók 25-30 %-ának a jelszavát is ki tudtuk így deríteni. Az irodalomban ettől nagyobb sikerrátával is találkoztunk. Nick Breese[10] 2007 novemberében számolt be egy konferencián arról, hogy a PlayStation játékkonzol processzorát használva 8 karakterből álló jelszavakat is meg tudott fejteni egy órán belül.

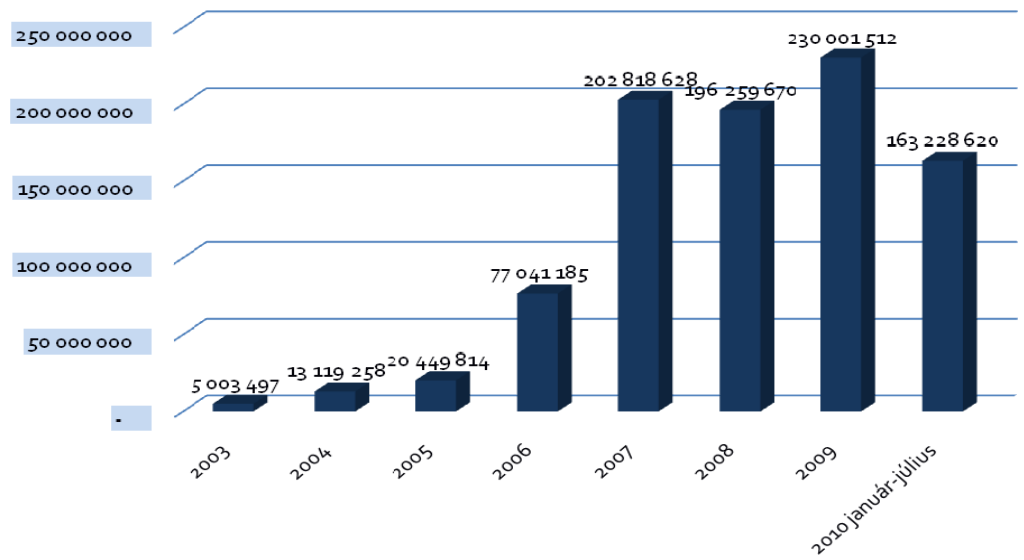
A fentiekből következik, hogy rövid, könnyen kitalálható jelszavakat nem szabad használni. A jelszavak legalább 7 karakterből álljanak és tartalmazzanak kis- és nagybetűket valamint különleges karaktereket (számok, írásjelek, stb.). Ezek már a mindennapi életben megfelelő biztonságot nyújtanak.

A következő két táblázatban összefoglaljuk a biztonsági osztályok és a hozzájuk tartozó azonosító valamint jogosultságkezelővel szemben támasztott követelményeket.

6.5 Ügyfélkapu

Miután áttekintettük az azonosítók használatának általános elméletét megvizsgáljuk, hogy hogyan valósulnak meg a megállapításaink hazánk legnagyobb szervezetében, az államigazgatásban. Az Ügyfélkapu a magyar kormányzat elektronikus ügyfélbeléptető és azonosító rendszere. Biztosítja, hogy felhasználói a személyazonosság igazolása mellett egyszeri belépéssel biztonságosan kapcsolatba léphessenek elektronikus közigazgatási ügyintézést és szolgáltatást nyújtó szervekkel. Ebben a fejezetben bemutatjuk az Ügyfélkapun keresztül történő regisztráció módjait és lépéseit, valamint az Ügyfélkapu azonosítási módszereit. Kormányzati portál látogatottságát mutatja a 6.6 ábra, amelyből kiderül, hogy a kezdeti hezitálás után a polgárok és a vállalkozások egyre jobban megbarátkoztak az elektronikus ügyintézéssel.

[10] Forrás: BBC News, Friday, 30 November 2007, http://news.bbc.co.uk/2/hi/technology/7118997.stm



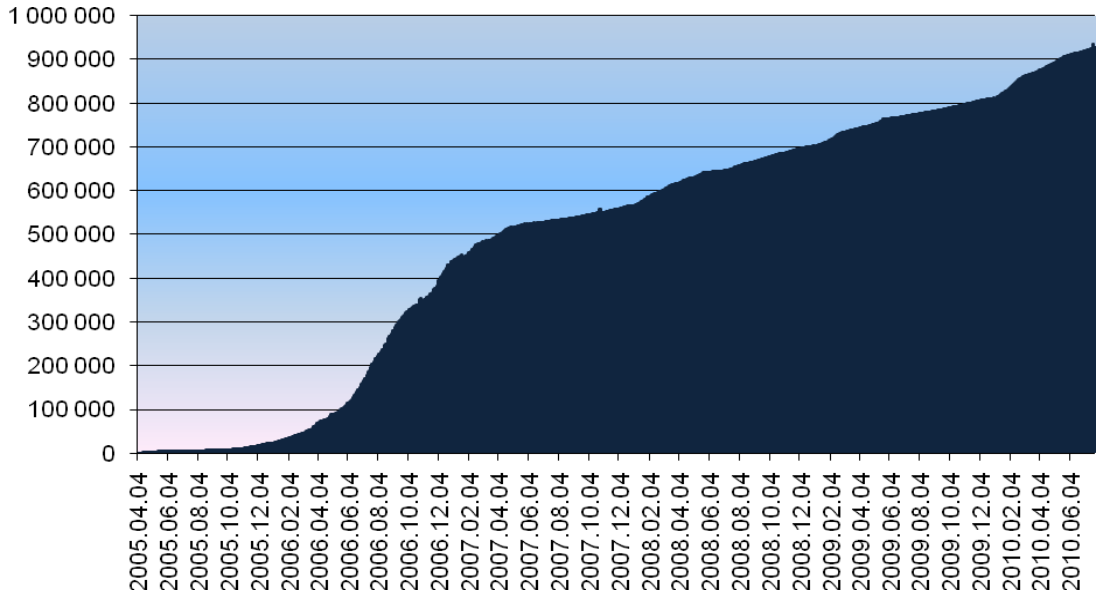
6.6 ábra[11]

6.5.1 Ügyfélkapus regisztráció - személyesen

Az elektronikus ügymenet országos szinten az Ügyfélkapun keresztül valósul meg. A rendszer megvalósításának köszönhetően egyszeri belépéssel (*single sign on*) az állampolgárok kapcsolatba tudnak lépni az elektronikus közigazgatási szolgáltatást nyújtó szervekkel. Az elektronikus ügymenetek indítása regisztrációt igényel. A regisztráció többféle módon történhet, az első esetben személyes jelenlétet kíván bármelyik okmányirodában. Ma már minden nagyobb városban működik okmányiroda, elhelyezkedésükről a Kormányzati Portálon keresztül tájékozódhatunk.

Az okmányirodába interneten keresztül is foglalhatunk időpontot. Nemcsak a természetes, hanem a jogi személyek is egyre inkább online keresik a kapcsolatot a hatóságokkal. Az online ügyintézésekben rejlő lehetőségek kiaknázása terén még messze állnak a 100%-os mutatótól. 2008. július elsejétől csak elektronikus úton intézhetők a cégbejegyzéssel kapcsolatos ügyek, így a 2005 óta létező elektronikus forma mára kizárólagossá vált. Ügyfélkapus regisztrációk számának rohamos növekedése jól követhető a 6.7 ábrán. Az is leolvasható, hogy a 2006-os rohamos növekedés után csökkent az érdeklődés intenzitása, de a növekedés napjainkig is töretlenül folytatódik.

[11] Forrás: Szittner Károly- E-kormányzás magyar megközelítése



6.7 ábra[12]

6.5.2 Debrecen - timeR

Debrecen a korszerű ügyfélfogadás érdekében új rendszert vezetett be mely időalapon működik. Két ügyféltérben, 53 ablaknál valósul meg a timeR funkcióinak használata. A polgárok számára helyben, telefonon és a város portálján keresztül is lehetővé teszi a tájékozódás vagy bejelentkezés lehetőségét. Interneten keresztül is be lehet jelentkezni a http://idopont.debrecen.hu oldalon. A projekt célja az ügyfelek felkészítésének és ügyeinek intézésének idő- és költséghatékony megvalósítása. 2008. február 1-től működik a rendszer felváltva és továbbfejlesztve a 2003-tól működő OSS sorszámozós módszert. A program bekerült a magyarorszag.hu „jó gyakorlatok adatbázisába" például szolgál az ország többi Polgármesteri Hivatalai és Okmányirodái számára.

6.5.3 Ügyfélkapus regisztráció – elektronikus formában

Az Ügyfélkapu regisztráció másik módjára elektronikus formában van lehetőség, ehhez szükség van egy olyan elektronikus aláírásra, mely teljes mértékben igazolni tudja az állampolgárok személyazonosságát. Személyes tapasztalat, hogy ideiglenes tanúsítvánnyal a regisztráció nem lehetséges, mert az ingyenes szolgáltatás keretében letölthető teszt tanúsítvány csak a technikai ellenőrzést teszi lehetővé. Ehhez semmi másra nincs szükség csak arra, hogy megadjunk egy érvényes e-mail címet és az erre a címre kiküldött linkre

[12] *Forrás: Szittner Károly- E-kormányzás magyar megközelítése*

lépve letöltsük a tanúsítványt. Így látható, hogy semmilyen személyes azonosítás nem történik meg, tehát ezzel az azonosítóval nem lehet regisztrálni.

Az Ügyfélkapu által elfogadott elektronikus tanúsítványok pl. a NetLock Tanúsítványkiadó Központnál a Minősített Közjegyzői (Class QA) személyes minősített tanúsítvány 20 000 Ft-tól kezdődik. A tanúsítványok díjai egy évre szólnak, a kibocsátástól számított egy év lejárta után, a tanúsítványok megújítása szükséges. Egy „átlag" állampolgár „átlag" ügyintézése személyes megjelenéssel egy Okmányirodában kisebb költségvetést igényel az ügyfél részéről. Igaz ez még akkor is ha figyelembe vesszük, hogy személyesen akár többször is be kell menni a Polgármesteri Hivatalba, és ez plusz parkolási vagy buszköltséget is jelenthet. Véleményünk szerint ez nagyban akadályozza az elektronikus kulccsal való regisztrálást. Napjainkban még csak néhány ezer kulcsot használnak (ezek között is legfőképpen jogi személyek, vagy vállalkozások a leggyakoribbak), a tömeges elterjesztés érdekében jó megoldás lenne a TB- kártyán elhelyezett privát kulcs. Ennek leolvasása a háztartásokban elég költséges lenne. A mobil eszközökön való megvalósítása például a kulcs elhelyezése SIM-kártyán már egy felhasználóbarát megoldás lehetne, ha megvizsgáljuk Magyarországon az egy főre eső mobiltelefonok számát.

A hitelesítés szolgáltatás költségeinek megosztása az állampolgárok és az általuk befizetett adókból működő államigazgatás között megoldásra váró probléma. Észtországban például a személyi igazolvány része a minősített tanúsítvány. A tanúsítványt egy magán hitelesítés-szolgáltató bocsátja ki, melyet két távközlési cég és két bank hozott létre. Az észtek nemcsak szavazhatnak interneten keresztül elektronikus személyigazolványuk segítségével, de már fizethetnek is vele a közlekedési eszközökön. Az igazolvány, kötelező minden állampolgár és minden, egy évnél hosszabb ideig az országban tartózkodó, külföldi számára. Az észt modell átvétele csak szűkebb körben képzelhető el Magyarországon a személyes adatok védelmének eltérő logikája miatt.

6.5.4 Ügyfélkapus regisztráció – ideiglenes

Az Ügyfélkapu lehetőséget ad ideiglenes regisztrációra is, mellyel néhány szolgáltatás (pl. ideiglenesen bejelentkezve a TAJ- kártya szolgáltatáshoz) elérhetővé válik. Ezt 30 napon belül aktiválni kell vagy a fent említett elektronikus módon, vagy személyesen bármelyik Okmányirodában egy személyigazolvány és egy e-mail cím megadásával. Tapasztalatunk szerint az eljárás nem felhasználó barát. Előfordul, hogy a kérelmező nem kap visszajelzést a kérelmének befogadásáról és teljesítéséről és később csak személyesen fejezheti be a regisztrációt.

5 napig van lehetőség az egyszer használatos linkre kattintva belépni, és egy új egyedi jelszót megadni. E jelszó egy úgynevezett „erős" jelszó, azaz tartalmaznia kell legalább két numerikus karaktert és egy nagybetűt. Minimum 8 karakternek kell lennie és az ügyintéző hölgy telefonon azt tanácsolta, hogy 3 karakternél több ne egyezzen meg a felhasználó névben szereplő karakterekkel. Az Ügyfélkapun keresztül indított ügymenet akár több héten keresztül is eltarthat, ami véleményünk szerint sokkal időigényesebb, mint a hagyományos papír alapú ügyintézés.

6.5.5 Ügyfélkapus azonosítási módszerek

Az informatikai biztonság követelménye a kétlépcsős azonosítás, egy olyan módszer, amely két egymástól független módon állapítja meg a személyazonosságot és jogosultságot. Tudás (jelszó) és birtoklás (tanúsítvány) alapján is hitelesíthető egy ügyfél. Ez az elektronikus aláírással együtt valósult meg, de a felhasználók 99%-a maradt az egylépcsős hitelesítésnél. Mára csak ez a lehetőség maradt, mert a kétlépcsős módszert, azaz a személyes azonosítás és elektronikus kulcs egyidejű kombinációját megszüntették, így a rendszer informatikai szempontból bizonyos támadások ellen védtelenné vált.

Az elmúlt időszakban több üzemzavar is fellépett a működés során. Az *Informatikai Biztonsági Felügyelő Részletes jelentésében* megállapításra került, hogy az üzemzavarok emberi mulasztásra vezethetők vissza. 2009 februárjában egy olyan szoftver okozta az Ügyfélkapu zavarát, ami a kormányzati portál teljesítményét volt hivatott növelni. A rendszert azért indították el, mert a portál túlterhelés miatt lelassult. A program teszt helyzetben jól működött, de élesben számos hibát produkált. Az üzemeltetők a hiba megjelenésekor sem állították le a rendszert. Az állampolgárok bejelentkezésekor, más polgárok személyes adataihoz juthattak véletlenszerűen. A titkos kulcsokkal készített adatokat nem láthatták illetéktelenek, az adófolyószámlákon sem lehetett változtatni.

2009.12.1-re tervezték az „Ügyfélkapu 2” bevezetését, amelyet a *KOPINT-DATORG Infokommunikációs Zrt* dolgozott ki. A fejlesztés célja az Ügyfélkapu megbízhatóságának növelése és ügyfélbarát jellegének további erősítése. Több hónapos késéssel végül 2010.03.01-én hajnalban indították be az új rendszert, amely újra összeomlott és a régi rendszert állították vissza. 2010.03.06-án indult el sikeresen az új arculat. A http://ujportal.magyarorszag.hu/bemutato/index.html - címen érhető el az új portál bemutatása, ill. összevetése a korábbi rendszerrel.

Az azonosítás területén bevezették az ún. „kétcsatornás” rendszert, mely azt jelenti, hogy a személyes azonosítás mellett, egy mobil eszközzel történő azonosítási módot is alkalmaznak. Ez a megoldás már néhány éve hatékonyan működik a banki ügyintézésnél.

Alkalmazzák a „kétfázisú” módszert is; az állampolgár belép az Ügyfélkapun és ezután a TAJ- számát is meg kell adnia a TAJ- szolgáltatások eléréséhez. Az a tapasztalatunk azonban, hogy e szolgáltatás elérése nem igényel teljes regisztrációt, mert ideiglenes regisztráció mellett, és a TAJ- szám megadásával be lehet lépni a rendszerbe. Mint azt már említettük az ideiglenes regisztráció nem követel meg személyes megjelenést, hanem elegendő interneten keresztül megadni a személyes adatainkat valamint e-mail címünket. Véleményünk szerint gondot okozhat, ha valaki elhagyja vagy eltulajdonítják az igazolványait, mert illetéktelen személyek a fent említett adatokkal ideiglenes regisztráció mellett hozzáférhetnek pl. az elmúlt 10 év egészségügyi adataihoz. Többek között a vényre felírt gyógyszerek listájához, és az OEP adatbázisában szereplő adatokhoz.

6.6 Debrecen e-Kormányzata

E fejezetben tárgyaljuk a Debrecen Megyei Jogú Város e-kormányzatához kapcsolódó legfontosabb kérdéseket. Hogyan történik az állampolgárok azonosítása? Hogyan érvényesül

a nyilvánosság elve? Milyen informatikai irányelveket követ a város? Bemutatjuk az adatkezelési, adathozzáférési és a webadó rendszert is.

6.6.1 Azonosítás

Mint láthatjuk Debrecen önkormányzatának nem feladata az állampolgárok azonosítása, e feladatot ugyanis az Ügyfélkapu üzemeltetője végzi el. A konkrétan Debrecen városhoz tartozó közérdekű információkhoz nem a https://magyarorszag.hu, hanem a http://www.debrecen.hu oldalon juthatnak hozzá az állampolgárok. Színvonalas honlap ahol minden fontos információ megtalálható a város vezetéséről.

6.6.2 Webadó

Debrecen városa informatikai rendszerének egyik problémája, hogy az egyes közigazgatási rendszerek szigetszerűen működnek, az ügyek egy részének intézése nem interaktív. Az egyik kiemelkedő alkalmazás a webadó. Az interneten beadott adóbevallások száma több uniós országban is meghaladta 2008-ban a hagyományos papíron beküldött bevallások számát. Magyarországon az adóhatóság által biztosított Internetes kitöltő programmal elkészített, majd kinyomtatva beérkezett személyi jövedelemadó bevallások az összes, nyilvántartott 3 797 956 db bevallásból 2010-ben is jelentős hányadot, 59,6%-ot képviselnek. Ez összesen 2 262 342 darab bevallást jelent. Tendenciájában folyamatos növekedés tapasztalható a személyi jövedelemadó bevallásukat elektronikusan teljesítők számában. Az elektronikus úton benyújtott személyi jövedelemadó bevallások száma 2010-ben 607 325 db volt, ami 6,2%-os növekedést jelent a 2009-es 572 164 db-hoz képest.

A webadó rendszernek köszönhetően Debrecen elsőként vezette be a digitális aláírás használatát és engedélyezését hazai szinten. Az adóügyi rendszer szigorúan bizalmas ezért hármas jelszóhasználatot vezettek be a jelszó, PIN- kód és adószám együttes használatával. A digitális aláírás elfogadásával pedig helyi adóbevallásra is lehetőség van. A rendszer funkcionalitásának a két alapvető részét az általános információk és szolgáltatások valamint az adózó specifikus funkciói adják. Fel van készítve arra, hogy hitelesített adatforgalmat bonyolítson le interaktív módon az adóhatóság és az ügyfelek között.

6.6.3 Nyilvánosság elve

Érvényesül a nyilvánosság elve, hiszen az elektronikus közigazgatás egyik fő jellemzője a tájékoztatási kötelezettség. A jelenleg elérhető információs tartalmak már nem csak magyar nyelven érhetőek el, hanem angolul is. Ezzel a lépéssel közelebb került a város az Európai Uniós elvárásokhoz. Az említett városi portálon kötelező megjeleníteni a pályázatokat, állásokat, azokat az adatokat, ill. információkat, amelyeket a „kötelező közzététel” jellemez. Vannak olyan információk, melyek csak akkor adhatóak ki, ha a helyi jegyző engedélyt ad rá. Ilyen például az önkormányzat szerződései.

A szervezetnek lehetősége van különböző adatok továbbítására, ill. kezelésére. Ezek az adatok lehetnek az állampolgárok adatai is. Az adatkezelőnek elsősorban a természetes

személyazonosító adataival kell tudni azonosítani a polgárokat (úgy, mint anyja neve, születési hely, idő, lakcím). Ezek persze helyettesíthetőek egyetlen azonosító kóddal is (pl. személyigazolvány szám). Ezt az azonosítót csak a törvény által meghatározott esetben használhatja fel, minden más esetben szükség van a polgár előzetes, írásbeli hozzájárulására.

6.6.4 Adatkezelés

Különböző adatkezelési szituációk léphetnek fel, ezek között lehet személyes adat igénylése, közérdekű adat igénylése, hatósági eljárás, kapcsolatfelvétel és nem utolsó sorban az adattovábbítás. Az adatkezeléssel kapcsolatban van néhány alapkötelezettség. Megfelelő tájékoztatás (jogi közlöny), az érintett, ill. törvény hozzájárulása az adatkezeléshez, technikai védelmi intézkedések, belső adatvédelmi szabályzat. A rendeltetésszerű működés elengedhetetlen feltétele néhány fontos tanács, így pl. a személyes és telefonos beszélgetést az ügyféllel ne halhassa illetéktelen személy, a feleslegessé vált adatokat meg kell semmisíteni, mielőtt kidobják őket. Minden új adatkezelést be kell jelenteni az Adatvédelmi Biztosnál.

6.6.5 Adathozzáférés, Active Directory

Az önkormányzat dolgozói számára különböző jogosultságokat osztanak ki, ezzel meghatározzák azt, hogy melyik belső felhasználó milyen adatokhoz férhet hozzá. A belső informatikai szabályozás hozzáféréséhez a jegyző ad engedélyt, míg a belügyminisztériumi (a továbbiakban BM) adatokhoz szükség van a jegyző valamint a BM hozzájárulására is. Az önkormányzaton belüli felhasználók jogosultsága egyébként munkakörfüggő. A hálózati jogosultságok kezelése a Microsoft Active Directory struktúráján keresztül történik. A rendszer menedzselése az önkormányzat rendszergazdájának és az IT szakemberek feladata. Az AD lehetővé teszi a belső hálózat minden publikált erőforrásának (fájlok, adatbázisok, felhasználói csoportok, perifériák stb.) központosított adminisztrálását és a rendszergazda számára egy központosított felügyeletet. A szoftverek és szoftverfrissítések is ezen keresztül történnek. A hivatalon belül csak érvényes írásbeli igazolással (licence szerződés) ellátott programok telepíthetők.

Az Active Directory-ba történő felvétel, törlés, módosítás a szervezeti egység vezetőjének kezdeményezésével és csak a Hivatal vezetőjének engedélyével történhet meg. Ezzel meghatározzák, hogy például a főkönyvi könyvelési rendszerbe, vagy a pénzügyi rendszerbe melyik szervezeti egységen belül ki férhet hozzá. Tehát jogosultságot csak az osztályvezető adhat, a jegyző felülbírásával, de például a gépjármű nyilvántartási rendszerhez történő hozzáféréshez belügyminisztériumi együttműködés is szükséges.

A hivatalon belüli felhasználók elleni védelem nincs megoldva a pénzhiány miatt, ezért a már meglévő szoftvereket próbálják alkalmazni. Ha a felhasználó névvel törölnek valamit, akkor csak a felhasználó név tulajdonosát vonhatják felelősségre. Ezt a jogszabályt most próbálják bevezetni. Tegyük fel, hogy egy irodában többen dolgoznak. Minden munkatársnak személyre szóló felhasználó neve és jelszava van, személyre szabott jogosultságokkal. Ha egy kávé vagy ebédszünet alkalmával, a felhasználó nem jelentkezik ki

és az ő számítógépén keresztül törölnek valamilyen fontos adatot, akkor jelen pillanatban még nem őt terheli a felelősség. Az új szabály bevezetésével arra kívánják majd ösztönözni a hivatalban dolgozókat, hogy ne adják ki a felhasználó nevüket és jelszavukat, valamint ha bekapcsolva hagyják a rendszert, legalább jelentkezzenek ki belőle. Jelenleg olyan megoldás létezik, hogy a hozzáférés géphez kötött és bevált módszer az autókiléptetés is, mely x idő múlva zárolja a gépet és visszalépésnél ismét be kell jelentkezni. Ez a beállítás egyébként az operációs rendszerek sajátossága tehát ez költségtöbbletet és új alkalmazásokat nem jelent a hivatal számára.

6.6.6 Irányelvek

A közigazgatás hatékonysága csak a résztvevők összehangolt, tervszerű működésével érhető el, melynek alapja az elektronikus közigazgatás rendszereinek összehangolása. Az alapkoncepció az, hogy a magyar közigazgatás minden szerve egységes elvekre és szabványokra épülő, hatékonyan és biztonságosan működő rendszereket működtessen. Célja a közszolgáltatások közötti átjárás biztosítása, melynek hiánya az elektronikus közigazgatás akadályozó tényezője.

A város önkormányzata 100%-osan követi az általános informatikai irányelveket. Az internet felhasználás követi az RFC szintű szabványokat (Request for Comments).

A régi és az új városháza között van egy átjáró, a két épület biztonsági és szerkezeti felépítése azonos elvekre épül. A többi szabvány viszont alkalmazásfüggő. Figyelembe veszik azokat az elveket, melyek már jól beváltak az adott probléma felmerülése során, de természetesen minden ajánlást vagy bevált gyakorlatot a saját lehetőségükhöz mérten használnak ki.

Az online közigazgatás egyre jobban felértékeli az informatikai biztonság kérdéskörét. Fontos megemlíteni az ebben a tárgyban előkészített törvényt:

- kimondja a személyiséglopás bűncselekménnyé nyilvánítását
- meghatározza a kritikus infrastruktúrák informatikai elemeinek védelmével kapcsolatos feladatokat
- meghatározza a hálózatbiztonsággal összefüggő tevékenységeket
- kimondja, hogy 2010. január 1-től a közigazgatásban csak informatikai biztonsági szempontból auditált elektronikus szolgáltatások indíthatók.

6.7 Az e-azonosító rendszerekkel kapcsolatos problémák[13]

Minden fejlett ország, így hazánk polgárai számára egyre nagyobb gondot okoz az azonosítók elburjánzása. Az államigazgatásnak és az azonosításban érdekelt más szereplőknek is nagyon sokba kerül a különböző azonosítók elkészítése és menedzselése. Technológiai szempontból lehetőség lenne egyetlen, vagy néhány biztonságos azonosító

[13] A 6.7 és 6.8 fejezetek „Az információs társadalom fejlődése és a munkaerőpiac”, szerkesztő: Dr. Várhelyi Tamás; Kiadó: Debreceni Egyetem Informatikai Kar és Debreceni Lokálpatrióta Egyesület, 2007 III. fejezetének felhasználásával készült.

elkészítésére. A ma használatos azonosítást kezelő rendszerek azonban távol vannak attól, hogy az azonosítók tetszőleges kombinációját, azaz valaminek a tudását, valaminek a birtoklását és a biometrikus azonosítókat továbbá tetszőleges kriptográfiai tokeneket (smart kártyákat, biztonságos memória kártyákat és biztonságos platform modulokat (trusted platform module), mobil alkalmazásokat, szövetségi azonosítókat (federated identities) felhasználó barát módon és a személyiségi jogok tiszteletben tartásával kezeljék.

A polgárok egyre gyakrabban találkoznak az e-egészségüggyel, e-közigazgatással, és más e-szolgáltatással. Ezeket egyszerűen, de a személyiségi jogokat tiszteletben tartó módon szeretnék használni. A smart kártyákat széles körben alkalmazzák azonosításra, például e-egészségügyi kártyaként, elektronikus bankkártyaként. Ezeknek a kártyáknak az a közös tulajdonsága, hogy – legalábbis elvileg – alkalmasak digitális aláírásra és a kártyák hitelesítésére is. Alternatív megoldást jelentenek az USB-alapú tokenek, az RFID-k és a mobiltelefonok.

Digitális azonosítót sok szervezet állít ki, pl. munkáltatók, bankok, szolgáltatók, de az államigazgatás által kiadott azonosítók helyzete különleges. Csak az államigazgatás képes ugyanis az egész populáció regisztrálását, a biometrikus azonosítók összegyűjtését és az adatok ellenőrzését elvégezni és ezeknek a bonyolult feladatoknak a költségét viselni. Az azonosítás menedzsment olyan biztonságos, mint az egész rendszer leggyengébb láncszeme, azaz az adatok felvétele. Olyan általános érvényű és megbízható adatfelvételt, mint amilyet az államigazgatás végre tud hajtani, a társadalmi élet más szereplőitől nem várhatunk el.

Ezért valószínű, hogy az állam által kiállított digitális azonosító kitüntetett szerepet fog játszani a belátható jövőben is az azonosítás menedzsmentben. Egyrészt közvetlenül használhatják azokat azonosításra, vagy pedig másodlagos azonosítók hitelesítésére, amint már ma is történik a hitelesítő szervezetek (CA) adatfelvételekor. Az Európai Unió tagjai már bevezették az elektronikus azonosítókat vagy tervezik azok rövid időn belüli bevezetését. Tekintettel arra, hogy az eID-ra vonatkozó formális szabványok (CEN TS 15480 "European Citizen Card" és ISO 24727) ratifikációs eljárása még folyamatban van és nem várható, hogy lefedi a kérdéskör minden ága-bogát, a jelenleg használt eID-k területén nagyon sok és lényegesen különböző megoldást találunk.

A 2005. novemberi Manchesteri Miniszteri Nyilatkozattal az identitás menedzsment interoperábilitásának fontossága formálisan is politikai szintre emelkedett. Az információs társadalmunk fejlődésének kulcsfontosságú elemeként ismerték el azt. A nyilatkozat elismeri a nemzetek autonómiáját a nemzeti azonosító dokumentumok és eID-k kiadásának tekintetében. Az i2010 akció terv és az eID roadmap, amelyeket a tagországok és az Európa Tanács dolgozott ki, ezért szükségesnek tekinti szövetségi interoperábilis eszközök kidolgozását a határokon átnyúló kormányzati szolgáltatások támogatására. Egy ilyen eszköznek a következő feladatokat kell ellátnia:

- Tetszőleges felhasználó azonosító kezelése (tudás, birtoklás és biometrikus azonosítók tetszőleges kombinációja).
- Mindenféle kriptográfiai token kezelése (smart kártya, biztonságos memória kártya, stb.) .

- Szövetségi azonosítók felhasználó barát és a személyiségi jogokat tiszteletben tartó kezelése.
- Biztonságos mobil hálózatok kiszolgálása mobil eszközök között.
- Tetszőleges hardveren bizalmas számítási környezet kiszolgálása.

A fejlődés trendje alapján várható, hogy a jövőben a mobil eszközök támogatni fogják a közeli kommunikációt (near field communication, NFC) és a felhasználók sok, különféle kriptográfiai tokent is használnak majd. Az eszköz és a külső kriptográfiai tokenek közötti kommunikáció fontos szerepet fog játszani a mindenütt használható eID szempontjából. Ezért a közeli kommunikációra nagyon fontos olyan megbízható protokollok kidolgozása, amelyek tetszőleges kriptográfiai tokenek alkalmazását támogatják.

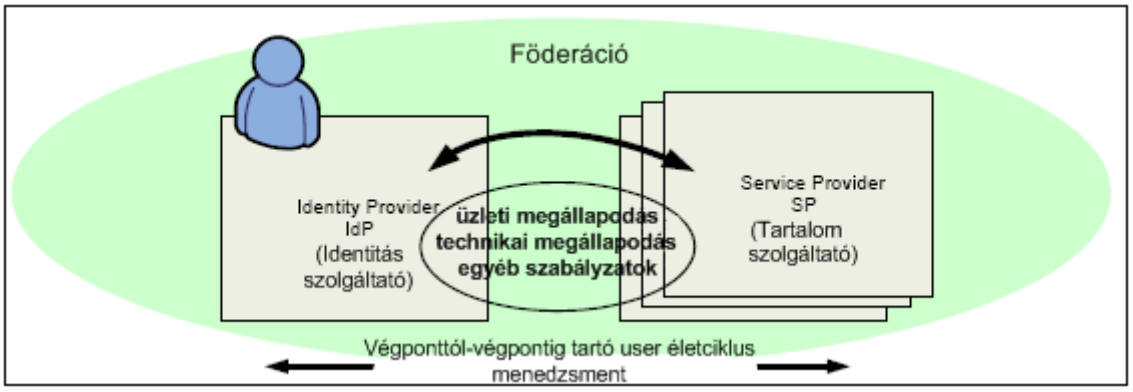
7 Szövetségi ID menedzsment

Ebben a fejezetben felhasználjuk Perge Zoltán, *Szövetségi (Federated) Azonosítás*, Debreceni Egyetem Informatikai Kar, 2009 szakdolgozatát.

Mint arra már korábban rámutattunk a polgárok egyre gyakrabban kerülnek olyan helyzetbe, amikor azonosítani kell magukat. Azonosítóik száma és bonyolultsága nem teszi lehetővé, hogy azokat fejben tartsák. A távmunka elterjedésének következtében gyakran többféle szerepben is azonosítani kell magukat egyidejűleg, a szerepeik gyorsan változnak. A hagyományos azonosítási technikák az ilyen emberek munkáját nagyon megnehezítik. Ezt az igényt észlelve dolgozták ki a *szövetségi ID menedzsment* (federated identity managment) eszközöket, amelyek lehetővé teszik, hogy a felhasználó egyetlen belépési ponton azonosítsa magát és innen lépjen tovább újabb azonosítási procedúra nélkül más védett munkahelyekre.

A szövetségi ID menedzsment feladata az, hogy biztonságos kapcsolatot teremtsen különböző biztonsági és védelmi rendszerrel rendelkező szervezetek között. A szövetségi ID eszközzel egy absztrakciós réteget implementálunk azonosító és biztonsági területek fölé. Szabványos módszereket használva minden terület megoszthatja a lokális azonosító és biztonsági információit, egyidejűleg azonban megtartja saját könyvtárait, metakönyvtárait, azonosító kezelését és nyilvános kulcs infrastruktúráját.

A szövetségi azonosítás technológiát globálisan együttműködő online azonosítási célokra, kapcsolatok vezetésére és cégek közti rokonsági alapú üzleti modellek létrehozására használják. Az ötlet lényegében nem új, minthogy vannak valós világbeli modelljeink az egyének szövetségi azonosításra – az útlevél nemzetközileg használható a személyazonosság igazolására; a bank kártya a tulajdonos bankszámlájáért kezeskedik; a vezetői engedély pedig a személy azon képességét igazolja, hogy tud gépjárművet vezetni és szintén használható személyazonosításra.



7.1. ábra – Szövetségi azonosítás menedzsment

Szövetségi azonosítás menedzsmentet az üzleti-, technikai megállapodások és szabályzatok alapozzák meg, melyek lehetővé teszik a cégeknek, hogy együttműködjenek a

megosztott azonosítás menedzsment megoldásokkal. Ezáltal a szervezetek csökkenthetik az azonosítás menedzsment költségeiket és nagyobb felhasználói élmény[14] érhető el. Hordozható azonosítókat használ, ezáltal egyszerűsítve a felhasználók adminisztrációját, illetve a biztonság és bizalom kezelését a szövetséges üzleti kapcsolatokban. Az adminisztráció és az életciklus management leegyszerűsödése a föderációban a következő eredményekhez vezet:

- Az azonosítás menedzsment költségek csökkenthetők mivel a cégeknek nem kell többé a *felhasználók* és *azonosítóik* kezelésével foglalkozniuk, beleértve az adminisztrátorok delegálását is. A szervezeteknek csak az adatokhoz való hozzáférési jogosultságokat kell kezelni.
- A felhasználói élmény azáltal növekszik, hogy a felhasználó könnyedén navigálhat a web-oldalak között miközben globálisan bejelentkezve marad.
- A végponttól-végpontig tartó biztonsági és bizalmi lehetőségeket hasznosítja a federáción belüli vállalatok-közti alkalmazás integráció.

Az integráció azért egyszerűsödhet, mert egységes út vezet a cégek, az alkalmazások és a hálózati identitások között. A szervezetek olyan üzleti stratégiákat valósíthatnak meg, amelyek pozitívan befolyásolják a piacot és az ügyfelek számának növekedéséhez vezet azáltal, hogy kiküszöbölik a súrlódást, melyeket az összeférhetetlen azonosítás- és biztonság menedzsment megoldások okoztak.

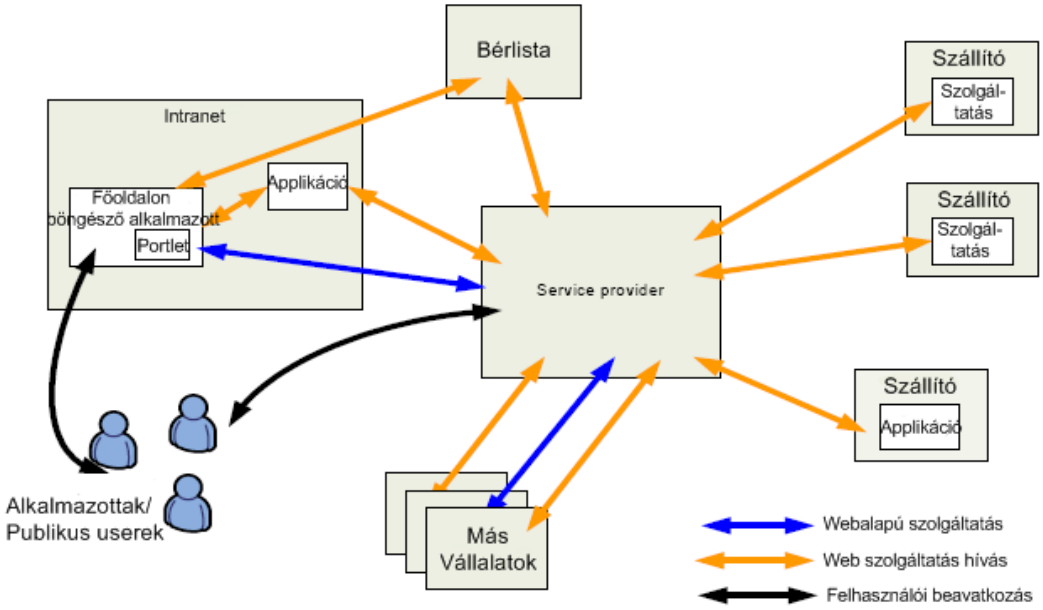
7.1 Magas szintű példa az újracsoportosításra

Világunkban egyre több szolgáltatás elérhető el a technológiának köszönhetően, beleértve olyan területeket ahol bizalmas és érzékeny információk cseréjére is szükség van. Az erőforrás (szolgáltatás) felhasználás jelenlegi reaktív megközelítése nem elégíti ki a valós-idejű, gördülékenyen csoportosuló szolgáltatás elvárásait mely optimálisan követni a változó piaci feltételeket.

A 3. ábra néhány integrációs pontot mutat be, amelyeket a szolgáltatások újra csoportosításának részeként úgy kell újra szervezni, hogy támogathassák az új vagy már meglévő üzleti folyamatokat. Egy cég (az ábrán Intranet) kiszervezi a bérszámfejtés (Bérlista) valamint a telekommunikációs és az azzal kapcsolatos szolgáltató részleg adminisztrációját (egy tartalomszolgáltatóhoz, Service Provider). A tartalomszolgáltatónak hasonló kapcsolatai vannak más ügyfelekkel, illetve a saját szállítóival. Megjegyzendő, hogy a Service Provider szintén kiszervezi a bérszámfejtést a Bérlista entitáshoz.

---

[14]User Experience **- Így nevezzük azokat a benyomásokat, élményeket, amiket egy felhasználó egy termék vagy rendszer használata során szerzett. Forrás: http://www.fatdux.com/hu/what/what-is-ux/**



7.2. ábra: A vállalatok összetett értékhálóvá szervezése megsokszorozza a kapcsolatokat

Ilyen kapcsolatok manapság számos vállalatnál léteznek, de nehéz feladat ezeket implementálni, testre szabni, fenntartani. Például egy üzleti folyamat, melyet szervezeti tartományokon keresztül kell megosztani, olyan problémákat vet fel, mint például a vállalati határokon keresztüli munkafolyamat. Az adminisztrációs szolgáltatónak magába kell gyűjtenie a szállítói által nyújtott szolgáltatásokat, egyesített szolgáltatások egy egybefüggő halmazába, mellyel cserébe ellátja annak ügyfeleit.

Ezen kívül a szállítónak gondoskodnia kell arról és garantálnia kell azt, hogy az információ biztonságos, szegmentált és bizalmas legyen az ügyfelek között. A szállítóknak egymással is kell kommunikálniuk az alkalmazottak érdekében, betartva a titoktartási előírásokat. A szállítónak ismernie kell a felhasználóit, akik száma nagy lehet.

A komplex dinamika és kollektív magatartás megértése és irányítása egyre fontosabbá válik, hogy elkerülhető legyen a rendszer instabilitása és azért, hogy előkészítse a terepet a globális és lokális optimalizálásnak. Alapjaiban új megközelítést kell tehát implementálni, hogy biztonságos alapot szolgáltasson az igények kielégítését, mely magában foglalja a föderációt és az elhatárolódást is.

Tehát az interakciók, melyek eleget tesznek az új üzleti folyamatok elvárásainak, keverékei lesznek az alkalmazás-alkalmazás és felhasználói interakcióknak, melyek igénylik a szövetségi azonosítás menedzsment képességeinek teljes tárházát, hogy kezelhessék az azonosítás, bizalom és biztonság problémáit.

7.2 A szövetségi azonosítás menedzsment és vállalatok fejlődése

A vállalatok mérete és szervezete a környezet hatásaira reagálva folyamatosan változik. Növekednek, ha olyan terméket sikerül piacra vinniük, amelyre sok igény van, leépítenek, ha az igény csökken. Szervezeti egységek olvadnak össze vagy válnak szét. Vállalatok összeolvadnak, leányvállalatokat hoznak létre. Sok más esemény is történik a vállalatokkal, de nem ezek elemzése a feladatunk. Az informatikai rendszerek, így azok fontos alkotórészei az azonosító és jogosultságkezelő mechanizmusok is, követik a vállalatban bekövetkezett változásokat. Az alábbiakban felsorolunk néhány tipikus példát vállalatok változásaira, amelyek követésére a szövetségi azonosítási menedzsment különösen alkalmas.

Egyesülés és felvásárlás. Ilyen esetben a cég növekedési stratégiája a más cégekkel való egyesülésen illetve azok felvásárlásán alapszik. Ekkor a siker kulcsa az, hogy milyen gyorsan tudják a cégek az IT infrastruktúrájukat összekapcsolni, ezáltal nyerve új ügyfélkört. Ilyen esetekben az ügyfelek azonosítás menedzsmentjének újraszervezése az egyik legkomplexebb probléma. A hagyományos megoldás szerint a megszerzett ügyfeleknek külön fiókot hoznak létre, amelyet az ügyfeleknek aktiválni kell. A szövetségi azonosításon alapuló integrációs stratégia egyszerűsíti a felhasználói élményt, ő legfeljebb a számla fejlécéből veszi észre, hogy a partnere megváltozott. Az egyesült cégek felhasználói könnyedén elérhetik minden cég erőforrásait, szolgáltatásait; az integráció során megoldott a gyors és akadálymentes ügyfél integráció.

Cégek közti együttműködés. Nagy cégnek vannak önállóan működő üzleti egységei, amelyek kapcsolatot tartanak az anyavállalattal, egymással és saját ügyfeleikkel is. Ennek okai lehetnek a szervezeti struktúra, regionális politikai megfontolások vagy a versenytársak. Egy nemzetközi termelő cégnek például lehetnek regionális képviseletei Amerikában, Európában, Ázsiában stb. és e képviseletek alkalmazottainak szüksége lehet a másik képviselet erőforrásaira. A szövetségi azonosítás menedzsment lehetővé teszi az üzleti egységeknek, hogy fenntartsák a függetlenségüket és rugalmas utat biztosít az adatok, erőforrások megosztására a vállalatok között.

Vásárlói bázis növekedése. Egy terjeszkedő cég stratégiája lehet az, hogy egyenként szerez új ügyfeleket, ami lassú megoldás. Választhat azonban olyan megoldást is, hogy társul egy másik céggel, amelyeknek meg akarja szerezni az ügyfeleit. Ilyenkor egyszerre sok új ügyfélre tehet szert. Például egy pénzügyi szolgáltató társul egy mobil távközlési szolgáltatóval (akinek milliónyi előfizetője van), hogy elektronikus számlázási szolgáltatást nyújtson az ügyfeleinek, papíralapú helyett. Az ösztönzés a mobil szolgáltatónak ebben a társulásban az, hogy a kiadásait csökkentheti a számlázási feladatok kiszervezésével a pénzügyi szolgáltatóhoz. Cserébe a mobil szolgáltató engedményt ajánl az új e-számlázási szolgáltatásra előfizető ügyfeleknek. Ezen társulással a pénzügyi szolgáltató egy millió új ügyfélre tett szert, akik az új szolgáltatás lehetséges előfizetői. A szövetségi azonosítás menedzsment lehetővé teszi a pénzügyi szolgáltatónak, hogy új ügyfélbázisokat érhessen el, akiknek már meglévő saját identitásuk van. (A különböző identitás menedzsment megoldásokkal rendelkező cégek közti integrációt megoldva.)

Kiszervezett szolgáltatások. Az alkalmazottak önkiszolgálása elsődleges feladat olyan vállalatoknál, akik csökkenteni kívánják a felhasználók kezelésével kapcsolatos költségeiket. Ilyenkor kiszervezik a nem kritikus kompetenciákat külső (harmadik fél) szolgáltatókhoz

Ilyenek kompetenciák lehetnek például az emberi erőforrás menedzsment, nyugdíjpénztár, egészségbiztosítás, utazás stb. A vállalati intranetes portál segítségével elérhetők ezek a külső szolgáltatók, így a vállalatnak csak a kiszervezett szolgáltatások adminisztrációját kell ellátnia. Az alkalmazottakat azonban nem kapcsolhatják közvetlenül a szolgáltatókhoz, így szükséges információs szolgálatot (help-desket) fenntartani az alkalmazottak iktatásához a magánnyugdíj-pénztár, egészségbiztosítás és bérlista szolgáltatásokhoz. A munkáltatók jelentős összeget fordítanak e szolgáltatások adminisztrációs költségeinek megtervezésére, de végül mégis a vállalatnak magának kell adminisztrálnia ezeket a szolgáltatásokat vagy alkalmaznia kell ügyfélszolgálati személyzetet.

A szövetségi azonosítás menedzsment lehetővé teszi az alkalmazottaknak, hogy elérjék és kezelhessék az adataikat a különböző tartalomszolgáltatók Web oldalain, az alkalmazotti portálon való bejelentkezést követően. Egy már meglévő portál használata egyszerűsíti a felhasználói élményt és lehetővé teszi, hogy a felhasználó elérje a különböző szolgáltatók weboldalait anélkül, hogy az üzleti partnereknél regisztrációt vagy hitelesítést igényelne. A munkáltató csökkentheti az alkalmazott támogatás- és adminisztrációs költségeket azáltal, hogy a dolgozók közvetlen elérhetik a szolgáltatókat.

Tartalomszolgáltatás automatizálása. Egy nagyobb tartalomszolgáltatónak, aki alkalmazottak magán nyugdíjpénztári accountjait kezeli, az ügyfelek alkalmazottainak felhasználói életciklus kezelése jelentős költségeket jelent. Ezek a költségek az ügyfelek alkalmazottainak account regisztrációjából és kezeléséből, a jelszavak kezeléséből illetve ügyfélszolgálat fenntartásából erednek. Ilyen feladatok például a felhasználók elfelejtett jelszavainak és bejelentkezési adatainak kezelése.

Tételezzük fel, hogy egy ilyen új jelszó kérő hívás $20-ba kerül, és adott egy tartalomszolgáltató, aki 100 ügyféllel rendelkezik, és minden ügyfélnek átlagosan 10000 alkalmazottja van. Ha ezeknek csak a negyede évente egyszer elfelejti a jelszavát, az 5 millió dolláros kiadást jelent az account és jelszókezelésben.

A tartalomszolgáltató tehát jelentősen érdekelt a szövetségi modellre váltásban ahol a szolgáltató felhasználja az alkalmazottak hitelesítését az egyesített portálon, így hozzáférve a szolgáltatásaikhoz. Ebben a modellben a munkáltató (ügyfél) felelős a felhasználóinak és jelszavainak kezeléséért, amely egyébként is a feladata, tehát nem jelent plusz költséget, a tartalomszolgáltató pedig az ügyfeleire hárítja a felhasználói adminisztráció költségeit. Ez a megközelítés az alkalmazottak számára is kedvező mivel nem kell több helyre is regisztrálnia illetve több jelszót fejben tartania, hogy kezelhesse a magán nyugdíjpénztári és egészség biztosítási adatait.

Portál alapú integráció. Az internet alapú szolgáltatók új generációja, a vállalatoknak és cégeknek kínál szoftver-mint-szolgáltatás (SaaS) megoldásokat. Ilyen szolgáltatók például WebEX, Salesforce.com, Travelocity.com és így tovább. Ezek a szolgáltatások lehetővé teszik, hogy a vállalatok hozzáférjenek Interneten hosztolt szolgáltatásokhoz anélkül, hogy az IT infrastruktúra költségeit magára kellene vállalnia melyet e szolgáltatások helyi kezelése jelentene. A szövetségi azonosításnak kritikus szerep jut ebben a rendszerben, mert lehetővé teszi a cégek alkalmazottainak, hogy különböző szoftver alapú szolgáltatásokat érjenek el a saját munkahelyi belépési azonosítóikkal. Miközben egyre több vállalat szervezi ki a nem

létfontosságú üzleti szolgáltatásait, a szövetségi azonosítás menedzsment tölti be egy olyan identitás integrációs technológia szerepét mely segítségével a felhasználók akadálymentesen elérhetnek harmadik-személy által nyújtott szolgáltatásokat melyek lehetnek helyileg- vagy távolról hosztoltak.

Közigazgatási együttműködés. A közigazgatásban nagy az igény a hatékonyságra és az együttműködésre. A folyamatok több kormányzatot, intézményt és hatóságot áthidalhatnak különböző régiókban, ahol szükséges az adatok megosztása, de politikai, intézményi vagy egyéb okokból nincs lehetőség az integrálódásra vagy egyesülésre. Az entitásoknak, a felhasználóik számára szükséges lehet a kormányközi entitások erőforrásainak elérhetővé tétele. Például egy európai ország valamely hatóságának szüksége lehet lényeges információra egy személyt illetően egy másik ország adatbázisából, azonban ehhez szükséges lenne egy ország hatóságának a másik ország hatósági felhasználóit kezelnie. A szövetségi azonosítás lehetővé teszi, hogy a nemzeti hatóságok megőrizzék függetlenségüket és saját felhasználóik kezelését, miközben flexibilis megoldást kínál az adatmegosztására a nemzetközi entitásoknak.

7.3 Bizalomi kapcsolat és biztosítása

A szövetségi üzleti modellben nagyon fontos a *bizalmi kapcsolat* az együttműködők között. A szövetségi modellben a szervezet, amely elérést szeretne biztosítani egy identitásnak, akit nem ellenőriz a szervezet saját belső biztonsági eljárása. Ehelyett a szervezet megbízik egy harmadik személy kijelentésében az identitást tekintve. A megoldás tehát amely rizikót és a bizonytalanságot visz, az üzleti tranzakciók bizalmasságába. Egy szervezet nem köt szövetségi üzleti megállapodást, ha nincs rálátása az üzleti partner identitás és hozzáférés kezelési rendszerére és folyamataira. A szervezetnek meg kell becsülnie az üzleti partnerekkel való együttműködés kockázatát. Fel kell mérnie a partner üzleti folyamatait és ellenőrző eljárásait az üzleti partner identitásigazolása, akkreditációja és (jó) hírneve vonatkozásában. Ezek az intézkedések biztosítják az átláthatóságot és minőségi értékelést adnak arról, hogy a harmadik fél identitásai miként vonhatók be a hozzáférés vezérlésről és a bizalmi kapcsolat szabályairól szóló üzleti döntésekbe.

Az üzleti partner identitásigazolása az a folyamat melyben ellenőrzik a leendő szövetséges üzleti partner fizikai identitását, mind az online üzleti kapcsolat létrejötte előtt és mikor már elkezdték a futásidejű tranzakciókat. Az identitásigazolás része a vállalat fizikai identitásának ellenőrzése – de ki is a vállalat?

- Létezik-e az adott néven törvényes vállalat?
- Az üzleti partner küldi a kérést?
- Az adott dolgozó jogosult erre a kérésre?

Amikor az adott fizikai identitást leellenőrizték, valamilyen online token-t bocsátanak ki az üzleti partnernek, ezután pedig összekapcsolják a vállalat adott fizikai identitásával. Az üzleti partner identitás ellenőrzés különböző módjai használatosak, beleértve:

- Önazonosítás

- Meglévő kapcsolat felhasználása
- Elektronikus vagy postai levélcím megerősítése
- Identitás ellenőrzés

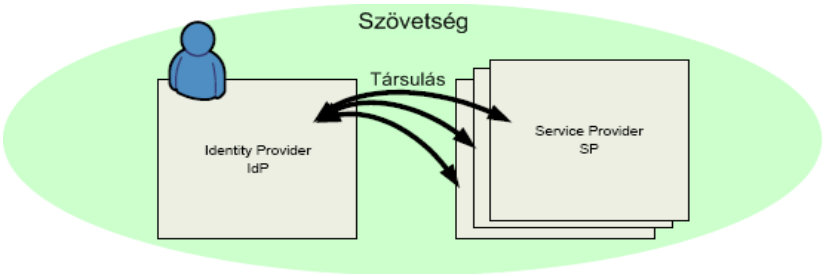
Az üzleti partner *akkreditáció*, arra a kérdésre ad választ, hogy mit tudunk a cégről? Különösen, hogy mit várhatunk ettől a cégtől? Az akkreditáció egy jól meghatározott szabályzaton alapul, mely azt írja le, hogy milyen elvárásoknak kell egy partnernek megfelelnie. Egy föderációt kiépíteni akaró cégnek ki kell adnia egy ilyen szabályzatot, ugyanígy a partner cégnek is meg kell határoznia mely kritériumoknak, felel meg a saját IT infrastruktúrája. A két szabályzat illeszkedésének kiértékelése egy megbízható harmadik fél feladata, aki az üzleti akkreditációra szakosodott. Példák a jellemzők típusára melyeket kiértékelnek az akkreditációs folyamatban:

- Hitelt érdemlő a vállalat?
- Jó hírnevű vállalatnak tartják a céget?
- A vállalatot elismerik a fontosabb szakmai szakszervezetek?
- A vállalat része a szövetségnek?
- A vállalat belépési azonosítóit szabványosított és megbízható formában bocsátja ki?

A *hírnév* alternatív eszköz arra, hogy értékelhető kiegészítő információt kapjunk a vállalatról. A fő különbség a jó hírnév és az akkreditáció között az, hogy az előbbit folyamatosan figyelik a vállalat tevékenysége alapján. Másik különbség, hogy a hírnevet tipikusan egy független entitás figyeli, és az alany nem vesz aktívan részt benne. A reputációmérésre napjainkban egy nyílt visszajelzés alapú mechanizmust használnak. A jó hírnév szolgálat általában egyszerű értéket határoz meg, melyet egy adott, könnyen érthető eljárás segítségével állapítanak meg.

7.4 Szerepek

Egy szövetségen belül, az üzleti partnerek két szerepet játszhatnak: *Identity Provider* (Identitás Szolgáltató, IdP) vagy *Service Provider* (Tartalomszolgáltató, SP) esetlegesen mindkettő. Az identitásszolgáltató a hitelesítő fél, hitelesíti a végfelhasználót és valamilyen megbízható formában identitást állít ki a usernek. Azok a partnerek, a tartalomszolgáltatók, akik szolgáltatásokat kínálnak, de nem identitásszolgáltatók. Az IdP vállalja magára a felhasználói életciklus menedzsmenttel kapcsolatos problémákat. A tartalomszolgáltató (SP) az IdP-re támaszkodik, hogy az hitelesítse a felhasználóval kapcsolatos információkat, és így az SP csak azon user attribútumokat kezeli melyek a saját maga számára fontosak.



7.3. ábra: Identitásszolgáltató és tartalomszolgáltató a szövetségi modellben

7.4.1 Identitásszolgáltató – IdP

Az IdP felelős az account létrehozásáért, a beállításokért, az azonosító és az általános account kezelésért továbbá gyűjtőpontként vagy kliensként viselkedik a megbízható identitásszolgáltatókhoz. Egy szövetségi partner, amely a felhasználók IdP-jeként működik, megszabadítja a többi partnert a felhasználókra vonatkozó ekvivalens adatokat kezelésének terhétől. Az SP-k az IdP-vel kialakított bizalmi kapcsolat alapján fogadják el az IdP által a felhasználókról kiadott hitelesített információkat. Ez lehetővé teszi a tartalomszolgáltatóknak, hogy az azonosítás és hozzáférés menedzsment költségeit átruházzák a federáción belül az identitásszolgáltatóra.

7.4.2 Tartalomszolgáltató – SP

A tartalomszolgáltatók védett tartalmakat szolgáltatnak a felhasználók számára. A felhasználók személyes adataival általában nem rendelkeznek, ezért nem szükséges a felhasználókat adminisztrálniuk sem.

A tartalomszolgáltató funkciói (a funkciók föderációs modelltől függően ezektől eltérhetnek):

- azonosított kapcsolat létrehozása az identitásszolgáltató segítségével
- az identitás szolgáltatótól kapott adatok értelmezése
- az identitás szolgáltatótól kapott adatok alapján meghatározni, hogy a felhasználó jogosult-e a művelet végrehajtására (autorizáció)

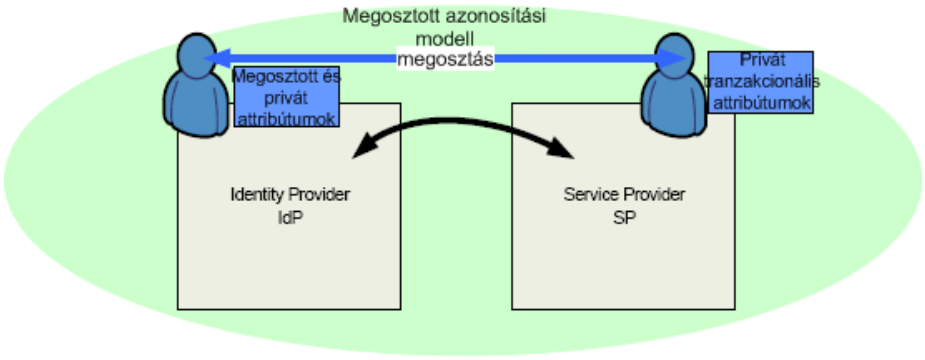
A tartalomszolgáltató kezelhet helyi információt a felhasználókról, még a szövetség kontextusán belül is. Például, belépve egy szövetségi azonosítás menedzsment kapcsolatba lehetséges, hogy egy tartalomszolgáltató átadja az accountkezelést, beleértve a jelszó menedzsmentet, egy IdP-nek míg az SP a saját user-specifikus adatok kezelésére fókuszál: például SP oldali szolgáltatás-specifikus attribútumok és a személyre szabással kapcsolatos információk. Általában a tartalomszolgáltató rábízza az azonosítás menedzsmentet az identitásszolgáltatóra, így minimalizálva az azonosítási követelményeket miközben változatlanul elérhetővé teszi a teljes tartalomszolgáltatói funkcionalitást.

7.5 Azonosítási modellek

Kétféle azonosítási modellt használnak a szövetségi azonosítási rendszerek. A megosztott és különálló azonosítási modellek az azonosítási adat menedzsment természetére utalnak. A megosztott azonosítási adat menedzsment megosztott volta arra utal, hogy az azonosítási információt egy üzleti partner kezelheti (az IdP). A különálló pedig, hogy az információ ismétlődik, és külön kezelik az üzleti partnerek között.

7.5.1 Megosztott

A megosztott azonosítás a szövetségi üzleti interakciókban akkor lehetséges, ha egy üzleti partner megbízik az identitásszolgáltató által a felhasználókról kiadott tanúsítványban. Ebben a modellben a szövetség lehetővé teszi a felhasználónak (és a federációs üzleti partnereknek), hogy létrehozzanak egy közös egyedi azonosítót, amellyel utalhatnak a felhasználóra. Az azonosító alapján az identitásszolgáltató képes kezelni a user azonosítási adatait, és ezen információ hiteles forrásaként működik a tartalomszolgáltatók számára.



7.4. ábra: Megosztott azonosítási modell

Az üzleti partnerek közötti azonosítás és attribútum kezelés módját tekintve alapvető kérdés, hogy milyen információk oszthatók meg és mik az előnyei a megosztásnak? A legoptimistább lehetőségként az IdP és SP minden információt megosztanak, ahogy az a 7.4. ábrán látható.

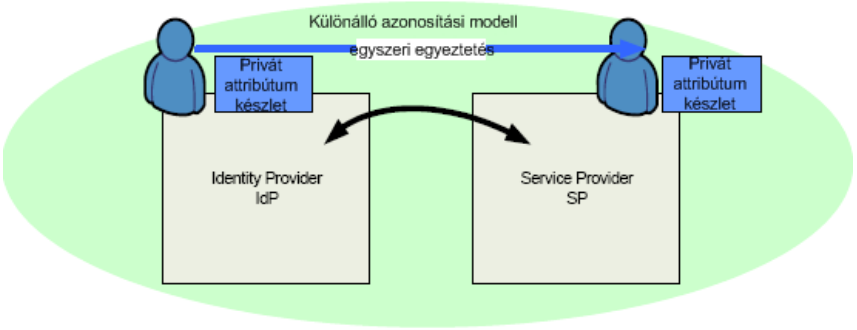
- A *bejelentkezési adatok megosztása* az identitásszolgáltató és a tartalomszolgáltató között azt jelenti, hogy a SP megbízik az IdP felhasználó hitelesítésében, így mentesítve a SP-t a jelszavak és felhasználónevek tárolása alól. Ha az account adatai nem oszthatók meg akkor mind az IdP-nek mind az SP-nek külön kell a felhasználói accountokat kezelni.
- A *tranzakció attribútumok megosztása* igényli, hogy az IdP és SP megegyezzen a szerepekről, jogosultságokról vagy a felhasználó csoporthoz tartozásáról. Ezt nehéz megvalósítani, mivel két egymástól független szolgáltató jellemzően különböző megoldást alkalmaz az identitások csoportosítására illetve a szerepek információinak kezelésére. A tranzakciós attribútumok megosztása helyett, egy szolgáltató leképezheti a tranzakciós

attribútumaikat, olyan alakban, amit az üzleti partnere megért. Ebben a megközelítésben az azonosítási meta-adatot külön kezelik az IdP-nél és az SP-nél.

- A *profil attribútumok megosztása* az IdP és az SP között általában felhasználói beleegyezés kérdése. A felhasználó preferenciáitól és a bizalmassági igényeitől függ. Ezen attribútumok megosztásához szükség van felhasználói beleegyezésre illetve képesség ennek igazolására. Gyakorlati értelemben, bizonyos attribútumok megoszthatók (ilyen például az e-mail cím) míg más attribútumok nem. Ha nem megoszthatóak, akkor duplikálni kell őket az IdP-nél és az SP-nél is. Ha például egy felhasználó lakás címe duplikálva van, akkor külön kell kezelnie az üzleti partnereknek. Ha a felhasználó elköltözik és az egyik üzleti partner ismeri az új címet, a különálló azonosítási modellben, az üzleti partner nem oszthatja meg ezt az információt a partnereivel.

7.5.2 Különálló

A különálló megközelítést a szövetségi üzleti interakciókban akkor alkalmazzák, ha a két szervezet nem oszthat meg bizonyos azonosítási információkat. Ennek oka lehet adatok elkülönítése, közvetítés-ellenesség (verseny okok miatt a vállalatok nem szeretnék megosztani az ügyfél-információkat), politikai okok vagy, mert a felhasználónak mindkét szolgáltatóval van külön kapcsolata.



7.5. ábra: Különálló azonosítási modell

A különálló azonosítási adat menedzsment modellben, azonosítási adatok kezdetben egyeztethetők az üzleti partnerek között a kezdeti account beállítás részeként, habár később külön fogják kezelni őket.

7.6 Szabványok és törekvések

Az egyszerűsített bejelentkezési technikákat és megoldásokat már évek óta alkalmazzák. A szövetségi azonosítás menedzsment gyökerei a bejelentkezési technológiában vannak. Ebben a fejezetben először áttekintjük az eID kialakítására tett kísérleteket, majd bemutatjuk a SAML (Security Assertion Markup Language) nyelvet.

7.6.1 Az eID szabványosítása

A szövetségi ID menedzsment szükségessé teszi, hogy az egyedi azonosítók mellett a polgároknak legyen egyetlen nagyon biztonságos eID-je is. Ilyen eID kidolgozására és bevezetésére, mint arra fentebb rámutattunk csak az államok képesek. Természetesen ezen a területen is szükség van az Európai Unió országainak együttműködésére, tapasztalataik, eredményeik folyamatos kicserélésére.

Állami eID kiadása Európában még kezdeti stádiumban van. Észtország volt az első, amelyik a teljes lakosságot ellátta ilyen azonosítóval. Az első védett szolgáltatások tipikusan a kormányzat területén maradnak és nemzeti jellegűek. Tekintettel arra, hogy sokszor az államot megelőzve, a vállalkozói szektor is jelentősen növeli az Interneten keresztüli online szolgáltatásait, valamint az Európai szolgáltatási piacnak komoly határokon átnyúló terjeszkedési lehetősége van, várható a privát szektor részéről is, hogy igénybe veszi az interoperábilisan használható, nemzetközi eID-t.

Információs társadalmunk gazdasági potenciálja csak akkor használható ki teljesen, ha az eID-t szektor semlegesen és minden körben használjuk. Meg kell jegyeznünk ugyanakkor, hogy az eID használati körének kiterjesztésével arányosan növekedik a veszélyforrások száma is. Jelenleg a kormányok az eID-t állami alkalmazások kiszolgálására kívánják használni, következésképpen nem nagyon foglalkoznak azzal a járulékos veszélynövekedéssel, amelyet a szélesebb körű használat jelentene. Az eID teljes körű elterjedését tehát csak egy hosszabb evolúciós folyamat eredményezheti. Az eID-vel kapcsolatos veszélyek elhárításának leghatékonyabb stratégiáját ezért ezen az evolúciós modellen lehet vizsgálni, az elemzést a jelenlegi helyzettel kell kezdeni és kiterjeszteni az elemzést a folyamatosan javuló veszély menedzsmentre. Az egyik kulcsfontosságú veszély a polgárok privát szféráját jelenti. Különösen fontos itt az alábbi két terület:

(i) személyre vonatkozó, egymástól független adatok összekapcsolása,

(ii) szükségtelen személyes adatok ellenőrizetlen közlése, terjesztése.

A mai állami eID-k többsége nem tudja ezeket a problémákat kezelni, így veszélyes lenne azoknak a széles körű elterjesztése. Például nagyon sok eID a polgárt a személyi igazolványában található egyedi, nemzeti jelsorozattal azonosítja. Határozott intézkedések nélkül könnyen előállhat az a helyzet, hogy nagyon sok, egymástól független adatbázis alkalmazza ezt az azonosítót az adatokhoz való hozzáférés kulcsaként és így lehetőséget teremt független adatok összekapcsolására. Tekintettel arra, hogy az adatbázisok kulcsát később nehéz megváltoztatni, ezért az ilyen helyzetet már kezdeti stádiumban el kell kerülni.

Az osztrák polgárkártya jó példa arra, hogy hogyan kerülhető el az eID-k alkalmazása során az összekapcsolhatóság. Ez a kártya dinamikus, szektor illetve alkalmazás specifikus személyi azonosítókat alkalmaz, amelyeket egyetlen, az eID-ben található gyökér-azonosítóból vezetnek le. Tekintettel arra, hogy ez a megoldás nem alkalmazható közvetlenül az X.509 eID-vel, ezért ezt a megoldást nem lesz könnyű kiterjeszteni más országra. Belgiumban, hazánkhoz hasonlóan, törvény tiltja, hogy a személyi számot adatbázisok kulcsaként tárolják.

A legtöbb ma használatos eID nagyon kevés ellenőrzési lehetőséget ad a személyes adatok ellenőrizetlen közlésére, terjesztésére. A személyes adatokat tipikusan egy hitelesítés tartalmazza és egy vagy több személyes adatot tartalmazó fájl található smart kártyákon. Személyes adatok ezen autentikus forrásának eredményes és biztonságos alkalmazásaihoz ma még hiányoznak a szabványos hozzáférési és jóváhagyási mechanizmusok. A hitelesítésben található adatok jelentik minimális kiolvasható információk körét. Amennyiben a hitelesítésen túl további adatokra van szükségünk, akkor a mai eID-k nagyon kevés ellenőrzési lehetőséget tartalmaznak az adatok kiolvasására. Az adatfájlokat csak egységes egészként lehet továbbítani, származtatott adatok közlésére, mint például egy korcsoporthoz való tartozás a születési dátum helyett, egyáltalán nem lehetséges.

7.6.2 Security Assertion Markup Language (SAML)

Az elsődleges szövetségi ID szabványért versengő jelöltek közül pillanatnyilag a SAML (Security Assertion Markup Language) tűnik a legesélyesebbnek. A SAML az OASIS Security Services Technical Committee nemzetközi konzorcium terméke. 2001-ben kezdték meg az XML alapú eszköz kidolgozását. 2002-ben készült el az első verzió V1.0. Később a Liberty Alliance, amely vállalkozások, non-profit és állami szervezetek konzorciuma javaslatot tett a SAML szabvány kibővítésére, amelyet Liberty Identity Federation Frameworknek (ID-FF) neveztek el. A Liberty ID-FF is szabványos, területek közötti, web-alapú és egyetlen azonosítást igénylő rendszer. Ezen felül a Liberty bevezette a bizalmi kör fogalmát, amely szerint minden résztvevő megbízható abban a tekintetben, hogy pontosan dokumentálja a felhasználó azonosítás eljárásait, a hitelesítő eljárások típusait és azokat a szabályokat, amelyek a hitelesített igazolványokkal kapcsolatosak. A bizalmi kör tagjai ellenőrizhetik egymást, hogy betartják-e az előírásokat. A tapasztalatokat figyelembe véve az OASIS is kibővítette a SAML nyelvet és 2005-ben megjelentette annak V2.0 verzióját, amely máig érvényes.

Az Európa Tanács szakmai szervezete, az IDABC (Interoperable Delivery of European eGovernment Services) jelenleg értékeli a szövetségi azonosítást menedzselő rendszereket, többek között a korábban már említett Liberty Alliance ID FF, WS-*, és TLS-Federation termékeket.

A következőkben a szövetségi azonosítás menedzsmenttel kapcsolatos szabványokat, mutatjuk be, a SAML-t, mint az egyik legalapvetőbb szabványt részleteiben is. A SAML egy XML-alapú szabvány mely autentikációs (hitelesítési) és autorizációs (engedélyezés) adatok cseréjét teszi lehetővé biztonságos web domain-ok, tehát az *identitásszolgáltató* IdP (tanúsítvány kiadója) és egy *tartalomszolgáltató* SP (tanúsítvány „fogyasztója”) között. Az elsődleges és legfontosabb probléma, amit a SAML kezelni próbál a *Web-böngésző Single Sign-On* (SSO) problémája.

Single sign-on (SSO) webes, egyszeri bejelentkezési módszer, amely olyan speciális formája a szoftveres azonosításnak, ami lehetővé teszi a felhasználó számára, hogy egy adott rendszerbe való belépéskor csak egyszer azonosítsa magát és ezután a rendszer minden erőforrásához és szolgáltatásához további autentikáció nélkül hozzáfér.

A Single sign-on megoldások bőségesek az intranet szintjén (cookie-k használata például) de ezen lehetőségek kibővítése az intraneten túlra problémás volt és a nem teljesen együttműködő saját technológiák elburjánzásához vezetett. A SAML vált a döntő szabvánnyá, alapjául szolgálva sok Single Sign-On megoldásnak a vállalati azonosítás menedzsment problémakörben.

A SAML feltételezi, hogy a hivatkozott felhasználó (principal) bejegyzett legalább egy azonosítás szolgáltatóhoz. Ez az azonosítás szolgáltató vélhetően a hivatkozott felhasználó helyi hitelesítési szolgáltatásait látja el. Azonban a SAML nem írja le ezeknek a helyi szolgáltatásoknak az implementációit; a SAML nem törődik azzal, hogy a helyi azonosítási szolgáltatások miként vannak implementálva.

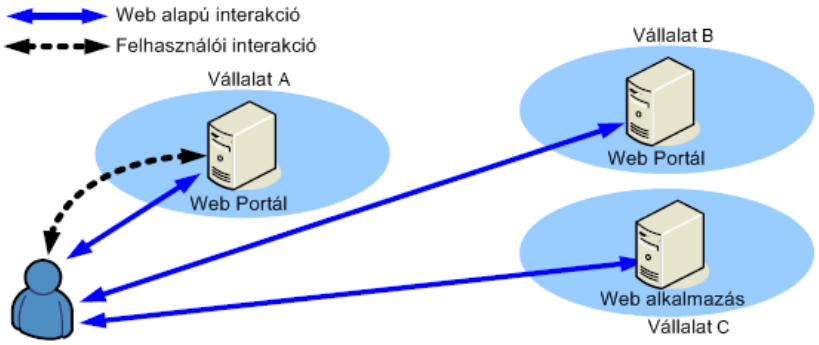
Így tehát a tartalomszolgáltató az azonosítás szolgáltatójára támaszkodik a hivatkozott felhasználó azonosítása érdekében. A hivatkozott felhasználó kérésénél az azonosítás szolgáltató SAML assertion-t küld az tartalomszolgáltatóhoz. Az assertion alapján, az tartalomszolgáltató egy hozzáférés vezérlési döntést hoz.

A SAML jó néhány létező szabványra épül, amelyek közül a fontosabbakat felsoroljuk:

- Extensible Markup Language (XML). A legtöbb SAML elem az XML egy szabványosított dialektusában íródott, amely a SAML nevének alapját is adta (Security Assertion Markup Language).
- XML Séma. SAML assertion-ök és protokollok (részben) XML sémában lettek specifikálva.
- XML Szignatúra. Mind a SAML 1.1 és a SAML 2.0 digitális aláírást használ (az XML Szignatúra szabványon alapulva) az autentikációra és az üzenetek integritásának megőrzésére.
- XML Kódolás. XML kódolás használatával, a SAML 2.0 elemeket szolgáltat a kódolt névazonosítók, kódolt attribútumok, és kódolt assertion-ök használatára (a SAML 1.1-nek nincsenek kódolási lehetőségei).
- Hipertext Transzfer Protocol (HTTP). A SAML erősen támaszkodik a HTTP-re, mint kommunikációs protokollra.
- SOAP. SAML részletezi a SOAP használatát, különösen a SOAP 1.1-t.

7.6.3 Föderációs Single Sign-On

A szövetségi SSO (F-SSO) az eljárás, ami által egy felhasználó hitelesíti magát egy szövetségi üzleti partnernél (identitásszolgáltató, IdP) és az IdP kibocsát egy hozzá tartozó identitást (és attribútumait) az egyik/az összes szükséges (és megbízható üzleti partner) tartalomszolgáltatónak, a felhasználó online szövetségi tapasztalatainak részeként. A globális bejelentkezést a szövetségi single sign-on protokoll biztosítja. Ezek a protokollok szabványos, együttműködésre képes eszközöket biztosítanak a szövetségi üzleti partnereknek ahhoz, hogy megegyezzenek a felhasználók bejelentkezési azonosítóinak megadásáról. A következőkben a szövetségi egyszeri bejelentkezést tanulmányozzuk.



7.6. ábra: Biztonságos felhasználói interakció – F-SSO

A 7.6. ábrán a felhasználói interakció egy egyszerűsített ábrázolása látható, ahol a felhasználó kommunikál az A jelű vállalattal, aki IdP-ként viselkedik, a két másik vállalat (B és C) SP-k. A kommunikáció Web böngésző alapú és F-SSO-t használnak az egyszeri bejelentkezésre. Az egyszerűsített bejelentkezés bármelyik SSO protokollal megoldható, SAML, Liberty ID-FF vagy WS-Federation.

A F-SSO szempontjából a leglényegesebb funkciók: pull és push SSO protokollok, account összekapcsolás, WAYF, sessionkezelés, kijelentkezés, bejelentkezési adatok eltakarítása, globális good-bye és account szétkapcsolás. Ezeket az alábbiakban részletesen is bemutatjuk.

7.6.3.1 Push és Pull SSO

Az SSO-nak két módja van, push és pull. A pull SSO a SAML 1.x és 2.0-ban, a Liberty-ben és a WS-Federation-ben elérhető, a push a SAML 1.x-ben és a WS-Federation-ben.

A push SSO azt jelenti, hogy a SSO adatcserét egy az IdP-hez érkező kérés indítja, amely küld (PUSH) egy biztonsági tokent a SP-nek.

A pull SSO esetén a SSO adatcserét az SP-hez érkező kérés indítja, amely kér (PULL) egy biztonsági tokent az IdP-től.

7.6.3.2 Account összekapcsolás

Az eddigiekben csak az egyszeri bejelentkezésről volt szó - nem beszéltünk arról az SP oldalon felmerülő problémáról, hogy esetleg plusz adatokat is kell tárolnia a felhasználóról (ami csak az adott alkalmazásra vonatkozik). Tovább bonyolítja a helyzetet, hogy a felhasználó személyiségi jogait is meg kell védeni a rendszerben, illetve probléma esetén az IdP oldalon visszakövethető kell legyen, hogy egy adott pillanatban melyik felhasználó melyik szolgáltatást vette igénybe.

Az első felmerülő lehetőség, hogy az IdP oldalon tároljuk az összes információt. Ez nyilvánvalóan több okból sem tehető meg. Egyrészt felesleges terhelést és adminisztrációt

jelent, ráadásul az IdP szempontjából lényegtelen információkról van szó. Másrészt, ilyenkor biztosítani kellene, hogy a többi SP ne jusson hozzá ezekhez az információkhoz. Sok esetben nem kerülhető ki tehát, hogy az SP is tároljon felhasználói adatokat, a felhasználó rendelkezzen lokális „account”-tal. Például már egy egyszerű webes közösségi alkalmazás esetén sem kerülhető meg, hogy néhány attribútumot (legalább a becenév) megjelenítsen a felhasználókról. Valahogy össze kell tehát kapcsolni az IdP által adott felhasználói identitást az SP oldali adatokkal. Erre a következő megoldásokat találták ki:

Összekapcsolás az IdP által tárolt attribútumok alapján - ez a megoldás az egyszerűbb esetekben használható, de rengeteg problémát indukál. Nem kezeli az esetleges attribútum változásokat, esetleg az attribútum-kiadási elvek változásait, és túl szoros csatolást visz a rendszerbe.

Összekapcsolás fix azonosító alapján - az IdP minden felhasználóról nyilván tart egy fix azonosítót, amit a felhasználó nem változtathat, és ezt az azonosítót elküldi minden egyes SP-nek, aki éppen be akar kapcsolódni a felhasználó munkamenetébe. Tipikusan ez a fix azonosító lehet a felhasználó e-mail címe, vagy a tanúsítványán szereplő név. Sajnos ez a megoldás sem védi meg a felhasználó személyiségi jogait, hiszen két SP a felhasználó tudta és beleegyezése nélkül képes a saját lokálisan tárolt adataikat egyeztetni, ezzel egy nagyobb képet kialakítani a felhasználó tevékenységéről.

Összekapcsolás fix, de alkalmazásonként változó pszeudoname azonosító alapján. Ez a megoldás egy „álnevet” visz a rendszerbe, ami konkrét megvalósításban egy véletlenül kisorsolt azonosítót jelent. Ráadásul minden SP más álnevet lát, de a többszöri látogatás során mindig ugyanazt. Ez a megoldás nem teszi lehetővé a személyes adatok előző pontban végiggondolt kiszivárgását. Az IdP felelőssége, hogy alkalmazásonként különböző véletlen álneveket adjon ugyanannak a felhasználónak, és gondoskodjon arról, hogy ezek az azonosítók perzisztensek maradjanak. Ez többlet adminisztrációval jár, ami miatt néhány IdP szoftver nem támogatja ezt a megoldást.

Összekapcsolás változó pszeudoname azonosító alapján. Ebben a megoldásban az IdP munkamenetenként más és más véletlen azonosítót rendel a felhasználóhoz, ami függ az SP-től is. Sajnos így elveszik az account összekapcsolás lehetősége, de probléma esetén a visszakövethetőség megmarad. Általában beállítható, hogy valamennyi ideig megőrződjenek az álneveket tároló naplófájlok, amiből rekonstruálható a felhasználó útja a rendszerben.

Ezeket az összekapcsolásokat többféleképp is megtehetjük. Attribútum alapú összekapcsolásnál az SP az attribútumok alapján kikeresheti a megfelelő lokális felhasználói profilt és elvégezheti automatikusan az összekapcsolásukat. Egyébként a felhasználónak explicit módon be kell jelentkeznie mindkét rendszerbe, és ezzel az egyidejű bejelentkezéssel kötheti össze a kétféle azonosítóját. Utóbbi megoldást általában a perzisztens álnevek esetén szokás használni. Ez a felhasználó által kezdeményezett összekapcsolás más szempontból is előnyös: magára a végfelhasználóra bízza a döntést, így az adatkezelést is teljesen tisztává teszi. A legtöbb ilyen módon összekötött rendszer lehetővé teszi az összekapcsolt, „federált” azonosítók szétkapcsolását is. Automatikus összekapcsolás esetén az SP akár dinamikusan is létrehozhatja a lokális accountot, az IdP által adott attribútumok figyelembe vételével. A fentieken kívül lehetőség van természetesen arra is, hogy a perzisztens azonosítók

alkalmazásával automatikusan, előre összekössünk néhány konkrét SP accountot a hozzájuk tartozó identitással. Ezt „bulk federation”-nek hívják, és az üzleti rendszereknél gyakran alkalmazott megoldás.

Vegyük RBTelkom-ot és RBBanking-et ahol Kiss Józsefnek külön (hitelesíthető) accountja van mindkét vállalatnál. Amikor a két vállalat megegyezik a federációba csatlakozásról, akkor nekik valamilyen módon lehetővé kell tenni, hogy az RBTelkom felhasználói SSO-val beléphessenek RBBanking-hoz. Ennek a megoldása RBBanking feladata. Ez két lépésben történik, jelen esetben az RBTelkom weboldaláról indulva. Az RBTelkom megváltoztatja a hivatkozást a portálján, így az egyszerű átirányítás helyett, a bank linkjére kattintás egy SSO kérést indít RBBanking-hoz. A pénzintézet megkapja a kérést, de nem tudja megfeleltetni egy helyi identitásnak. Ez azt eredményezi, hogy RBBanking-nak el kell kérnie a bejelentkezési adatait Kiss úrtól. Sikeres hitelesítés esetén, RBBanking-nál hozzárendelődik a RBTelkom által kibocsátott CUID-hez (az SSO kérésből) a saját helyi felhasználó reprezentáció (József direkt bejelentkezéséből). RBBanking most már képes account összekapcsolást létesíteni, így Kiss úr SSO-val bejelentkezhet RBTelkom-tól.

Ha a felhasználó a roll-over[15] időszakban akarja közvetlenül elérni RBBanking-ot, akkor őt a szokásos módon hitelesítik. Ezután, RBBanking SSO-t fog kérni RBTelkom-tól (a már hitelesített felhasználónak). A megfelelő SSO válasz tartalmazni fogja a közös felhasználói azonosítót (CUID), így RBBanking mind a RBTelkom által kibocsátott CUID-vel (az SSO kérésből) mind a saját helyi felhasználó reprezentációval (József direkt bejelentkezéséből) rendelkezni fog. RBBanking most már képes account összekapcsolást létesíteni, így Kiss úr SSO-val bejelentkezhet RBTelkom-tól.

RBBanking kikapcsolhatja a felhasználó helyi jelszavának kérését, így a közvetlen hitelesítés RBBanking-nál már nem lehetséges, addig ameddig a felhasználó accountja össze van kapcsolva RBTelkom-mal. Legközelebb, amikor a felhasználó megpróbál közvetlenül hozzáférni RBBanking-hez, a bank SSO-t fog kérni RBTelkom-tól.

Általában az account összekapcsolás részeként, létrehoznak valamilyen hosszú távú/állandó információt, mint például egy http cookie, amely ennek a felhasználónak az identitásszolgáltatójaként azonosítja RBTelkom-ot. A roll-over időszak alatt ezt arra is használják, hogy megkülönböztessék a már összekapcsolt és „még nem összekapcsolt” felhasználókat. Amint a roll-over periódus befejeződött minden felhasználót aki nem rendelkezik ezzel az állandó információval, meg kell kérdezni, hogy eldöntsék, hogy RBTelkom e a valódi identitásszolgáltatójuk.

7.6.3.3 Where Are You From? (WAYF)

Szolgáltatóknak több identitásszolgáltatóval is lehetnek bizalmi kapcsolatai. Ez azt jelenti, hogy a felhasználó kezdeményezhet SSO-t az egyik IdP-től. A tartalomszolgáltató számára azt az eljárást, amellyel meghatározza, hogy melyik IdP-től kell kérnie a SSO-t,

[15] Egy adott pozíció, szolgáltatás lejáratkori lezárása és egyidejű megújítása további időszakra.

Where are you from? (WAYF) szolgáltatásnak nevezzük. Ez egy olyan eljárás, amivel egy felhasználó megadhatja a preferált IdP-jét. Ezt az információt a SP kezeli, így egyszerűen, felhasználói beavatkozás nélkül meghatározhatja, hogy a jövőben melyik IdP-től kell kérnie a SSO-t.

RBBanking ügyében, a WAYF információt a roll-over időszakban hozzák létre. Ebben a periódusban, RBBanking mind tartalomszolgáltatóként (a már szövetségi felhasználók számára), mind identitásszolgáltatóként (a nem federációs felhasználóknak) viselkedik. Tehát RBBanking és RBTelkom is identitásszolgáltatóként viselkedik, az egyetlen tartalomszolgáltatónak RBBanking-nak.

Ha RBBanking egyetlen SP-je volna több IdP-nek, akkor támaszkodnia kellene valamilyen állandó információra a felhasználóval kapcsolatban (mint például http cookie), azzal kapcsolatban, hogy egy SSO kérést melyik identitásszolgáltatónak kell elküldeni. Ha a cookie hiányzik, akkor RBBanking-nak kezdeményeznie kell, valamilyen felhasználó általi WAYF feldolgozást. RBBanking felkéri Józsefet, hogy válasszon ki egy identitásszolgáltatót az ismert/megbízható IdP-k listájáról.

Némely esetben a tartalomszolgáltató nem hajlandó felfedni a megbízható IdP-k listáját. Ekkor RBBanking utasítást adna Kiss úrnak, hogy miként érheti el közvetlenül az identitásszolgáltatóját (RBTelkom) és hogyan kezdeményezhet SSO kérést egy IdP alapú mechanizmuson keresztül.

7.6.3.4 Session menedzsment és hozzáférési jogosultságok

Amint a felhasználó bejelentkezett egy tartalomszolgáltatóhoz, a SP felelős azért, hogy kezelje a felhasználó helyi munkamenetét. Ebbe beletartoznak a felhasználó tevékenységeivel kapcsolatos jogosultsági döntések, a munkamenet-kezelés maga, továbbá a kijelentkezés és a biztonsági időkorlát lejárta (session time-out).

Ez azt jelenti, hogy az SP valamilyen szinten képes kezelni a felhasználó attribútumait és bejelentkezési adatait. Ezeket az attribútumokat arra használják, hogy egy felhasználó helyi hozzáférési jogait meghatározzák. Hozzáférési jogokat az IdP adhat ki, a felhasználóról szóló kibocsátott (asserted) attribútumok formájában, ilyen például a csoporttagság.

7.6.3.5 Kijelentkezés

Néhány szövetségi forgatókönyvben, a globális vagy egyetlen kijelentkezés szintén szükséges, ami lehetővé teszi a felhasználónak, hogy az IdP által kijelentkezési kérést küldjön minden munkamenethez. Globális kijelentkezést kérhet a felhasználó IdP-től és SP-től is, a globális kijelentkezés folyamatát azonban mindig az identitásszolgáltató irányítja. Az IdP felelős azért, hogy kezelje azon SP-k listáját, akikhez a felhasználó az adott munkamenetben SSO-val bejelentkezett. Az IdP ekkor egy kijelentkezés-kérést küld a felhasználó nevében mindezeknek az SP-knek.

Ha József kijelentkezik például RBTelkom portáljáról, akkor az RBTelkom már nem veszi figyelembe azokat a tranzakciókat, amibe József belekezd. Ebben az esetben RBTelkom elindít egy kijelentkezési kérést minden üzleti partnernek, amihez SSO kérést bocsátanak ki József aktuális munkamenetén belül.

A globális kijelentkezés nem utal arra a tényre, hogy helyileg is kijelentkezés történik. Előfordulhat, hogy egy felhasználó ki kíván jelentkezni egy tartalomszolgáltatónál lévő munkamenetből, de az IdP-nél lévő munkamenetet nem akarja megszakítani. Másik alternatíva egy SP-nél levő helyi kijelentkezésre, hogy rövidebb munkamenet össz/inaktivitási időtúllépés keretet kell beállítani, mint az alapértelmezett közvetlenül hitelesített munkamenetben. Egy rövidebb tétlenségi időtúllépés, az SSO felhasználónak elfogadhatóbb lehet, mivel így nem kényszerítik explicit újra hitelesítésre. Helyette a SP egyszerűen újra kér egy SSO-t a felhasználó identitásszolgáltatójától.

7.6.3.6 Bejelentkezési adatok eltakarítása

A kijelentkezés, legyen az globális vagy helyi, gyakran magába foglalja a session megszakítását az SP-nél. Ez a munkamenet független lehet a kiszolgáló oldali alkalmazásokkal rendelkező munkamenetektől. A kiszolgáló oldali alkalmazás munkameneteit arra használhatják, hogy fenntartsanak egy státuszt a több lépésből álló tranzakciók kérés/válaszai között. Kijelentkezéskor biztosítani kell, hogy mind az identitásszolgáltatónál, mind a tartalomszolgáltatónál mindenféle sessiont és az ahhoz tartozó attribútumokat megsemmisítsék.

Nézzük meg mi történik, amikor József kijelentkezik a RBTelkom portálról és ezzel egyúttal a RBBanking weboldaláról. Ha József elindított egy tranzakciót (eszközök átvitelére például) aztán elfelejtkezett róla, akkor ezt a tranzakciót el kell takarítani (ez lényegében, egy szemétgyűjtő). Ha ez nem történik meg, akkor RBBanking-nek olyan árva munkamenetei maradnak, amik erőforrásokat köthetnek le a kiszolgáló oldali alkalmazásainál.

7.6.3.7 Globális good-bye

A globális good-bye kezeli a felhasználó hozzáférési jogainak és felhatalmazásainak visszavételét egy szövetségi forgatókönyvön belül. Akkor használják, amikor egy IdP és SP közti kapcsolat megszűnik és minden felhasználói attribútum - beleértve a tranzakció, profil és szolgáltató specifikus attribútumokat - ami fontos a megszűnő kapcsolat szempontjából, szintén megszűnik. Vegyük figyelembe, hogy a szövetségi kapcsolatok többféle módon fejeződhetnek be: a felhasználó dönthet úgy, hogy megszakítja a kötését az IdP és a SP között vagy az identitásszolgáltató és a tartalomszolgáltató nem kívánja folytatni az együttműködést, így megszakítva az IdP felhasználóinak kötéseit.

Például, ha a kedvenc alkalmazottunk, Első úr új állás után néz (és a KisCég-nél dolgozik ezután) akkor hozzáférési jogait és jogosultságait valamint a NagyCég által szponzorált utazási kedvezményeit el kell távolítani a NagyCég és RBTravel közti globális

good-bye részeként. Megjegyzendő, hogy ez nem jelenti azt, hogy eltávolítanák Első úr accountját - beleértve a szolgáltató specifikus attribútumokat – RBTravel-nél. Ez csak annyit tesz, hogy minden NagyCég-gel kapcsolatos attribútumot (beleértve a tranzakció és profil attribútumokat) törölnek Első úr RBTravel accountjából.

Általában a globális good-bye az account szétkapcsolással együtt megy végbe.

7.6.3.8 Account szétkapcsolás

Az account szétkapcsolás az az eljárás, amely a közös egyedi azonosítót megsemmisíti, megszüntetve annak a lehetőségét, hogy az IdP és SP egyedileg utaljon egy adott felhasználóra. A szétkapcsolás egyik eredménye, hogy a felhasználó már nem használhatja az egyszeri bejelentkezést IdP-től az SP felé. Megjegyzendő, hogy az account szétkapcsolás független attól, hogy az SP-nél miként hozták létre az accountot/regisztrációs bejegyzést. Tehát a szétkapcsolás lehetséges akkor is, ha az accountot explicit létrehozta a felhasználó vagy az IdP, SP provisioning eredményeként jött létre. A szétkapcsolás után a felhasználó vagy a SP választhat egy másik IdP-t az account összekapcsolás céljából, vagy a tartalomszolgáltató úgy dönthet, hogy folytatja a user közvetlen hitelesítését.

Kiss József dönthet úgy, hogy megszünteti RBTelkom-os számláját. Ez történhet költözés miatt vagy, mert szolgáltatót vált stb.. Esetünkben József már nem lesz képes SSO-val elérni RBBanking-ot RBTelkom-tól, mert már RBTelkom-hoz sem fog tudni belépni. Ebben az esetben József információit RBTelkom-nál és RBBanking-nél is szét kell kapcsolni („de-federálni”). A folyamat eredményeként József közös egyedi azonosítóját megsemmisítik és az egyszeri bejelentkezési képességét RBTelkom-nál elveszti, továbbá visszahelyezik olyan felhasználónak, akit közvetlenül hitelesít RBBanking.

8 Kriptográfiai alapismeretek.

Ma az adatfeldolgozás döntően számítógépeken, megfelelően tervezett és kódolt programokkal történik. Az adatokat jogi vagy ügyviteli eszközökkel a feldolgozás folyamatában résztvevő emberekkel szemben lehet védeni. A számítógépnek ezek az előírások semmit sem jelentenek mindaddig, amíg nem fogalmazzuk meg számukra érthető nyelven. Ez a nyelv algoritmusokból és protokollokból áll, amelyeket program formájában teszünk érthetővé a számítógépekkel. Az adatvédelem „számítógépbarát" elemeit algoritmikus adatvédelemnek nevezzük.

Ezek alapját olyan matematikai módszerek képezik, amelyek lehetővé teszik adatok bizalmas tárolását és továbbítását. Korábban már rámutattunk, hogy adatvédelmi szempontból a tárolás is információtovábbítást jelent csak nem térben, azaz például Debrecenből Budapestre, hanem időben, azaz máról holnapra vagy egy évvel későbbre. Természetesen a tárolással kapcsolatban felmerülnek olyan problémák, amelyek a továbbításnál nem fontosak. Ilyen speciális igény információk hosszú időtartamú tárolása, azaz archiválása. Másrészt, a továbbításnál általában lényeges a kommunikáció sebessége, ami tárolás esetén általában nem kiemelt igény.

Elfogadva tehát, hogy a tárolás és továbbítás adatvédelmi szempontból egységesen kezelhetőek, azt vizsgáljuk meg, hogy miért van szükség bizalmas adattovábbításra. A jegyzet korábbi fejezeteiben sokat foglalkoztunk az azonosítással. A 6.3 fejezetben magyaráztuk meg részletesen, hogy amikor egy számítógépbe vagy egy alkalmazásban be akarunk lépni, akkor azonosítani kell magunkat, azaz be kell bizonyítani a számítógépnek, hogy ismerjük a felhasználói nevünkhöz tartozó jelszót. A belépés általában nem a számítógéppel összekötött terminálon történik, hanem egy nyilvános hálózaton keresztül. Ha a jelszót eredeti formájában küldjük át a hálózaton, akkor ahhoz illetéktelenek könnyen hozzáférhetnek és kaméleont játszva, a nevünkben léphetnek be a számítógépbe, hozzáférve ezzel minden hozzánk rendelt erőforráshoz és információhoz. A jelszót tehát úgy kell kódolni a nyilvános hálózaton történő küldés előtt, hogy azt illetéktelen ne tudja dekódolni.

Amikor egy vállalat vezető tisztviselője távoli terminálról, otthonról, szállodai szobából vagy a gépkocsijából bejelentkezik a vezetői információs rendszerbe, akkor nem kívánatos, hogy a forgalmazott adatok kódolás nélkül utazzanak a nyilvános hálózaton. Azt sem szeretnénk, ha banki tranzakcióink tartalma a hálózaton bárki számára olvasható legyen. Hasonló példákat hosszan lehetne sorolni, de ennyinek is elegendőnek kell lenni a bizalmas üzenettovábbítás fontossága alátámasztására. Bizalmas üzenettovábbítás olyan kódolással érhető el, amikor a kódolt üzenetet csak az arra illetékesek tudják dekódolni. Az ilyen kódolást *titkosításnak* nevezzük.

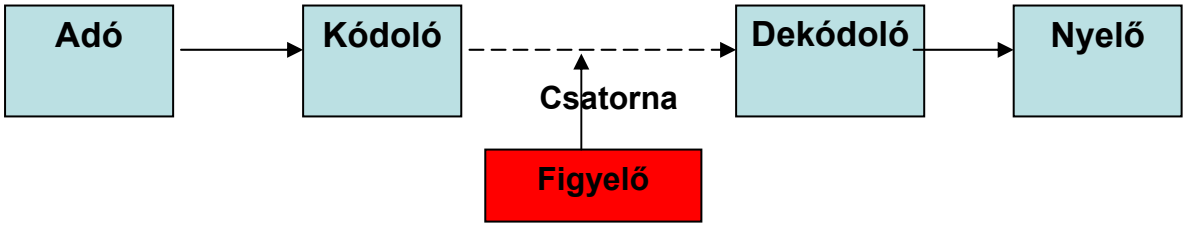
A 4.2 fejezetben foglalkoztunk az elektronikus aláírással, amelynek az információs társadalomban betöltött kiemelt szerepét mutatja, hogy szinte minden ország törvényben szabályozza az alkalmazását. Nem nyilvánvaló, de később meg fogjuk mutatni, hogy a digitális aláírás is titkosítási eljáráson alapul.

A kriptográfiai szakirodalom elsősorban angolul érhető el, ezért a magyar kifejezéseknek, azok első előfordulásakor megadjuk az angol megfelelőjét is.

8.1 Alapfogalmak

A bizalmas üzenettovábbítás elvének bemutatására tökéletesen alkalmazható Claude Shannon (1916 – 2001) amerikai matematikus modellje, amelyet a 8.1 ábrán jelenítünk meg. A modell három szereplője: az Adó, aki bizalmas üzenetet akar küldeni a Nyelőnek és végül a Figyelő, aki az üzenetet minden rendelkezésére álló eszközzel meg akarja szerezni. Az üzenetet szokás *nyílt szövegnek* (plain text) is nevezni. A Figyelő elsősorban a csatornán férhet hozzá az üzenethez, de egyéb fontos információkat is szerezhet az Adó és a Nyelő oldalán is. A Figyelő dolgát megnehezítendő, az Adó az üzenetet nem az eredeti formájában küldi át a csatornán, hanem egy *titkosító eljárásnak* (encryption) veti alá. A csatornán tehát már nem a nyílt szöveg, hanem annak kódolt változata a *titkos üzenet* (ciphertext) megy át. Közvetlenül a Nyelő sem tud mit kezdeni a titkos üzenettel, de ismerve a *megfejtő,* dekódoló *eljárást* (decryption) vissza tudja állítani az eredeti szöveget és értelmezni tudja azt.

Bizonyos alkalmazásoknál a modellből hiányozhat a dekódoló egység, illetve a kódoló átkerülhet a nyelő oldalára. Ilyen példákkal találkoztunk a 6.3 fejezet jelszavas azonosításról szóló részében, illetve látni fogunk a digitális aláírásnál is (hash függvény).



8.1 ábra

Jelöljük a lehetséges üzenetek halmazát P-vel, a titkosított üzenetek halmazát pedig C-vel. Ekkor a titkosító eljárás egy E: $P \rightarrow C$, a visszafejtés pedig egy D: $C \rightarrow P$ leképezés. Bizonyos esetekben a titkosító és a visszafejtő eljárás nemcsak a nyílt üzenettől, hanem egy további paramétertől, kulcstól (key) is függ. Ha a lehetséges kulcsok halmazát K-val jelöljük, akkor a titkosító, illetve visszafejtő leképezések definíciója a következőképpen alakul: E: $P \times K \rightarrow C$, D: $C \times K \rightarrow P$, ahol most $\times$ halmazok direkt szorzatát jelöli.

A titkosítás klasszikus alkalmazásainál arra törekedtek, hogy a kódoló eljárás és a kulcs is titokban maradjon. Ekkor persze az adónak és vevőnek a kommunikáció megkezdése előtt meg kell állapodnia a titkosító módszerben és a kulcsban. Ez nagyon lecsökkenti a potenciális partnerek számát. A következő fejezetben ismertetünk ilyen példát. Internetes világunkban ez az út nem járható. Ha például az APEH minden adózó állampolgárral más-más titkosító algoritmussal kommunikálna, akkor néhány millió, jól tesztelt eljárást kellene alkalmaznia, ami sem anyagi sem technikai szempontból nem realizálható. A kriptográfiában ma a titkosító és visszafejtő függvényt ismertnek, sőt szabványosnak tételezzük fel, így a titkosítás minősége a kulcstól függ. A szabványosítás nagyon fontos követelmény. Gondoljuk

tovább az előbbi APEH-es példát. Napjainkban néhány százezer adóalany nyújt be elektronikus úton adóbevallást. Ezeket egy kulcscsere után, DES-el kódolva juttatják el az APEH szerverének. Az adózók számítógépei sokféle operációs rendszert használhatnak, és sokféle alkalmazással végezhetik az adóbevallás kódolását. Ha ezek valamelyike nem a szabványos DES kódolást végezné, akkor az APEH-es szerver nem tudná helyesen dekódolni az adatokat.

A továbbiakban azt vizsgáljuk, hogy az *E* és *D* függvényeknek milyen tulajdonsággal kell rendelkeznie, hogy alkalmasak legyenek titkosításra. Mint fentebb megállapítottuk a titkos üzenetnek olyannak kell lenni, hogy a Figyelő csak nagyon nehezen tudja azt megfejteni. Ez annyit jelent, hogy tetszőleges *u* üzenetre és *k* kulcsra, ha $m = E(u,k)$, akkor a Figyelő *E* és *m* ismeretében nagyon nehezen tudja *u*-t, esetleg *k*-t is meghatározni. A Figyelő közvetlen célja a bizalmas üzenet kiderítése, de ha az alkalmazott kulcsot is megismeri, akkor más üzeneteket is dekódolhat, illetve megszemélyesítheti az Adót. Az eléggé homályos „nagyon nehéz" heurisztikusan annyit jelent, hogy a meghatározás igen nagy számítási erőt feltételezve, az ismert módszerekkel néhány száz évig is eltarthat.

Egy titkosító függvény csak akkor használható a gyakorlatban, ha a kódolást gyorsan el tudja végezni. Olyan eljárást, amely néhány kilobájtnyi adatot percekig kódol, nyugodtan el lehet felejteni.

A követelményrendszer heurisztikus ismertetését a dekódoló függvény elemzésével tesszük teljessé. Mint említettük néhány fontos alkalmazásnál erre a függvényre nincs is szükség. A dekódolás azt jelenti, hogy vissza akarjuk állítani az eredeti üzenetet. Ehhez persze egy dekódoló kulcs is kell. A *D* függvénynek tehát olyannak kell lennie, hogy minden k_t titkosító kulcshoz legyen egy k_d visszafejtő kulcs úgy, hogy minden *u* üzenetre

$$D(E(u,k_t),k_d) = u.$$

Szavakban kifejezve az előző egyenlőség annyit jelent, hogy ha az *u* üzenetet a k_t kulccsal titkosítjuk, majd a titkos üzenetet a k_d visszafejtő kulccsal dekódoljuk, akkor visszakapjuk az eredeti üzenetet.

Két bekezdéssel korábban, az E-vel kapcsolatban megfogalmaztuk, hogy „a Figyelő *E* és *m* ismeretében nagyon nehezen tudja *u*-t, esetleg *k*-t is meghatározni", amit most kiegészítünk a következőre: a Figyelő *E, m* és *D* ismeretében nagyon nehezen tudja *u*-t, esetleg k_t-t is meghatározni. Végezetül a k_d ismeretében a dekódolásnak is gyorsnak kell lenni.

Pontosabb definícióhoz először a modellünkben szereplő halmazokat kell precízebben meghatározni. Vegyük észre, hogy a gyakorlatban a *P, C* és *K* halmazok véges hosszúságú bináris szavakból állnak, így maguk is véges halmazok. Az *E* illetve *D* függvényeket könnyű kiszámítani, ha maximális bonyolultságuk kis kitevőjű polinommal becsülhető. A legjobb, ha a kitevő egy, azaz a kiszámítás bonyolultsága lineáris. Ha *E* bonyolultsága polinomiális, akkor persze m hossza is becsülhető *u* és k_t összhossza polinomiális függvényével. Ugyanennek kell teljesülnie k_d -re is, mert különben a visszafejtés k_d ismeretében sem lehet gyors. A Figyelő tehát exponenciális időben mindig vissza tudja fejteni az eredeti üzenetet. A jó kriptográfiai függvény tehát olyan, amelyre a dekódolás egyetlen inputra sem történhet meg exponenciálisnál lényegesen gyorsabban. A fentiekben leírt *E* függvényeket szokás *egyirányú függvény*nek (one way function) nevezni. Azokat az egyirányú függvényeket pedig, amelyeknek van a fentiekben leírt dekódoló *D* párja, *egyirányú csapóajtó függvény*nek (one way trapdor function) nevezzük.

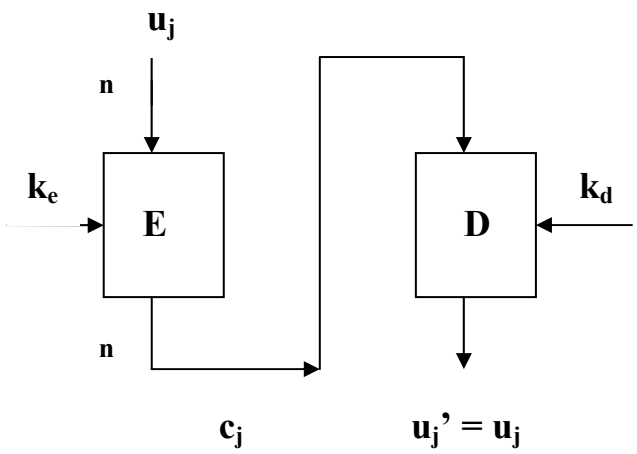
Digitális információk kódolása kétféleképpen történhet vagy az egész üzenetet egyszerre kódoljuk, amit folyamkódolásnak (stream cipher) nevezünk, vagy pedig az üzenetet feldaraboljuk, a darabokat külön-külön kódoljuk és az eredményt a nyelő oldalán ismét összefűzzük. Az utóbbit blokk kódolásnak (block coding) nevezzük.

Folyamkódolásra példa a Vernam titkosítás. A digitális u üzenetet a $\{0,1\}$ abc feletti szónak tekintjük. Ezután generálunk az u-val egyforma hosszúságú véletlen és egyenletes eloszlású k bitsorozatot. Végezetül az u és v bitjeire bitenként a kizáró vagy műveletet alkalmazva nyerjük az m titkosított üzenetet. A dekódoláshoz a nyelőnek ismernie kell a v kulcsot. Ekkor az m és a v bitjeire ismét bitenként alkalmazza a kizáró vagy műveletet és visszanyeri u-t. A dekódolás korrekt, hiszen a kizáró vagy művelete tetszőleges a, b bitre rendelkezik az $(a \oplus b) \oplus b = a$ tulajdonsággal.

A Vernam titkosítás a lehető legbiztonságosabb, ha a v-t minden üzenetre egyedileg állítjuk elő valamint bitjei egyenletes eloszlásúan és véletlenek. Nagy problémát jelent viszont, hogy nemcsak a titkosított üzenetet, hanem az egyedi dekódoló kulcsot is el kell juttatni a nyelőnek. Ez az üzenet hosszának megduplázását jelenti, ami gazdaságtalanná teszi az eljárást. A biztonságosság mellett a Vernam titkosítás jó tulajdonsága az is, hogy rendkívül egyszerű és gyorsan implementálható. A bitenkénti kizáró vagy műveletet ezért a modern titkosítási eljárásokban gyakran alkalmazzák.

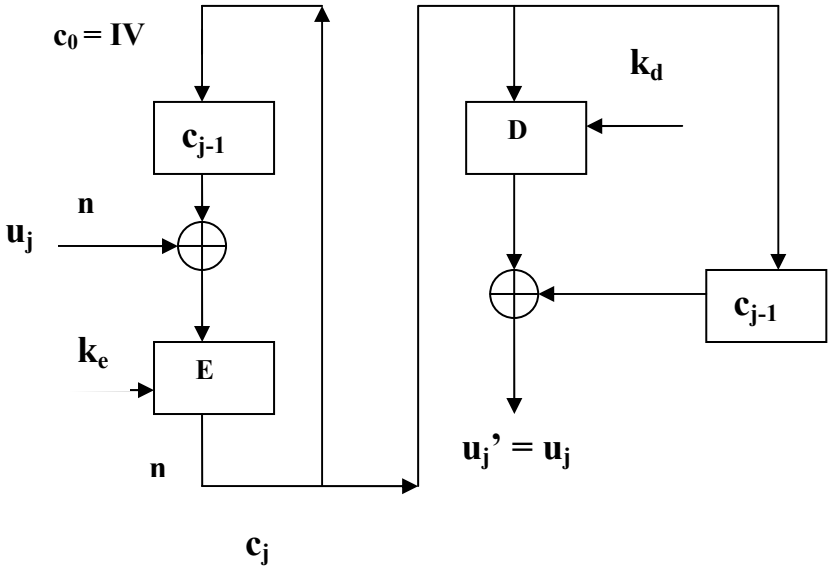
Mint fentebb írtuk a blokk titkosítás során az üzenetet feldaraboljuk és az egyes darabokra külön-külön alkalmazzuk a titkosítási eljárást. A blokkok hossza lehet fix vagy változó. Titkosításra szinte csak a fix blokkhosszú változatot alkalmazzák. Hosszabb üzenetek továbbítására három módszer áll rendelkezésünkre, amelyeket a 8.2 – 8.4 ábrákon mutatunk be. Az üzenetblokkokat, melyeknek hossza n, u_j-vel, a titkosított blokkokat c_j-vel jelöljük.

A legegyszerűbb az ECB (Electronic Codebook) módszer, amely a 8.2 ábrán található. Ennél az üzeneteket egymás után titkosítva közvetlenül küldjük a nyelőnek. Adott kódoló eljárás mellett nyilván ez a legegyszerűbb és leggyorsabb továbbítási módszer. A csatorna azonban általában zajos, különböző hibát véthet az átvitel során. Ha a hiba olyan természetű, hogy néhány bittel lerövidíti vagy meghosszabbítja az átvitt blokkot, akkor vagy a következő blokkból kerül át néhány bit a megelőzőbe, illetve a következő blokk néhány bitje kerül át az előzőbe. A dekódolóra tehát nem c_j, hanem egy ettől picit különböző blokk érkezik és ezért a dekódolás hibás eredményt ad. Egy ilyen hiba nemcsak azt a blokkot érinti, amelynél jelentkezett, hanem minden további blokkot is, így az üzenet jelentős része használhatatlanná válik. Hosszabb üzenetek átvitelére tehát nem javasolt.



8.2 ábra

Az ECB hibáját küszöböli ki a CCB (Cipher-block Chaining) módszer, amely a 8.3 ábrán látható. A módszer lényege, hogy a kódolt blokk egyrészt átkerül a nyelő oldalára, de vissza is csatoljuk a következő blokk kódolásához. Ilyenkor tehát a következő üzenetblokkot először bitenként xor-ozzuk az előző titkosított blokkal és az eredményre alkalmazzuk a kódoló eljárást. Persze a visszacsatolást a nyelő oldalán is el kell végezni. Az első üzenet blokk kódolásakor még nincs mit visszacsatolni, ezért szükség van egy IV-vel jelölt kezdőblokkra, amelyet persze a nyelő oldalán is hozzá kell adni az első titkosított blokkhoz.



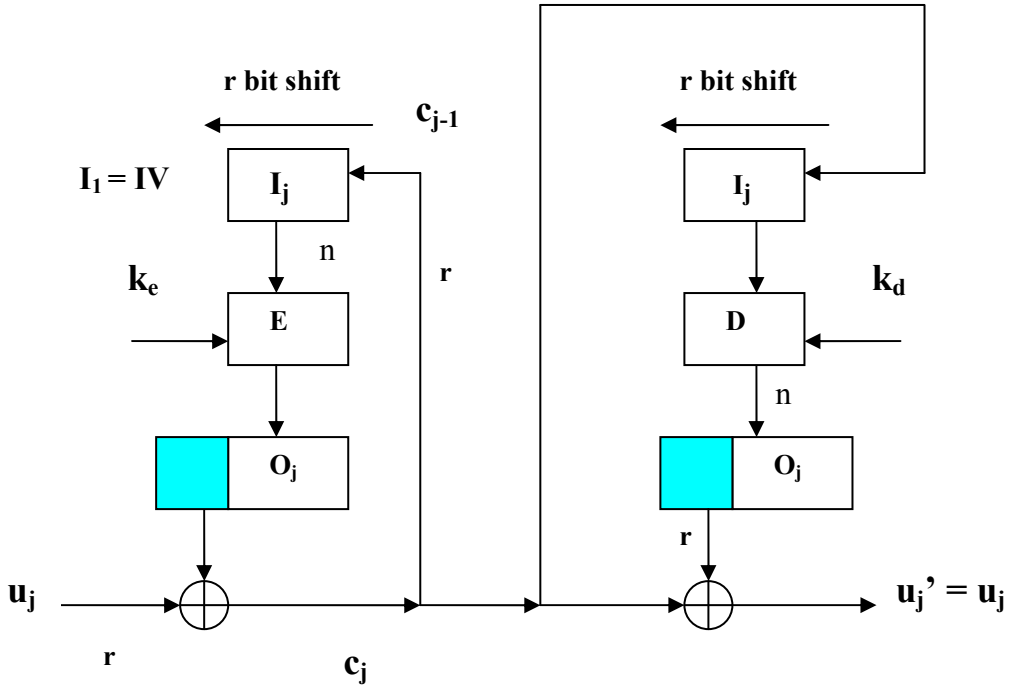
8.3 ábra

A CCB módszer kiküszöböli az ECB hiányosságát, mert egy esetleges hiba csak az érintett blokk korrekt dekódolását teszi lehetetlenné. Ennek ára az, hogy a következő blokk

feldolgozásához csak akkor lehet hozzákezdeni, ha az előző visszacsatolása megtörtént. Amennyiben a blokkhossz nagy, ami például az aszimmetrikus titkosításnál általános, akkor ez komoly késleltetéshez vezet.

A 8.4 ábrán bemutatott CFB módszer kompromisszumot jelent a visszacsatolás adta átviteli biztonság és a késleltetés miatti hátrány között. Az üzenetblokk hossza most *r*, de a kódoló ettől hosszabb, $n > r$ bites blokkokat titkosít. A titkosított információ első r bitjét maszkoljuk az aktuális üzenetblokkhoz. Ha az r lényegesen kisebb, mint n, akkor a késleltetést lényegesen lehet csökkenteni. A CCB módszerhez hasonlóan most is szükség van egy inicializáló blokkra.

Cipher feedback Mode (CFB), r bites blokk/ r bites visszacsatolás



8.4 ábra

8.2 Klasszikus titkosítási eljárások

A titkosítás több ezer éves múltra tekint vissza. Államférfiak és hadvezérek tudták, hogy ellenfeleik mindent megtesznek azért, hogy terveiket mielőbb megtudják és megelőzzék a lépéseiket. Elképzeléseiket ezért csak közvetlen bizalmasaikkal tárgyalták meg és utasításaikat a lehető legkésőbbi időben juttatták el alvezéreikhez. Modern terminológiával élve, a kommunikációra bizalmas csatornát, megbízható küldöncöt használtak. Még ezt sem tartották elég biztonságosnak, mert az üzenetet is kódolva adták át a küldöncnek. A címzett

persze csak akkor tudta megfejteni az üzenetet, ha korábban közölték vele a visszafejtő eljárást.

Julius Caesar római császárról jegyezték fel, hogy üzeneteit úgy kódolta, hogy a szövegben előforduló betűket az abc-ben az adott betűtől előre megállapított távolságban levő másik betűvel helyettesítette. A helyettesítést úgy kell persze elképzelni, hogy az abc betűit egy kör kerületére írjuk fel, így az abc végén álló karakterek helyett az abc elején állókat kell írni. Ezt a módszert ma Caesar titkosításnak nevezzük. Tekintsük például az angol abc-t és tegyük fel, hogy a távolság 7 akkor a megfelelő helyettesítést az alábbi táblázat mutatja:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g

8.5 ábra

A következő üzenet: „tizenegy orakor tamadunk” kódolt formája tehát így alakul: „apglnlnf vyhrvy ahthkbur”. A 8.1 fejezetben bevezetett terminológiát használva a Ceasar titkosítás a következőképpen írható le. A P és C az angol abc betűinek halmaza, míg $K = \{0,1,\ldots, 25\}$. Az E kódoló függvény értékét az u üzenetre és a k_t kulcsra úgy kell kiszámítani, hogy meghatározzuk u sorszámát az abc-ben, ehhez hozzáadjuk k_t –t. Ha a kapott érték nagyobb, mint 26, akkor levonunk belőle 26-ot. Ezek után megkeressük az így kapott sorszámú betűt az abc-ben. A dekódoló kulcs megegyezik k_t –vel, ha az 0 és $26 - k_t$ –vel különben.

Az előbbi leírás egy ember számára érthető, de számítógépnek nehézkes. Sokkal jobb eredményt érhetünk el, ha figyelembe vesszük, hogy a P és C halmazok végesek, így nem a betűket, hanem csak az abc-beli sorszámaikat vesszük figyelembe. A sorszámozást 0-val kezdve $P = C = K = \{0,1,\ldots, 25\}$ és ekkor $E(u, k_t) = u + k_t \bmod 26$. Továbbá $k_d = 26 - k_t \bmod 26$ és $D(m, k_d) = m + k_d \bmod 26$.

Világos, hogy a Caesar titkosítás kulcstere nagyon kicsi, csak 26 elemet tartalmaz, így számítógéppel nagyon gyorsan ki lehet próbálni az összes lehetőséget. Nagyobb kulcsterű eljárás az affin titkosítás, amelyre

$$P = C = \{0,1,\ldots, 25\} \text{ és } K = \{(a,b) : 0 \leq a,b, \leq 25,(a,26) = 1\}.$$

A titkosító függvény $E(u,(a,b)) = au + b \bmod 26$. Jelölje a^- azt a 0 és 25 közötti egész számot, melyre $aa^- \bmod 26 = 1$. Ilyen szám a xx fejezet szerint létezik és a kiterjesztett euklideszi algoritmussal meghatározható. A dekódoló kulcs (a^-,b) és a visszafejtő függvény $D(m,(a^-,b)) = a^-(m-b) \bmod 26$. Ebben az esetben a kulcstér mérete $26\ \varphi(26) = 26\ 13 = 338$, ami lényegesen nagyobb, mint a Caesar titkosításnál, de a gyakorlatban még mindig nagyon kicsi.

A Caesar és az affin titkosítás közös általánosítása a helyettesítéses titkosítás. Ekkor is teljesül, hogy $P = C$ a kulcstér pedig a P permutációinak, azaz kölcsönösen egyértelmű leképezéseinek halmaza. Ha π a P egy permutációját jelenti, akkor $E(u,\pi) = \pi(u)$. jelölje π^- a π inverzét, akkor ez lesz a dekódoló kulcs és így $D(m, \pi^-) = \pi^-(m)$. Helyettesítéses titkosításnál

a kulcstér nagy; ha $|P|$ jelöli a P elemeinek számát, akkor K elemszáma nyilván $|P|!$. A korábbi példáinkat tovább folytatva, ha P az angol abc betűinek halmaza, akkor $|P| = 26$ és így K-é $26! = 403291461126605635584000000 \approx 4{,}03\ 10^{27}$. Ez már a gyakorlatban is elegendően nagy lenne, ha csak a kulcstér méretét tekintjük. Természetes nyelvekben készült szövegekre azonban a helyettesítéses titkosítás nem elég biztonságos, mert a karakterek előfordulásának a gyakorisága szigorú szabályoknak tesznek eleget, amelyek egyszerűvé teszik a kulcs és meghatározását és így a szöveg visszafejtését is. A részletekbe, a jegyzet keretei miatt nem térünk ki.

Az eddigiekben a *betűnkénti*, más terminológiával monoalfabetikus, titkosítás néhány egyszerű módszerét ismertettük. Ezek hiányosságai már a középkor végén is ismertek voltak, ezért biztonságosabb módszereket kerestek. Egy ilyen a *Vigenère titkosítás*, amelyet Blaise de Vigenère (1523–1596) francia diplomatáról neveztek el. A helyettesítéses titkosítás fő hibája, hogy a betűknek, azok minden előfordulásakor, ugyanazt a betűt feleltetik meg. Ki lehetne küszöbölni ezt a hiányosságot például úgy, hogy nem betűket, hanem betűcsoportokat helyettesítünk. Az ilyenek előfordulási gyakorisága már egyenletesebb, de a kódolás, különösen, ha azt kézzel végzik, nagyon elbonyolódik.

A Vigenère titkosítás során a simítást sokkal egyszerűbb eszközzel érjük el. A módszert ismét az angol abc-re ismertetjük. Készítsünk el egy olyan 26x26-os táblázatot, amelynek i-dik sora az abc i-dik betűjével kezdődik és az abc többi betűjével folytatódik úgy, mint azt a 8.5 ábrán bemutattuk. A kódoláshoz szükségünk van egy kulcsra, amely egy magunk választotta n betűs szó. Az üzenetet bontsuk fel n karakterből álló blokkokra úgy, hogy a szóközöket figyelmen kívül hagyjuk. Ha az üzenet hossza nem n többszöröse, akkor az utolsó blokk nem teljes. Írjuk a kódszót annyiszor egymás után az üzenet alá, ahány blokkot kaptunk (beleértve a nem teljes blokkot is). Ezek után egy blokkot úgy kódolunk, hogy megkeressük a blokk aktuális karakterét a táblázat első sorában, majd az alatta levő kódszó karaktert keressük meg a táblázat első oszlopában. Az a betű, amelyik a kiválasztott sor és oszlop találkozásában van, lesz a titkosított szöveg következő karaktere.

Tekintsük példaként ismét a „tizenegy orakor tamadunk” üzenetet és legyen a kulcs: „roham”. A kódolás eredményét a következő táblázatban találjuk:

t	i	z	e	n	e	g	y	o	r	a	k	o	r	t	a	m	a	d	u	n	k		
r	o	h	a	m	r	o	h	a	m	r	o	h	a	m	r	o	h	a	m	r	o	h	a	m
k	w	g	e	z	v	u	f	o	d	r	y	v	r	f	r	a	h	d	g	e	y		

8.6 ábra

A titkosított üzenetet az utolsó sorban találjuk. Látható, hogy most ugyanazon betű különböző előfordulásaihoz más betűt rendeltünk. Például a t betű első előfordulásához a k-t, míg a másodikhoz az f-et. Bár a Vigenère titkosítás során lényegesen egyenletesebb lesz a betűk előfordulásának gyakorisága, mit a helyettesítéses titkosítás után, a kódszó periodikus alkalmazása mégis elegendő statisztikai információt szolgáltat ahhoz, hogy finomabb elemzéssel nagyon gyorsan feltörhető legyen.

A Vigenère titkosítás formális leírásakor a betűk helyett, ugyanúgy, mint a Caesar titkosításnál, azok sorszámai halmazával dolgozunk. Ha az abc-ben m karakter van, akkor az abc-t azonosíthatjuk a $\{0,1,\ldots,m-1\}$ halmazzal, így $P = C = K = \{0,1,\ldots,m-1\}^*$, azaz az abc feletti véges szavak halmazával. Ha a titkosító kulcs, k_t, hossza n, és az üzenet, u, hossza h, akkor bontsuk fel u-t n hosszúságú szavak konkatenációjára, azaz legyen $u=u_1\ldots u_k$. Ekkor teljesül, hogy $n(k-1) < h \leq nk$. Tegyük fel, hogy $k_t = k_{t1}\ldots k_{tn}$. Ha az u egyik részszava $u_i = u_{i1}\ldots u_{in}$, akkor

$$E(u_i,k_t) = (u_{i1} + k_{t1} \bmod m) \ldots (u_{in} + k_{tn} \bmod m).$$

Az E függvényt minden egyes részszóra alkalmazni kell és az eredmények konkatenációja adja a titkosított szöveget. A dekódolás nyilvánvalóan ugyanúgy történik, mint a kódolás azzal a különbséggel, hogy nem hozzáadjuk, hanem kivonjuk a kódszó egyes „karaktereit”.

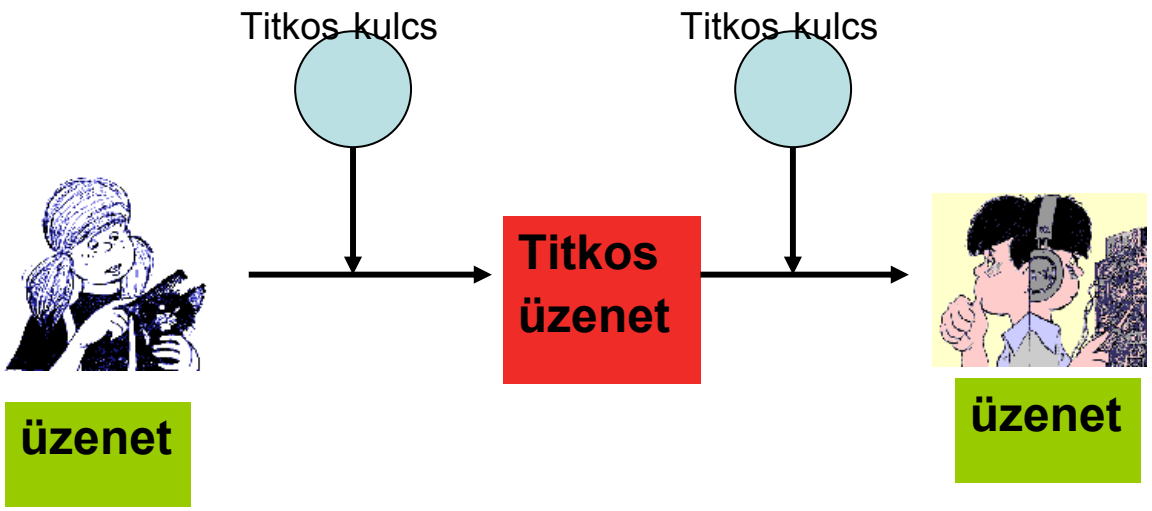
A 4. fejezetben említettük a Jefferson kereket, amely az első mechanikus titkosító eszköz volt. A XIX. majd a XX. században ezt az eszközt lényegesen tovább fejlesztették. A II. világháborúban a német hadsereg híres titkosító berendezése volt az ENIGMA. A szövetséges hadsered Normandiai partraszállásának sikerében fontos szerepet játszott, hogy az angol hadsereg zsákmányolt egy ENIGMÁ-t. A kriptoanalitikusoknak sikerült megérteni a működési elvét és így egyrészt megfejthették a németek üzeneteit, másrészt félrevezető üzeneteket küldhettek nekik. Teljesen más módszert választott az amerikai hadsereg a Csendes Óceáni hadműveletek során. Egy kis indián törzs, a navajok, tagjait alkalmazták üzenetek titkosítására. A törzs nyelvét nagyon kevesen beszélték, de elég kifejező volt ahhoz, hogy a hadvezetés üzeneteit le lehetett fordítani a navajo nyelvre. A fontos kommunikációs központokba tehát egy-egy navajo indiánt küldtek. Ők lefordították a parancsokat és elküldték azokat. A vevő oldalon is volt egy indián, aki megértette az üzenetet és visszafordította angolra, amit a helyi parancsnokok végre tudtak hajtani. Az amerikai hadsereg titkosítási módszerét nem tudták feltörni a világháborúban. Hasonlóan „titkosított” édesanyám és nagymamám. Amikor olyanról beszélgettek, amit nem akartak a gyerekek orrára kötni, akkor németre fordították a szót. Persze ez már nem működött akkor, amikor mi is megtanultunk németül. A titkosítás művészetéről szól Simon Singh nagyon olvasmányos könyve[16].

8.3 A szimmetrikus kriptográfia alapjai.

Az előző fejezetben láttuk, hogy az elmúlt évszázadok során sokféle titkosítási technikát dolgoztak ki, de a XX. század végéig főként az egykulcsos vagy szimmetrikus algoritmusokat használták. Ezek, persze lényegesen bonyolultabb formában, még ma is jelentős szerepet játszanak a kriptográfiában. Egy titkosító eljárást szimmetrikusnak nevezünk, ha a kódoló és a dekódoló kulcsok megegyeznek, vagy a dekódoló kulcs a kódolóból könnyen - polinomiális időn belül - megkapható. Ilyen módszert használva persze

[16] Simon Singh, The Code Book. How to Make It, Break It, Hack It, Crack It, Delacorte Press, New York, 2001.

mind a kódoló, mind a dekódoló kulcsot titokban kell tartani. A szimmetrikus titkosítás sémáját mutatja a következő ábra.



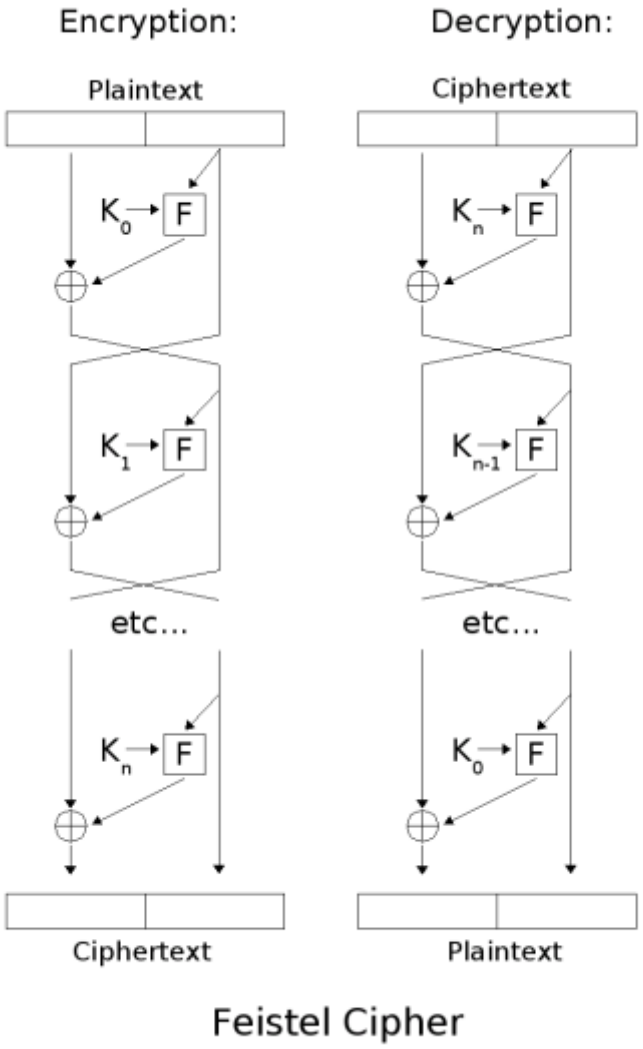
8.7 ábra

Az előző fejezetben tárgyalt klasszikus titkosító algoritmusok: eltolásos, affin, helyettesítéses és a Vigenére eljárások mind szimmetrikusak.

A szimmetrikus titkosítás előnye az egyszerűség és gyorsaság, hátránya viszont a legfőbb tulajdonságából származik: a titkosításhoz és kódoláshoz ugyanaz a kulcs használatos, így azt a feladónak és a címzettnek is ismernie kell. A jelszót (kulcsot) biztonságosnak ítélt úton - például személyes találkozáskor - kell eljuttatni a másik félhez. A személyes találkozás persze nem minden esetben kivitelezhető, hiszen gyakran fordul elő, hogy a partnerek igen nagy távolságban vannak egymástól, amikor sürgősen bizalmas üzenetet kell váltaniuk. Például az ország másik szögletében vagy külföldön tartózkodva egy tranzakciót kezdeményezünk a bankunknál. Más megoldás nem lévén ez az út valószínűleg ugyanaz a nem biztonságos csatorna lesz, amelyen a további kommunikáció is folyna. Ez a tény azonban megnehezíti az azonnali és globális kommunikációt. A modern rendszerekben a kulcsokat aszimmetrikus titkosítással juttatjuk el a partnereknek, akik aztán a jóval gyorsabb szimmetrikus módszerrel folytathatják a kommunikációt. A kulcscsere problémájáról és megoldási lehetőségéről később írunk.

A szimmetrikus titkosítás néhány modern képviselője a DES, a TripleDES (168 bit), AES, TwoFish, GOST 28147-89 és az IDEA (128 bit). Ezek sok, egyszerű transzformáció egymás utáni végrehajtása után érik el a kívánt titkosítási szintet. Kétféle tervezési elv kristályosodott ki: a Feistel és az SP-hálózatok.

Az elsőt Horst Feistel (1915-1990) német származású, de életének nagy részét az USA-ban töltő kriptográfus dolgozta ki. A Feistel titkosítás blokkdiagramját a 8.8 ábrán mutatjuk be. A módszerhez szükségünk van egy nem feltétlenül invertálható F függvényre és ha n-szer iteráljuk a kódolási menetet, akkor $n+1$ menetkulcsra: $K_0,\ldots,K_n$. A különböző titkosítási eljárások ezek megválasztásában térnek el egymástól.



8.8 ábra[17]

A kódoláskor az üzenetet (Plaintext) először két egyforma hosszúságú blokkra – L_0, R_0 – bontjuk. Általában, ha már kiszámoltuk L_i és R_i-t, akkor

$$L_{i+1} = R_i \text{ és } R_{i+1} = L_i \oplus F(R_i,K_i),\ i=0,\ldots,n.$$

Végezetül a titkosított üzenetet úgy kapjuk, hogy R_{n+1}-et és L_{n+1}–et konkatenáljuk. A 8.8 ábra bal oldali oszlopa mutatja a kódolás folyamatát, a jobb oldali pedig a dekódolását. A dekódolás során keletkező félszavakat jelöljük l_i és r_i-vel. Világos, hogy $l_0 = R_{n+1}$ és $r_0 = L_{n+1}$. Tegyük fel, hogy $l_i = R_{n+1-i}$ és $r_i = L_{n+1-i}$ teljesül valamely i ≥ 0-ra. Akkor

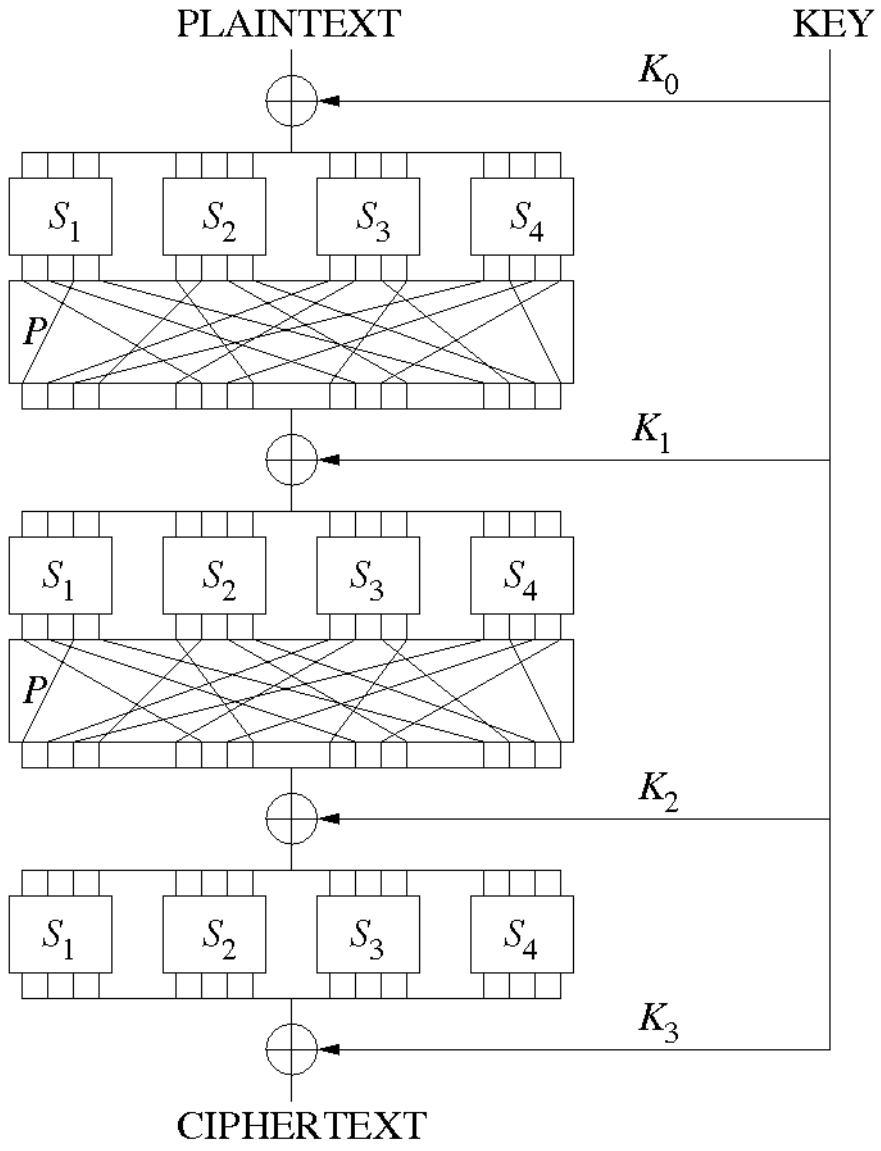
$$l_{i+1} = r_i = L_{n+1-i} = R_{n-i} \text{ és}$$

$$r_{i+1} = l_i \oplus F(r_i,K_{n-i}) = R_{n+1-i} \oplus F(L_{n+1-i},K_{n-i}) = L_{n-i} \oplus F(R_{n-i},K_{n-i}) \oplus F(R_{n-i},K_{n-i}),$$

amelyből közvetlenül következik $r_{i+1} = L_{n-i}$. A bizonyított relációt i=n+1-re alkalmazva kapjuk, hogy $l_{n+1} = R_0$ és $r_{n+1} = L_0$. A dekódoló algoritmus kimenete $r_{n+1}||l_{n+1} = R_0||L_0$, ami éppen az eredeti üzenet. Itt $x||y$ az x és y szavak konkatenációját jelöli. A bizonyításból

[17] Forrás: http://en.wikipedia.org/wiki/File:Feistel_cipher_diagram.png

látható, hogy Feistel módszere valóban független az *F* függvénytől és a menetkulcsoktól. Az egyetlen feltétel az, hogy *F* értéke olyan szó legyen, amelynek hossza megegyezik az üzenet hosszának felével.



8.9 ábra[18]

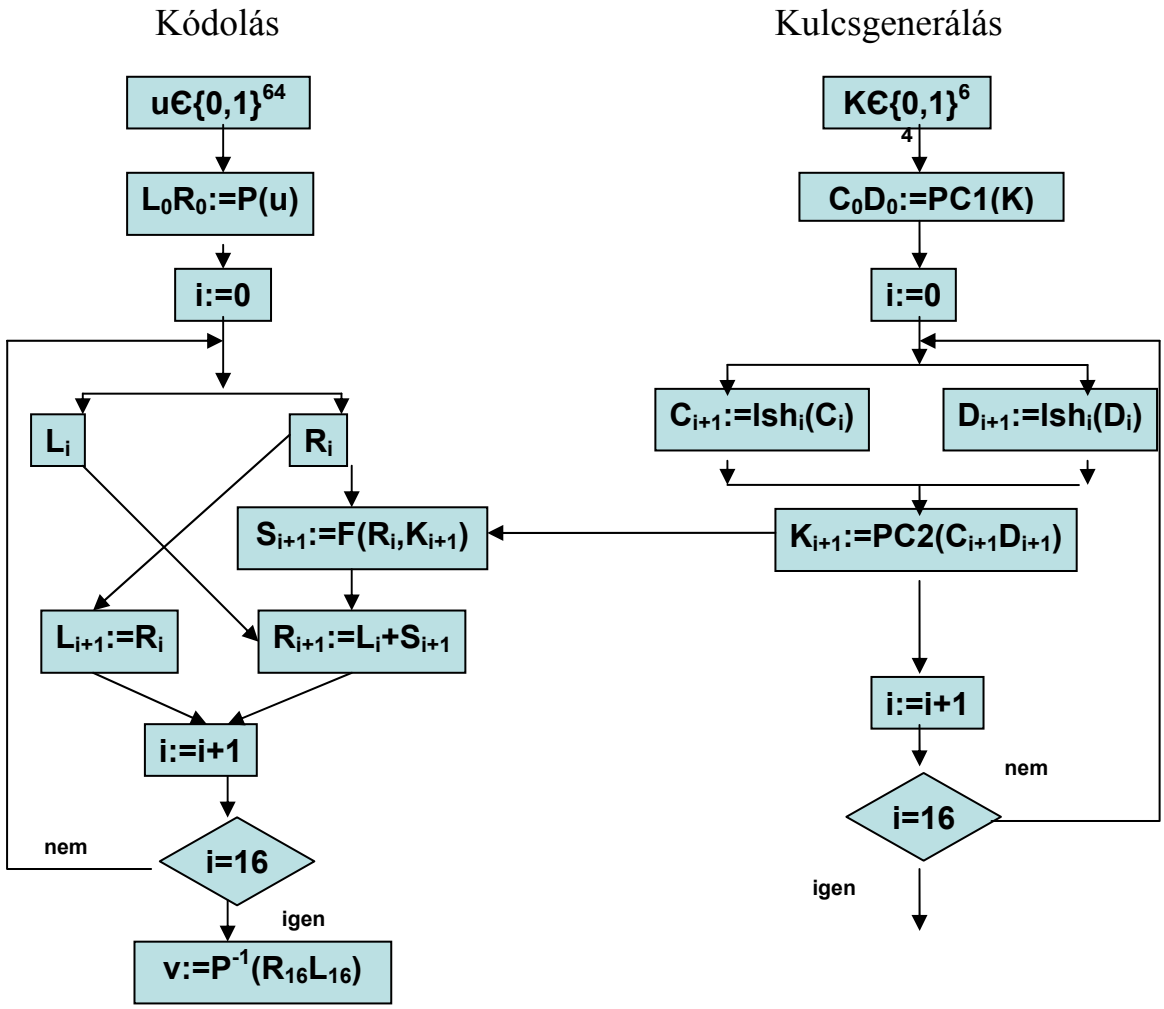
A másik gyakran használt módszer szimmetrikus titkosító eljárás készítésére látható a 8.9 ábrán. Ennek neve helyettesítő-keverő hálózat (substitution-permutation network), amelyet az angol neve után SP-hálózatnak neveznek. Az egyes iterációs lépésekben először a feldolgozandó szót kisebb blokkokra bontjuk és rájuk egy helyettesítést alkalmazunk. Ezután az egyes blokkok bitjeit szisztematikusan összekeverjük, majd az eredményt xor-ozzuk a menetkulccsal. A módszer előnye a Feistel hálózattal szemben, hogy a nyílt üzenetet már az

[18] Forrás: http://upload.wikimedia.org/wikipedia/commons/c/cd/SubstitutionPermutationNetwork2.png

első iteráció előtt xor-ozzuk az első menetkulccsal. A Feistel hálózatban az üzenet jobb félszava csak a második iterációban módosul. Az SP hálózat másik előnye, hogy kevesebb iterációra van szükség ugyanolyan titkosítás eléréséhez, mint a Feistel hálózatnál.

8.4 DES (Data Encryption Standard)

Az egyik legrégebbi és polgári alkalmazásoknál leggyakrabban használt szimmetrikus titkosítási algoritmus a DES, amely a Data Encryption Standard rövidítése. A DES egy Feistel hálózat és 1976-ban szabadalmaztatta az IBM. Az eljárás 64 bites üzenetblokkokat kódol 64 bites kulccsal, ebből azonban 8 bit paritásellenőrzésre szolgál, így a kulcs csak 56 bites. A DES folyamatábrája a 8.10 ábrán látható.



8.10 ábra

Bal oldalon a titkosító eljárást, a jobbon pedig a menetkulcsok generálását mutatjuk be. Az eljárás bemenete a 64 bites K kulcs, amelyből minden nyolcadik bit paritásellenőrzésre szolgál és az ugyancsak 64 bites u kódolandó információblokk. (Több blokkból álló üzenetet

úgy kódolunk, hogy a blokkokat külön-külön ugyanazzal a kulccsal titkosítjuk, majd a kapott blokkokat összefűzve juttatjuk el a fogadó félhez.) A DES algoritmus leírása:

Input: $u \in \{0,1\}^{64}$, $K \in \{0,1\}^{64}$.
Output: $v = DES(u,K) \in \{0,1\}^{64}$.
Paraméterek: P: $\{0,1,\ldots,63\} \rightarrow \{0,1,\ldots,63\}$ bijektív leképezés (permutáció) és ennek inverze P^{-1}.

1. $u_0 := P(u)$, $u_0 = L_0R_0$, ahol $L_0, R_0 \in \{0,1\}^{32}$.
2. for i := 0 to 16 do {
   $L_{i+1} := R_i$;
   $R_{i+1} := L_i \oplus F(R_i,K_{i+1})$;
   i := i+1;
   }
3. $v := P^{-1}(R_{16}L_{16})$;

Az algoritmusban $\oplus$ bitenkénti xor műveletet jelent, az F függvényt és a menetkulcsok kiszámítását pedig az alábbiakban írjuk le.
Input: $A \in \{0,1\}^{32}$, $J \in \{0,1\}^{48}$.
Output: $F(A,J) \in \{0,1\}^{32}$.
Paraméterek: E: $\{0,1,\ldots,31\} \rightarrow \{0,1,\ldots,47\}$ többértékű leképezés
R: $\{0,1,\ldots,31\} \rightarrow \{0,1,\ldots,31\}$ permutáció
S1,…,S8: $\{0,1\}^6 \rightarrow \{0,1\}^4$, amelyeket S-dobozoknak nevezünk.

1. D := E(A); (*Ennek során minden bitet felhasználunk, de 16 bitet kétszer.*)
2. $B := D \oplus J$;
3. B-t bontsuk fel nyolc darab 6 bites szóra, $B = B_1\ldots B_8$.
4. for i := 1 to 8 do $C_i := S_i(B_i)$; (* A B_i hatbetűs szó első és hatodik bitjéből álló kétbites szám megadja, hogy az S_i doboz melyik sorából, a maradék négy bitből álló szám pedig azt, hogy melyik oszlopából kell az eredményt kivenni. *)
5. $C := C_1\ldots C_8$;
6. F(A,J) := R(C);

A folyamatábrából látható, hogy a DES minden ciklusban 48 bites menetkulcsokat használ, amelyeket a mesterkulcsból származtat. Ennek algoritmusa:

Input: $K \in \{0,1\}^{64}$ mesterkulcs.
Output: $K1,\ldots,K16 \in \{0,1\}^{48}$ menetkulcsok.
Paraméterek: PC1: $\{0,1,\ldots,63\} \rightarrow \{0,1,\ldots,55\}$ paritásbitek eltávolítása és keverés
PC2: $\{0,1,\ldots,55\} \rightarrow \{0,1,\ldots,47\}$ injektív leképezés

1. K0 = C0D0 := PC1(K);

2. for i := 0 to 16 do {
   C_i := $lshift_i(C_{i-1})$;
   D_i := $lshift_i(D_{i-1})$; (* $lshift_i$ ciklikus balra tolást jelent i-től függően 1 vagy 2 pozícióval. Ha i = 1, 2, 9, 16, akkor 1-el kell eltolni, különben 2-vel. *)
   Ki := $PC2(C_iD_i)$;
   output K_i;
   i := i+1
   }

Az algoritmusban előforduló paraméterek szabványos értékei megtalálhatóak például a DATA ENCRYPTION STANDARD, FIPS PUB 46-3, http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf kiadványban.

A DES-el titkosított adatok dekódolása úgy történik, hogy a titkosító algoritmust alkalmazzuk a kódolt szóra, azonban a menetkulcsokat fordított sorrendben generáljuk.

A DES az 56 bites kulccsal ma már nem tekinthető biztonságosnak. A Bochumi Egyetem IT-Security intézete Horst Götz vezetésével néhány évvel ezelőtt kifejlesztett egy FPGA alapú eszközt, amelyet COPACOBANA-nak neveztek el. Az eszközzel elvégezték többek között a DES kriptoanalízisét[19] is. 2007-ben egy hétnél rövidebb idő alatt találtak meg DES kulcsokat. Hasonló eredmények után azt mondhatjuk, hogy a mai technológiával 64 bites kulcsokat tetszőleges titkosítási eljárásnál meg lehet találni. Ezért csak olyan eljárásokat érdemes konstruálni, amelyeknél a kulcsméret legalább 128. Bár az 1990-es évek elején a technika még nem tette lehetővé egyszerű és olcsó DES-kulcs-törők készítését, azok lehetőségét a szakértők előre látták. A DES azonban annyira elterjedt, hogy sokáig nem akarták új eljárással kiváltani, hanem a kulcshosszat növelték meg. Ez úgy történik, hogy nem egy hanem három DES futamot, három különböző kulccsal egymás után alkalmaznak az üzenetre. Így a kulcshossz 168 bitre nő. Ezt a formát háromszoros vagy tripla DES-nek, TDES-nek nevezzük.

8.5 GOST 28147-89

Ez a szimmetrikus titkosító algoritmus körülbelül egyidős a DES-szel, de csak a múlt század végén hozták nyilvánosságra. Nem polgári célra készítették, hanem a Szovjetunió hadseregében és felső párt- és államigazgatásában alkalmazták. A DES-hez hasonlóan ez is egy Feistel hálózat. Az alábbiakban megadjuk az eljárás pszeudokódját[20].

---

[19] Sandeep Kumar, Christof Paar, Jan Pelzl, Gerd Pfeiffer, Andy Rupp, Manfred Schimmler, How to break DES for € 8980, http://www.copacobana.org/paper/copacobana_SHARCS2006.pdf

[20] Forrás: А.А. Молдовян, Н.А. Молдобян, Б.Я. Собетов, Криптография, Лахь, Цанкт-Петербург, 2000.

Input: u Є $\{0,1\}^{64}$, $K_0,\ldots,K_7$ Є $\{0,1\}^{32}$.
Output: v = GOST(u,K) Є $\{0,1\}^{64}$.
Paraméterek: F: $\{0,1\}^{32} \rightarrow \{0,1\}^{32}$.

1. u = L||R, ahol L,R Є $\{0,1\}^{32}$. (*Bontsuk fel u-t 32 bites részszavak konkatenációjára.*)
2. for i := 1 to 32 do {
    a. V := R;
    b. if i < 25 then j := (i - 1) mod 8 else j := (32 – i) mod 8;
    c. R := $(R + K_j) \bmod 2^{32}$;
    d. R := F'(R);
    e. R := lshift(R,11);
    f. R := R ⊕ L;
    g. L := V;
    h. i := i+1;
    }

Az algoritmus c. lépésében az R és K_j szavakat 32 bites bináris számoknak tekintjük, és az összegüket képezzük moduló 2^{32}. Tekintettel arra, hogy egy 32 bites szó egy $[0, 2^{32}-1]$ intervallumba eső egész számot reprezentál, így $0 \leq R + K_j \leq 2^{33}-2$. A c. lépés tehát helyettesíthető az alábbival:

c'. if $R + K_j \geq 2^{32}$ then $R := R + K_j - 2^{32}$ else $R := R + K_j$;

Az e. lépésben az lshift(R,11) függvény azt jelenti, hogy az R szót 11 bittel ciklikusan balra kell shiftelni.

Az F' függvény megadásához szükséges nyolc darab S_7, S_6, S_5, S_4, S_3, S_2, S_1, S_0 táblázat, amelyek a {0,1,…,15} számok egy permutációját jelenti úgy, hogy minden számot négy bites szóként ábrázolunk. Bontsuk fel az R szót nyolc darab, négy bites részszóra, azaz legyen $R = r_7||r_6||r_5||r_4||r_3||r_2||r_1||r_0$. Ezek után helyettesítsük r_i helyére az S_i táblázat r_i-dik elemét. Az F'(R) így tényleg egy 32 bites szó. A Feistel hálózat definiálásánál használt F függvényt most a 2.b. – 2.f. utasítások határozzák meg. Ezért a dekódolásnál alkalmazható az általános módszer. Megjegyezzük, hogy a táblázatok is változók a GOST algoritmusban, így az aktuális kulcshossz 256 bitnél nagyobb.

A GOST algoritmust eddig nem vetették alá olyan részletes elemzésnek, mint a DES-t, így biztonságáról kevesebbet tudunk.

8.6 AES (Advanced Encryption Standard)

1997-ben a NIST (National Institute of Standard and Technology) pályázatot írt ki új szimmetrikus titkosító szabvány készítésére, amelyet Advanced Encryption Standardnak röviden AES-nek neveztek el. Eredményt 2000 őszén hirdettek, a győztes a Rijndael fantázianevű algoritmus lett, amelynek alkotói Vincent Rijmen és Joan Daemen, két belga

mérnök. A Rijndael a DES-szel és a GOST-tal ellentétben nem Feistel, hanem SP hálózatot alkalmaz a titkosításra.

Az AES 128/192/256 bites blokkokat 128/192/256 bites kulccsal titkosít, minden párosításban. Először csak a 128-128 bites párosítást, később mindegyiket elfogadták szabványnak. Az alábbiakban a 128 bites üzenetblokkokat, 128 bites kulccsal titkosító algoritmust ismertetjük. A Rijndaelról részletesen olvashatnak a Joan Daemen and Vincent Rijmen, "The Design of Rijndael: AES - The Advanced Encryption Standard." Springer-Verlag, 2002 könyvben.

A Rijndael a 128 bites input szót 16 bájtra bontja és ezeket egy 4x4-es táblázatba rendezi, amelyet állapotnak (state) nevez. Az eljárás függvényei az állapottáblázatokon operálnak. Négy függvényt használ, úgymint:

- ByteSub(State): az állapot minden bájtját kicseréli egy S-box által meghatározott bájtra. Az S-boxot matematikai függvényként is ki lehet számítani.
- ShiftRow(State): az állapot i-dik sorát i-1 pozícióval ciklikusan balra tolja.
- MixColumn(State): az állapot oszlopait, mint vektorokat megszorozza egy mátrixszal.
- AddRoundKey(State, RoundKey): bitenkénti xor az aktuális állapot és a menetkulcs között.

Az input szóra először az AddRoundKey függvényt alkalmazza, majd 9-szer a ByteSub, ShiftRow, MixColumn és AddRoundKey függvényekből álló blokkot. Az eljárást végül a ByteSub, ShiftRow és AddRoundKey blokk zárja. Látható, hogy az AES-nél a DES-szel szemben már az első lépésben megkezdődik a titkosítás a menetkulcs hozzáadásával. A kódolás során 11 menetkulcsot használ, amelyeket a mesterkulcsból számít ki. A paraméterek a fent idézett könyvben megtalálhatóak vagy az internetről letölthetőek.

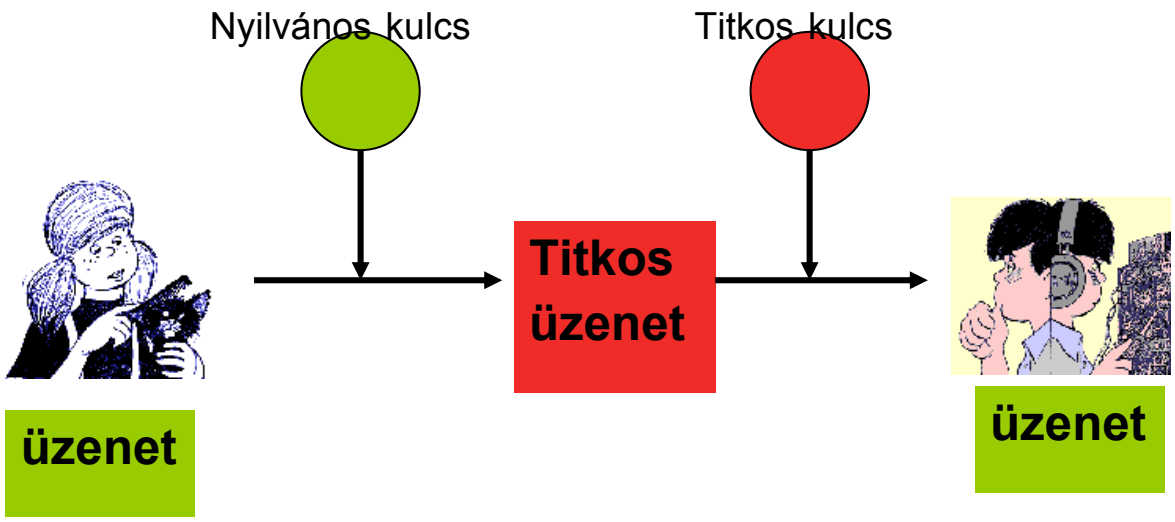
Megjegyezzük, hogy a ByteSub függvényhez használt S-box (helyettesítés táblázat) előállításának módja szintén része a szabványnak. Úgy választották ki, hogy a helyettesítés a lehető legtávolabb legyen a lineáris leképezésektől. A DES-nél alkalmazott S-boxok előállításának algoritmusa ezzel szemben máig ismeretlen, ami miatt sok szakértő kritizálta is a tervezőket.

8.7 Nyilvános kulcsú vagy aszimmetrikus titkosítás.

A szimmetrikus titkosítás évezredeken keresztül kielégítette az igényeket, mert a bizalmas üzenetcserét csak nagyon korlátozott körben és jól szervezett közösségekben – hadsereg, rendőrség, titkosszolgálatok – alkalmazták. A kommunikációs, majd az informatikai hálózatok megjelenésével és elterjedésével a bizalmas üzenetcserére megnőtt az igény. Banki vagy egészségügyi adatokat például nem szabad nyilvános csatornán, titkosítás nélkül továbbítani. A szimmetrikus algoritmusok, például a DES elég gyors és biztonságos volt a múlt század hetvenes éveiben, de volt egy gyenge pontja: a titkosító (és visszafejtő) kulcsot mindkét partnernek ismernie kell az üzenetcseréhez. A kulcsot tehát biztonságos módon kell eljuttatni a partnerhez, más nem szerezheti meg és a fogadó félnek biztosnak kell lenni abban,

hogy attól kapta a kulcsot, aki ezt állítja magáról. A klasszikus módszerek: a kulcs előzetes egyeztetése, futár vagy postagalamb alkalmazása lassú, egyedi és nagyon drága megoldás, az infokommunikációs hálózatok korában nem használható.

A problémát Whitfield Diffie és Martin E. Hellman fogalmazta meg egy 1976-ban megjelent dolgozatukban. Egyben megfogalmazták a megoldás elvét is, amelynek lényege: minden felhasználó (feladó és címzett egyaránt) rendelkezik egy kulcspárral, ami egy nyilvános (public) és egy titkos (private) kulcsot tartalmaz. A mindenki számára elérhető nyilvános kulcs a titkosításhoz, a csak a tulajdonosa által ismert privát kulcs pedig a visszafejtéshez használatos. Fontos megjegyezni, hogy a nyilvános kulcsból a titkos kulcs nem számítható ki még az előállításukra szolgáló algoritmus ismeretében sem.



Az előző ábra a nyilvános kulcsú titkosítás folyamatát mutatja. Amikor Kriszta bizalmas üzenetet akar küldeni Aladárnak, akkor elkéri vagy megkeresi Aladár *nyilvános* kulcsát. Ezzel kódolja az üzenetet és elküldi például e-mailben Aladárnak. Aladár a saját *titkos* kulcsával fejti meg – dekódolja – Kriszta üzenetét. Ha válaszolni akar, akkor persze Kriszta nyilvános kulcsával titkosít.

A két kulcs tehát egymást kiegészítve működik; a címzett nyilvános kulcsával titkosítjuk az üzenetet, amit rajta kívül más nem tud elolvasni, hiszen csak ő rendelkezik a visszafejtéshez szükséges titkos kulccsal. A módszer erősségét a szimmetrikus kulcsos titkosítás hátrányának kiküszöbölése adja: azok is tudnak titkosított üzeneteket váltani, akik nem ismerik egymást (elég, ha előzőleg kicserélték nyilvános kulcsaikat). Ez a csere történhet Interneten keresztül is, hiszen attól, hogy valaki megszerzi a nyilvános kulcsunkat még nem fér hozzá bizalmas információkhoz. További előnyös tulajdonsága, a digitális aláírás készítésének lehetősége, mely opciót a hitelességvizsgálat céljából érdemes kihasználni. Ezzel egy későbbi fejezetben részletesen foglalkozunk.

Diffie és Hellman dolgozata nyitva hagyta azt a problémát, hogy van-e nyilvános kulcsú kódolási eljárás. Cikkük megjelenése után azonban számos javaslat jelent meg a

probléma gyakorlati megvalósítására a tudományos irodalomban. Ezek közül néhányról több-kevesebb idő után kiderült, hogy nem felel meg a követelményeknek. Az egyik legelső algoritmus, az RSA, azonban máig feltörhetetlennek bizonyult és széles körben elterjedt.

8.7.1 Az RSA algoritmus.

Az RSA-t 1977-ben publikálta Ronald Rivest, Adi Shamir és Leonard Adleman és családi neveik kezdőbetűiből lett az RSA betűszó. Algoritmusuk elemi számelméleti ötleten alapszik.

Legyenek p és q különböző prímszámok, azaz olyan természetes számok, amelyeknek 1-en és önmagukon kívül nincs más osztójuk. Prímszámok például a 2,3,5,7,… Az ókori görög matematikus Eukleidész Elemek című könyvében szerepel annak bizonyítása, hogy végtelen sok prímszám létezik. A XIX. sz. vége óta azt is tudjuk, hogy a prímszámok elég gyakoriak. Annak a valószínűsége ugyanis, hogy egy véletlenszerűen kiválasztott x-nél kisebb szám prímszám legyen $1/\ln x$. A Miller-Rabin teszttel gyorsan eldönthető, hogy egy szám prímszám-e. (Pontosabban, ha egy szám átmegy a Miller-Rabin teszten, akkor csak azt mondhatjuk, hogy nagy valószínűséggel prímszám. Ez azonban a gyakorlatban elegendő. 2002-ben Manindra Agrawal, Neeraj Kayal és Nitin Saxena indiai informatikusok publikáltak egy polinom idejű determinisztikus prímszámtesztet, amelynek hatékonysága azonban ma még nem vetekedik a Miller-Rabin teszttel.)

Nagy prímszámokat tehát könnyű találni, de ha két ilyet összeszorzunk, akkor csak a szorzatot ismerve nagyon nehéz a tényezőket megtalálni. Ezt a faktorizáció problémájának nevezik, ami mai tudásunk szerint egy nagyon nehéz algoritmikus probléma.

Legyenek p és q különböző prímszámok és $n=qp$. Ekkor az n-nél kisebb, n-hez relatív prím természetes számok száma $\varphi(n) = (p-1)(q-1)$. Ezt az értéket p és q ismeretében könnyű kiszámítani. A φ függvényt Euler függvénynek nevezzük. Legyen most e egy olyan $\varphi(n)$-hez relatív prím természetes szám, amelyik kisebb $\varphi(n)$-nél. Akkor pontosan egy olyan $1 \le d < \varphi(n)$ természetes szám létezik, amelyre $ed \bmod \varphi(n) = 1$. Itt és a továbbiakban a *mod m* jelenti az a természetes szám maradékát m-mel osztva. Ezek után a *nyilvános kulcs* az (e,n) számpár, a titkos kulcs pedig a d szám. A kulcsok meghatározása után a p és q értékét is titokban kell tartani vagy ezeket a számokat meg kell semmisíteni.

A kódolás során az üzenetet először számok sorozatává alakítjuk olyan módon, hogy a számok mindegyike kisebb legyen, mint n. Ez könnyen megtehető, hiszen az üzenetet a számítógépben bináris alakban tároljuk és most ezt a bináris sorozatot, mint egy kettes számrendszerben megadott szám számjegyeit értelmezzük. Ezután az egyes m számokat az

$$M = m^e \bmod n$$

képlettel kódoljuk, előállítva a rejtjelezett M üzenetet. A kódoláshoz csak a nyilvános kulcsot, az e,n számpárt kell ismerni! A titkos M üzenetet az

$$m = M^d \bmod n$$

képlet alapján lehet dekódolni. A visszafejtéshez tehát a titkos *d* kulcs ismerete kell!

A kódolás és dekódolás során is moduláris hatványozást kell végezni, amelyik az „intelligens” hatványozó algoritmussal elfogadható gyorsasággal elvégezhető. Az elemi számelméletből jól ismert Euler-Fermat tételből következik, hogy a dekódolás után tényleg az eredeti üzenetdarabot kapjuk vissza.

Az algoritmus ismertetése után néhány megjegyzést teszünk az RSA paraméterek megválasztásával kapcsolatban. A titkos kulcs – *d* – mai ismereteink szerint csak a *φ(n)* birtokában számítható ki, *φ(n)* meghatározása viszont ugyanolyan nehézségű, mint *n* prímtényezőkre bontása. Az RSA biztonsága tehát azon múlik, hogy milyen gyorsan tudjuk az *n* számot faktorizálni. Ha *p* és *q* és így *n* is kicsi, akkor ez egyszerű feladat. Növelve azonban *p*-t és *q*-t egyre nehezebb, ma még megoldhatatlan problémához jutunk. A felhasznált számoknak olyan nagyoknak kell lenniük, hogy az *n* számot ne lehessen prímtényezőkre bontani. Ma azt mondhatjuk, hogy *n*-nek legalább 1024 bináris, azaz kb. 308 decimális jegyű számnak kell lennie. A faktorizáló algoritmusok pillanatnyi csúcsteljesítménye az RSA-200, egy 200 decimális jegyű szám tényezőkre bontása, amelyet közel két év munka után 2005-ben fejezett be egy 80 számítógépből álló klaszter.

A *p* és *q* megválasztása során nemcsak a nagyságukra kell figyelni, hanem arra is, hogy a különbségük is nagy legyen. Ellenkező esetben ugyanis a Fermat faktorizációs algoritmus gyorsan megtalálja a tényezőket. Feltéve, hogy az n 1024 bites szám, *p* és *q*-t 512 bitesnek célszerű választani úgy, hogy a különbségük legalább 400 bit nagyságú legyen.

Az *n* szám megválasztása után áttérünk *e* és *d* közelebbi elemzésére. Fentebb leírtuk, hogy *d* értékét az *e* és *φ(n)* egyértelműen meghatározza. Az is jól ismert, hogy *d* értékét a kiterjesztett euklideszi algoritmussal könnyen ki tudjuk számítani. A nyilvános kulcsdarab – *e* – megválasztására általában kétféle módszert követnek: az *e*-t véletlenszerűen választjuk ki az *[1, φ(n)-1]* intervallumból vagy olyan kis, páratlan számnak választjuk, amelynek bináris felírásában kevés 1-es számjegy található, például 17 vagy 65537. Az első esetben nem biztos, hogy rögtön olyan számot választunk, amelyik relatív prím *φ(n)*-hez, ilyenkor meg kell ismételni a választást, amíg ez a feltétel nem teljesül. Be lehet bizonyítani, hogy néhány választás után ez igen nagy valószínűséggel teljesül.

Az aszimmetrikus titkosítás elvének megfogalmazása óta eltelt több, mint 30 évben számos más algoritmust is javasoltak, például a diszkrét logaritmus kiszámításának nehézségén alapuló ElGamal, a diszkrét elliptikus görbéket alkalmazó algoritmus vagy a hibajavító kódok dekódolásának bonyolultságára építő Mc Ellise módszer. Ezekkel most nem foglalkozunk.

A nyilvános kulcsú titkosítás előnye, hogy a titkos kulcsot csak egy ember ismeri, így titkosított üzenetet nyilvános csatornán is lehet vele küldeni. A gyakorlatban azonban ez az előny csak korlátozottan aknázható ki, mert a jelenleg ismert módszerekkel a kódolás (és

dekódolás) nagyságrendekkel tovább tart, mint a szimmetrikus algoritmusokkal. Ezért az aszimmetrikus módszereket csak rövid üzenetek kódolására célszerű alkalmazni. Ezen az észrevételen alapulnak az úgynevezett hibrid kriptorendszerek, amelyeknél egy szimmetrikus és egy aszimmetrikus módszert – pl. AES és RSA – kombinálnak a következőképpen:

1. Kriszta választ egy K kulcsot a szimmetrikus algoritmushoz,
2. K-t Aladár nyilvános kulcsával kódolva elküldi Aladárnak,
3. Aladár a titkos kulcsával visszafejti K-t,
4. A bizalmas információcsere K használatával a szimmetrikus algoritmussal történik.

A kulcscserének vannak olyan variánsai is, amelyekben a szereplők egyforma mértékben veszik ki részüket a közös kulcs kiszámításában. A hibrid kriptorendszer működését úgy is felfoghatjuk, hogy a klasszikus szcenárióban alkalmazott futár szerepét az aszimmetrikus titkosítás veszi át. Ilyen elven működik a távoli számítógépre való biztonságos bejelentkezésre szolgáló ssh (secure shell) szabványcsalád.

Az aszimmetrikus titkosításnak a kulcscserén kívül vannak más, fontos alkalmazásai is. Korábban már hangsúlyoztuk, hogy a titkos kulcsot csak egyetlen ember, a tulajdonosa ismeri, így a titkos kulcs alkalmas a tulajdonos egyértelmű **azonosítására**. A titkos kulcsot persze nem kérhetjük el igazoltatáskor a tulajdonosától, mert ha odaadná, akkor az igazoltató is megismerné azt és a tulajdonos többé már nem is használhatná. Olyan módszert kell tehát kitalálni, amely során a tulajdonos nem fedi fel titkos kulcsát, hanem csak bizonyítja, hogy ő rendelkezik a titkos kulccsal. Mivel az eljárás nagyon hasonlít a digitális aláírásnál alkalmazott módszerhez, ezért a következő fejezetben foglalkozunk vele.

8.8 Szimmetrikus és aszimmetrikus titkosítás összehasonlítása

	Kulcsméret	Sebesség(kulcs-méret)	Haté-konyság	Kulcs tárolás
Szimmetrikus: DES, TDES, AES, …	64(56), 112, 128/192/256	~kulcshossz	1	nincs
Aszimmetrikus: RSA, ElGamal, …	1024/2048, 512/1024	~kulcshossz^3	1000	Amíg nem kompromit-tálódik.

	Előny	Hátrány

Szimmetrikus: DES, TDES, AES, …	Közérthető, Egyszerű programozni, Rövid kulcshossz, Gyors	Legalább két személy a titokgazda, A kulcsot rövid ideig lehet tárolni, Kulcscsere.
Aszimmetrikus: RSA, ElGamal, …	Matematikai eszközökkel elemezhető, Egy személy a titokgazda! A kulcs tárolható. Nyilvános/titkos kulcs	Lassú, Komplikált, Nehéz programozni.

9 Hash függvények és a digitális aláírás

9.1 Hash függvények

9.1.1 Hash függvények fogalma

A hash függvények fogalmával az informatika más területén már találkozhattunk, adatbázisok rendszerezésére alkalmaztuk. A kriptográfiában az adatok integritásának biztosítására szolgál. Ahelyett, hogy a tetszőleges hosszú és nagyméretű adatsor integritásának védelmét biztosítjuk, inkább mindössze egy fix hosszúságú, igen kisméretű bitsztringre (kb. 160 bit) koncentrálunk. A tetszőleges méretű üzenetre egy hash függvényt hajtunk végre, melynek eredményeként egy fix méretű *hash érték*et (*üzenetkivonat*ot vagy *lenyomat*ot) kapunk. A hash függvény tehát egy $H:\{0,1\}^* \rightarrow \{0,1\}^n$ függvény, azaz egy tetszőleges hosszúságú bitsorozatot egy fix hosszúságú bitsorozatba képez. Amennyiben sikerül az üzenetkivonat integritását megőrizni, akkor könnyen ellenőrizhető, hogy a nagyméretű eredeti üzenetünk változott–e. Az ellenőrző fél lefuttatja a hash függvényt az eredeti üzenetre és az eredményként kapott üzenetkivonatot összehasonlítja a korábbi üzenetkivonattal. Ha a lenyomatok megegyeznek, akkor az üzenet nem módosult. A hash függvények alkalmazásával a nyilvános, nem biztonságos csatornán integritásvédelmet lehet megvalósítani.

Egy jó példa a hash függvény használatára, valamely programkód változatlanságának ellenőrzése. Tekintsünk például egy titkosító eljárást, melyet a kliens gép merevlemezén tárolunk. Első alkalommal kiszámítjuk a program kódjának lenyomatát és egy smart kártyára másoljuk, melyet állandóan magunknál tartunk. Későbbiekben minden egyes használat előtt kiszámítja a merevlemezen levő programkód hash értékét és összehasonlítja a smart kártyán levő lenyomattal. Ha megegyeznek, akkor a kód nem módosult és biztonságosan használható.

Ahhoz, hogy ez a megoldás biztonságos legyen, garantálni kell, hogy az üzenet egyetlen bitjének módosulása maga után vonja a lenyomat változását. Ami valójában azt jelenti, hogy ne lehessen megadni két olyan üzenetet, melynek lenyomata megegyezik. A hash függvények nem injektívek, hiszen tetszőleges méretű üzenetekhez egy fix méretű bitsorozatot rendelünk. Amennyiben a lenyomat hossza k, akkor 2^k db különböző lenyomat lehetséges, míg az eredeti lehetséges üzenetek száma sokkal több. Ezért léteznek üzenetek, melyeknek lenyomata megegyezik. Ezek alapján célunk az, hogy *nehéz* (polinomiális idő alatt ne lehessen) legyen olyan üzeneteket találni, melyek hash értéke megegyezik. Az ilyen hash függvényeket *ütközésmentes hash függvény*eknek nevezzük. A különbség a kriptográfia, illetve általánosan az informatikában használt hash függvények között, hogy az előbbi esetben *nehéz* ütközéseket találni, míg az utóbbinál az ütközés előfordulása csak nem valószínű.

A kriptográfiában használt hash függvényeket az *elkötelezettségi rendszerek*nél (commitment scheme) is alkalmazzuk. Ha valaki szeretné elkötelezni magát egy x adathoz, anélkül, hogy megmondaná az x értékét, akkor kiszámítja $H(x||r)$ értéket, ahol H egy hash függvény és r egy véletlen bitsorozat. Később felbontja elkötelezettségét, azaz megadja az x

és az *r* értékeket. Az ilyen esetekben fontos az, hogy *H(x||r)* ismeretében az *x* értékről ne tudjunk meg hasznos információkat, azaz a *H* hash függvénynek egyirányúnak kell lennie. Az egyirányúság biztosítja, hogy a lenyomatból az eredeti üzenet kiszámítása *nehéz.* Az egyirányú hash függvény lehetővé teszi, hogy egy entitás az üzenet lenyomatának megadásával „borítékolja" az üzenetet, azaz elrejtse, de ugyanakkor elkötelezze magát amellett.

Az elkötelezettségi rendszerek alkalmazására egy jó példa a pénzfeldobás telefonon. A játék lényege az, hogy a két résztvevő nem látja egymást, csak telefonon keresztül beszélgetnek. Az egyik fél feldob egy pénzérmét, és megkéri a másikat tippeljen, hogy fej avagy írást kapott. A másik fél megtippeli, hogy fej. A pénzt feldobó azt fogja mondani, hogy nem nyert, hiszen írás lett. Mi a garancia arra, hogy írás lett? A pénzt feldobó személy akár füllenthetett is arról, hogy mit dobott. Ez a játék szabályossá válik, ha valamely elkötelezettségi rendszert használunk. Az egyik személy feldobja az érmét és az eredményt egy véletlen értékkel együtt hash-eli. Az így kapott lenyomatot megmondja a másik félnek és kéri, hogy tippeljen. A tipp után megadja az eredményt és a véletlen értéket is, melyek alapján a tippelő fél ellenőrizheti, hogy tényleg az lett–e a pénzfeldobás eredménye. A rendszer biztonságához két dolog is szükséges: az egyirányúság, illetve, hogy nehéz legyen a lenyomathoz egy másik megfelelő ősképet találni.

9.1.2 Támadások

A hash függvényekkel szembeni támadásokat két fő kategóriába sorolhatjuk: az *őskép elleni és ütközéses támadások.*

1. *Őskép elleni támadások*

Kétféle támadást különböztetünk meg, az első és a második őskép elleni támadást.

- Az *(első) őskép elleni támadás* célja, hogy a támadó az *y* lenyomat esetén találjon egy olyan *x* értéket, hogy *H(x)=y.* Ha egy hash függvény az első őskép elleni támadással szemben biztonságos, akkor *őskép ellenálló hash függvény,* a függvény egyirányúságára utal.
- A *második őskép elleni támadás* célja, egy adott *x* érték estén olyan *x'* érték megadása, hogy *H(x)=H(x'),* ahol *x'≠x.* Amennyiben egy hash függvény a második őskép elleni támadással szemben védettséget ad, akkor *gyengén ütközésmentes* vagy *második őskép ellenálló* hash függvénynek nevezzük.

2. *Ütközéses támadások*

Az ütközéses támadás célja, hogy a támadó meghatározzon két különböző tetszőleges üzenetet, melyek hash értéke megegyezik, azaz találjon *x* és *x'* bitsorozatot, ahol *x'≠x* és *H(x)=H(x').* Jelentős különbség az ütközéses és az őskép elleni támadások között, hogy az ütközéses esetben a lenyomat sem és az inputok egyike sem áll a támadó rendelkezésére. Speciálisan *adott prefixű ütközéses támadás* célja, hogy a támadó meghatározzon két különböző p_1 és p_2 bitsorozathoz, ahol $p_1 \neq p_2$, olyan m_1 és m_2 bitsorozatokat, hogy

$H(p_1||m_1)=H(p_2||m_2)$. A || szimbólum a konkatenációt jelöli. Az ütközéses támadással szemben biztonságos hash függvényeket *(erősen) ütközésmentes*nek nevezzük.

Különböző szituációkban a releváns támadások különbözőek. Integritásvédelem estén a támadó célja, hogy egy adott *x* üzenethez olyan *x'* üzenetet találjon, melyek lenyomatai megegyeznek. Ez tipikusan a második őskép elleni támadás. Az elkötelezettségi rendszerek esetén, mint ahogy azt már fentebb részleteztük fontos az egyirányúság, azaz a támadó sikeres első őskép elleni támadást akar végrehajtani. Illetve, a támadó próbál azzal csalni, hogy az *x* üzenetet kicseréli egy *x'* üzenetre még az elkötelezettség felfedése előtt, azaz ütközéses támadást végez. A fent részletezett pénzfeldobós játék esetén a támadó adott prefixű ütközéses támadást indít, hiszen célja az, hogy olyan „véletlen" m_1 és m_2 értékeket „találjon ki", hogy $H(„fej"||m_1)=H(„írás"||m_2)$ teljesüljön.

Bizonyítható, hogy az erősen ütközésmentes függvények gyengén ütközésmentesek, és bizonyos feltételek mellett őskép ellenállók is [1]. A digitális aláírásoknál erősen ütközésmentes, őskép ellenálló hash függvényeket alkalmazunk.

9.1.3 MD5

Az MD5 (Message-Digest algorithm 5) egy 128 bites hash értékkel rendelkező hash függvény. Az MD5-öt az RSA egyik alkotója Rivest fejlesztette ki 1991-ben. 2004-ben több biztonsági rést fedeztek fel az algoritmusban, mely alapján 2005 óta digitális aláírásoknál használata nem javasolt.

Hash függvények készíthetőek kompressziós függvényekből. Az $CF:\{0,1\}^n \rightarrow \{0,1\}^m$ függvényt *kompressziós függvény*nek nevezzük, ha $m<n$. Látható, hogy a kompressziós függvények egy fix hosszúságú üzenethez egy rövidebb fix hosszúságú üzenetet rendelnek.

Az MD5 konstrukciójában is kompressziós függvényt alkalmazunk, melyet egymástól függetlenül Ralph Merkle és Ivan Damgard tervezett 1989-ben. Ennek a *CF* kompressziós függvénynek két inputja van, egy 128 bit hosszú *Y* sztring és egy 512 bit hosszú *B* blokk, a függvényérték pedig egy 128 bites sztring. Egy tetszőleges hosszúságú *x* üzenet lenyomatának generálása a következőképpen történik:

1. A tetszőleges hosszú *m* üzenetet kitöltjük úgy, hogy a hossza az a legkisebb érték legyen, mely osztható 512-vel. A kitöltést egy *1*-es bittel kezdjük, majd tetszőleges számú *0*-t írunk. A kitöltést az eredeti *m* üzenet 64 biten ábrázolt hosszával zárjuk. Az így kitöltött üzenetet jelöljük *M*-mel.
2. Vágjuk fel az *M* üzenetet 512 bit hosszú blokkokra, jelölje $B_1, B_2, \ldots B_n$ az így keletkezett blokkokat.
3. Vegyünk egy fix 128 bit hosszú kezdeti vektort: *IV,* és legyen $Y_0=IV$.
4. Kiszámítjuk $Y_i=CF(Y_{i-1},B_i)$ kompressziós függvényértékeket, ahol $i=1,\ldots,n$.
5. Az eredeti *m* üzenet lenyomata: $H(m)=Y_n$, azaz a keletkezett hash érték az utolsó blokkra vonatkozó kompressziós függvényérték lesz, mely 128 bit.

Meg kell, hogy jegyezzük, hogy ez a konstrukció maximálisan 2^{64} bit hosszú üzenetekre alkalmazható, hiszen az üzenet hosszát *64* biten ábrázoljuk. Ez a gyakorlatban a szám nagysága miatt nem jelent megkötést.

Bizonyítható, hogy az előbbi konstrukció esetén, ha a *CF* kompressziós függvény ütközésmentes, akkor a hash függvény is az.

Az MD5 *CF* kompressziós függvénye a Davies-Meyer kódoló algoritmusán alapszik, mely egy 128 bites és egy 512 bites sztringet 128 bites sorozatba képez. Jelölje CF_0 ezt a kódoló algoritmust, $Y=(A,B,C,D)$ a 128 bites input bitsztringet és B_i az 512 bites blokkot. Ekkor $CF(Y,B_i)=CF_0(Y,B_i)+(A,B,C,D)$, ahol az összeadást modulo 2^{32} végezzük. Első körben az *A,B,C,D* értékek definiált fix konstansok.

A CF_0 kódoló algoritmus főbb lépéseit ismertetjük csak, a részletes leírást az RFC 1321 szabvány **[24]** tartalmazza. Minden egyes B_i blokkot 16 db 32 bites sztringre bontunk. Az algoritmus 4 körből áll. Minden egyes körben a *következő* transzformáció hajtódik végre:

$$ROTL^s(a+f_i(b,c,d)+x+k)+b$$

A $ROTL^s()$ bitenkénti balra történő rotációt jelöl, a felső indexben levő *s* érték a rotáció mértékét adja meg. Az *s* és *k* fix, definiált konstansok. Az *x* jelöli az 512 bites B_i blokk 32 bit hosszú részét. Az *a,b,c,d* kezdőértékei *A,B,C,D,* és az algoritmus befejezése után ez négy érték együttesen fogja adni a hash értéket. Az f_i $(i=1,...,4)$ bitenkénti logikai függvény, mely a következőképpen van definiálva:

- $f_1(b,c,d)=(b \text{ AND } c) \text{ OR } ((\text{NOT } b) \text{AND } d)$
- $f_2(b,c,d)= (d \text{ AND } b) \text{ OR } ((\text{NOT } d) \text{AND } c)$
- $f_3(b,c,d)=b \text{ XOR } c \text{ XOR } d$
- $f_4(b,c,d)=c \text{ XOR } (b \text{ AND } (\text{NOT } d))$

Az MD5 ellen több *gyakorlatban* is alkalmazható támadást is bemutattak. Az MD5 nem mutat védettséget az ütközéses támadással szemben, illetve sikeres adott prefixű ütközéses támadást is végre lehet hajtani. 2005-ben sikeres támadást mutattak be tanúsítványokon, ugyanazon két különböző nyilvános kulcs MD5 lenyomata megegyezett. Tehát a gyakorlatban már nem javasolt alkalmazni.

9.1.4 SHA

A másik gyakorlatban elterjedt hash függvénycsalád az SHA (Secure Hash Algorithm). Az SHA az MD5 *rendszeren* alapszik, 1993-ban az amerikai FIPS 180 szabványban lett ismertetve. Tipikusan digitális aláírási rendszereknél alkalmazzák. Az SHA lenyomat *160* bit hosszú, a Merkle-Damgard konstrukciót használja, a kompressziós függvény *160* bites és *512* bites sztringhez *160* bitet rendel $(160\times512\rightarrow160)$. A kompressziós függvény a következőképpen adott:

$$CF(Y,B_i)=CF_0(Y,B_i)+(A,B,C,D,E).$$

Hasonlóan az MD5-höz az SHA is egy CF_0 kódoló függvényt futtat, mely egy *160* bit hosszú *Y=(A,B,C,D,E)* és *512* bites B_i blokkhoz *160* bitet rendel. További részleteket a **(3)** dokumentumban olvashatunk.

Az amerikai szabvány az SHA és SHA-1 algoritmuson kívül tartalmazza az SHA224, SHA256, SHA384 és SHA512 algoritmusokat is, melyeket közös néven SHA-2-nek is neveznek. Ez a négy algoritmus nagyon hasonlít az SHA-hoz, csak komplexebb. Az elnevezésekben a háromjegyű számok az algoritmus által generált lenyomat hosszát jelzik. Az SHA256 egy *256* bit hosszú üzenetkivonatot eredményez. Az SHA384 és SHA512 a *32* bites részblokkok helyett *64* bitesekkel számol és az *512* bites blokkméret helyett *1024* bites üzenetblokkokat kezel.

9.1.5 Születésnapi paradoxon

A születésnapi paradoxon alapvető kérdése a következő. Tegyük fel, hogy egy szobában véletlenül választott emberek egy csoportja tartózkodik. A kérdés az, hogy mennyi a valószínűsége, hogy legalább két személynek megegyezik a születésnapja. A meglepő válasz az, hogy annak valószínűsége, hogy legalább két ember ugyanazon a napon született több mint *50%,* ha a csoport létszáma 23. Emiatt a meglepő eredmény miatt hívjuk paradoxonnak, hiszen a *23* fő nagyon kis létszám az év *365* napjához képest.

A valószínűség kiszámítása a következő képlet alapján történik, ha az év *365* napját tekintjük és a véletlenül választott személyek száma *n*:

$$P=1-\frac{\binom{365}{n}n!}{365^n}$$

Ez a valószínűség jól közelíthető a

$$P=1-e^{-\frac{n^2}{2\cdot 365}},$$

A születésnapi paradoxon a hash függvényeknél nagy jelentőséggel bír, hiszen a paradoxon segítségével megadhatjuk egy ütközés véletlenszerű megtalálásának a valószínűségét. Ugyanis, ha egy *n* bit hosszú lenyomatot adó hash függvényt tekintünk, akkor a születésnapi paradoxon szerint $2^{\frac{n}{2}}$ véletlenül választott üzenet között annak valószínűsége, hogy ütközés lesz, azaz két érték hash értéke megegyezik 50%-hoz közeli érték. Ez alapján a hash érték méretét úgy kell meghatározni, hogy $2^{\frac{n}{2}}$ elég nagy legyen, hogy a gyakorlatban az ilyen jellegű támadás kivitelezhetetlen legyen. Jelenleg ez az *n* érték *160.*

Következésképpen az MD5 által generált *128* bites sztring az elég kicsi ahhoz, hogy ilyen jellegű támadást lehessen megvalósítani. Az MD5CRK projekt keretén belül ütközést találtak a születésnapi támadás felhasználásával.

9.1.6 Üzenethitelesítés

Az üzenethitelesítő kódokra a MAC (Message Authentication Code) rövidítést is használják. Ezek a kódok egy dokumentum hitelességét garantálják egy nem biztonságos csatorna használatakor. A küldő és a fogadó fél biztonságos csatornán kicserél egy titkos kulcsot, majd a küldő fél a titkos kulccsal elkészíti üzenetének MAC kódját. Az üzenet is és a kód is továbbítódik. A fogadó fél a titkos kulcs ismeretében szintén kiszámítja a kapott üzenet MAC kódját és ellenőrzi, hogy a két MAC kód megegyezik–e.

Az üzenethitelesítő kódok, mint ahogy a neve is mutatja, az üzenet hitelességét garantálják, ami magába foglalja az üzenet adatintegritását, azaz változatlanságát és az üzenet eredetét is. Következésképpen MAC használatával ellenőrizni tudjuk, hogy a kapott üzenet megegyezik az elküldöttel, valamint az üzenetet ténylegesen az a személy vagy entitás küldte, akitől várjuk.

A következő alfejezetben a digitális aláírással foglalkozunk, mely szintén a hitelesítés eszköze. Az alapvető különbség a hitelesítő kódok és a digitális aláírások között, hogy a digitális aláírás letagadhatatlanságot is garantál, azaz bárki a nyilvános kulcs ismeretében ellenőrzést végezhet, míg a MAC esetén csak a két fél képes ellenőrzést végezni, hiszen csak ők ismerik a titkos kulcsot.

Egyik legelterjedtebb hitelesítő kód a CBC-MAC, mely az ISO/IEC 9797 szabvány. Ugyanakkor hitelesítő kódot kaphatunk hash függvényekből is a következőkben ismertetett megoldást HMAC-nek nevezik.

9.1.6.1 HMAC

A HMAC (Hash-based Message Authentication Code), az RFC 2104 Internet szabvány. A HMAC egy a kriptográfiában alkalmazott hash függvény és egy titkos kulcs kombinációja. A konstrukció lehetővé teszi az MD5 és a SHA-1 hash függvény alkalmazását is, ekkor HMAC-MD5-nek vagy HMAC-SHA-1-nek nevezzük a hitelesítő kódot.

A HMAC az üzenetet szintén blokkokra bontja, melyek mérete MD5 és SHA-1 esetén *512* bit. A blokkokra kompressziós függvényt alkalmazva a HMAC egy *128* vagy *160* bit hosszú sztringet ad attól függően, hogy milyen hash függvényt alkalmaz. Legyen *m* az üzenet, melynek a hitelesítő kódját kívánjuk meghatározni, jelölje *K* a kicserélt titkos HMAC kulcsot és legyen *H* egy választott hash függvény. A lépések a következők:

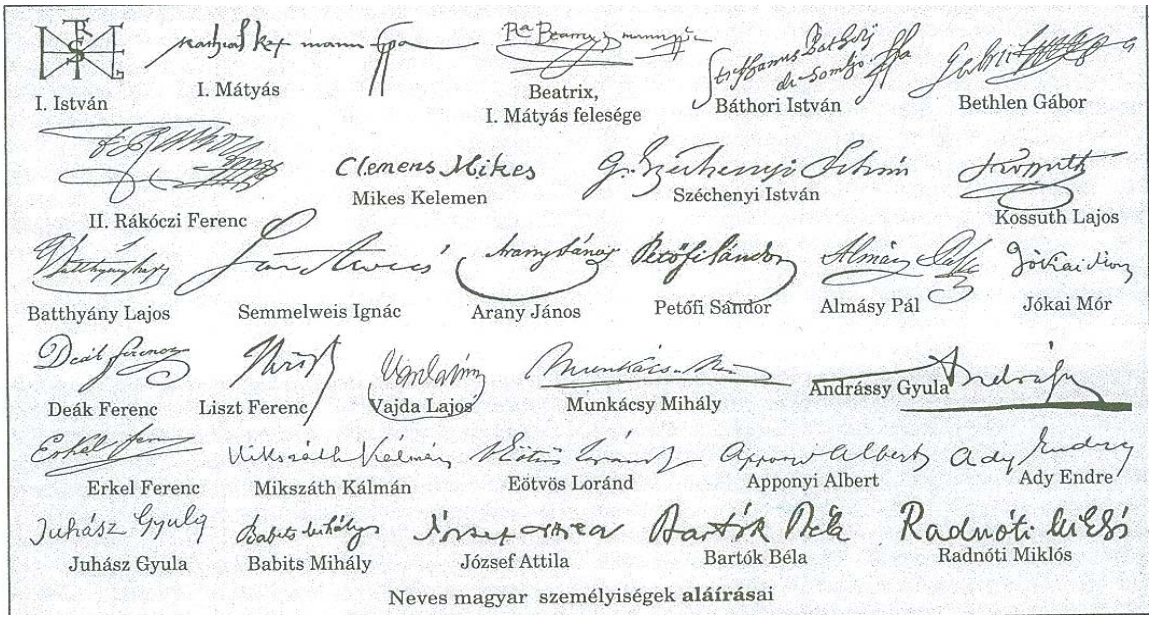
1. Ha *K* hosszabb, mint a hash függvény blokkmérete, akkor legyen *K* a *H(K)* érték, így *K* rövidebb, mint a blokkméret.
2. Ha *K* rövidebb, mint a blokkméret, akkor kitöltjük *0*-val míg a mérete megegyezik a blokkmérettel.
3. Kiszámítjuk *H((K□opad)||H((K□ipad)||m))*, ahol *ipad* és *opad* két fix bitsztring. Az *opad* (outer padding) külső kitöltést, míg az *ipad* (inner padding) belső kitöltést jelöl. Az *ipad* egy blokkméretnyi hexadecimális konstans: *0x363636…36*. Az *opad* szintén egy blokkméretnyi hexadecimális konstans: *0x5c5c5c…5c5c*.

9.2 Digitális aláírás

9.2.1 Digitális aláírásokról általában

9.2.1.1 Hagyományos és digitális aláírások összehasonlítása

A hagyományos aláírási rendszer mindennapi életünk fontos részét képezi. Aláírásunkkal látjuk el a különböző szerződéseket, hivatalos dokumentumokat, kérvényeket, űrlapokat, valamint személyes leveleinket is. Az aláírt dokumentum igazolja az aláíró személyét vagy intézményét, valamint azt, hogy az adott személy/intézmény a dokumentum tartalmát ismeri, elfogadja, egyetért vele. Ezen kívül az aláírás igazolja, hogy a dokumentum tartalma nem változott meg; nem írtak bele új mondatokat és nem is töröltek belőle.



9.3 ábra

A 9.3 ábrán néhány neves magyar személyiség aláírása látható, amelyet a Magyar Nagylexikon aláírás címszavából kölcsönöztünk. Érdemes megfigyelni, hogy az aláírások képe milyen sokat egyszerűsödött az évszázadok alatt. A 9.4 ábra kilenc aradi vértanú aláírását mutatja[21].

A hagyományos és digitális aláírások több szempontból is eltérnek egymástól. A hagyományos aláírás a fizikai dokumentum részét képezi, a fizikai hordozóhoz (pl. papírhoz) tartozik, ahhoz hozzáfűzött egyedi kézjegy. Módosítása és másolása csak az adathordozó (papír) manipulálásával érhető el. A digitális aláírás ezzel szemben az üzenthez csatlakozik,

[21] Forrás: Aradi Vértanúk Albuma, Szerkesztette: Varga Ottó, Lampel R. (Wodianer F. és fia) könyvkiadása, Budapest, 1890.

azaz az aláíró algoritmus hozzáfűzi az aláírást a konkrét elektronikus dokumentumhoz. Ha hagyományosan írunk alá egy hosszabb dokumentumot, akkor minden egyes oldalt kézjegyünkkel el kell látni, viszont mindössze egy digitális aláírást csatolunk bármilyen méretű elektronikus üzenethez.

Jelentős különbség van az aláírás ellenőrzésének folyamatában is. A hagyományos aláírást egy – szükség esetén grafológus segítségével – hitelesített, aláírási mintához hasonlítják. Digitális aláírás érvényessége ellenőrző algoritmus futtatásával igazolható, mely bárki számára elérhető. Aláírás hamisítási szempontból a digitális aláírások sokkal megbízhatóbbak.

9.4 ábra

A harmadik jelentős különbség a már aláírt dokumentum másolhatóságában van. A hagyományosan aláírt dokumentum fénymásolata megkülönböztethető az eredetitől. Digitálisan aláírt dokumentum könnyen másolható és a másolat megegyezik az eredetivel. Egy megoldás speciális digitális aláírási rendszerek használata, a *letagadhatatlan aláírások* alkalmazása, mikor az aláírás ellenőrzéséhez az aláíró személy részvétele is szükséges.

9.2.1.2 Az elektronikus aláírások kategóriái

Napjainkban mindennapi életünk lebonyolítása egyre inkább elektronikusan történik, így egyre több elektronikus dokumentumot használunk. Természetes igény a hagyományos aláírás eszközének megfelelő elektronikus megoldás alkalmazása.

Az elektronikus aláírásokat három különböző kategóriába sorolhatjuk, a normál elektronikus aláírás, a fokozott biztonságú, valamint a minősített elektronikus aláírások kategóriájába.

A *normál elektronikus aláírás* a legtágabb kör, bármely elektronikus formájú aláírás idesorolható. Példaként említhetjük az e-mailjeink végére begépelt nevünket, vagy a különböző digitális tollakkal írt aláírást is.

Az elektronikus aláírásokról szóló törvény alapján a *fokozott biztonságú elektronikus aláírás* olyan "elektronikus aláírás, amely megfelel a következő követelményeknek":

1. Alkalmas az aláíró azonosítására, és egyedülállóan hozzá köthető,
2. Olyan eszközzel hozták létre, mely kizárólag az aláíró befolyása alatt áll,
3. A dokumentum tartalmához olyan módon kapcsolódik, hogy minden - az aláírás elhelyezését követően az iraton, illetve dokumentumon tett - módosítás érzékelhető

A fokozott biztonságú elektronikus aláírás tehát a nyilvános kulcsú technológia, azaz a *digitális aláírási* technika felhasználásával készül. Amennyiben a fokozott biztonságú elektronikus aláírás minősített tanúsítványon alapul, akkor minősített *elektronikus aláírás*ról beszélünk. A minősített tanúsítványokról a 0 fejezetben olvashatunk.

Egyes bírósági eljárásokban legalább fokozott biztonságú elektronikus aláírással ellátott elektronikus iratokat fogadnak el, a minősített elektronikus aláírással ellátott elektronikus dokumentum teljes bizonyító erejű magánokiratnak minősül.

9.2.2 Digitális aláírási séma

A digitális aláírási séma ismertetéséhez három algoritmust kell megadni. Szükség van egy olyan algoritmusra, mely az üzenet aláírására szolgál, és szükséges az üzenethez csatolt aláírás ellenőrzését végrehajtó eljárás is. A harmadik algoritmus a kulcsgeneráló algoritmus, mely során meghatározódnak az aláírás során, illetve az ellenőrzéshez használt kulcsok. Biztosítani kell, hogy az aláírás ellenőrzését bárki elvégezhesse, azaz az ellenőrző algoritmus és a szükséges paraméterek mindenki számára elérhetőek legyenek. Az aláíró entitás egy tetszőleges $m \in P$ nyílt szöveghez – ahol P jelöli a lehetséges üzenetek halmazát – generál egy $s \in A$ aláírást – ahol A jelöli az aláírások halmazát – az aláíró algoritmus segítségével úgy, hogy azt csak ő tudja kiszámítani. Ezt úgy lehet biztosítani, hogy felhasznál egy aláíró kulcsot, amit csak ő ismer. Következésképpen az $s \in A$ aláírás értéke függ az $m \in P$ nyílt szövegtől és a kulcsgeneráló algoritmus által létrehozott titkos aláíró kulcstól. Az így keletkezett aláírás a nyílt üzenethez csatolódik. Az aláírás tulajdonképpen egy a nyílt szöveghez csatolt bitsorozat. Az aláírást ellenőrző entitás, pedig a kapott $(m,s) \in P \times A$ elempár és a rendelkezésre álló nyilvános információk alapján lefuttatja az ellenőrző algoritmust, melynek visszatérési értéke *igaz*, amennyiben az aláírás érvényes, és *hamis*, ha nem. Az ellenőrző entitás tehát az $s \in A$ aláírás érvényességét a kapott $m \in P$ nyílt szöveg és egy nyilvános ellenőrző kulcs segítségével dönti el.

A digitális aláírási séma jellemzői alapján kézenfekvő, hogy a konkrét megoldások aszimmetrikus rendszereket alkalmaznak. Hiszen, mint ahogy azt a 8.7 fejezetben ismertettük, aszimmetrikus rendszerek esetén minden résztvevő egy kulcspárral rendelkezik, ami egy titkos és egy nyilvános kulcsból áll. Mivel a nyilvános kulcsból a titkos kulcs kiszámítására polinomiális algoritmus nem ismert, így nyugodtan – mint ahogy a nevében is szerepel – nyilvánosságra is hozható, a titkos kulcs titkossága nem sérül, biztonságosan alkalmazható. Digitális aláírási sémáknál a titkos kulcsot aláírásra használjuk, hiszen alapvető cél az aláíró entitás egyértelmű beazonosítása, a nyilvános kulcs pedig az aláírás ellenőrzésére szolgál, hiszen garantálni kell, hogy bárki képes legyen arra, hogy eldöntse az aláírás hitelességét.

A digitális aláírási séma definíciója a következő:

Definíció. Jelölje P a lehetséges üzenetek halmazát és A az aláírások halmazát. Az AS=*(K, Sign, Ver)* digitális aláírási séma három algoritmusból áll:

- A *K kulcsgeneráló algoritmus* egy polinom idejű algoritmus, melynek inputja a k biztonsági paraméter, outputja egy véletlen (*SK*,*PK*) kulcspár, ahol *SK* jelöli a titkos kulcsot, *PK* pedig a nyilvános kulcsot. A generált kulcsok mérete függ a k biztonsági paramétertől, aminek értékét a konkrét rendszerrel szembeni sikeres támadások határozzák meg.
- Az *Sign aláíró algoritmus* az $m \in P$ szöveg és az *SK* titkos kulcs felhasználásával egy $s \in A$ aláírást generál. Vannak olyan rendszerek is, amelyek az aláírás generálásánál további véletlen paramétereket is alkalmaznak, így a kapott aláírás randomizált.
- A *Ver ellenőrző algoritmus* az $(m,s) \in P \times A$ elempárhoz, a *PK* nyilvános kulcs alkalmazásával *igaz* vagy *hamis* értéket rendel aszerint, hogy az s érvényes aláírása–e az m üzenetnek.

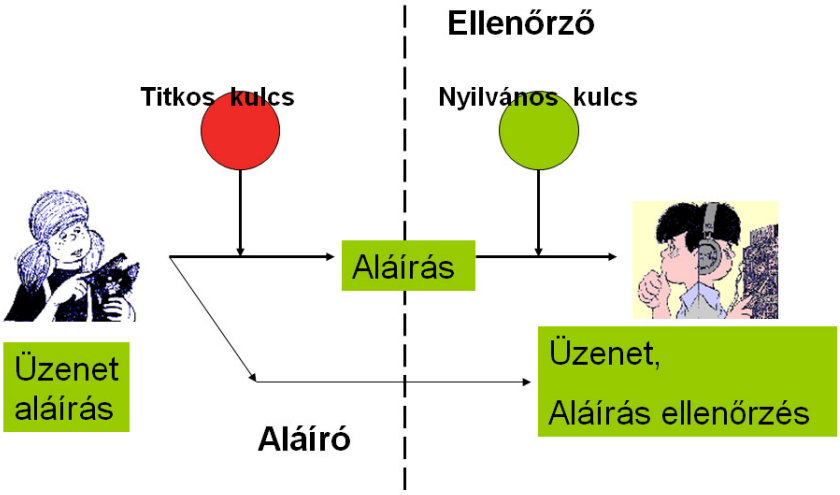
Az előbbi definícióban szereplő *Sign* algoritmus az *SK* titkos kulcs ismeretében egy adott m üzenethez polinomiális időn belül számítja ki a megfelelő s aláírást. Viszont, ha a titkos kulcs nem ismert, akkor fontos az, hogy egy adott m üzenethez ne lehessen érvényes s aláírást létrehozni polinomiális időn belül. Ez a kitétel azt is jelenti, hogy a *PK* nyilvános kulcsból ne lehessen polinomiális időn belül a hozzátartozó *SK* titkos kulcsot kiszámítani. Természetesen az is feltétel, hogy tetszőleges szabályosan generált aláírásra lefuttatott ellenőrző algoritmus visszatérési értékének *igaz*nak kell lennie.

Amennyiben a *Sign* algoritmus randomizált, akkor ugyanazon $m \in P$ üzenetnek, ugyanazon *SK* titkos kulcs esetén, különböző véletlen paraméter mellett különböző $s \in A$ érvényes aláírása lesz. A különböző érvényes aláírások mindegyikének ellenőrzése ugyanazon PK nyilvános kulccsal és ugyanazon m üzenet alapján történik, ahol az ellenőrző algoritmus determinisztikus.

Az ellenőrző algoritmus az $(m,s) \in P \times A$ elempár vizsgálatához a megfelelő *PK* nyilvános kulcsot használja fel. Csak abban az esetben lesz érvényes az s aláírás az adott m üzenettől függően, ha az aláíró entitás nyilvános kulcsával történik az ellenőrzés. Az aláíró

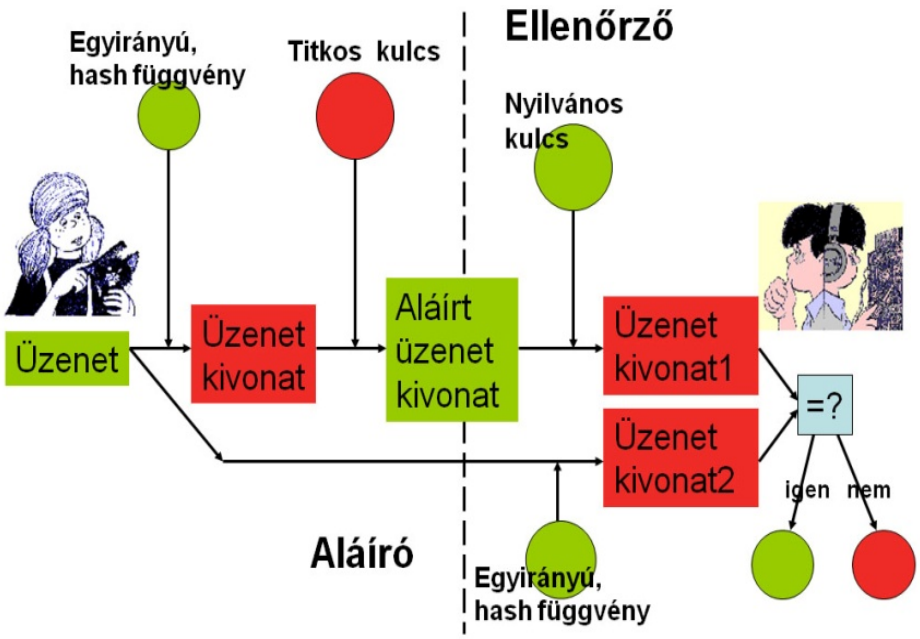
entitás nyilvános kulcsát egy hitelesítés-szolgáltató által kibocsátott aláíró tanúsítvány tartalmazza. Mint ahogy ezt a 0 fejezetben ismertetjük, a megbízható hitelesítés-szolgáltató szigorú regisztrációs folyamat után egy adott személynek vagy cégnek kiállít egy aláíró tanúsítványt, mely a tulajdonos adatain kívül a nyilvános kulcsát is tartalmazza. Így a nyilvános tanúsítvány alapján biztosak lehetünk abban, hogy az aláírás ellenőrzése során alkalmazott nyilvános kulcs a megfelelő kulcs.

A 9.5 ábra a digitális aláírás folyamatát mutatja be. Az ábrán is jól látható, hogy Kriszta, az aláíró entitás üzenetét titkos kulccsal írja alá, és az így kapott aláírást és az üzenetet is elküldi Aladárnak. Aladár, az aláírást ellenőrző személy az aláírásra Kriszta nyilvános kulcsát alkalmazza, majd az elküldött üzenetet is figyelembe véve ellenőrzést végez.



9.5 ábra Digitális aláírás

A 9.6 ábra a digitális aláírás gyakorlatban való megvalósítását mutatja. Az alapvető különbség az egyirányú hash függvény használatában van. Itt az üzenetre egy egyirányú hash függvényt alkalmazunk, melynek eredménye egy fix hosszúságú bitsorozat lesz. Majd erre az üzenetkivonatra alkalmazza Kriszta a titkos kulcsát. Az aláírt üzenetkivonat és az üzenet kerül elküldésre. Aladár az aláírt üzenetkivonatra alkalmazza Kriszta nyilvános kulcsát, illetve kiszámítja az üzenet egyirányú hash értékét. Amennyiben az így kapott üzenetkivonatok megegyeznek, akkor az aláírás érvényes. Az egyirányú hash függvény alkalmazásával a folyamat jelentősen felgyorsul, hiszen tetszőleges hosszúságú üzenet esetén is ugyanolyan hosszú kivonat (kb. 160 bit) kerül aláírásra, ami egy hatékony, de lassú folyamat. Az egyirányú hash függvény nemcsak felgyorsítja a folyamatot, de a rendszer biztonságát is jelentősen növeli.



9.6 ábra Digitális aláírás labormodellje

9.2.3 Digitális aláírás jellemzői

Amennyiben az aláírás érvényes, akkor a következő tulajdonságok teljesülnek:

1. *Hitelesítés (authentication)* (4):
   Hitelesítés során a fogadó félnek biztosítékot adunk arra, hogy az adott információ valamely támadó által nem lett megváltoztatva, vagy kicserélve. Ha a következő két tulajdonság egyszerre teljesül, akkor az adott üzenet hiteles.
   - *Üzenet adatintegritása*:
     A dokumentum aláírás után nem változtatható meg. Amennyiben az elküldött üzenet tartalmát egy támadó megváltoztatja, akkor annak lenyomata különbözni fog az eredetiétől, így ellenőrzésnél ez kiderül.
   - *Üzenet eredetének igazolása:*
     Az aláíró kiléte beazonosítható a kulcspárja segítségével, hiszen csak akkor lett szabályosan legenerálva az aláírás, ha az ellenőrzésnél felhasznált nyilvános kulcs titkos párjával lett aláírva. Azt hogy az adott nyilvános kulcs az aláíró személyéhez kötődik, valamely hitelesítés szolgáltató által hitelesített aláíró tanúsítvány igazolja. A digitális aláírás egy eszköz arra, hogy egy adott üzenet eredetét bárki ellenőrizhesse, azaz az ellenőrző entitás akár egy harmadik félnek is bizonyíthatja az üzenet küldőjének személyazonosságát (A MAC üzenethitelesítéssel ellentétben).
2. *Letagadhatatlan (non-repudiation):*
   Mivel az aláírás érvényességét bárki ellenőrizheti (az adott üzenet és a nyilvános kulcs felhasználásával), így bárki meggyőződhet arról, hogy az adott dokumentumot vagy üzenetet egy adott entitás aláírta. Következésképpen a digitálisan aláírt dokumentum nem letagadható. Ha a dokumentumon időpecsét is szerepel, akkor az aláírás időpontjáról is van információnk.

3. *Hamisíthatatlan*
   Adott dokumentumhoz egy *A* entitás helyett egy *B* entitás nem képes olyan digitális aláírást készíteni, amelyet elfogadnának az *A* entitás aláírásaként. Hasonlóképpen a dokumentum sem hamisítható, hiszen két különböző dokumentum aláírásának különböznie kell. Az aszimmetrikus kriptográfiai rendszer jellemzői alapján a nyilvános információkból a titkos kulcs kiszámítása *nehéz*. Amennyiben csak a tulajdonos ismeri titkos kulcsát, akkor az aláírás nem hamisítható.
4. *Az aláírás nem átruházható*
   Az aláírt üzenet kivonat egy adott üzenethez tartozik, más üzenethez nem csatolható, hiszen lenyomatuk különbözik.

9.2.4 Támadások

Áttérünk a digitális aláírási rendszer biztonsági fogalmaira. Elsőként a rendszerrel szembeni támadó lehetséges céljait adjuk meg, majd a támadásokat kategorizáljuk. Természetesen minden esetben a támadó alapvető célja az aláírás meghamisítása. Mégis a célokat négy fő kategóriába sorolhatjuk:

- *Teljes feltörés*:
  Ez a legkomolyabb támadási cél. A támadó képes meghatározni az aláíró fél titkos kulcsát. Következésképpen a támadó tetszőleges üzenetet tud aláírni a titkos kulcs tulajdonosa nevében.
- *Univerzális hamisítás*:
  A támadó célja egy olyan hatékony algoritmus konstruálása, mely *tetszőleges üzenet*ek aláírására képes az aláíró nevében a titkos aláíró kulcs ismerete nélkül.
- *Szelektív hamisítás*:
  A támadó célja egy olyan hatékony algoritmus konstruálása, mely az általa *kiválasztott üzenet*hez képes az aláíró fél nevében érvényes aláírást generálni.
- *Egzisztenciális hamisítás*:
  A támadó célja egy olyan algoritmus generálása, mely segítségével *egy üzenet*hez tud érvényes aláírást készíteni az aláíró fél nevében. Az alapvető különbség a szelektív és az egzisztenciális hamisítás között, hogy míg a szelektív esetén a támadó egy általa választott üzenet aláírására képes, addig itt a támadó valamely (nem feltétlen a számára legelőnyösebb) üzenethez tud aláírást konstruálni.

Sokszor az egzisztenciális hamisítás nem minősül veszélyes támadásnak, hiszen a hamisított aláírt szöveg nem értelmes, a támadó számára nem felhasználható üzenet lesz. Ennek ellenére mégis egy üzenethez érvényes aláírás tartozik, tehát az üzenet eredetét képes igazolni. Így az egzisztenciális hamisítás eredményeképpen létrejött aláírt bitsorozat magában igazolja az üzenet eredetét. Következésképpen az egzisztenciálisan hamisítható digitális aláírási sémák nem alkalmasak randomizált értékek – például kulcsok – aláírására. Tehát annak ellenére, hogy sok szituációban ez a támadástípus nem veszélyezteti a rendszer biztonságát, mégis ezzel a támadással a digitális aláírási sémák alapvető jellemzője, a hamisíthatatlanság nem teljesül, hiszen bárki generálhat érvényes aláírást az aláíró fél nevében.

Hasonlóan a kriptorendszerekhez, a digitális aláírási rendszerekkel szemben is többféle támadást különböztetünk meg. Passzív támadások esetén a támadó csak lehallgatja a

csatornát, vagy esetleg valamilyen úton-módon (pl. aláíró mechanizmus, statisztikai elemzések futtatásával) hozzáfér további információkhoz és az így megszerzett adatok felhasználásával próbál aláírást hamisítani. Attól függően, hogy a támadónak milyen információk állhatnak rendelkezésére a támadásokat a következőképpen kategorizáljuk:

- *Csak a nyilvános kulcs ismert* (key-only attack):
  A támadó rendelkezésére mindössze a nyilvános elérhető információk állnak, azaz az aláíró fél nyilvános kulcsa.
- *Ismert üzenet alapú támadás* (known-message attack - KMA):
  A támadó rendelkezésére áll a nyilvános kulcson kívül egy üzenetekből és a hozzájuk tartozó aláírásokból álló lista. A listán szereplő üzeneteket nem a támadó választotta.
- *Választott üzenet alapú támadás* (chosen-message attack - CMA):
  A támadó rendelkezésére áll egy az aláíró fél nevében aláíró mechanizmus, mely képes a támadó által választott üzenetek aláírására úgy, hogy a támadó nem fér hozzá a titkos aláíró kulcshoz. A támadó az aláíró mechanizmustól kéri az általa előre kiválasztott üzenetek aláírását. Az így generált üzenet-aláírás elempárokból álló lista és a nyilvános információk alapján a támadó próbál egy a listán nem lévő üzenet aláírását elkészíteni.
- *Adaptívan választott üzenet alapú támadás* (adaptive chosen-message attack):
  Ez a digitális aláírási séma elleni legerősebb támadási forma. Ebben az esetben a támadó hozzáférhet egy aláíró mechanizmushoz, amely az általa választott üzentekhez legenerálja az érvényes aláírásokat. A különbség a nem adaptívan és az adaptívan választott üzenet alapú támadás között, hogy adaptív esetben a támadó a már kapott aláírások alapján választja ki a következő üzenetet, melynek aláírását kéri. A támadó célja az így elkészült lista és a nyilvános kulcs ismeretében egy a listán nem szereplő üzenet érvényes aláírását kiszámítani az aláíró fél nevében.

Akkor fogunk egy digitális aláírási sémát biztonságosnak mondani, ha a leggyengébb támadási cél esetén a legerősebb támadással szemben is védett, azaz, ha az egzisztenciális hamisításra irányuló adaptívan választott üzenet alapú támadással szemben védettséget mutat. Egy digitális aláírási séma biztonsági elemzésének megadása azt jelenti, hogy azt vizsgáljuk, hogy egy adott típusú támadási cél elérhető–e egy kiválasztott támadási formával szemben.

Általában azt bebizonyítani nem tudjuk, hogy nem lehet sikeres támadást generálni egy adott digitális aláírási sémával szemben. Ehelyett a séma egy adott támadással szembeni védettségét egy *nehéz*nek bizonyuló problémára vezetjük vissza, mint például a 8. fejezetben ismertetett prímfaktorizáció, illetve diszkrét logaritmus problémájára.

Randomizált aláírások esetén, mivel az aláíró algoritmus nem determinisztikus ugyanannak az üzenetnek több érvényes aláírása is lehet. Az ilyen aláírásoknál a fent ismertetett biztonsági fogalmak könnyen félreérthetőek. Adaptívan és nem adaptívan választott üzenet alapú támadás esetén a szigorúbb megközelítés a cél, azaz az elvárás az, hogy a támadó egy a listán nem szereplő üzenet aláírását hamisítsa. Az engedékenyebb megközelítés az, hogy akár a listán szereplő üzenet aláírását is hamisíthatja, csak egy, a listán nem szereplő érvényes aláírást kell megadnia. Amennyiben a támadó képes egy a listán

szereplő üzenethez egy új aláírást hamisítani, akkor azt mondjuk, hogy a digitális aláírás *alakítható* (malleable). A megfelelő biztonsági cél *nem-alakíthatóság* (non-malleability). Adaptív esetben még az is megengedhető, hogy a támadó ugyanazt az üzenetet többször kiválasztja és kéri a hozzátartozó érvényes aláírás legenerálását.

Digitális aláírási séma kialakítható aszimmetrikus titkosító eljárásokból abban az esetben, ha egy tetszőleges m üzenetre vonatkozóan az $Enc_{PK}(Dec_{SK}(m))=m$ teljesül. Nyilvánvalóan titkosító rendszereknél az alapvető elvárás az, hogy $Dec_{SK}(Enc_{PK}(m))=m$, amennyiben bármely üzenet esetén a titkosító és visszafejtő algoritmus tetszőleges sorrendben vett egymás utáni végrehajtásával az eredeti üzenetet kapjuk vissza, akkor alkalmazható digitális aláírásra is. Ilyen rendszer például az RSA titkosító eljárás, hiszen $Enc_{PK}(Dec_{SK}(m))=(m^d)^e=m \pmod n$. Viszont az ElGamal rendszer esetén ez a tulajdonság nem teljesül, hiszen a titkosító és a visszafejtő algoritmus nem ugyanazon a halmazon van értelmezve.

Mint azt az 8.7 fejezetben kifejtettük az aszimmetrikus titkosításhoz szükséges kulcsok nagy bináris számok, ezért azokat nem lehet úgy memorizálni, mint a mindennapos PIN kódokat. A digitális aláíráshoz is ilyen kulcsokat használunk. Az ellenőrzéskor használt nyilvános kulcs elhelyezhető egy hitelesítő szervezet adatbázisában, tehát ugyanúgy, mint az aszimmetrikus titkosító rendszereknél, itt is szükséges a PKI (nyilvános kulcsú infrastruktúra) használata. A PKI-ről részletesen a 0 fejezetben olvashatunk. Az aláíró, titkos kulcsot azonban a tulajdonosának kell olyan könnyen kezelhető adathordozón tárolni és őrizni, amelyik egyszerűen felhasználható. Ezeknek a feltételeknek a smart kártyák vagy penndrive-ok felelnek meg. Normál biztonsági szintű digitális aláíráshoz használhatunk szoftkulcsot (softkey) is, amelyet a számítógép memóriájában, PIN kóddal védett állományban tárolunk.

A következő alfejezetek konkrét aláírási rendszereket részleteznek. Az itt megadott aláírási sémák mind aszimmetrikus kriptográfián alapulnak. Először a faktorizáció problémáján alapuló RSA aláírási sémával foglalkozunk, majd a diszkrét logaritmus problémáján alapuló ElGamal és DSA sémát részletezzük.

9.2.5 RSA aláírási séma

Az algebrai struktúrája miatt az RSA kriptorendszer könnyen alakítható digitális aláírási sémává **(5)**. Az RSA aláírási sémát a definíció szerinti három algoritmussal adjuk meg, a kulcsgeneráló, az aláíró és az ellenőrző algoritmussal.

A *kulcsgenerálás* teljesen megegyezik az RSA titkosításnál tanultakkal. A következő négy lépést kell végrehajtani:

1. véletlenül választunk két nagy prímet, jelöljük p-vel és q-val, majd kiszámítjuk az $n = p \cdot q$ értéket, amit RSA modulusnak hívunk.

2. kiszámítjuk $\varphi(n) = (p-1)(q-1)$, ahol φ, az Euler-féle φ függvény

3. választunk véletlenül egy e számot, mely $1 < e < \varphi(n)$ és $(e, \varphi(n))=1$

4. meghatározzuk *d*-t, ahol $1 < d < \varphi(n)$ és $e \cdot d \equiv 1 \pmod{\varphi(n)}$.

A kulcsgenerálás során kapott *(n,e)* értékek nyilvánosak, az *e*-t nyilvános exponensnek hívjuk, a *d, p, q, φ(n)* számok titkosak, ahol *d* az aláíró titkos kulcs. Azaz *SK=d* és *PK=(n,e)* és *p, q, φ(n)* titkos paraméterek.

Az *aláíró algoritmus* egy tetszőleges $m \in P$ üzenethez a *d* titkos aláíró kulcs segítségével egy $s \in A$ aláírást számít ki, azaz $Sign_d(m)=s$, ahol $s \equiv m^d \pmod{n}$.

Az *ellenőrző algoritmus* az $(m,s) \in P \times A$ elempárhoz *igaz* vagy *hamis* értéket rendel aszerint, hogy $m \equiv s^e \pmod{n}$ teljesül –e, ahol *e* a generált nyilvános kulcs.

Az alap RSA aláírási sémát több szempontból sem tekintjük biztonságosnak. Egy lehetséges támadás az *egzisztenciális hamisítás*, ha a támadó véletlenül választ egy *s* értéket és kiszámítja az $m \equiv s^e \pmod{n}$ értéket, ahol az *e* és *n* nyilvánosak. Így a támadó generált egy érvényes *(m,s)* aláírt üzenetet a titkos *d* aláíró kulcs ismerete nélkül. Természetesen az így generált *m* üzenet nem egy a támadó által kiválasztott üzenet, hanem egy olyan bitsorozat, melynek értelme bármi – akár értelmetlen is – lehet, a támadó által nem szabályozható.

Az alap RSA egy másik gyengesége az, hogy *alakítható (malleable),* azaz a támadó a rendelkezésére álló (m_1,s_1) és (m_2,s_2) aláírt üzenetek alapján egy új aláírt üzenetet tud kialakítani, kiszámítani. Ugyanis az $m_3 \equiv m_1 m_2 \pmod{n}$ üzenet szabályos aláírása az $s_3 \equiv s_1 s_2 \pmod{n}$, hiszen $s_3 \equiv m_1^d m_2^d \equiv (m_1 m_2)^d \pmod{n}$.

A fent részletezett támadások kriptográfiai hash függvény alkalmazásával kivédhetőek.

9.2.5.1 RSA-FDH

1996-ban Bellare és Rogaway [2] javasolt egy lehetséges digitális aláírási sémát, mely során az üzenet lenyomata kerül aláírásra. Ez a „hash-and-sign” paradigma, RSA esetén az RSA-FDH (Full-Domain Hash) digitális aláírási séma. A kulcsgeneráló algoritmus megegyezik az alap RSA rendszernél megadottakkal. Az *m* üzenet aláírásához választunk egy nyilvános, $H: \{0,1\}^* \rightarrow \mathbb{Z}_n^*$ egyirányú ütközésmentes hash függvényt és

- elkészítjük a *H(m)* lenyomatot,
- kiszámítjuk az $s \equiv H(m)^d \pmod{n}$ aláírást

Az ellenőrző algoritmus az *(m,s)* értékek alapján kiszámítja a *H(m)* lenyomatot, valamint az $s^e \pmod{n}$ értéket. Amennyiben a $H(m) \equiv s^e \pmod{n}$ kongruencia áll, akkor az aláírás érvényes, ha nem teljesül, akkor az aláírás nem érvényes.

Bizonyítható, hogy az RSA-FDH séma az egzisztenciális hamisításra irányuló adaptívan választott üzenetekre épülő támadással szemben biztonságos a véletlen orákulum modellben[22] (random oracle model). A bizonyítással, illetve a modell ismertetésével itt nem foglalkozunk, további ismereteket a **(3)** könyvben találhatunk.

---

[22]Véletlen orákulum alatt olyan véletlen függvényt értünk, mely nyilvánosan hozzáférhető, ugyanazon inputra ugyanazon outputot ad, és különböző inputokra viszont az output térben egyenletesen választott outputot ad.

Azon kívül, hogy az alap RSA digitális aláírási sémánál megadott támadásokkal szemben az RSA-FDH védettséget biztosít, jelentős gyakorlati előnye is van. Ugyanis a tetszőleges méretű üzenet helyett az üzenet fix hosszúságú (kb. 160 bit) hash értéke kerül aláírásra. Következésképpen a lassú aszimmetrikus számításokat mindössze csak néhány bit hosszú sztringre kell alkalmazni, ezzel jelentősen felgyorsítva az aláírás generálásának és az ellenőrzésének folyamatát.

Természetesen az is fontos, hogy a hash függvény használata nehogy csökkentse az RSA aláírási séma biztonságát. Következésképpen megfelelő tulajdonságú hash függvényeket alkalmazunk.

Tegyük fel, hogy a támadó rendelkezésére áll egy *(m,s)* aláírt üzenet, ahol $s \equiv H(m)^d \pmod n$. Például a támadó megtudott egy korábban, ugyanazzal a titkos kulccsal aláírt üzenetet. Ekkor kiszámítja a *H(m)* lenyomatot és keres egy $m' \neq m$ üzenetet, melyre $H(m)=H(m')$. Ha talál ilyen *m'* üzenetet, akkor az *(m',s)* egy érvényes aláírt üzenet lesz. Így egy egzisztenciális hamisításra irányuló ismert üzenet alapú támadást végeztünk. Azért, hogy ezt a hamisítást megelőzzük elvárjuk, hogy a hash függvény gyengén ütközésmentes legyen.

Másik lehetséges támadás, ha a támadó talál két üzenetet, melynek hash értéke megegyezik, azaz *m* és *m'*, ahol $H(m)=H(m')$. A támadó megkéri az aláíró személyt, hogy az *m* üzenetet írja alá titkos kulcsával. Jelölje *(m,s)* az így aláírt üzenetet. Ekkor a támadó rendelkezésére áll egy érvényes aláírt üzenet az *(m',s)*. Ez egy egzisztenciális hamisításra irányuló választott üzenet alapú támadás. Ez a támadás is megelőzhető, amennyiben az alkalmazott hash függvény (erősen) ütközésmentes.

A harmadik lehetséges támadás, ha a támadó csak a nyilvános kulcsot ismeri. Ekkor választ egy tetszőleges véletlen *s* értéket, alkalmazza rá a nyilvános kulcsot: $s^e \equiv y \pmod n$. Amennyiben a támadó az *y* lenyomathoz meg tud adni egy *m* üzenetet, hogy $y=H(m)$, akkor az *(m,s)* egy érvényes aláírt üzenet lesz. Ez egy egzisztenciális hamisítás, ahol csak a nyilvános kulcs ismert a támadó számára. Ez a támadás megelőzhető, ha a hash függvény egyirányú.

Az előbbiek alapján is láthatjuk, hogy a digitális aláírási sémáknál alkalmazott hash függvényeknek ütközésmentesnek és egyirányúnak kell lenniük.

9.2.6 ElGamal aláírási séma

Taher ElGamal 1985-ben PhD dolgozatában **(32)** a diszkrét logaritmus alkalmazási lehetőségeit tanulmányozta kriptográfiai rendszerekben. Az ElGamal aláírási séma egy egész digitális aláírási séma család alapjául szolgál.

A következőkben az ElGamal aláírási séma kulcsgeneráló, aláíró és ellenőrző algoritmusát ismertetjük.

A *kulcsgeneráló algoritmus* lépései:

1. Választunk egy nagy *p* prímet és egy $g \in \mathbb{Z}_p^*$ primitív gyököt
2. Véletlenül választunk egy $a \in \mathbb{Z}_{p-1}$ értéket és kiszámítjuk $A \equiv g^a \pmod p$

A kulcsgenerálás során kapott $a \in \mathbb{Z}_{p-1}$ érték a *titkos kulcs*, azaz $SK=a$, az *A* érték a *nyilvános kulcs,* tehát $PK=A$, a *p* prím és a *g* primitív gyök pedig nyilvános paraméterek.

Az aláírandó üzenetek halmaza: $\mathbb{Z}_p^*$, és az aláírások halmaza: $\mathbb{Z}_p^* \times \mathbb{Z}_{p-1}$.

Az *aláíró algoritmus* lépései:

1. Véletlenül választunk egy $k \in \mathbb{Z}_{p-1}^*$ értéket, amit titokban tartunk.
2. Kiszámítjuk: $r \equiv g^k \pmod p$
3. Meghatározzuk: $t \equiv (m-ar)k^{-1} \pmod{p-1}$, ahol $m \in \mathbb{Z}_p^*$ az üzenet, melyet aláírunk.
4. Az m üzenet aláírása: $s=(r,t)$

Az *ellenőrző algoritmus* lépése:

1. Ellenőrizzük, hogy az $A^r r^t \equiv g^m \pmod p$ kongruencia és $0<r<p$ teljesül–e. Amennyiben teljesülnek, akkor az aláírás érvényes, ellenkező esetben nem.

Vegyük észre, hogy az alap ElGamal aláírási séma az aláírandó üzeneteit a $\mathbb{Z}_p^*$ halmazból veszi, az aláírások halmaza pedig $\mathbb{Z}_p^* \times \mathbb{Z}_{p-1}$. Ennek következtében az aláírás mérete duplája az aláírandó üzenet méretének. Ha például tekintünk egy *1024* bit méretű p prímet, akkor az *1024* bites üzenethez tartozó aláírás mérete *2048* bit, ami elég nagy.

Ennél a sémánál is alkalmazunk hash függvényeket. A gyakorlatban előforduló változata csak néhány lépésben különbözik. A kulcsgenerálás algoritmusa változatlan. Az aláíró algoritmus során meghatározzuk az m üzenet lenyomatát: $h=H(m)$, ahol $H:\{0,1\}^* \rightarrow \mathbb{Z}_{p-1}$.hash függvény. Majd az így kapott h lenyomat aláírása történik meg, azaz $t \equiv (h-ar)k^{-1} \pmod{p-1}$. Az aláírás továbbra is az $s=(r,t)$ elempár lesz. Az ellenőrző algoritmus során az ellenőrző fél kiszámítja a kapott m üzenet h lenyomatát, majd ellenőrzi, hogy a $A^r r^t \equiv g^h \pmod p$ kongruencia és $0<r<p$ teljesül –e.

Az ElGamal digitális aláírási séma nemdeterminisztikus (a 8.7 fejezetben már említettük, hogy az ElGamal titkosítás is nemdeterminisztikus), azaz ugyanannak az üzenetnek több érvényes aláírása is lehetséges, és az ellenőrző algoritmus valamennyi szabályosan generált aláírást érvényesnek fog tekinteni.

Amennyiben az aláírás szabályosan generált, akkor az ElGamal digitális aláírási séma ellenőrző algoritmusa elfogadja azt. Mivel az (r,t) aláírás szabályosan generált az m üzenethez, így $0<r<p$ teljesül és mivel $A \equiv g^a \pmod p$ és $r \equiv g^k \pmod p$, így

$$A^r r^t \equiv g^{ar} g^{kt} \equiv g^{kt+ar} \pmod p.$$

Mivel g primitív gyök a $g^{kt+ar} \equiv g^m \pmod p$ kongruencia teljesül, ha az exponensek kongruensek modulo $p-1$, azaz $m \equiv kt+ar \pmod{p-1}$, ami akkor és csak akkor teljesül, ha

$t \equiv (m-ar)k^{-1} \pmod{p-1}$.

Az aláíró személy két titkos érték segítségével írja alá az üzenetet, az egyik maga az a titkos kulcs, a másik pedig a k titkos paraméter. Az ellenőrzés viszont a A nyilvános kulcs és a g, p nyilvános paraméterek felhasználásával történik.

Vizsgáljuk meg az ElGamal aláírási séma biztonsági kérdéseit. Elsőként tekintsük azt az esetet, mikor a támadó szelektív hamisítást kezdeményez, azaz egy érvényes aláírást akar hamisítani egy adott m üzenethez az a titkos kulcs ismerete nélkül. A támadó választ egy r értéket és keresi a megfelelő t-t, azaz $r^t \equiv g^m A^{-r} \pmod p$ ismeretében keresi t-t. Ehhez diszkrét

logarimust kell számolnia, mégpedig a $g^m A^{-r}$ érték r alapú *diszkrét logaritmus*t *mod p,* ami *nehéz* probléma.

Egy másik lehetséges támadás, ha véletlen *(r,t)* elempárhoz megkeressük az *m* üzenetet. Ebben az esetben is diszkrét logaritmust kell számolni, azaz az $A^r r^t \equiv g^m \pmod p$ ismeretében az $A^r r^t$ érték g alapú diszkrét logaritmusát kell megadni a p modulusra nézve. Ez a típusú támadás már egzisztenciális.

Az alap ElGamal aláírási séma esetén sikeres egzisztenciális hamisítást tudunk generálni, ha csak a nyilvános kulcsot ismeri a támadó, vagy ha választott üzeneten alapuló támadást hajt végre. Ezeket a támadásokat itt nem ismertetjük, a [35] könyvben részletesen megtalálhatóak. Mindezen egzisztenciális támadásokkal szemben hash függvények alkalmazásával védekezhetünk.

Sikeres támadásokat lehet végrehajtani akkor is, ha a k véletlen paramétert nem megfelelően alkalmazzuk. Komoly probléma, ha nem tartjuk titokban, azaz a támadó a $t \equiv (m-ar)k^{-1} \pmod{p-1}$ kongruenciában ismeri t,m,r,p és k értékeket, így egyedül a az ismeretlen. A a meghatározása könnyű: $a \equiv (m-kt)r^{-1} \pmod{p-1}$. Tehát ekkor a rendszert teljesen feltörtük, hiszen meghatároztuk a titkos kulcsot. Hasonlóan támadást lehet intézni a rendszerrel szemben a titkos kulcs meghatározására akkor is, ha kétszer ugyanazt a k értéket használjuk különböző üzenetek aláírására.

9.2.7 DSA

Az ElGamal digitális aláírási séma egyik hátránya, hogy az aláírás mértéke kétszerese az üzenetnek (pl. 2048 bit). A Digital Signature Algorithm (DSA) esetén az aláírás mérete jelentősen lecsökken. A DSA az ElGamal rendszer egy módosított változata. A DSA 1994-ben lett szabvány.

A *kulcsgeneráló algoritmus* a következő lépésekből áll:

1. Véletlenül választunk egy q prímet,
2. Megadunk egy p prímet, melyre $q|p-1$.
3. Választunk egy $g \in \mathbb{Z}_p^*$ elemet, melynek rendje q.
4. Választunk egy véletlen $a \in \mathbb{Z}_q$ értéket.
5. Kiszámítjuk az $A \equiv g^a \pmod p$ számot.

Az $a \in \mathbb{Z}_q$ érték lesz a titkos kulcs ($SK=a$), $A \in \mathbb{Z}_p^*$ a nyilvános kulcs ($PK=A$), valamint a p,q prímek és $g \in \mathbb{Z}_p^*$ nyilvános paraméterek.

Az *aláíró algoritmus* a következő lépésekből áll:

1. Választunk egy $H:\{0,1\}^* \rightarrow \mathbb{Z}_q$ hash függvényt.
2. Veszünk egy véletlen $k \in \mathbb{Z}_q^*$ értéket, amit titokban tartunk.
3. Kiszámítjuk: $r \equiv (g^k \bmod p) \bmod q$
4. Meghatározzuk: $t \equiv (H(m)+ar)k^{-1} \pmod q$, ahol $m \in \mathbb{Z}_p^*$ az üzenet, melynek lenyomatát aláírjuk.

5. Az *m* üzenet aláírása: *s=(r,t)*

Az *ellenőrző algoritmus* lépései:

1. Ellenőrizzük, hogy $0<r<q$ és $0<t<q$ teljesül –e
2. Jelölje $e \equiv H(m)t^{-1} \pmod q$ és
3. $f \equiv rt^{-1} \pmod q$
4. Amennyiben $(r \equiv g^e A^f \bmod p) \bmod q$ teljesülnek, akkor az aláírás érvényes, ellenkező esetben nem.

1991-ben a szabvány első verziójában a *q* hossza *160* bit, *p hossza 512* és *1024* bit közötti, valamint *64*-gyel osztható. Későbbiekben a javasolt értékpárok: (1024,160), (2048,224), (2048,256) és (3072,256).

Az ElGamal aláírási sémához képest az aláírás mérete kisebb, a *q* méretének kétszerese, azaz például, ha *p* mérete *1024* bit nagyságú, akkor az ElGamal aláírás *2048* bit nagyságú, míg a DSA aláírás csak *320* bit méretű.

9.2.8 A digitális aláírás fajtái és alkalmazása

Mindennapi életünkben az elektronikus szolgáltatások egyre nagyobb teret hódítanak, mely természetes módon a biztonságos elektronikus aláírás egyre gyakoribb használatát is eredményezi. Több területen is lehet hivatalos okiratainkat elektronikusan benyújtani, tárolni és digitális aláírással hitelessé tenni, ilyen területek napjainkban például az adóbevallás, cégeljárás, ügyvédi ellenjegyzés, közigazgatási hatósági eljárás, elektronikus számlázás és vizsgajelentések.

A digitális aláírási technika egy alkalmazásának tekinthetjük az időbélyeg-szolgáltatást is, mely során az adott elektronikus dokumentum lenyomatát egy Időbélyegző Szolgáltató digitális aláírásával hitelesít. Az időbélyegzésről részletesebben később lesz szó.

Digitális aláírást alkalmazhatunk minden olyan szituációban, amikor valamely információ hitelességét, érvényességét kívánjuk biztosítani, akár valamely másik fél által. Úgynevezett vak aláírási technikához folyamodhatunk, a digitális aláírás egy módosított változatához, ha a másik féllel nem akarjuk tudatni, hogy mit ír alá. Tipikus alkalmazási területe az elektronikus szavazások és az elektronikus pénz. Elektronikus szavazás esetén úgy kell érvényesíteni egy szavazatot, hogy a hitelesítő szervezet ne tudja meg, hogy az adott szavazó kire voksolt. A vak aláírási technikát a 9.2.8.2 alfejezetben részletezzük.

Az online nyereményjátékoknál, például a Puttónál, időbélyeg-szolgáltatással hitelesítik egy adott időpontban beérkezett tippeket. A játék lényege, hogy 5 percenként 8+1 nyerőszámot sorsolnak ki, így a játék adatai egy minősített szolgáltató által kibocsátott időbélyegzővel hitelesített állományba kerülnek, tehát egy külső megbízható szervezet igazolja a beérkezett tippek időpontját.

A digitális aláírások egy gyakori alkalmazása a programkódok aláírása. A kód aláírása egyértelműen igazolja az adott szoftver készítőjét, illetve garantálja, hogy a programkód a kibocsátás óta nem módosult. Ha a felhasználó bármely, a szoftver által generált rosszindulatú

működést azonosít be, akkor igazolható, hogy az adott szoftver kódját senki más nem változtatta meg, egyértelműen a szerző a felelős.

Az azonosítás terén jelentős mérföldkő a digitális aláírás alkalmazása. A digitális aláírás segítségével egyértelműen meg tudjuk határozni a másik fél identitását. Ez a technika sokkal nagyobb biztonságot nyújt, mint a gyakran használt felhasználói név-jelszó alapú azonosítás. Ilyen technika például nagyon jól használható védett virtuális terekbe való belépésre. Konkrét azonosító rendszereket később ismertetünk.

9.2.8.1 Időbélyegzés

Általában a hagyományos, sajátkezű aláírással ellátott dokumentumokon az aktuális dátum feltüntetésével igazoljuk a dokumentum készítésének időpontját. Elektronikus dokumentumok esetén az időbélyeg-szolgáltatás hitelesíti a dokumentum létrejöttének idejét. Ennek megfelelően az elektronikus aláírásokhoz szorosan kapcsolódik az időbélyeg-szolgáltatás.

Az *időbélyegzés* egy olyan elektronikus igazolás, mely bizonyítja, hogy egy elektronikus dokumentum egy adott időpontban már létezett, és annak tartalma az időbélyegzés óta nem változott meg. Az időbélyeg általában az elektronikus aláírás létrehozásának időpontját igazolja, de használatos akkor is, ha valamely dokumentum vagy állomány adott időben való létezésének későbbi bizonyítása a cél.

Az időbélyeget *Időbélyegző Szolgáltató* állítja ki. Az időbélyeg tartalmazza az adott dokumentum kivonatát (hash értékét), valamint a bélyegzés időpontját, melyet az Időbélyegző Szolgáltató elektronikus aláírásával hitelesít. Minősített időbélyegzés csak Internet hozzáféréssel valósítható meg, általában az alkalmazás automatikusan kapcsolatba lép a megadott időbélyegzés szolgáltatóval.

Az időbélyegzés folyamata:

1. Véglegesítjük az adott dokumentumot.
2. Megfelelő program segítségével elkészül az adott dokumentum lenyomata.
3. A lenyomatot a program elküldi az Időbélyegző Szolgáltatónak.
4. Az Időbélyegző Szolgáltató elkészíti az időbélyeget, aláírásával hitelesíti azt, és visszaküldi a programnak.
5. A program csatolja a dokumentumhoz az időbélyeget.

Az Időbélyegző Szolgáltató nem ismeri a dokumentum tartalmát, hiszen annak csak a lenyomatát kapja meg, melyből a dokumentumot visszaállítani nem tudja. Elektronikus aláírások esetén az időbélyeg azt az időpontot igazolja, amikor az adott aláírás már megtörtént. Amennyiben pontos időpontot, vagy időintervallumot kívánunk megadni, akkor összesen két időbélyegre van szükség. Az első igazolja azt az időpontot, amitől korábban nem keletkezhetett, a másik pedig azt az időpontot, ami előtt már megtörtént az aláírás.

9.2.8.2 Vak aláírások

A vak aláírás a digitális aláírás egy módosított változata, melynek ötlete David Chaumtól ered az 1980-as évek elejéből [6]. A „vak” jelző arra utal, hogy az aláíró úgy hitelesíti a dokumentumot, hogy nem ismeri annak tartalmát. Tipikusan olyan alkalmazásoknál használatos, ahol valamilyen bizalmas információt kell hitelesíttetni. Gyakorlati alkalmazása általában az elektronikus szavazásoknál és a digitális pénznél fordul elő.

A vak aláírás egy interaktív algoritmus, két résztvevővel. Az egyik résztvevő, nevezzük Aladárnak, a másik résztvevővel, mely általában egy szervezet (pl. bank), aláírattat valamely bizalmas dokumentumot. Tehát az algoritmus során Aladár a „vakított” üzenetet elküldi a szervezetnek aláírásra, miután a szervezet hitelesítette a dokumentumot, visszaküldi Aladárnak, aki a „vakítás” megszüntetésével visszanyeri az aláírt, eredeti dokumentumot.

A vak aláírás technikája szemléletesen egy indigós boríték aláírásához hasonlítható. Aladár a dokumentumát belehelyezi egy átlátszatlan, indigós bevonatú borítékba, és lezárja azt. A lezárt borítékot elküldi a szervezetnek, aki kézjegyével látja el a boríték külső oldalát. Mivel a boríték indigós felületű, az aláírás megjelenik a borítékban levő dokumentumon is. Az aláíró személy nem ismeri a dokumentum tartalmát, hiszen a boríték átlátszatlan és sértetlen. Aladár miután kiveszi a dokumentumot, ellenőrizheti a hiteles aláírást.

Vak aláírási rendszerek esetén is minden egyes aláíró fél rendelkezik titkos-nyilvános kulcspárral, ahol a titkos kulcsot természetesen csak az aláíró ismeri, a nyilvános kulcsot pedig egy megbízható hitelesítés-szolgáltató által hitelesített aláíró tanúsítvány tartalmazza. A vak aláírással hitelesített dokumentumot, üzenetet bárki a nyilvános kulcs felhasználásával ellenőrizheti. Ugyanúgy, mint ahogy a digitális aláírásoknál is történik, az üzenet és a nyilvános kulcs ismeretében bárki ellenőrizheti az aláírás érvényességét.

A különbség a digitális aláírások és a vak aláírások között, hogy az utóbbinál aláíró fél az aláírás időpontjában nem ismeri az üzenet tartalmát, míg az előbbinél igen. Viszont az aláírás ellenőrzése mindkét esetben ugyanúgy történik, tehát akkor már akár az aláíró fél is, aki eddig „vak” volt, megtekintheti az üzenet tartalmát.

Az RSA vak aláírás kulcsgenerálása teljesen megegyezik az alap RSA aláírásával. Nyilvános kulcs lesz az e exponens, és jelölje n az RSA modulust. A titkos kulcsot jelöljük d-vel, és legyen a titkos két nagy prím p és q.

Először az üzenet „vakítása” történik meg, majd az így elfedett üzenetet a szervezet aláírja:

1. A kliens választ egy $b \in Z_n$ véletlen „vakító” faktort.
2. A kliens kiszámítja az $m \equiv b^e \bullet H(m') \pmod n$ üzenetet, ahol m' az eredeti üzenet, H pedig egy megfelelő hash függvény.
3. A kliens elküldi az m „vakított” üzenetet a szervezetnek.
4. A szervezet kiszámítja $s \equiv m^d \pmod n$ aláírást, és visszaküldi a feladónak.
5. A kliens a kapott aláírásról „leveszi a vakítást”, azaz megszorozza a vakító faktor multiplikatív inverzével, így megkapja az eredeti m' üzenet érvényes aláírását.

6. A kliens a kapott *s* aláírásból kiszámítja az eredeti üzenet aláírását az $s' \equiv sb^{-1} \pmod n$ kongruenciával, ahol *s'* az eredeti üzenet hiteles aláírása.
7. Az aláírás ellenőrzését a már ismert módon végezhetjük el:
8. A rendelkezésre álló *(m', s')* üzenet és aláírás alapján az ellenőrző fél kiszámítja *H(m')* és $(s')^e \pmod n$ értékeket, és ellenőrzi, hogy kongruensek –e modulo *n*.

Amennyiben a fenti lépések szabályosan hajtódnak végre a $H(m') \equiv (s')^e \pmod n$ kongruencia teljesül:

$$(s')^e \equiv (sb^{-1})^e \equiv (m^d b^{-1})^e \equiv m^{de} b^{-e} \equiv mb^{-e} \equiv b^e H(m') b^{-e} \equiv H(m') \pmod n$$

9.2.8.3 Letagadhatatlan aláírások

A letagadhatatlan aláírások fogalmát 1989-ben Chaum és van Antwerpen vezette be. Az ilyen aláírások több jó tulajdonsággal is rendelkeznek. Közülük a legjelentősebb, hogy egy aláírást csak az aláíró fél közbenjárásával lehet ellenőrizni. Ez lehetővé teszi, hogy az aláíró fél védje az aláírt üzenetet attól, hogy bárki elektronikusan lemásolhassa, illetve szétküldhesse. Az aláírás ellenőrzése az úgynevezett *kihívás-és-válasz* technika alkalmazásával végezhető el.

Jogos a kérdés, ha az aláírás ellenőrzése nem történhet meg az aláíró fél nélkül, mi akadályozza meg őt attól, hogy későbbiekben letagadja azt. Az aláíró fél irányíthatja úgy a protokoll lefutását, hogy egy érvényes aláírást hamisítványnak állítja be, illetve az ellenőrzés nem történik meg. Ennek kiküszöbölése miatt minden letagadhatatlan aláírás része a *tagadó protokoll*, mely segítségével bizonyítani tudja, hogy egy aláírás hamisítvány.

A letagadhatatlan aláírási séma négy algoritmusból áll, a *kulcsgeneráló*, az *aláíró*, az *ellenőrző* és a *tagadó algoritmus*ból. A Chaum-van Antwerpen letagadhatatlan aláírási séma négy algoritmusa a következő:

Kulcsgeneráló algoritmus:

1. Választunk két nagy, véletlen prímet jelöljük őket *p*-vel és *q*-val úgy, hogy $p=2q+1$.
2. Választunk véletlenül egy $g \in \mathbf{Z}_p^*$ elemet, melynek rendje *q*.
3. Véletlenül generálunk egy $1 \le a \le q-1$ értéket.
4. Kiszámítjuk $h \equiv g^a \pmod p$.

A *p, g és h* értékek nyilvánosak, speciálisan a *h*-t hívjuk nyilvános kulcsnak, míg az *a* szám a titkos kulcs.

Aláíró algoritmus:

Jelöljük *G*-vel a $\mathbf{Z}_p^*$ multiplikatív részcsoportját, melynek rendje *q*.

1. Jelölje $m \in G$ az aláírandó üzenetet.
2. Az aláíró fél kiszámítja az *m* üzenet aláírását: $s \equiv m^a \pmod p$

Ellenőrző algoritmus:

1. Az ellenőrző fél választ két véletlen értéket: $e_1, e_2 \in Z_q$

2. Az ellenőrző fél kiszámítja $c \equiv s^{e_1} h^{e_2} \pmod p$ és elküldi az aláíró félnek. (kihívás)
3. Az aláíró fél kiszámítja: $d \equiv c^{a^{-1} \bmod q} \pmod p$ és elküldi az ellenőrzőnek. (válasz)
4. Az ellenőrző fél pontosan akkor tekinti az s értéket m érvényes aláírásnak, ha $d \equiv m^{e_1} g^{e_2} \pmod p$.

Tagadó algoritmus:

1. Az ellenőrző fél választ két véletlen értéket: $e_1, e_2 \in Z_q^*$
2. Az ellenőrző fél kiszámítja $c \equiv s^{e_1} h^{e_2} \pmod p$és elküldi az aláíró félnek. (kihívás)
3. Az aláíró fél kiszámítja: $d \equiv c^{a^{-1} \bmod q} \pmod p$ és elküldi az ellenőrzőnek. (válasz)
4. Az ellenőrző fél megkapja, hogy $d \not\equiv m^{e_1} g^{e_2} \pmod p$.
5. Az ellenőrző fél választ két másik véletlen értéket: $f_1, f_2 \in Z_q^*$
6. Az ellenőrző fél kiszámítja $C \equiv s^{f_1} h^{f_2} \pmod p$ és elküldi az aláíró félnek. (kihívás)
7. Az aláíró fél kiszámítja: $D \equiv C^{a^{-1} \bmod q} \pmod p$ és elküldi az ellenőrzőnek. (válasz)
8. Az ellenőrző fél megkapja, hogy $D \not\equiv m^{f_1} g^{f_2} \pmod p$.
9. Az ellenőrző fél az s aláírást hamisítványnak tekinti pontosan akkor, ha $(dg^{-e_2})^{f_1} \equiv (Dg^{-f_2})^{e_1} \pmod p$.

A Chaum-van Antwerpen letagadhatatlan aláírási séma biztonsága a diszkrét logaritmus problémáján alapszik, hiszen a $h \equiv g^a \pmod p$ alapján h,g és p alapján az a titkos kulcs kiszámítása nehéz.

Elsőnek azt bizonyítjuk, hogy a szabályosan generált aláírást az ellenőrző fél elfogadja. Azaz az $s \equiv m^a \pmod p$ aláírás, ahol $m \in G$, pontosan akkor érvényes, ha az ellenőrző fél a $c \equiv s^{e_1} h^{e_2} \pmod p$ kihívásra a $d \equiv c^{a^{-1} \bmod q} \pmod p$ választ kapva $d \equiv m^{e_1} g^{e_2} \pmod p$, hiszen $d \equiv c^{a^{-1} \bmod q} \equiv (s^{e_1} h^{e_2})^{a^{-1} \bmod q} \equiv (m^{ae_1} g^{ae_2})^{a^{-1} \bmod q} \equiv m^{e_1} g^{e_2} \pmod p$.

A tagadó protokoll valójában az ellenőrző algoritmus kétszeri lefutása, természetesen nem megegyező véletlen értékekkel és egy kongruencia ellenőrzésből áll. Ha $s \not\equiv m^a \pmod p$ és mind az aláíró, mind az ellenőrző fél követi a protokoll lépéseit, akkor $(dg^{-e_2})^{f_1} \equiv (Dg^{-f_2})^{e_1} \pmod p$. A $d \equiv c^{a^{-1} \bmod q} \pmod p$, $h \equiv g^a \pmod p$, $c \equiv s^{e_1} h^{e_2} \pmod p$, $C \equiv s^{f_1} h^{f_2} \pmod p$, $D \equiv C^{a^{-1} \bmod q} \pmod p$ a feltételek mellett:

$$\begin{aligned}(dg^{-e_2})^{f_1} &\equiv \left(c^{a^{-1} \bmod q} g^{-e_2}\right)^{f_1} \equiv \left((s^{e_1} h^{e_2})^{a^{-1} \bmod q} g^{-e_2}\right)^{f_1} \\ &= \left(s^{e_1 a^{-1} \bmod q} h^{e_2 a^{-1} \bmod q} g^{-e_2}\right)^{f_1} = \left(s^{e_1 a^{-1} \bmod q} g^{e_2} g^{-e_2}\right)^{f_1} \\ &= s^{f_1 e_1 a^{-1} \bmod q} \pmod p\end{aligned}$$

$$(Dg^{-f_2})^{e_1} \equiv \left(C^{a^{-1} \bmod q} g^{-f_2}\right)^{e_1} \equiv \left((s^{f_1} h^{f_2})^{a^{-1} \bmod q} g^{-f_2}\right)^{e_1}$$
$$\equiv \left(s^{f_1 a^{-1} \bmod q} h^{f_2 a^{-1} \bmod q} g^{-f_2}\right)^{f_1} \equiv \left(s^{f_1 a^{-1} \bmod q} g^{f_2} g^{-f_2}\right)^{f_1}$$
$$\equiv s^{f_1 e_1 a^{-1} \bmod q} \pmod p$$

A fentiek alapján, tehát tagadó protokoll segítségével az „elvileg aláíró fél” meg tudja győzni az ellenőrző felet, hogy az aláírás hamisítvány. Ugyanakkor az érvényes aláírást nem tudja letagadni. Ebben az esetben feltételezzük, hogy az aláíró fél nem követi a protokoll lépéseit, azaz a *d* és *D* értékeit nem az adott módon számítja ki. A következő tétel szerint egy érvényes aláírás nem letagadható.

Tétel. Feltételezve, hogy $s \equiv m^a \pmod p$ és az ellenőrző fél követi a tagadó protokoll lépéseit, valamint $d \not\equiv m^{e_1} g^{e_2} \pmod p$ és $D \not\equiv m^{f_1} g^{f_2} \pmod p$, akkor annak valószínűsége,

hogy $(dg^{-e_2})^{f_1} \not\equiv (Dg^{-f_2})^{e_1} \pmod p$ az $1\text{-}1/q$.

A tétel bizonyítását itt nem adjuk meg, megtalálható a [35] könyvben.

10 Alkalmazások

10.1 Azonosítási technikák

A digitális aláírások egyik gyakori alkalmazási területe az *azonosítás*, mely során a résztvevő egyértelműen tudja bizonyítani identitását valamely más résztvevőnek. Mint ahogy azt már a 6. fejezetben részleteztük, az azonosítás három megvalósítási módját különböztetjük meg. Az első kategória, amikor fizikai és viselkedési jellemzők alapján, a második esetén birtoklás alapján, a harmadik esetén pedig valamely tudásanyag alapján történik az azonosítás. A gyakorlatban előforduló rendszerek esetén általában több kategóriából vett megoldások keverednek. Például ATM automatáknál történő pénzfelvétel esetén, a bankkártyán kívül PIN kód megadása is szükséges. Ez a példa a birtoklás alapú (bankkártya), és a tudás alapú (PIN kód) megoldás kombinációja.

Tegyük fel, hogy egy P személyt egy V fél kívánja azonosítani. A folyamat során a támadó célja, hogy a P résztvevőt *megszemélyesítse,* azaz úgy viselkedjen, válaszoljon V kérdéseire, mintha P lenne. A támadó azt próbálja elérni, hogy V az azonosítási folyamat végére azt higgye, hogy P-vel vette fel a kapcsolatot. Azt a speciális esetet sem szabad figyelmen kívül hagyni, hogy miután V beazonosítja P-t, ne tudja P-t megszemélyesíteni, azaz V ne rendelkezzen elegendő információval ahhoz, hogy valamely harmadik fél felé úgy tudjon az azonosítás kérdéseire válaszolni, mintha P lenne.

A támadó rendelkezésére áll az összes adat, ami P és V között a nyilvános csatornán keresztül továbbítódott, valamint feltételezzük, hogy pontosan ismeri az azonosítási technika általános folyamatát. A támadó célja azonosítás során mindig a megszemélyesítés. Sikeres támadásnak azt tekintjük, mikor a **támadó aktívan részt vesz**. *Passzív* esetben a támadó folyamatosan figyeli a felek közötti beszélgetést, azaz a nyilvános csatornán folyó kommunikációt, valamint rendelkezésére áll az összes nyilvános információ, paraméter. A *lehallgatás*sal összegyűjtött információkat egyszer egy későbbi időpontban felhasználva (aktívan) megszemélyesíti valamely résztvevőt. *Aktív* esetben a támadó megváltoztatja a csatornán továbbított üzeneteket. Több támadás is idesorolható. *Módosítás* során a támadó megváltoztatja a küldött üzenetet, *visszajátszásos támadások* során a támadó olyan üzeneteket küld valamely résztvevőnek, melyek szerepeltek már valamely korábbi üzenetben. Egyik példa támadásra például, amikor a támadó közvetlenül az ellenőrzővel kommunikál, és többször egymásután megpróbál „megfelelően” válaszolni a feltett kérdésekre. Az ilyen típusú támadások elleni védekezés egyik lehetséges módja például a próbálkozások számának maximalizálása. Például egymásután legfeljebb háromszor adhat meg a felhasználó rossz jelszót.

Sok biztonságos, és a gyakorlatban is alkalmazható azonosítási rendszer létezik [35]. Egyik fontos szempont azonban, hogy az azonosítási technika implementálható legyen smart kártyán, így a számítási bonyolultságnak is és a memória felhasználásának is minimálisnak

kell lennie. Természetesen fokozott figyelmet kell szentelnünk ebben az esetben a kártyára, hiszen, a kártya elegendő ahhoz, hogy az azonosítási kérdésekre megfelelően válaszoljon a támadó. Mindazonáltal, hogy igyekszünk nem elveszíteni, vagy elvesztés esetén egyből letiltatni, szükséges a több faktorú rendszer megvalósítása, azaz a kártya mellett szükséges valamely más adat ismerete is (pl. PIN kód).

Az azonosítás lehet *egyoldalú*, illetve *kölcsönös*. Az előbbi esetben csak a résztvevők egyikének azonosítása történik meg, míg az utóbbinál valamennyi résztvevő identitása ellenőrzésre kerül. Egyoldalú azonosításra jó példa, amikor egy szerver valamely klienst azonosítja be. Kölcsönös azonosítás történik az Internetes banki felületeknél, amikor a bank SSL tanúsítványával igazolja magát, míg az ügyfél felhasználói név és jelszó párossal, vagy más technikával bizonyítja identitását.

Sok olyan rendszer áll a rendelkezésünkre, mely az entitás beazonosításával egyidejűleg kulcscserét is megvalósít. A kulcscsere rendszerekkel itt nem foglalkozunk.

10.1.1 Jelszó alapú rendszerek

A gyakorlatban leggyakrabban előforduló azonosítási technika a jelszó alapú azonosítás. Ez többnyire azért van így, mert könnyű implementálni és felhasználóbarát abban az értelemben, hogy nem feltételez extra hardvert és könnyen kezelhető. Viszont több biztonsági kérdést is felvet, amit már a 6.4 fejezetben ismertettünk.

Egyik jelentős probléma az alap jelszó alapú rendszerekkel, a statikusságuk. Fontos feltétel ahhoz, hogy egy azonosítási rendszer biztonságos legyen, a véletlenszerűség biztosítása. Amennyiben az adatsor, amit P küld V-nek, hogy azonosítsa magát mindig ugyanaz, azaz nem változik, akkor a támadó *visszajátszásos támadással* meg tudja személyesíteni P-t. Tehát minden egyes azonosítás során alkalmazni kell valamely véletlenül generált értéket, így minden alkalommal más és más lesz az elküldött adatsor.

Jelszó alapú azonosítás esetén, ha SSL/TLS kapcsolatot alakítunk ki a két fél között, és a felhasználói név a jelszóval már a titkos csatornán továbbítódik, akkor az SSL/TLS protokoll miatt a kódolt jelszó már nem lesz statikus.

10.1.2 Egyszer használatos jelszavak

Egy másik megoldás az egyszer használatos jelszavak (One-time password ─ OTP) használata. Mint ahogy a neve is mutatja, minden jelszó csak egyszer használható. Az egyszer használhatóság miatt a visszajátszásos támadás nem releváns. Ugyanis ha a támadó valahogy megtudja az egyszer használatos jelszót, akkor azt hiába próbálja újra elküldeni, az ellenőrző fél nem fogja elfogadni, hiszen az már nem lesz érvényes. Tehát az egyszer használatos jelszavak biztonságosabbak a statikusoknál, viszont az egyszer használhatóságuk miatt nehezebb is az ember számára megjegyezni.

Alapvetően két kategóriát különböztetünk meg: *időszinkronizációs* és *egyirányú függvényen alapuló* rendszerek kategóriáját.

- *időszinkronizációs*: Az ilyen rendszerek általában egy hardvereszközzel, tokennel együtt járnak. A token tartalmaz egy beépített órát, mely szinkronizálva van az ellenőrző órájával. Az egyszer használatos jelszavak generálásának alapja az aktuális idő.
- *egyirányú függvényen alapuló:* Ezek a rendszerek általában vagy egy megelőző jelszón, vagy egy kapott véletlen értéken alapulva generálják le az egyszer használatos jelszót, egyirányú függvények segítségével.

Az egyszer használatos jelszavak egyik megvalósítása az RSA Security cég által kifejlesztett SecureID token. A rendszer és az algoritmus részletes felépítése nem publikus. Ismereteink szerint minden token tartalmaz egy kriptográfiai processzort, mely szimmetrikus titkosítást hajt végre, egy beépített órát és minden egyes tokenen van egy titkos kulcs is és egy kijelző. Az egyszer használatos jelszó az aktuális időn alapuló érték szimmetrikus titkosítása lesz. Bizonyos időintervallumonként, pl. percenként, generál egy jelszót, melyet megjelenít a kijelzőn. A tulajdonos általában az azonosítás során valamely más jelszóval együtt használja.

A legelső egyszer használatos jelszavakat generáló rendszerek egyike a *Lamport séma,* mely a következő lépésekből áll.

- A kliens egy kezdeti w jelszóhoz a h egyirányú függvény segítségével kiszámítja az $(n,h^n(w))$ elempárt, ahol $h^n(w)$-t a w szóból a h egyirányú függvény n-szeri alkalmazásával kapjuk. A kliens eljuttatja a $(n,h^n(w))$ párt az ellenőrző félnek.
- Amikor a kliens i-edik alkalommal azonosítja magát, akkor elküldi $w_i=h^{n-i}(w)$ az ellenőrző félnek, ami megvizsgálja, hogy szerepel –e az adatbázisában az $(n-i+1,f(w_i))$ pár, ha igen, akkor az azonosítás sikeres és felülírja az $(n-i+1,f(w_i))$ párt $(n-i,w_i)$-re.

Az ellenőrző oldalon elegendő mindig egy függvényértéket kiszámolni és ellenőrzést végezni. A kliens vagy letárolja az összes jelszót és így nem kell számítást végezni, vagy letárolja w-t és számításokat végez. Vegyük észre, hogy a $h^n(w)$ érték megadásával a w titokban marad, hiszen a h egyirányúsága miatt a függvényértékből az w-t nehéz kiszámítani. A Lamport sémán alapulva két rendszert is implementáltak az S/Key és OPIE rendszereket.

10.1.3 Kihívás-és-válasz alapú rendszerek

Az eddig ismertetett jelszó alapú rendszereknél az azonosítás során a kliens felhasználói név és jelszó párost küld az ellenőrző félnek. Az ellenőrzés során bizonyos előre egyeztetett adatok alapján végzi el, mint például biztonságosan tárolt jelszó vagy szinkronizált óra, de nem kommunikál többet a klienssel. A *kihívás-és-válasz* (challenge-and-response) rendszerek esetén az azonosítás az ellenőrző fél és a kliens interakciója során történik meg. Mint ahogy a neve is mutatja, V egy kihívást (challenge), azaz egy kérdést tesz fel P számára, ami általában egy véletlen érték, melyet sokszor *nonce*-nak (number used only once) hívunk. Amennyiben P megfelelően válaszolja meg (response) a kérdést, azaz a véletlen értékkel megfelelő számításokat végez, akkor P azonosítása sikeres. Vannak olyan megoldások, mely során az ellenőrző fél több kihívást is küld, melyre várja a válaszokat.

Kriptográfiai megoldások alapján az azonosítási rendszerek két nagy csoportját különböztetjük meg. Az egyik csoportba azok a megoldások tartoznak, melyek kriptográfiai

primitívek, pl. hitelesítő kódok, digitális aláírások etc., közvetlen alkalmazásai. A másik kategória, amikor nem valamely primitívre épülnek, hanem közvetlenül, valamely nehéz problémára. Az utóbbi kategóriába tartozó sémákat itt nem ismertetjük, több megoldást is talál az olvasó a [35] könyvben.

Az alábbiakban ismertetett rendszerek valamennyien *kihívás-és-válasz* alapú megoldások. Két fő kategóriát különböztetünk meg a *szimmetrikus* és az *aszimmetrikus* kulcsú rendszereket.

10.1.3.1 Szimmetrikus kulcsú rendszerek

Ebben a fejezetben azokat a megoldásokat ismertetjük, melyek szimmetrikus kulcsokat használnak. Ezt a szimmetrikus kulcsot, melyet mindkét fél ismer, jelöljük *K*-val. Tekintsük először azt az esetet, amikor az alkalmazott kriptográfiai primitív hitelesítő kód, melyet MAC_K-val jelölünk.

Hitelesítő kód alapú azonosítási rendszerek esetén valamely résztvevő azonosítása az alapján történik meg, hogy képes–e szabályosan kiszámolni egy véletlenszám hitelesítő kódját. Mivel csak az identitását bizonyító és az ellenőrző résztvevő ismeri a szimmetrikus kulcsot, így csak ők tudják megadni a megfelelő kódot.

Két lépésben fogjuk bemutatni a *hitelesítő kódon alapuló, egyoldalú* azonosító rendszert. Az első lépésben ismertetett rendszer még nem biztonságos, a sikeres támadást is részletezzük. A második lépésben pedig a végleges, biztonságos rendszert adjuk meg.

Ahhoz, hogy a sémánk biztonságos legyen, feltételezzük, hogy a titkos *K* kulcsot *csak* az azonosításban résztvevő két fél ismeri, valamint az ellenőrző fél számára rendelkezésre áll egy jó tulajdonságokkal rendelkező álvéletlenszám generátor, valamint az alkalmazott üzenethitelesítő kód biztonságos.

Az azonosítási rendszer két résztvevőjét továbbra is jelöljük *P*-vel és *V*-vel, ahol *P* jelöli azt az entitást, aki igazolja magát és *V* pedig az ellenőrző felet. A **nem biztonságos** rendszer lépései a következők:

1. *V* generál egy véletlen *r* számot, a kihívást és elküldi *P*-nek.
2. *P* kiszámítja $m=MAC_K(r)$ és visszaküldi *V*-nek.
3. *V* kiszámítja $m'=MAC_K(r)$ és ellenőrzi az $m=m'$ egyenlőséget, ha teljesül, akkor az azonosítás sikeres.

A támadó ismeri az *r* véletlen értéket, és célja az $MAC_K(r)$ kiszámítása. Mivel a titkos *K* kulcsot csak *P* és *V* ismeri a támadó nem, így a hitelesítő kódot kiszámítani nem tudja, még akkor sem, ha megelőzően több $(r_i,MAC_K(r_i))$ elempár a birtokába került. Ennek ellenére mégis lehet támadást indítani a rendszer ellen. A támadót *A*-val jelöljük. A támadás a következőképpen történik:

1. *V* generál egy véletlen *r* számot, a kihívást és elküldi *P*-nek.
2. *A* a csatornát lehallgatva megtudja az *r* értéket és visszaküldi *V*-nek.
3. *V* kiszámítja $m=MAC_K(r)$ értéket és visszaküldi *A*-nak.
4. *A* elküldi $m=MAC_K(r)$ kódot *V*-nek.

5. *V* kiszámítja $m'=MAC_K(r)$ és ellenőrzi az $m=m'$ egyenlőséget.

Talán az olvasó azt gondolja, hogy ez a támadás nem elég realisztikus. Az igazság az, hogy előfordulhatnak olyan helyzetek, amikor ez a támadás releváns lehet, célunk, hogy egy azonosítási rendszer biztonságos legyen minden szituációban.

Az előbbi protokoll javítása, azaz a **biztonságos** megoldás lépései a következő:

1. *V* generál egy véletlen *r* számot, a kihívást és elküldi *P*-nek.
2. *P* kiszámítja $m=MAC_K(ID(P)||r)$ és visszaküldi *V*-nek.
3. *V* kiszámítja $m'=MAC_K(ID(P)||r)$ és ellenőrzi az $m=m'$ egyenlőséget, ha teljesül, akkor az azonosítás sikeres.

Láthatjuk, hogy a különbség az $m=MAC_K(ID(P)||r)$ üzenetben rejlik. Amennyiben a támadó az előbb ismertetett támadást próbálja végrehajtani, akkor *V*-től az $m=MAC_K(ID(V)||r)$ értéket kapja, amit később *V* vissza fog utasítani.

Ahhoz, hogy azt mondhassuk egy rendszer biztonságos, fontos pontosan megadni, hogy mit értünk támadáson. Például a következő nem minősül támadásnak:

1. *V* generál egy véletlen *r* számot, a kihívást és elküldi *P*-nek.
2. *A* lehallgatja *r*-et és továbbítja *P*-nek.
3. *P* kiszámítja $m=MAC_K(ID(P)||r)$ és visszaküldi *A*-nak.
4. *A* továbbítja $m=MAC_K(ID(P)||r)$ üzenetet *V*-nek.
5. *V* kiszámítja $m'=MAC_K(ID(P)||r)$ és ellenőrzi az $m=m'$ egyenlőséget, ha teljesül, akkor az azonosítás sikeres.

A támadó itt nem aktív, az azonosítás folyamata pontosan ugyanúgy hajtódott végre, mintha a támadó részt sem vett volna a folyamatban. *P* azonosítása sikeresen megtörtént, amit *P* el akart küldeni *V*-nek el is jutott abban a formában, ahogy elküldte, és ugyanez igaz *V*-re is. Támadás akkor történik, ha vagy megváltozik az üzenet, vagy valamely más résztvevőnek továbbítódik.

Most tekintsük a *hitelesítő kódon alapuló, kölcsönös* azonosítási rendszert. Ebben az esetben az azonosítás sikeresen fut le, ha *mindkét fél* azonosítása megtörténik. A támadó célja akár *P,* akár *Q,* akár mindkettőjük megszemélyesítése.

Vizsgáljunk meg itt is egy lehetséges megoldást a kölcsönös azonosításra, mely **nem biztonságos**.

1. *Q* generál egy véletlen r_1 számot, a kihívást, és elküldi *P*-nek.

2. *P* generál egy véletlen r_2 számot, a kihívást, kiszámítja $m_1=MAC_K(ID(P)||\ r_1)||\ r_2$ és elküldi *Q*-nak.

3. *Q* kiszámítja $m_1'=MAC_K(ID(P)||\ r_1)$ és ellenőrzi az $m_1=m_1'$ egyenlőséget, ha teljesül, akkor *P* azonosítása sikeres. *Q* kiszámítja $m_2=MAC_K(ID(Q)||\ r_2)$ és elküldi *P*-nek.

4. *P* kiszámítja $m_2'=MAC_K(ID(Q)||\ r_2)$ és ellenőrzi az $m_2=m_2'$ egyenlőséget, ha teljesül, akkor *Q* azonosítása sikeres.

Ez a protokoll nem biztonságos, hiszen a támadó sikeres támadást tud indítani. Csak az egyik fél (Q) megszemélyesítését mutatjuk meg, a támadás hasonlóan végrehajtható a másik féllel szemben is.

1. A generál egy véletlen r_1 számot, és elküldi P-nek.
2. P generál egy véletlen r_2 számot, kiszámítja $m_1=MAC_K(ID(P)||\ r_1)||\ r_2$ és elküldi A-nak.
3. A az r_2 értéket elküldi Q-nak, aki kiszámítja $m_2=MAC_K(ID(Q)||\ r_2)$-t és generál egy r_3 véletlen értéket, majd $m_2||r_3$ üzenetet visszaküldi A-nak.
4. A továbbítja $m_2=MAC_K(ID(Q)||\ r_2)$-t P-nek.
5. P kiszámítja $m_2'=MAC_K(ID(Q)||\ r_2)$ és ellenőrzi az $m_2=m_2'$ egyenlőséget, ha teljesül, akkor Q azonosítása sikeres.

Látható, hogy a támadó a 3. pontban egy új folyamatot indít el Q-val P-től kapott r_2-vel, aki kiszámítja a megfelelő $m_2=MAC_K(ID(Q)||\ r_2)$ hitelesítő kódot. Így a támadó sikeresen megszemélyesíti a Q résztvevőt. A protokollt a következőképpen lehet biztonságossá tenni:

1. Q generál egy véletlen r_1 számot, a kihívást, és elküldi P-nek.
2. P generál egy véletlen r_2 számot, a kihívást, kiszámítja $m_1=MAC_K(ID(P)||\ r_1||\ r_2)$ és elküldi Q-nak.
3. Q kiszámítja $m_1'=MAC_K(ID(P)||\ r_1||\ r_2)$ és ellenőrzi az $m_1=m_1'$ egyenlőséget, ha teljesül, akkor P azonosítása sikeres. Q kiszámítja $m_2=MAC_K(ID(Q)||\ r_2)$ és elküldi P-nek.
4. P kiszámítja $m_2'=MAC_K(ID(Q)||\ r_2)$ és ellenőrzi az $m_2=m_2'$ egyenlőséget, ha teljesül, akkor Q azonosítása sikeres.

Szimmetrikus kulcsú azonosítási sémákat nemcsak üzenethitelesítő kódból tervezhetünk, hanem szimmetrikus titkosítás felhasználásával is. Egyik megoldás, hogy az ellenőrző fél által generált véletlen értéket, a kihívást, a szimmetrikus kulccsal titkosítania kell a kliensnek. A másik lehetőség, hogy a kihívás egy a szimmetrikus kulccsal titkosított érték, és a kliensnek vissza kell fejtenie a titkos kulcsot felhasználva. Valamennyi megoldás esetén mindkét fél a titkos szimmetrikus kulcs segítségével végez számításokat, ellenőrzéseket. Mivel a szimmetrikus kulcsot csak a két fél ismeri, így, ha az adott résztvevő a titkos kulcsot szabályosan alkalmazza, akkor azonosítása sikeres.

A szimmetrikus *titkosítás* alapú azonosítás **váza** a következő:

1. $V \rightarrow P$: r véletlen
2. $P \rightarrow V$: $Enc_K(r)$

A szimmetrikus *visszafejtés* alapú azonosítás **váza** a következő:

1. $V \rightarrow P$: $Enc_K(r)$, ahol r véletlen
2. $P \rightarrow V$: r

10.1.3.2 Aszimmetrikus kulcsú rendszerek

Szimmetrikus kulcsú azonosítás esetén az ellenőrző félnek az első alkalommal minden egyes résztvevővel kulcsot kell cserélnie, majd a titkos kulcsot biztonságosan tárolnia kell.

Ezzel szemben aszimmetrikus kulcsú rendszereknél minden résztvevő egy kulcspárral rendelkezik, azaz egy titkos és egy nyilvános kulccsal. Az identitását bizonyító *P* résztvevő a titkos kulcsát felhasználva *(SK_P)* végez számításokat, az ellenőrző fél, *Q* pedig *P* nyilvános kulcsával *(PK_P)* végzi az ellenőrzést. Az alapvető különbség a szimmetrikus és aszimmetrikus kulcsú azonosítási rendszerek között, hogy az azonosítási séma lépésein túl aszimmetrikus esetben nem szükséges megelőző kulcscsere és nem kell tárolni a bizonyító fél kulcsát sem, hanem mindössze a bizonyító fél tanúsítványát kell ellenőrizni. A tanúsítványokról részletesen a 11.7 fejezetben olvashatunk.

Kétféle aszimmetrikus kulcsú azonosítási rendszer is létezik. Az egyik esetben az identitását bizonyító fél aláírja a kihívásként generált véletlen számot, a másik esetben pedig titkos kulcsával visszafejti a kihívásként küldött aszimmetrikusan titkosított véletlen értéket. Aszimmetrikus esetben is, ha az adott fél egy véletlen értékre megfelelően alkalmazza titkos kulcsát, akkor sikeresen igazolja identitását.

Az *egyoldalú azonosítás* során két résztvevő szerepel: *P* a bizonyító személy, aki bizonyítani kívánja személyazonosságát, és *V* az ellenőrző személy, aki meggyőződik *P* identitásáról.

A *digitális aláírás alapú, egyoldalú azonosítási rendszer* esetén a protokoll lépései a következőek:

1. *V* generál egy véletlen r_V számot, a kihívást és elküldi *P*-nek.
2. *P* kiszámítja $m=r_P||r_V||ID(V)||\mathbf{Sign}_{SK_P}(r_P||r_V||ID(V))||Cert_p$ és visszaküldi *V*-nek, ahol r_P a *P* által generált véletlen szám és $Cert_p$ a *P* tanúsítványa.
3. *V* ellenőrzi *P* tanúsítványát, azaz a $Cert_p$ érvényes és hiteles –e.
4. *V* ellenőrzi az aláírás érvényességét, azaz ténylegesen *P* titkos kulcsával történt -e az aláírás, valamint az elküldött r_V és a kapott r_P véletlen értékek és *ID(V))* konkatenációja lett–e aláírva.

Amennyiben a 3. és 4. lépésben a tanúsítvány és az aláírás is hiteles, akkor *P* sikeresen igazolta identitását. A *P* által generált r_P véletlen szám az üzenet egyedi voltát jelzi, ugyanis *P* ugyanazt az aláíró tanúsítványt többször, több alkalmazásnál is felhasználhatja. Tegyük fel, hogy ha *P* csak a $m=r_V||ID(V)||\mathbf{Sign}_{SK_P}(r_V||ID(V))||Cert_p$ üzenetet küldi el, melyet a támadó eltárol. Amennyiben valamely más alkalmazás ugyanazt az r_V értéket küldi *P*-nek, akkor a támadó – például *V* maga – is tudná igazolni, hogy ő *P*.

A fenti egyoldalú azonosítás könnyen kölcsönössé alakítható, mely során mindkét fél igazolja identitását a másiknak. Tekintsük a *digitális aláíráson alapuló, kölcsönös azonosítási sémát*.

Először nézzünk egy olyan rendszert, mely **nem biztonságos**:

1. *V* generál egy véletlen r_V számot, a kihívást, és elküldi *P*-nek.
2. *P* kiszámítja $m=r_P||r_V||ID(V)||\mathbf{Sign}_{SK_P}(r_P||r_V||ID(V))||Cert_p$ és visszaküldi *V*-nek, ahol r_P a *P* által generált véletlen szám és $Cert_p$ a *P* tanúsítványa.
3. *V* ellenőrzi *P* tanúsítványát, azaz a $Cert_p$ érvényes és hiteles–e.

4. *V* ellenőrzi az aláírás érvényességét, azaz ténylegesen *P* titkos kulcsával történt-e az aláírás, valamint az elküldött r_V és a kapott r_P véletlen értékek és *ID(V)* konkatenációja lett –e aláírva.
5. *V* generál egy másik véletlen r_V' számot, és az

$$m'=r_P||r_V'||ID(P)||\mathbf{Sign}_{SK_V}(r_P||r_V'||ID(P))||Cert_V$$

üzenettel együtt elküldi *P*-nek, ahol *V* tanúsítványát $Cert_V$ jelöli.
6. *P* ellenőrzi *V* tanúsítványát, azaz a $Cert_V$ érvényes és hiteles–e.
7. *P* ellenőrzi az aláírás érvényességét, azaz ténylegesen *V* titkos kulcsával történt-e az aláírás, valamint az elküldött r_P és a kapott r_V' véletlen értékek és *ID(P)* konkatenációja lett–e aláírva.

Amennyiben a 3. és 4. lépésben a tanúsítvány és az aláírás is hiteles, akkor *P* sikeresen igazolta identitását, ha a 6. és 7. lépésben az ellenőrzések eredménye pozitív, akkor *V*-t sikerült azonosítani. Az előbbi séma az egyoldalú azonosítási rendszer kétszeri végrehajtása, hiszen az 5. lépés a 2. lépés analógja. Milyen sikeres támadást adhatunk meg a fenti sémával szemben?

A rendszer gyenge pontja az 5. lépésben generált r_V' véletlen érték, melyet *V* titkos kulcsával aláír. A támadó *P*-vel és *V*-vel futtatja le a kölcsönös azonosítási rendszert úgy, hogy hol *V*-t, hol *P-t* személyesíti meg. A támadás lépései a következők:

1. *A* megszemélyesíti *V*-t, generál egy véletlen r_V számot, a kihívást, és elküldi *P*-nek.
2. *P* kiszámítja $m=r_P||r_V||ID(V)||\mathbf{Sign}_{SK_P}(r_P||r_V||ID(V))||Cert_p$ és visszaküldi *A*-nak, ahol r_P a *P* által generált véletlen szám és $Cert_p$ a *P* tanúsítványa.
3. *A* azonosítási folyamatot kezdeményez *V*-vel, megszemélyesíti *P*-t, tehát elküldi neki az r_P véletlent.
4. *V* kiszámítja $m'=r_P||r_V'||ID(P)||\mathbf{Sign}_{SK_V}(r_P||r_V'||ID(P))||Cert_V$ és visszaküldi *A*-nak, ahol r_V' a *V* által generált véletlen szám, és $Cert_V$ a tanúsítványa.
5. *A* elküldi $m'=r_P||r_V'||ID(P)||\mathbf{Sign}_{SK_V}(r_P||r_V'||ID(P))||Cert_V$ üzenetet *P*-nek.
6. *P* ellenőrzi *V* tanúsítványát, azaz a $Cert_V$ érvényes és hiteles–e.
7. *P* ellenőrzi az aláírás érvényességét, azaz ténylegesen *V* titkos kulcsával történt -e az aláírás, valamint az elküldött r_P és a kapott r_V' véletlen értékek és *ID(P)* konkatenációja lett–e aláírva.

P a 6. és a 7. lépésben elvégzi az ellenőrzéseket. Mind a tanúsítvány, mind az aláírás érvényes, hiszen *V* tanúsítványát küldte el *A,* illetve maga *V* titkos kulcsával írta alá az adott üzenetet.

A **biztonságos** kölcsönös, digitális aláíráson alapuló azonosítási folyamat lépései a következők:

1. *V* generál egy véletlen r_V számot, a kihívást, és elküldi *P*-nek.
2. *P* kiszámítja $m=r_P||r_V||ID(V)||\mathbf{Sign}_{SK_P}(r_P||r_V||ID(V))||Cert_p$ és visszaküldi *V*-nek, ahol r_P a *P* által generált véletlen szám és $Cert_p$ a *P* tanúsítványa.

3. V ellenőrzi P tanúsítványát, azaz a $Cert_p$ érvényes és hiteles–e.
4. V ellenőrzi az aláírás érvényességét, azaz ténylegesen P titkos kulcsával történt -e az aláírás, valamint az elküldött r_V és a kapott r_P véletlen értékek és $ID(V)$ konkatenációja lett–e aláírva.
5. V kiszámítja az $m'=r_P||ID(P)||Sign_{SK_V}(r_P||ID(P))||Cert_V$ üzenetet és elküldi P-nek, ahol V tanúsítványát $Cert_V$ jelöli.
6. P ellenőrzi V tanúsítványát, azaz a $Cert_V$ érvényes és hiteles–e.
7. P ellenőrzi az aláírás érvényességét, azaz ténylegesen V titkos kulcsával történt-e az aláírás, valamint az elküldött r_P és $ID(P)$ konkatenációja lett–e aláírva.

Ha a 3., 4., 6. és 7. lépésben végzett ellenőrzések sikeresen megtörténtek, akkor mind P, mind V azonosítása befejeződött. A résztvevők jelenlétét a megfelelő véletlen számok aláírása garantálja.

Aszimmetrikus azonosítási séma generálható aszimmetrikus titkosítás alkalmazásával is. Ebben az esetben az identitását bizonyító félnek vissza kell fejtenie titkos kulcsával a számára titkosított véletlenszámot. Az ilyen típusú azonosító sémák **váza** a következő:

1. $V \rightarrow P$: $Enc_{FK_P}(r)$, ahol r véletlen
2. $P \rightarrow V$: r

10.1.4 Nulla-ismeretű protokollok

A napjainkban alkalmazott azonosítási rendszerek (pl. jelszó vagy kihívás-és-válasz alapú) általában kiszivárogtatnak valamely részinformációt a titkos információról, melyet az identitását bizonyító fél felhasznál. Például, a digitális aláíráson alapuló azonosítási rendszerek kiszivárogtatják a kihívásként küldött üzenet digitális aláírását. Így az aláírt kihívást a támadó felhasználhatja. Sikeres támadást lehet megvalósítani abban az esetben, ha az identitását bizonyító fél azt a titkos aláíró kulcsot használja azonosításra, mint amit az üzenetek digitális aláírására, akkor az ellenőrző fél kihívásként valamely dokumentum hash értékét is küldheti. Így megszerzi egy általa létrehozott üzenet aláírását.

Ha azt szeretnénk, hogy az azonosítási rendszer semmilyen részinformációt ne adjon ki, akkor *nulla-ismeretű azonosítási rendszer*t kell használnunk. Leegyszerűsítve az ilyen rendszer azon kívül, hogy bizonyítja valamely állítás helyességét, más információt nem ad ki. Az ellenőrző fél azon kívül, hogy bizonyítékot kap arról, hogy az adott állítás helyes, más információhoz nem jut. Ez azt is jelenti, hogy minden, ami polinomiális időn belül kiszámítható a nulla-ismeretű bizonyítás üzeneteiből, az polinomiális időn belül kiszámítható az érvényes állításból magából is. Egy protokoll nulla-ismeretű tulajdonsága tehát azt jelenti, hogy bármit, amit az ellenőrző fél „láť” a bizonyító féllel való interakciója során, mindazt polinomiális időn belül tudja szimulálni maga is a bizonyító fél részvétele nélkül. Azt fontos megjegyezni, hogy a nulla-ismeretű tulajdonságnak akkor is teljesülnie kell, ha az ellenőrző fél nem követi a protokoll előírt lépéseit, hanem eltér tőle.

Amos Fiat és Adi Shamir létrehozott egy nulla-ismeretű azonosítási rendszert, melynek biztonsága a modulo összetett szám szerinti négyzetgyökvonás kiszámításának nehézségén alapszik.

Definíció. Ha az $x^2 \equiv a \pmod n$ kongruencia megoldható, akkor az a számot *kvadratikus maradék*nak nevezzük modulo n.

Az $a \equiv 0 \pmod n$ számokat nem soroljuk sem a kvadratikus maradékok, sem a kvadratikus nemmaradékok közé.

Az $x^2 \equiv a \pmod n$ kongruencia megoldásainak száma függ a modulustól. Ha a modulus prím, azaz $n=p$, ahol p prím, akkor a kongruenciának legfeljebb két megoldása lehet. Ha $b \pmod p$ megoldása a $x^2 \equiv a \pmod p$ kongruenciának, akkor $p-b \pmod p$ is az.

Amennyiben $n=pq$, ahol p és q prímszámok, akkor, ha az $x^2 \equiv a \pmod n$ kongruenciának van megoldása, akkor négy megoldása is van. A megoldásokat az

$$x^2 \equiv a \pmod p$$
$$x^2 \equiv a \pmod q$$

kongruenciákból a kínai maradéktétel segítségével számíthatjuk ki. A megoldásokat $b_1 \pmod n$, $n-b_1 \pmod n$, $b_2 \pmod n$, $n-b_2 \pmod n$ formában kapjuk meg.

Ha az $x^2 \equiv a \pmod p$ kongruencia megoldható, akkor megoldásainak meghatározására létezik polinomiális idejű algoritmus ($O(\log^4 p)$). Az $x^2 \equiv a \pmod n$, ahol $n=pq$, kongruencia megoldásait p és q ismeretében polinomiális időn belül kiszámíthatjuk a kínai maradéktétel és az $x^2 \equiv a \pmod p$ kongruencia megoldására vonatkozó hatékony algoritmus alkalmazásával. Viszont, ha az n modulus faktorai nem ismertek, akkor az $x^2 \equiv a \pmod n$ kongruencia megoldása *nehéz* probléma.

A Fiat-Shamir azonosítási protokoll biztonsága azon alapszik, hogy az $x^2 \equiv a \pmod n$ kongruencia megoldása *könnyű*, ha n faktorai ismertek, ellenkező esetben *nehéz*.

A protokollnak két résztvevője van, P az identitását bizonyító fél és V az ellenőrző entitás. V véletlenszerűen választ két olyan nagy prímet, hogy a szorzatuk faktorizálása nehéz legyen. V kiszámítja $n=pq$ modulust, melyet nyilvánosságra hoz. A bizonyító fél generál egy véletlen $x \in_R \mathbf{Z}_n^*$, melyet titokban tart és kiszámítja $y \equiv x^2 \pmod n$ értéket, melyet nyilvánosságra hoz. Így az x a bizonyító fél titkos kulcsa, míg az n és y a nyilvános kulcsa. A Fiat-Shamir azonosítási rendszer a következő kör többszöri végrehajtása.

1. P generál egy véletlen számot: $r \in_R \mathbf{Z}_n^*$, ahol $\in_R$ jelöli, hogy véletlenül választunk. P kiszámítja $t \equiv r^2 \pmod n$ és elküldi t-t V-nek.
2. V generál egy véletlen bitet: $c \in_R \{0,1\}$. V elküldi c-t P-nek.
3. P kiszámítja $s \equiv r x^c \pmod n$ számot, és elküldi V-nek.
4. V ellenőrzi, hogy az $s^2 \equiv t y^c \pmod n$ kongruencia teljesül–e.

Ha a 4. pontban a kongruencia teljesül, akkor P igazolta identitását V felé, ha nem, akkor V elutasítja. Mint ahogy azt már említettük, az előbb részletett 4 lépést ugyanazon felek között többször is végrehajtódik. Minden egyes körben új r és c véletlen értékek generálódnak.

Vegyük észre, hogy a Fiat-Shamir protokoll is tartalmaz egy kihívás-és-válasz részt. A 2. pontban küldött $c \in_R \{0,1\}$ véletlen bit valójában egy *kihívás*, melyre *P*-nek, azaz az identitását bizonyító félnek megfelelő választ kell adnia a 3. pontban. A 4. lépésben az ellenőrző fél elvégzi a megfelelő ellenőrzéseket.

A protokoll minden szabályosan generált *s* válaszra igaz értékkel tér vissza, azaz a *V* ellenőrző személy elfogadja az identitását bizonyító *P* fél válaszát, hiszen

$$s^2 \equiv r^2 (x^c)^2 \equiv t(x^2)^c \equiv t y^c \pmod n.$$

Azt bizonyítani, hogy minden nem szabályosan generált válasz esetén a protokoll hamis értékkel tér vissza, azaz a *V* fél nem fogadja el a bizonyítást, meg kell vizsgálnunk, hogy a támadó milyen támadásokat indíthat a rendszerrel szemben.

A lehetséges támadások a következők:

- A támadó generál egy véletlen $t \in_R \mathbf{Z}_n^*$ és vár *V* véletlen *c* bitjére, majd megpróbálja kitalálni a megfelelő *s* válaszértéket. Ennek valószínűsége elég nagy *n* esetén kicsi.
- A támadó megpróbálja megjósolni a *c* véletlen bitet és ennek megfelelően adja meg a *t*, illetve *s* számokat. Ha $c=0$, akkor véletlenül választ egy $r \in_R \mathbf{Z}_n^*$ számot, majd kiszámítja $t \equiv r^2 \pmod n$ számot, és a 4. lépésben az $s=r$ válaszértéket adja meg. Amennyiben azt jósolja meg, hogy $c=1$, akkor a támadó véletlenül választ egy $s \in_R \mathbf{Z}_n^*$ számot és kiszámítja a $t \equiv s^2/y \pmod n$. Az így meghatározott értékeket a megfelelő lépésekben a támadó elküldi *V*-nek. Mindkét esetre a támadó nem tud felkészülni, mert akkor ki tudja számítani az *x* titkos kulcsot. Ugyanis, ha valahogy a támadó meg tudná adni az s_0 értéket, ha $c=0$ és s_1-t, ha $c=1$, azaz tudja, hogy milyen értéket vesz fel s_0, hogy $s_0=r$, és s_1-t, hogy $s_1=rx$, akkor könnyen meg tudja határozni az *x* titkos kulcsot is, hiszen $x=s_1/s_0$. Annak valószínűsége, hogy a támadó eltalálja a *c* véletlen bitet *1/2* minden körben. Ha a lépéseket *k*-szor hajtjuk végre, akkor annak valószínűsége, hogy a támadó sikeresen adja meg az értékeket $1/2^k$.

A Fiat-Shamir protokollnak több verziója is létezik. Egyik lehetőség a körök számának csökkentése úgy, hogy az *t,c,s* számok mindegyike egy-egy *k* elemű vektor. Így a bizonyító fél egy lépésben küldi vissza *k* db *t, c* és *s* értékeket. Ezzel felgyorsítjuk az azonosítást, csak a bizonyítható, hogy a protokoll nulla-ismeretű volta viszont megszűnik, ugyanis *V* hatékonyan nem tudja szimulálni a protokoll üzeneteit *P* részvétele nélkül. Következésképpen ugyanazon titkos és nyilvános kulcs mellett a *k* db érték kiszámítása nem ajánlott.

Viszont egy másik megoldás, ha *k* db titkos-nyilvános kulcspárt generál az identitását bizonyító fél. A párhuzamosított verziója a Fiat-Shamir protokollnak:

1. *P* generál egy véletlen számot: $r \in_R \mathbf{Z}_n^*$, ahol $\in_R$ jelöli, hogy véletlenül választunk. *P* kiszámítja $t \equiv r^2 \pmod n$ és elküldi *t*-t *V*-nek.
2. *V* generál *k* db véletlen bitet: $c_i \in_R \{0,1\}$, ahol $i=1,..,k$. *V* elküldi c_i-t $(i=1,..,k)$ *P*-nek.
3. *P* kiszámítja $s \equiv r \prod_{i=1}^{k} x_i^{c_i} \pmod n$ számot, és elküldi *V*-nek.

V ellenőrzi, hogy az $s^2 \equiv t\prod_{i=1}^{k} y_i^{e_i} \pmod n$ kongruencia teljesül–e. Így a protokoll egyszer fut le, csak *k* db értékkel, és annak valószínűsége, hogy a támadó sikeresen tudja *P*-t megszemélyesíteni $1/2^k$.

A Fiat–Shamir protokollnak többféle verziója is van. Az eddig ismertetett verzió esetén a nyilvános kulcs érvényességét tanúsítvány igazolja, tehát valamely nyilvános kulcsokat tartalmazó könyvtárból nyeri az ellenőrző fél. Egy másik lehetséges megvalósítás, egy megbízható kulcskiosztó központ alkalmazása, mely minden igénylőnek ad egy vagy több titkos kulcsot, mely függ az igénylő biometrikus adataitól. Például *P* személyesen felkeresi a központot, ahol igazolja személyazonosságát és ujjlenyomata és/vagy más biometrikus adatot vesznek tőle. Jelöljük *I*-vel azt az egyedi bitsorozatot, mely alapján *P* egyértelműen beazonosítható. A megbízható központ választ j_i $(i=1,..,k)$ értékeket úgy, hogy $f(I,j_i)=y_i$ kvadratikus maradék legyen, ahol *f* nyilvános függvény. A központ kiszámítja a titkos kulcsot és *P* smart kártyájára írja. A kártya lehetővé teszi *P*-hez tartozó *I* és j_i értékek ellenőrzését, *V* ellenőrző fél a nyilvános *f* függvény segítségével kiszámítja az $f(I,j_i)=y_i$ nyilvános kulcsot. Majd a protokoll lépései ugyanúgy hajtódnak végre.

A kriptográfiai primitíveket általában nem önmagukban alkalmazzuk, hanem valamely komplex kriptográfiai protokoll részei. A gyakorlatban sok olyan alkalmazás van, mely magas szintű biztonságot követel meg, így a kriptográfiai rendszerek használata elengedhetetlen. Ilyen protokollok például az elektronikus fizetési vagy szavazó protokollok. A következőkben egy szavazórendszert mutatunk be.

10.2 Az észt szavazórendszer

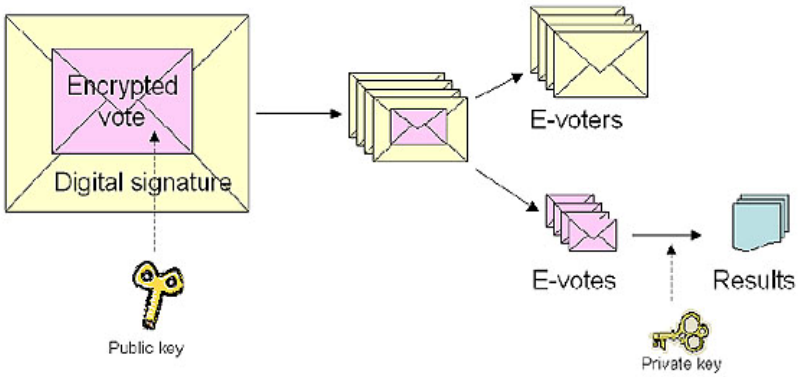
Az eddig ismertetett kriptográfiai primitívek egy alkalmazását ismertetjük, az EstEVS (Estonian E-Voting System) rendszert, azaz az észt szavazórendszert [22]. Az itt ismertetett rendszert a 2007-ben megtartott parlamenti választások során alkalmazták Észtországban.

Az elektronikus szavazási séma hasonlít a boríték módszerhez, melyet a kihelyezett (a szavazó otthonában végzett) szavazás esetén alkalmaznak:

- A szavazó azonosítja magát a szavazó bizottság tagjainak
- A szavazó kitölti a szavazócédulát, majd egy belső borítékba teszi
- A boríték egy másik, külső borítékba kerül, melyen a szavazó adatai szerepelnek
- A borítékot elviszik a szavazó helyiségbe, ahol ellenőrzik a szavazó jogosultságát, majd a külső boríték kinyitása után az anonim boríték az urnába kerül

Boríték módszer

A szavazó protokoll a fenti ötletet követi, a belső boríték a titkosított szavazatnak felel meg, a külső boríték egy digitális aláírásnak. A következő ábra szemlélteti a folyamatot.



10.1 ábra – Boríték módszer

EstEVS

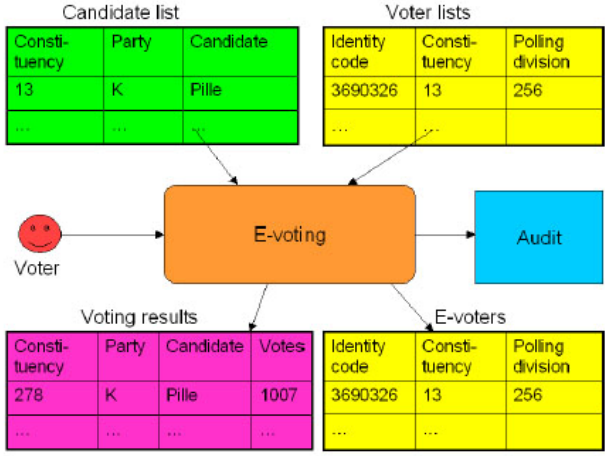
Az elektronikus szavazó rendszer input információi:

- szavazók listája
- jelöltek listája
- szavazatok

A rendszer output információi:

- szavazatok összesített eredménye
- e-szavazást igénybevettek listája

A 10.2 ábra illusztrálja a rendszer input és output információit.



10.2 ábra – EstEVS input és output információi

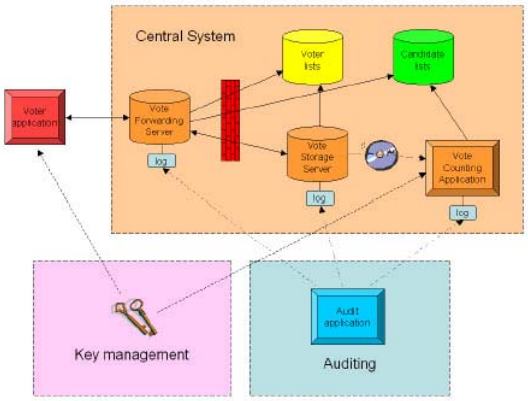
A séma résztvevői:

- *szavazó alkalmazás (Voter application)* – e-szavazó, PC-jével együtt. Egy titkosított, majd digitálisan aláírt szavazatot generál, melyet a központi szavazórendszernek küld el.
- *központi szavazórendszer (Central System)* – Nemzeti Választási Bizottság felügyelete alatti rendszer, mely megkapja és feldolgozza a szavazatokat.

- *kulcs menedzser (Key management)* – A rendszer kulcspárjait generálja le, a nyilvános kulcs a szavazó alkalmazásba, a titkos a szavazat-számláló alkalmazásba van integrálva. A kulcspár egy biztonságos hardver modul segítségével generálódik úgy, hogy a titkos kulcs sosem hagyja el az eszközt. A hardver modulhoz több résztvevő egyidejű hozzáférése szükséges ahhoz, hogy bármilyen biztonságos műveletet végrehajtson (dekódolás, kulcsgenerálás). A hozzáférő résztvevők beazonosítása kétfázisú módszerrel (kártya+PIN-kód) történik.
- *ellenőrző alkalmazás (Audit application)* – reklamációk kezelése (log állományok alapján)

A központi szavazórendszer elemei:

- *szavazat-továbbító szerver (Vote Forwarding Server, VFS)* – beazonosítja a szavazót tanúsítványa alapján, kiírja a szavazónak a választható jelöltek információit, majd megkapja a titkosított és aláírt szavazatot. A kapott e-szavazócédulát továbbítja a szavazat-tároló szervernek és a szervertől kapott visszaigazolást elküldi a szavazónak.
- *szavazat-tároló szerver (Vote Storage Server, VSS)* – a szavazat-továbbító szervertől kapott e-szavazócédulát tárolja, a szavazó fázis után a dupla szavazatokat törli, feldolgozza a szavazat törlési igényeket, törli a szavazásra nem jogosult személyek által elküldött szavazatokat. Végül elválasztja a külső és belső 'borítékokat', és a titkosított szavazatokat továbbküldi a szavazat-számláló alkalmazásnak.
- *szavazat-számláló alkalmazás (Vote Counting Application, VCA)* – Offline alkalmazás, a rendszer titkos kulcsával visszafejti a szavazatokat, majd összesíti, és nyilvánosságra hozza az eredményt.



10.3 ábra – EstEVS központi szavazó rendszerének elemei

A séma alapvetően két nagy fázisból áll, a szavazó fázis és az összeszámláló fázisból. A szavazó fázis lépései a következők:

1. A szavazó HTTPS protokollon keresztül tanúsítványával beazonosítja magát a VFS-nek.
2. VFS a szavazó azonosítója alapján ellenőrzi annak jogosultságát és választókerületét. Ha nem jogosult, akkor hibaüzenetet küld a szavazónak.
3. VFS megkérdezi VSS-t, hogy a szavazó szavazott–e már, ha igen, akkor a szavazót informálja erről.

4. VFS lekérdezi a jelöltek adatbázisából a megfelelő körzethez tartozó jelöltek listáját és megmutatja a szavazónak.
5. A szavazó választ egy jelöltet.
6. Az alkalmazás titkosítja a választott jelöltet és egy véletlen értéket a VCA nyilvános kulcsával. A szavazó aláírja a titkosított üzenetet.
7. A szavazó alkalmazás továbbítja az aláírt, titkosított szavazatot a VFS-nek, aki formálisan ellenőrzi a kapott információt, illetve hogy ugyanaz a személy küldte-e az adott borítékot, mint aki azonosította magát.
8. VFS továbbítja az e-szavazatot VSS-nek, aki az aláírás hitelességét bizonyító tanúsítványt a szavazathoz csatolja. Érvényes aláírás esetén visszaigazolást küld a VFS-nek, ami továbbítódik a szavazónak is.
9. A szavazat beérkezéséről bejegyzés kerül egy log állományba (személy azonosító, hash(szavazat)) formájába.
10. A szavazó többször is szavazhat.

Összeszámláló fázis:

1. VSS megszünteti a többszörös szavazatokat, csak az időben utoljára elküldött szavazatokat veszi figyelembe. A digitális aláírás érvényességi idejének összhangban kell lennie a szavazat leadási idejével. Minden törölt szavazatról bejegyzés kerül egy log állományba.
2. Az aláírás leválasztása után a titkosított szavazatok egy külső tároló egységre kerülnek (pl. CD lemez), mely a VCA-hoz jut el.
3. VCA dekódolja a szavazatokat és ellenőrzi a szavazatok helyességét.
4. Szavazókerületenként összeszámlálja a szavazatokat és nyilvánosságra hozza.

Előnyök. A séma egyszerű, könnyen érthető és követhető. Párhuzamosságot lehet vonni a hagyományos kihelyezett szavazás folyamatával. A résztvevők száma minimális.

Hátrányok. Egy résztvevő sem rendelkezhet egyidejűleg a digitálisan aláírt szavazattal és a rendszer titkos kulcsával. Biztosítani kell a titkos kulcs megfelelő hozzáférési szabályait. Anonimitást procedúra során egy megbízható fél, a szavazat-tároló szerver biztosítja, aki szétválasztja a szavazó aláírását a titkosított szavazattól.

További jellemzők:

- Az azonosítás a szavazó digitális aláírásának ellenőrzésével történik meg.
- Vizsgálja a szavazók jogosultságát.
- Minden választó csak egyszer szavazhat.
- A szavazatok titkosak.
- Anonim.
- Az ellenőrizhetőség a log állományokon keresztül van biztosítva.

11 Nyilvános kulcs infrastruktúra, hitelesítő szervezetek.

A fejezet a nyilvános kulcsú infrastruktúráról szól, amelyet angolul public key infrastructure-nak neveznek és általánosan használt rövidítése a PKI. Szó lesz arról, hogy hogyan épül fel a nyilvános kulcsú infrastruktúra, tárgyalásra kerülnek a különböző PKI modellek és persze az infrastruktúrát felépítő hitelesség szolgáltatók szerkezetéről, feladatairól és szervezeti egységeiről. Áttekintjük a jelenlegi jogi szabályozást, az elterjedtebb üzleti modelleket. Részletesen kifejtjük az egész infrastruktúra egyik legjellegzetesebb elemét, a tanúsítványokat szabályozó X509 szabványt. Az X509 tanúsítvány az az elem amivel a PKI kapcsolódik a nyilvános kulcsú kriptográfiához és gyakorlatilag tükröződik rajta a teljes infrastruktúra felépítése, lehetőségei, korlátai és fejlődése. Külön alfejezetet szentelünk a PKI gyakorlati alkalmazásainak, azoknak a környezeteknek ahol sikerrel alkalmazható a nyilvános kulcsú infrastruktúra. Ebben a bevezető részben mindazokról az általános tulajdonságokról lesz szó, amelyek a PKI-t jellemzik illetve azokról az információkról, amiket gazdasági döntéshozóként vagy akár végfelhasználóként tudni érdemes a nyilvános kulcsú infrastruktúráról. Habár a PKI fogalma és lehetőségei a publikus kulcsú kriptográfiával egy időben, több mint 20 évvel ezelőtt jelent meg, a gyakorlati megoldások csak napjainkban kezdenek széles körben elterjedni. A nyilvános kulcsú infrastruktúra erős azonosítást tesz lehetővé olyan feltételek mellett amely sok szervezet számára vonzóvá teszi az alkalmazását. Azonban a nyilvános kulcsú infrastruktúra üzembe helyezéséhez és üzemeltetéséhez a PKI-hez értő személyzetre is szükség van. Ez nem csak pár alapvető PKI tudással rendelkező adminisztrátort jelent, hanem magasan képzett szakembereket is, akik a rendszer megtervezését, a működési szabályzatot, az ezzel kapcsolatos dokumentumokat, a szabályzatok leképezését elkészítik, illetve elvégzik továbbá az esetleges kereszthitelesítéssel kapcsolatos döntéseket meghozzák és a folyamatot megtervezik illetve levezénylik.

A PKI gyors elterjedésének útjában áll, hogy nem áll rendelkezésre ezen munkakörök számára elegendő szakképzett munkaerő. A fejezet célja többek között, hogy a fent említett feladatokból ízelítőt adjon és a leendő PKI adminisztrátoroknak és tanácsadóknak egy alapos bevezetést jelentsen. A PKI terjedésének egy másik akadálya, hogy a PKI használatához nem elegendő egy nyilvános kulcsú infrastruktúrát üzemeltetni, természetesen szükség van a PKI-t használó alkalmazásokra is. A PKI képes alkalmazások száma egyre nő, de például az egyik mintapélda, a böngésző szerver architektúra is csupán részben tekinthető PKI képesnek, hiszen nem közvetlenül, hanem csupán egy alsóbb hálózati rétegben használják azt. Egy alkalmazás PKI képessé tételéhez szükség van az olyan alapvető műveletek implementációja mellett, mint például a digitális aláírás, az aláírás ellenőrzése, publikus kulcsú kódolás és dekódolás, a tanúsítványok kezelésére is. A fejezetben lévő anyag hasznos tudást jelent mindazon programozók és leendő programozók számára akik PKI képes alkalmazás fejlesztésével, vagy már meglévő alkalmazás PKI képessé tételével találkoznak pályájuk során. Szintén lassítja a PKI elterjedését a nyilvános kulcsú infrastruktúrát körülvevő vállalati szintű bizalmatlanság és értetlenség. A fejezet célja az is, hogy segítséget nyújtson

mindazoknak az (aktív vagy leendő) döntéshozóknak illetve asszisztenseknek, akik munkájuk során kapcsolatba kerülnek a nyilvános kulcsú infrastruktúrával.

11.1 Bevezetés

A PKI szerepe, hogy egy nevet kössön valamely kulcspárhoz a nyilvános kulcsú kriptográfiában. A nyilvános kulcsú infrastruktúra, mint ahogy a neve is jelzi egy olyan kiépített rendszer amit a felhasználók igénybe vehetnek. Az infrastruktúra szolgáltatói, a hitelességszolgáltatók biztosítják ezt a lehetőséget a felhasználók, fogyasztók számára, mint ahogy az áramszolgáltatók elektromos energiát biztosítanak az ügyfeleik részére.

11.1.1 Nevek

A nyilvános kulcsú infrastruktúra neveket köt az aszimmetrikus kriptográfiában alkalmazott algoritmusok és protokollok kulcspárjaihoz, ezért először a nevek fogalmát tisztázzuk és az ezzel kapcsolatos fogalmakat definiáljuk.

A név tágabb értelemben valamilyen adat, vagy adatok együttese, amik az adott környezetben a többi szereplőtől megkülönböztetik, egyértelműen azonosítják a név birtokosát. A név az adott szereplő azonosítója. Az hogy az adott környezetben az egyértelmű azonosításhoz mennyi és milyen jellegű információ (mekkora adatmennyiség) szükséges vagy áll rendelkezésre ilyen célra, nagyban függ az adott környezet jellegétől. Az alkalmazott név nagyban függ a környezet méretétől (például család, osztály, város) illetve annak jellegétől is (más formátumú elnevezéseket használnak például egy operációs rendszeren, mást egy kormányzati adminisztrációs rendszerben és az e-mailezésnél), és a méret növekedésével az elnevezések egyediségének biztosítása is egyre nagyobb problémákat vet fel.

Mint ahogy azt a későbbiekben látni fogjuk a fentiek alapján a név (nym) többféle dolgot takarhat a valós világbeli szereplőhöz való kapcsolatától függően. Lehet valamilyen azonosító, amelyet olyan szándékkal rendeltek az adott szereplőhöz, hogy a név és a valós világbeli tulajdonosa közötti kapcsolat ne legyen nyilvánvaló (pseudonym - álnév). Jelentheti továbbá az azonosító vagy név teljes hiányát (anonym - névtelen) illetve jelentheti a legkonvencionálisabb értelemben vett nevet, azaz olyan adatok összességét, amelyek alapján a tulajdonosa valós világbeli identitása egyértelműen meghatározható (veronym - igaz név). Nagyon gyakran a PKI-vel az a szándékunk, hogy az adott kulcsot valamely valós világbeli objektum identitásához, igaz nevéhez kössük, valójában azonban a PKI a kulcspárok bármilyen névhez való kötésére ad lehetőséget.

Ezek alapján megkülönböztethetjük az azonosítást (identification) a hitelesítéstől (authentication). A hitelesítés esetében a partner valamilyen tágabb értelemben vett nevéről, a környezetében lévő többi szereplőtől egyértelműen megkülönböztető azonosító információról van szó. Az azonosítás ezzel szemben valamilyen valós világbeli eszközhöz vagy személyhez köti a szereplőt, akivel kommunikálni szándékozunk. Például ha a hitelesítés lépése során a név szerepét igaz név játssza, akkor az egyben azonosítás is.

A nevek és azonosítók használata felveti az egyedi elnevezés illetve azonosító kiosztás problémáját. Ezt az alkalmazások különböző területein különféleképpen oldják meg, az egyediséget más és másféleképpen biztosítják. A domain nevek esetében az egyes domain név regisztrátorok felelnek a felügyeletük alá tartozó domaineken az egyediség biztosításáért. Az e-mail címek esetében a vonatkozó domain kezelője hivatott a nevek egyediségét biztosítani. A gépek ip címét az adott alhálózatot felügyelő rendszergazda vagy a címeket kiosztó szerver felügyeli. Az egyes környezetekben az azonosítók a legkülönfélébbek lehetnek. A tanúsítványokról szóló fejezetben említett X500 Directory szintén bevezetett egy egyedi elnevezési konvenciót ám ez nem terjedt el, csupán a tanúsítványok DN mezőiben láthatjuk nyomait. A PKIX szabvány pedig egyenesen lehetővé teszi ezeknek a mezőknek a kihagyását. Az egyedi azonosítók, nevek kiosztása nem a PKI feladata. A nyilvános kulcsú infrastruktúra csupán összeköti az adott nevet valamely kulcspárral.

11.1.2 Felhatalmazás

Az azonosítással kapcsolatban gyakorta felmerülő fogalom a felhatalmazás (authorisation). A rendszerben az egyes szereplőknek különböző hatásköreik lehetnek, és egy tevékenységet, amit az egyik szereplő elvégezhet, nem feltétlenül megengedett egy másik szereplőnek. Az informatikai biztonsági rendszereknek az egyes szereplők jogköreire vonatkozó döntéseket is meg kell hozniuk és az ezzel kapcsolatosan felmerülő problémákat is kezelniük kell. Habár a PKI lehetővé teszi felhatalmazásra vonatkozó információk továbbítását ez a gyakorlat nem ajánlott. A nyilvános kulcsú infrastruktúra csupán összeköti az adott nevet valamely kulcspárral: a PKI önmagában nem ad választ a fenti kérdésre, a felhatalmazás jogosságára való vizsgálatokat a rendszernek más eszközökkel kell elvégeznie.

11.1.3 Bizalom

Mint ahogy arról majd a későbbi részfejezetekben is szó lesz, a PKI segítségével a bizalmat adhatjuk tovább és vezethetjük le a bizalmi horgonyoktól a végfelhasználókig. A bizalmat mint olyat szintén többféleképp értelmezhetjük. Tágabban vett értelemben a bizalom jelentheti magába a végfelhasználóba vetett bizalmat, valamiféle elvárást a protokoll során való viselkedésével kapcsolatban. A szűkebb értelemben vett bizalom egy adott kulcsba vetett bizalom, azaz annak az elfogadása, hogy az adott kulcs párja valóban annak a birtokában van akivel a tranzakciót végrehajtani szándékozzuk. A különbség a kétfajta bizalmi felfogás között nagyon fontos. A PKI csupán a szűkebb értelemben vett bizalom továbbítására szolgál, semmilyen biztosítékkal nem szolgál a tágabb értelemben vett bizalmat illetően. A nyilvános kulcsú infrastruktúra alkalmazása során a kezdetben egyetlen szereplőbe, a bizalmi horgonyba vetett bizalom vándorol a végfelhasználókig. (Ez egy leegyszerűsített szemléletmód, a bizalmi modellekről szóló részben ismertetésre kerülnek olyan modellek is ahol nem egyetlen pontból indul ki a bizalom) A nyilvános kulcsú infrastruktúra semmit nem mond a szereplők megbízhatóságáról, kizárólag az egyes kulcsokba vetett bizalmat hivatott megalapozni.

11.1.4 Biztonság

A PKI csupán az azonosítást segíti elő. Önmagában nem teszi varázslatos módon biztonságossá az egyes informatikai rendszereket, nem foltozza be a szoftverhibákat, biztonsági réseket, nem fogja megoldani a túlterheléses támadások problémáját és elfedni a rosszul konfigurált szoftverekből adódó gyengeségeket, nem véd meg a vírusoktól, trójaiaktól, férgektől. Továbbra is szükség van a biztonsági rendszer körültekintő megtervezésére, tűzfalakra, behatolás észlelő rendszerekre, és úgy általában mindarra, amiről a korábbi fejezetekben szó volt. Minden esetben egy a már ismertetett költség-hatékonyság elemzés illetve kockázatkezelés szükséges annak eldöntésére, hogy az azonosítást elősegítésére egy PKI rendszert állítanak üzembe, vagy inkább megpróbálkoznak a hagyományos, szimmetrikus kulcsú technikákkal. A nyilvános kulcsú infrastruktúra nem helyettesíti az egész biztonsági rendszert, csupán részét képezi annak.

11.1.5 Megbízhatóság

A PKI sem küszöböli ki a rendszeradminisztrátorok, illetve a felhasználók gondatlanságát, nemtörődömségét, lustaságát vagy éppen hiányos képzettségét. A nyilvános kulcsú infrastruktúra nem küszöböli ki az emberi hibákat és tehetetlen a rosszindulat ellen is. A PKI által hitelesített egyén cselekedhet nem megfelelően, visszaélhet az azonosságához kapcsolt hatáskörrel. A nyilvános kulcsú infrastruktúra csupán a kulcsba vetett bizalmat alapozza meg, a kulcshoz tartozó név mögött lévő szereplő viselkedésével, gondosságával, szándékaival kapcsolatban semmilyen biztosítékot nem ad.

11.1.6 A PKI előnyei

A nyilvános kulcsú infrastruktúrának több előnye is van a hagyományos titkos kulcsú rendszerekkel szemben. Az első legszembeszökőbb előny, hogy lehetővé teszi az egyszeri bejelentkezést (Single Sign On). Nevezetesen, hogy a felhasználók a rendszer vagy éppen több rendszer használata során csupán egyetlen alkalommal jelentkezzenek be a biztonságos használat fenntartása mellett. Rendszerint ez azt jelenti, hogy a felhasználó PKI segítségével kerül azonosításra, az egyetlen jelszó amit meg kell jegyeznie a szimmetrikus titkosítással kódolt privát kulcs jelszava ahol rendszerint a jelszó egy hash értéke a tényleges titkos kulcs. (Emlékeztetőül: a privát kulcsot az aszimmetrikus kriptográfiában használjuk és a publikus kulcs párja, a titkos kulcs kifejezés pedig a szimmetrikus kriptográfiában használatos kulcsra vonatkozik) Ez jelentősen redukálja a rendszerben használatos jelszavak számát, megkönnyítve annak megjegyzését ezáltal könnyítve a felhasználók terhein illetve csökkentve annak a kockázatát, hogy a jelszavakat megjegyezni nem tudván papírlapokra jegyezzék fel amiket esetleg még a monitorra is kiragasztanak. A szervezet szempontjából ez a tulajdonság egy további előnyt is magában hordoz, hiszen a kevesebb jelszó nem csak a felhasználói oldalon csökkenti az adminisztratív terheket.

A másik nagy előny, hogy a PKI használata közben jóval kisebb mennyiségű adatot kell titokban tartani, mint a szimmetrikus megoldások esetében. A teljes rendszer

biztonságának a szempontjából csupán két kritikus adat van és ezek a bizalmi horgony aláíró kulcs és a visszahívási listájának az aláíró kulcsa. Az egyes felhasználók és erőforrások a rendszeren belül felelősek a bizalmi horgony tanúsítványának tárolásáért, de ezeknek a kompromittálódása csupán az infrastruktúra érintett részére jelent veszélyt és nem az egész rendszerre nézve, másrészt ezeknek csupán az integritását kell biztosítani és nem a titkosságát, ami jóval könnyebb feladat.

A PKI lehetővé teszi, hogy a privát kulcsokat való világbeli entitásokhoz kössük, ezáltal lehetőséget biztosít a rendszer szereplőinek azonosítására, kizárva ezzel a rendszer elemeinek megszemélyesítését illetve a közbülső ember (man-in-the-middle) típusú támadások lehetőségét. Ez természetesen mindaddig érvényes, amíg a rendszer szereplői titokban tartják a privát kulcsukat. Ezt a feladatot megkönnyíti, hogy a privát kulcsok sohasem közlekednek a hálózaton keresztül a tranzakciók során, és mint ahogy a publikus kulcsú titkosításról szóló részben erről már szó volt, a publikus kulcsból a privát kulcs kiszámítása praktikusan lehetetlen.

Mindemellett, habár közvetlenül nem ajánlott a felhatalmazások (authorisation) kezelésére, megvalósítására használni (annak ellenére, hogy mint majd azt látni fogjuk, a szabványok lehetőséget nyújtanak rá), lehetőséget nyújt az aláírások révén a rendszer szereplőinek valamely felhatalmazásokat kezelő rendszerhez való kötésére. Íly módon megvalósítható például a végfelhasználók egy SAML szolgáltatóhoz való kötése, abban kiosztott attribútumaik igazolása.

11.1.7 Tanúsítványok a gyakorlatban

A nyilvános kulcsú infrastruktúra termékei és eszközei a tanúsítványok. A tanúsítványok kapcsolják össze a valós világbeli entitások igaz neveit az egyes kulcspárokkal. A tanúsítványok természetüknél fogva nem alkalmasak közvetlen emberi felhasználásra, nem is arra tervezték őket, mégis elképzelhetőek olyan esetek, amikor a végfelhasználónak közvetlenül kell tanúsítványokkal kapcsolatos döntéseket hoznia. Az ilyen esetek több oknál fogva sem szerencsések: egyrészt, mint ahogy az már említésre került a tanúsítványok formátuma és az egyes adatok jellege kifejezetten nehézkessé teszi az emberi feldolgozást, másrészt a végfelhasználók előképzettsége kevés kivételtől eltekintve elégtelen az ilyen jellegű döntések meghozatalára.

Az egyik ilyen tipikus eset a nyilvános kulcsú titkosítással titkosított, esetleg digitális aláírással ellátott levelezés. Ebben az esetben, ha a szoftver nem tudja a PKI-n keresztül megalapozni a szóban forgó kulcsba vetett bizalmat, akkor a döntést a felhasználóra bízza. Szerencsésebb esetekben a levél megjelenítése után teszi fel a kérdést. Ez esetben a felhasználó a levél tartalma alapján esetleg tud következtetni az levélíró valódi identitására. Ez az eset természetesen teret nyit a social engineering típusú támadásoknak. Ha a szoftver a döntést a levél megjelenítése előtt várja, akkor a helyzet még rosszabb: a felhasználónak semmilyen plusz információja sincs ami alapján a döntést meghozhatná, ilyenkor az átlagos felhasználó az OK gombra klikkel és ezáltal véglegesen hozzáadja a potenciálisan hamis tanúsítványt a megbízható tanúsítványok listájához.

A másik eset ennél is gyakoribb és veszélyesebb. Az online fizetési tranzakciók és a személyes adatokat is érintő kérdőívek rendszerint PKI alapú technológiával közvetítik az adatokat (https). Ebből az átlagos felhasználó csak a böngésző sarkában található kis lakat ikont veszi észre. Ha nem ellenőrizhető tanúsítvánnyal találkozik, a böngésző szintén a felhasználóra hárítja a döntést, amit a felhasználó a megfelelő információk vagy képzés híján rendszerint az OK gomb lenyomásával elfogadja a potenciálisan rosszindulatú tanúsítványt. Minden olyan honlap, amely a böngészők által nem hitelesíthető tanúsítványt használ vagy csalás, vagy pedig potenciális veszélynek teszi ki a felhasználóit azzal, hogy a megbízhatatlan tanúsítványok elfogadására szokatja a látogatóit.

Röviden összefoglalva: a PKI az azonosításban játszik szerepet, egy nevet köt egy kulcspárhoz. Ennyit biztosít. Nem többet, nem kevesebbet.

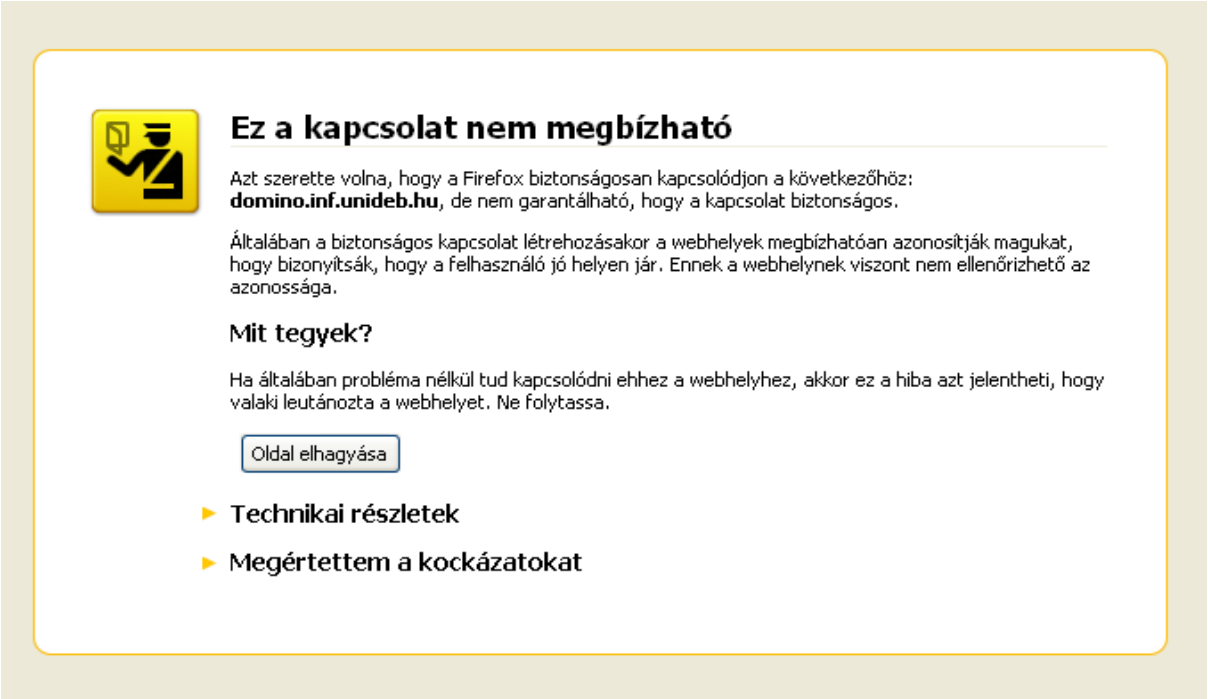
11.2 A nyilvános kulcs infrastruktúra alkalmazásai

Az aszimmetrikus kriptográfia lehetőségeire informatikai eszközök és megoldások széles rendszere épül. A nyilvános kulcsú protokollok megbízhatósága komoly matematikai háttéren alapszik és a gyakorlati megvalósítás és alkalmazás során sem szenvedett csorbát. Az aszimmetrikus kriptográfia széles eszköztára lehetővé teszi olyan biztonsági szempontból kényes ügyletek elektronikus lebonyolítását, amelyek mindez idáig csak papíron voltak lehetségesek, és amik legtöbb esetben a felek saját kezű aláírását is megkövetelték. A kriptográfiai algoritmusaink, protokolljaink azonban csak azt teszik lehetővé, hogy a résztvevő felek birtokában lévő kulcsokat azonosíthassuk. Ezt a hiányosságot hivatott orvosolni a nyilvános kulcs infrastruktúra (public key infrastructure, PKI), amely lehetővé teszi, hogy az egyes kulcsok birtokosainak személyazonosságáról is megbizonyosodhassunk. A PKI az ami lehetővé teszi ezen alkalmazások számára a megbízható működést. Mint ahogy az elektromos hálózat szolgáltatja az áramot villamos készülékeink számára, úgy biztosítja a PKI a kulcsok birtokosainak személyazonosságát a nyilvános kulcsú alkalmazásaink számára.

A PKI egyik legelső alkalmazásával gyakran találkozhatunk webes környezetben: minden alkalommal, amikor a „biztonságos kapcsolat”-ként emlegetett lehetőséget választjuk, egy ilyen alkalmazással szembesülünk. Ez teszi lehetővé, hogy vásároljunk az interneten, hogy bankunk netbankár szolgáltatását igénybe véve, kényelmesen, otthonról intézhessük banki ügyeinket, és egyáltalában rendelkezésünkre áll, amikor érzékeny, személyes adatainkat akarjuk megadni valakinek az interneten keresztül és biztosak akarunk lenni abban, hogy a másik tényleg az akinek mondja magát. Valójában ez az úgynevezett „https” protokoll alkalmazását jelenti, ami tulajdonképpen annyit tesz, hogy az eredeti http protokollt különböző kriptográfiai módszerekkel biztonságos csomagokba zárjuk. Ezt a módszert széles körben alkalmazzák az informatika világában és gyakran veszünk igénybe olyan szolgáltatásokat, amikről észre sem vesszük, hogy a PKI is szerepet játszik a feladatok végrehajtásában. A PKI ezen alkalmazása egy ponton jelentősen eltér a későbbiekben sorra kerülő megoldásoktól. Nevezetesen, ebben az esetben a PKI -t nem jogi személyek, hanem

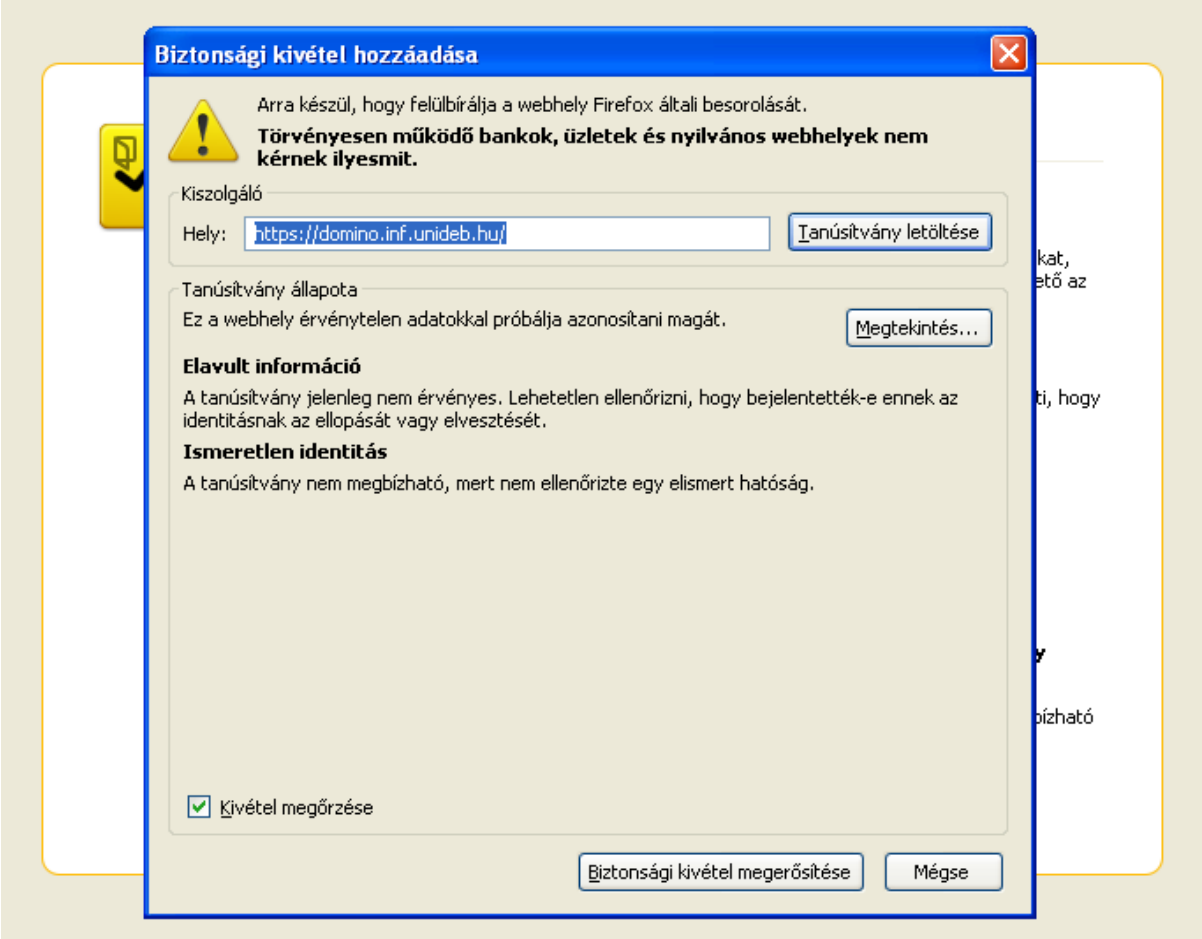
egyes, a világhálón megtalálható szerverek azonosítására használjuk.

Ha a tanúsítvány nem hitelesíthető, akkor a használt alkalmazás rendszerint figyelmezteti a felhasználót.



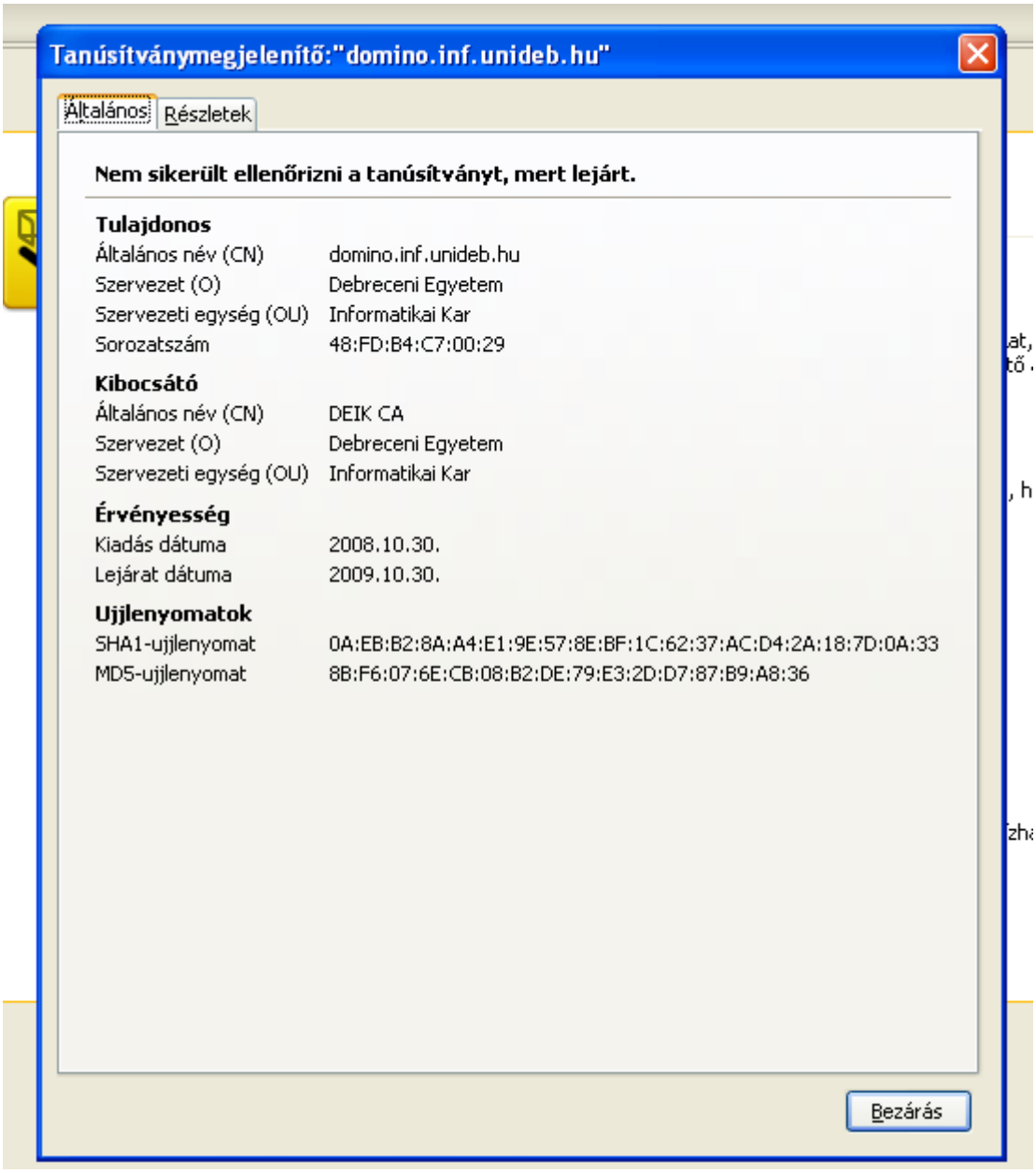
11.1 ábra A Mozilla Firefox figyelmeztetése

Legtöbb alkalmazás a tanúsítvány megtekintését is lehetővé teszi és felkínálja, hogy ha az adott tanúsítványt a felhasználó valamilyen okból a hitelesítési problémák ellenére is elfogadhatónak minősíti, akkor biztonsági kivételt adjon hozzá:

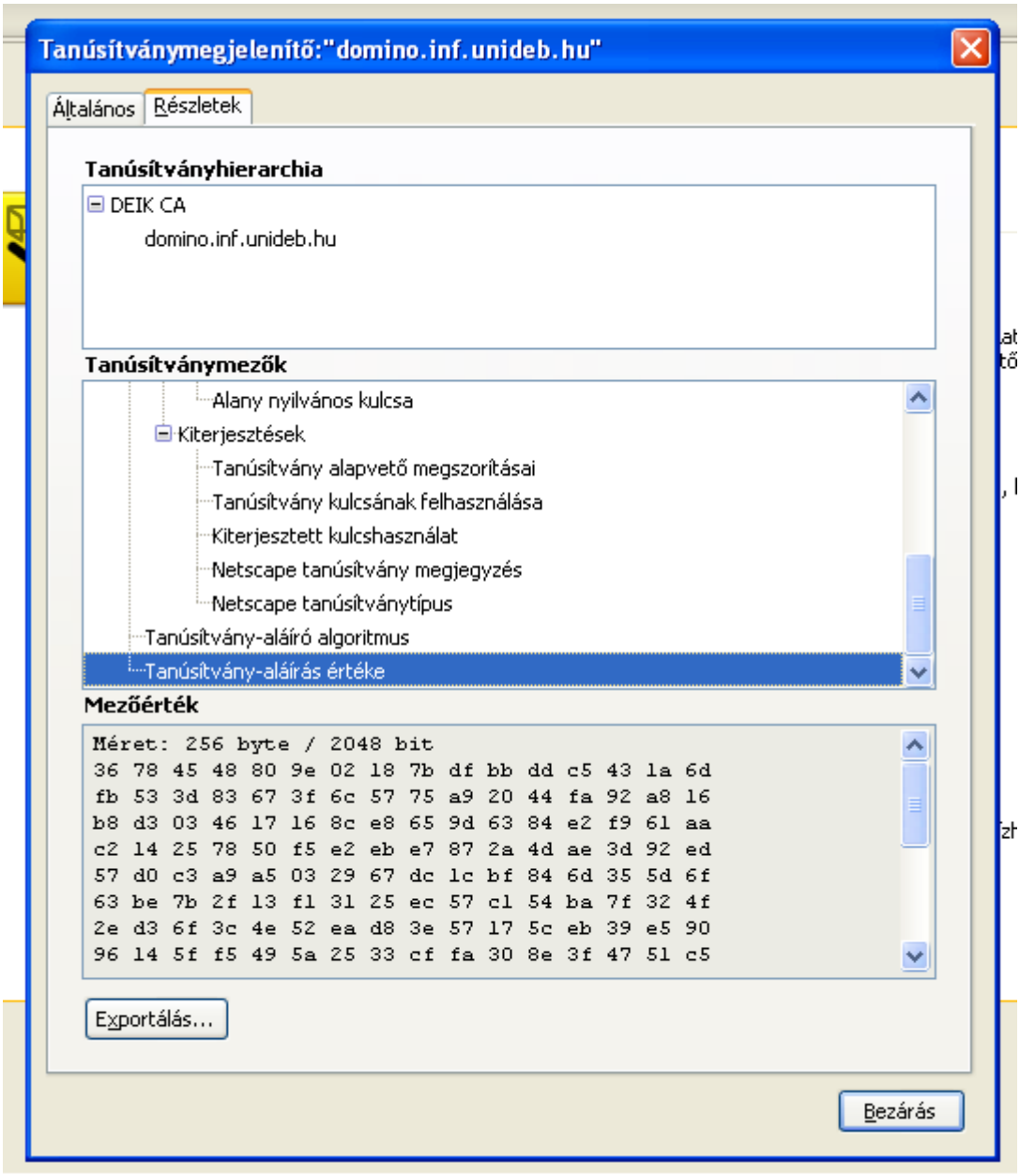


11.2 ábra A Mozilla Firefox felkínálja a biztonsági kivétel hozzáadását

Ahhoz, hogy a felhasználó döntsön az adott tanúsítvány megbízhatóságának megítéléséről, a felhasználónak szüksége van a tanúsítvány adataira is. Némely alkalmazás az alapvető információkon kívül a teljes tanúsítvány megtekintését lehetővé teszi.



11.3 ábra A Mozilla Firefox megjeleníti a tanúsítvány alapvető információit



11.4 ábra A Mozilla Firefox megjeleníti a teljes tanúsítványt

A PKI nem csak hálózati elemek és szereplők azonosítására használható. A PKI széleskörű alkalmazási lehetőségeit illusztrálják az alábbi példák:

- cégbejegyzési kérelmek beadása,
- a PSZÁF adatközlésekre és tőkepiaci közzétételre kötelezettek elektronikus közzétételei,
- a szakmai vizsgák szervezőinek elektronikus teljesítésű adatközlési kötelezettségei,
- szerződések elektronikus aláírása,
- elektronikus számlázás,
- elektronikus levelezés,
- elektronikus adóbevallás.

Ehhez természetesen elengedhetetlen a kérelmeket benyújtó személyazonosságának a meghatározása, ez, tekintettel a folyamat elektronikus jellegére minden esetben a PKI segítségével történik.

11.2.1 PKI képes szolgáltatások

A korábbi részekben áttekintettük a nyilvános kulcsú infrastruktúra felépítését és lehetőségeit. Ebben a részben azokról a szolgáltatásokról lesz szó, amelyek valamilyen módon a PKI segítségével is kivitelezhetőek.

11.2.1.1 Biztonságos kommunikáció

A biztonságos kommunikáció során két fél közötti adatátvitel történik, a korábbi fejezetekben már ismertetett feltételek: a hitelesség, az integritás és a bizalmasság megvalósítása mellett. Mint ahogy korábban arról már szó volt a PKI csupán a hitelesítést (ezen belül is első sorban az azonosítást) teszi lehetővé. A kommunikáció során az adatátvitelt már rendszerint hagyományos, titkos kulcsú technikákkal valósítják meg. Tipikus alkalmazása például a webes kommunikáció során az egyes weboldalakhoz való kapcsolódás. Ezt rendszerint valamilyen biztonságos protokollréteg alkalmazásával oldják meg, mint például a TLS vagy az SSL. Ezek a protokollok a hálózati protokollverem szállítási rétege felett működnek és így bármilyen felettük lévő rétegben tevékenykedő alkalmazás vagy protokoll számára biztosítani lehet általuk a biztonságos kommunikációt. Manapság bevett gyakorlat például, hogy a levelezőszerverekhez is ilyen módon védett kapcsolaton keresztül csatlakoznak a levelezőkliensek. A levelezést a PKI technikákkal ennél közvetlenebb módon is biztonságossá tehetjük: a levél tartalmát valamilyen biztonságos levelezési protokoll segítségével küldjük el, mint például az S/MIME valamelyik verziója. A vezeték nélküli hálózatoknál illetve némely vezetékes vállalati rendszereknél is széles körben alkalmazott virtuális privát hálózatok (VPN - Virtual Private Network) is felhasználhatják a nyilvános kulcsú infrastruktúra által nyújtott szolgáltatásokat.

11.2.1.2 Biztonságos időbélyegzés

A digitálisan aláírt szerződések esete mellett, több kriptográfiai protokoll vagy eljárás esetén is szükségünk lehet megbízható időbélyegek használatára. Az a megoldás, hogy minden egyes szereplő számára saját biztonságos órát biztosítsunk meglehetősen problémás és nehezen kivitelezhető. A nyilvános kulcsú kriptográfia használatával elegendő csupán néhány szereplőt ellátni hiteles és biztonságos órával és ezek, mint időbélyegző hatóságok szolgáltatják a hiteles időt mindenki más számára. Az időbélyegzés rendszerint az adatok valamilyen kriptográfiai hash értékének és a pontos időnek az időbélyegző hatóság által való aláírását jelenti.

11.2.1.3 Adathitelesítés (Notarization)

Valamely alkalmazásban valamely adat hitelesítése. Azaz a rendszer valamely szereplője tanúsítja, illetve kezeskedik azért, hogy a szóban forgó adatok, hitelesek, érvényesek, valamilyen szempontból korrektnek mondhatóak. Az itt említett hitelesség, érvényesség fogalma, az aktuális alkalmazástól és az adott környezettől függően bármit magában foglalhat. A korrektséget a rendszerben valamely megbízható, a közjegyző (notary) szerepét ellátó entitás ellenőrzi, és annak fennállásáról a saját digitális aláírásával nyilatkozik. Az ilyen jellegű alkalmazásoknál rendszerint szükség van hiteles időbélyegre is, hogy bizonyítani lehessen, hogy az adatok azon korrekt formában legalább mióta léteznek.

11.2.1.4 Letagadhatatlanság

Ez a szolgáltatás a rendszerek emberi komponensének őszinteségét hivatott szavatolni, elősegíteni illetve ellenőrizni. A probléma természeténél fogva igen komplex és a technológia csupán elősegíti illetve lehetőséget ad az ellenőrzésre bizonyos esetekben, de az emberi tényező viselkedését természetesen nem tudja meghatározni. A letagadhatatlanság több célt szolgálhat: beszélhetünk az eredet, a kézhezvétel, az elkészítés, leszállítás illetve hozzájárulás letagadhatatlanságáról.

A szolgáltatáshoz szükség van a biztonságos és hiteles tárolás lehetőségére, hiszen ahhoz, hogy valamely tett letagadhatatlanságát igazoljuk egy későbbi időpontban, a céltól és a körülményektől függően több különböző hiteles adatra is szükség lehet. Ezek az adatok jellemzően az adott akcióra vonatkozó időbélyegzett tanúsítványok visszavonási listák és tanúsítványláncok lehetnek, de a konkrét esettől függően tetszőleges egyéb hiteles adatot is letárolhatunk a bizonyíthatóság érdekében.

A technológia azt teszi lehetővé, hogy vitás helyzetben az egyik fél hanyagságát vagy rosszindulatú magatartását bizonyítsuk. Nevezetesen, hogy vagy valaki a tudtán kívül megszerezte a titkos kulcsát, vagy pedig hazudik az adott eseménnyel kapcsolatban. Ahhoz, hogy a technológia által nyújtott bizonyítékokat kiértékeljük, legtöbb esetben emberi közreműködésre van szükség (például mi a helyzet abban az esetben, ha a kérdéses kulcs tulajdonosa bizonyítani tudja, hogy csak a szóban forgó eset után jött rá, hogy már előtte ellopták a titkos kulcsát, esetleg azt is alá tudja támasztani, hogy mindez annak ellenére történt, hogy betartotta az előírásokat).

11.2.1.5 Jogosultságkezelés

A nyilvános kulcsú infrastruktúra önmagában csupán a hitelesítésért felelős, semmit nem mond az egyes résztvevők jogosultságairól. Jogosultságok alatt azon szabályok összességét, azt az eljárásrendet értjük, amely meghatározza, hogy az adott szereplő a rendszer mely elemeihez, mely részeihez férhet hozzá, és azokkal milyen műveleteket végezhet. A szóban forgó jogosultságok a fájlok egyszerű olvasási írási és futtatási jogosultságaitól kezdve az egyes alhálózatokhoz való hozzáférésen át akár a pénzügyi műveletekre való felhatalmazásig illetve az ezekre vonatkozó korlátokig terjedhetnek. A

jogosultságkezelés (privilige management, authorisation) feladata ezen jogok entitásokhoz rendelése és betartatása. A jogosultságkezeléshez az esetek többségében az egyedek előzetes hitelesítésére van szükség. A PKI kapcsolt jogosultságkezelés alkalmazása esetén a nyilvános kulcsú infrastruktúra végzi az entitások hitelesítését.

Ahol erre lehetőség van a jogosultságokat elegendő lehet letárolni valamely védett helyen. Erre azonban csak a legritkább esetben van lehetőség és a jogosultságok kezeléséhez és a hálózaton való biztonságos továbbításához erős kriptográfiára van szükség. A bonyolultabb üzleti környezetekben ráadásul a szükséges jogosultságok és a szabályozások rendszere is nagy összetettséget érhet el. Ilyen esetekben a jogosultságokat egy vagy több felhatalmazó szervezet kezeli és tartatja be. A több felhatalmazó hatóságot tartalmazó rendszerek esetében a szervezetek az egy konkrét esetre vonatkozó jogosultságokat szükség szerint egymás között kommunikálva határozzák meg a saját szabályozásaik alapján.

A rendszerek működése során szükség lehet a jogok átruházására is. Ennek megvalósítására alkalmazható a vak vagy a nemvak delegálás. Vak delegálás esetén a felhatalmazó hatóság számára rejtve marad, hogy az adott jogosultság delegálásra került illetve az is, hogy ki ruházta át a szóban forgó jogokat. A nemvak vagy más néven a vizsgálható delegálás esetén a felhatalmazó hatóság számára átlátható a jogosultság átadása illetve a teljes átadási útvonalat fel tudja építeni. Üzleti környezetben ez lehetővé teszi az anyagi károkat okozó tranzakciók és tevékenységek felelőseinek a meghatározását.

11.2.1.6 Személyes adatok biztonsága

A nyilvános kulcsú infrastruktúra segítségével biztonságosan hitelesíthetőek nem csak az igaz nevek, de az álnevek vagy a névtelen szereplők is. Ehhez csupán annyi szükséges, hogy az entitások tanúsítványaiban az igaz név helyett az álnév vagy az anonim azonosító szerepeljen. Ezáltal van lehetőség a PKI-ban a hitelesítés és az azonosítás szétválasztására. A névtelen hitelesítésre van szükség az olyan bonyolultabb kriptográfiai protokollokban, mint például az elektronikus szavazás vagy az elektronikus vizsgáztatás, illetve akkor is, ha a hitelesítést egyidejűleg több fél felé kell biztosítani, mint például az egyszeri bejelentkezés (Single Sign On - SSO) esetében. A PKI segítségével minden ilyen alkalmazásban megőrizhető a szereplők személyazonosságának, személyes adatainak illetve bejelentkezési információinak a biztonsága, titkossága.

11.3 Jogi háttér

A nyilvános kulcsú infrastruktúra lehetőséget biztosít a titkosításon túl számos elektronikus tranzakció biztonságos végrehajtására a felek között. A gyakorlatban a közreműködő felek lehetnek magánszemélyek, jogi társaságok, kormányzati szervek illetve ez utóbbiak képviselői. Ezen túlmenően a PKI, mint a globális Internet egyik tipikus terméke, határokon, sőt tengereken és óceánokon átnyúló alkalmazásokat is lehetővé tesz.

Az egyes felek közötti tranzakció, legyen az akár pénzügyi vagy jogi vonatkozású alapvetően két típusba sorolható: amikor a felek egyazon szervezethez tartoznak és egy belső

szabályzat érvényes rájuk, illetve amikor több különböző szervezet tagjai vagy képviselői kívánnak olyan tranzakciót végrehajtani, amire valamilyen jogi szabályozás vonatkozik illetve valamilyen jogilag is elismert, jogi következményekkel járó megállapodás vagy ügylet végrehajtása a cél. A helyzet ez utóbbi esetben kifejezetten bonyolult: az egyes felekre egyidejűleg több szabályozás is vonatkozhat és gyakran az sem egyértelmű, hogy melyik az érvényes és a szabályozások egyidejű megkövetelése ellehetetlenítené az egész tranzakciót. A nemzeti szabályozások a szokásos néhány éves késéssel követték a PKI létesítésére jelentkező igényeket. A PKI iránti igény globalitását jelzi, hogy a szabályozás a Föld legtöbb országában kis késéssel megtörtént és nagyon hasonló eredményt hozott.

Hogy az egyes elektronikus tranzakciók valamiféle jogi következménnyel járjanak, szükséges az elektronikus tranzakciók jogi elismerése és szabályozása. Ebben a részben át fogjuk tekinteni azokat a kezdeményezéseket, amelyek elindították azt a folyamatot, amely még ma is tart és célja az elektronikus tranzakciók jogi gyakorlatba való beültetése és általánosan használhatóvá tétele mind nemzeti mind nemzetközi szinten.

11.3.1 Amerikai Törvényszéki Egyesület - Digitális Aláírási Irányvonalak

Az első a sorban az Amerikai Törvényszéki Egyesület (American Bar Association - ABA) Digitális Aláírási Irányvonalak elnevezésű 1995-ben megszületett dokumentuma, amely megalapozta a digitális aláírás jogi eljárásokban való felhasználhatóságát és lehetővé tette a digitális aláírás törvényi szabályozását.

Az Amerikai Törvényszéki Egyesület irányvonalai alapján azóta majdnem minden államban született valamiféle szabályozás a digitális aláírásokat illetően.

Az egyes államok gyakorlatát követve és a digitális aláírás gyakorlatának egységes nemzeti jogi kereteket adva, 2000-ben megszületett az *E-Sign* törvényi szabályozás (U.S. Electronic Signatures in Global and National Commerce Act). A szabályozás az USA-ban minden nemzetközi vagy belföldi kereskedelmi tranzakcióban lehetővé teszi a digitális aláírás használatát. A szabályozás nem nyilatkozik az egyes aláírások bizonyítóerejéről, csupán azt köti ki, hogy a bírósági eljárások során nem lehet figyelmen kívül hagyni az érvényességét, jogosságát vagy bizonyítóerejét egy digitális aláírásnak, pusztán annak elektronikus jellege miatt.

Az E-Sign szabályozás szerint az elektronikus aláírás:"elektronikus hang, szimbólum, vagy eljárás, amely a szerződéshez vagy más okirathoz van csatolva vagy azzal összefüggésbe hozható és amelyet az okiratot aláírni szándékozó személy hajt végre vagy alkalmaz."

Ez a definíció meglehetősen tág teret ad az elektronikus aláírás fogalmának. Ennek alapján elektronikus aláírásnak tekinthető a digitális aláíráson túl egy sor egyéb dolog is. Megfelel a definíciónak például a konvencionális aláírás digitalizált változata vagy egy digitálisan tárolt ujjlenyomat, a dokumentumot aláírni kívánó személy hangja, esetleg valamiféle jelszót vagy nyilatkozatot felmondva, vagy akár szimplán az aláíró nevének digitálisan tárolt, kiírt változata. Az egyes típusok között természetesen hatalmas eltérések vannak biztonság szempontjából. Az E-Sign nem nyilatkozik arról, hogy melyek milyen körülmények között tekinthetőek bizonyítóerejűnek, ennek eldöntését a tranzakciót végző felekre hagyja.

11.3.2 **EU** Elektronikus Aláírás Irányelv

Az Európai Unió 1999. december 13.-án kiadott egy a digitális aláírásokra vonatkozó irányelvet (Electronic Signature Directive). Az irányelv célja, hogy összehangolja a tagállamok elektronikus aláírásokra vonatkozó jogi szabályozását. Az EU irányelv (mint ahogy azt a magyar törvényeknél is látni fogjuk) három kategóriába sorolja az elektronikus aláírásokat. A legtágabb kategória az elektronikus aláírások kategóriája és az E-Sign -hez nagyban hasonló álláspontot képvisel: "olyan elektronikus formájú adat, amely más elektronikus adathoz kapcsolódik vagy logikailag összefüggésbe hozható vele, és amely az azonosítás módszereként szolgál."

Az E-Sign-hez hasonlóan az EU ajánlás is kiköti, hogy az elektronikus aláírások bizonyítóerejét nem lehet megtagadni kizárólag azért, mert elektronikus formájúak, továbbá kiegészíti azzal, hogy önmagában az sem lehet kizáró ok, hogy az aláírás nem minősített tanúsítványon alapszik vagy, hogy a minősített tanúsítványt nem akkreditált hitelesítő szervezet adta ki továbbá, hogy az aláírást nem biztonságos aláíró eszközzel készítették. Ez természetesen nem jelenti azt, hogy bármilyen jellegű elektronikus aláírást egyforma bizonyító erejűnek kell elfogadni, csupán annyit, hogy jogi eljárásban a felhasználhatóságukat nem lehet csak az elektronikus mivoltukra alapozott érvek hatására megtagadni.

Az EU irányelv ezen felül definiálja az úgynevezett Fejlett Elektronikus Aláírás (Advanced Electronic Signature) fogalmát is, amely további követelményeket támaszt az elektronikus aláírásokkal kapcsolatban. Nevezetesen, szükséges, hogy a Fejlett Elektronikus Aláírás egyértelműen összeköttetésbe hozható legyen az aláíróval, annak alapján megoldható legyen az aláíró azonosságának meghatározása. Olyan körülmények között szülessen, amelyeket az aláíró teljes mértékben és kizárólagosan felügyel. Végezetül, oly módon kötődjön az aláírt adatokhoz, hogy azok későbbi módosítása felismerhető legyen.

A szabályozásban említésre kerülnek olyan fogalmak, mint minősített tanúsítvány, hitelesített tanúsítványkiadó, illetve biztonságos aláíró eszköz. A dokumentumban ezek is definiálásra kerülnek és ezzel is jobban meg lesznek határozva a technikailag alátámasztott és a gyakorlatban is biztonságosnak ítélt aláírások. Amint az alapelvben is szerepel, ezen kritériumok hiánya önmagában nem lehet az aláírás jogi alkalmazásának kizáró oka. (ref eucom)

A minősített tanúsítványnak a következőket kell tartalmaznia:

a) utalás arra, hogy a tanúsítványt minősített tanúsítványként bocsátották ki;

b) a hitelesítés szolgáltató azonosítója, továbbá az az ország, amelyben székhellyel rendelkezik;

c) az aláíró neve, vagy pedig egy álnév, amely ilyenként azonosítandó;

d) olyan rendelkezés, amely alapján az aláíró valamely egyedi jellemzője – a tanúsítvány felhasználási céljától függően – adott esetben feltüntethető;

e) az aláíró által birtokolt aláírás-létrehozó adatnak megfelelő aláírás-ellenőrző adat;

f) a tanúsítvány érvényességi idejének kezdete és vége;

g) a tanúsítvány azonosító kódja

h) a tanúsítványt kibocsátó hitelesítés szolgáltató fokozott biztonságú elektronikus aláírása,

i) adott esetben a tanúsítvány felhasználásának korlátai,

j) adott esetben azon ügyletek értékhatára, amelyekre vonatkozóan a tanúsítvány felhasználható.

A minősített tanúsítványokat kiállító hitelesítés szolgáltatóra vonatkozó követelmények:A hitelesítés szolgáltató köteles

a) bizonyítani a hitelesítés szolgáltatás nyújtásához szükséges megbízhatóságot;

b) gyors és biztonságos nyilvántartási szolgáltatás, valamint biztonságos és azonnali tanúsítvány-visszavonási szolgáltatás fenntartását biztosítani,

c) biztosítani azt, hogy a tanúsítványok kibocsátásának és visszavonásának dátuma és időpontja pontosan meghatározható legyen,

d) megfelelő eszközökkel a nemzeti jogszabályokkal összhangban igazolni annak a személynek az azonosságát, és – adott esetben – egyedi jellemzőit, akinek a részére a minősített tanúsítványt kibocsátották,

e) olyan munkatársakat alkalmazni, akik rendelkeznek a nyújtott szolgáltatásokhoz szükséges szaktudással, tapasztalattal és képesítéssel, valamint különösen hozzáértéssel a vezetők szintjén, szakértelemmel az elektronikus aláírási technológiában, és ismeretekkel a megfelelő biztonsági eljárásokat illetően; ezenfelül köteles olyan ügykezelési és ügyvezetési eljárásokat alkalmazni, amelyek az elismert szabványoknak megfelelnek,

f) megbízható rendszereket és termékeket használni, amelyek a változtatásokkal szemben védettek és biztosítják az általuk támogatott eljárások műszaki és titkosítási biztonságát,

g) intézkedni a tanúsítványok hamisítása ellen, valamint abban az esetben, ha a hitelesítés szolgáltató hozza létre az aláírás-létrehozó adatokat, az ilyen adatok létrehozása során a bizalmas kezelést biztosítani;

h) megfelelő pénzügyi erőforrásokkal rendelkezni az irányelvben megfogalmazott előírásoknak megfelelő működéshez, és különösen – például megfelelő biztosítás kötése által – vállalni a kártérítési felelősség kockázatát;

i) az egy adott minősített tanúsítványra vonatkozó valamennyi lényeges információt megfelelő időszakon keresztül rögzíteni, elsősorban a tanúsításra vonatkozó bizonyíték bírósági eljárások során történő szolgáltatása céljából. Az ilyen rögzítés végezhető elektronikus úton;

j) tartózkodni azon személy aláírás-létrehozó adatainak a tárolásától, illetve másolásától, akinek kulcskezelési szolgáltatásokat nyújtott;

k) – mielőtt az elektronikus aláírásának tanúsítvánnyal történő hitelesítését igénylő személlyel szerződéses jogviszonyra lépne – tartós kommunikációs eszköz segítségével tájékoztatni az igénylőt a tanúsítvány használatának pontos feltételeiről, így többek között a használat esetleges korlátairól, önkéntes akkreditációs rendszer létezéséről, továbbá a panasztételre és a jogviták rendezésére szolgáló eljárásokról. Az ilyen, elektronikusan is továbbítható információt írásban és közérthetően kell rögzíteni. Az információ megfelelő részeit a tanúsítványra hivatkozó harmadik személy számára is – kérésére – rendelkezésre kell bocsátani,

l) megbízható rendszert használni a tanúsítványok ellenőrizhető formában történő tárolására, oly módon, hogy:

- kizárólag engedéllyel rendelkező személyek végezhessenek bejegyzéseket és változtatásokat,

- ellenőrizhető legyen az információ hitelessége,

- a tanúsítvány kizárólag a tanúsítvány jogosultjának hozzájárulásával legyen nyilvánosan kereshető, és

- az e biztonsági előírásokat veszélyeztető műszaki változások az üzemeltető számára érzékelhetővé váljanak.

A biztonságos aláírás-létrehozó eszközökre vonatkozó követelmények

1. A biztonságos aláírás-létrehozó eszközöknek megfelelő műszaki és eljárási módok segítségével garantálniuk kell legalább azt, hogy:

a) az aláírás létrehozásához használt aláírás-létrehozó adatok gyakorlatilag csak egyszer jöhessenek létre, és titkosságuk ésszerű mértékig biztosított legyen,

b) az aláírás létrehozásához használt aláírás-létrehozó adatok kikövetkeztethetősége ésszerű mértékig kizárt legyen, az aláírás pedig a jelenleg rendelkezésre álló technológiát alkalmazó hamisítás ellen védett legyen;

c) az aláírás létrehozásához használt aláírás-létrehozó adatokat a jogszerűen aláíró személy megbízhatóan védeni tudja a mások általi felhasználással szemben.

2. A biztonságos aláírás-létrehozó eszközök nem módosíthatják az aláírással ellátandó adatokat, és nem akadályozhatják meg, hogy az adatokat az aláíró az aláírási eljárás előtt megtekintse.

Ezeknek a kezdeményezéseknek a végső célja, hogy az elektronikus kereskedelmet minden szinten lehetővé tegye. Ennek fényében az EU ajánlás például teljes mértékben technológia független és ezt a megközelítést jelöli ki irányvonalnak a tagállamok számára is.

Az E-Sign esetében már nem ilyen kedvező a helyzet. Az E-Sign -t időben megelőzték az egyes államok elektronikus aláírásra vonatkozó szabályozásai, ezért - bár az E-Sign maga technológia független - több államban is léteznek valamely technológiai megoldást nevesítő vagy előnyben részesítő törvényi szabályozások. Ezt a problémát megoldandó az E-Sign szabályozás kötelezi az egyes államokat az Egységes Elektronikus Tranzakciókról szóló Törvény (UETA - Uniform Electronic Transactions Act) alkalmazására vagy pedig egy technológia semleges jogi álláspont felvételére. Fontos megjegyezni, hogy mindezek csupán irányvonalak és hogy az egyes tagállamok konkrét törvényi szabályozásai meglehetősen különbözőek lehetnek. Ezért a gyakorlatban az alkalmazott szabályok megválasztása gondot okozhat, hiszen könnyen előfordulhat, hogy a felekre más szabályozás vonatkozik, amik esetleg nagyon távol álnak egymástól követelményekben, sőt akár ellentmondásosak is lehetnek.

A különböző szabályozások meglehetősen zavarossá teszik a helyzetet abban az esetben is ha az egyes PKI szolgáltatók megállapodnak valamely elnevezési terület felosztásában. A helyzetet tovább bonyolítja, hogy a technológia (lásd a tanúsítványokról szóló fejezetet) többféle ezzel kapcsolatban felmerülő igényt (lásd a PKI architektúrákról szóló fejezetet) is kielégít, ennek megfelelően többféle megoldás és kapcsolat is előfordulhat, amelyekre az esetlegesen eltérő jogi szabályozásokat egyeztetni kell.

A fent említett szabályozások az egyes gazdasági szereplők, jogi személyek közötti tranzakciók szabályozására szolgálnak. Felmerülhet a kérdés, hogy milyen szabályozásra

illetve jogi megfontolásokra van szükség abban az esetben, ha egy belső biztonsági rendszerről van szó és csupán az adott szervezet tagjai, alkalmazottjai használják belső kommunikációra, belső tranzakciók elvégzésére. Logikusnak tűnhet az álláspont miszerint belső biztonsági ügyekről lévén szó, az ilyen esetekre vonatkozzon a szokásos belső szabályzat és eljárásrendszer. Felmerülhetnek azonban olyan esetek, amikor általános jogorvoslatra van szükség. Mint például akkor, amikor egy gazdasági társaság által hozott elégtelen óvintézkedések, rosszul meghatározott biztonsági paraméterek vagy hibás biztonsági szabályzat miatt az alkalmazottak személyes adatai kompromittálódnak. Vagy amikor valamely alkalmazott gondatlan eljárása miatt az azonosítására szolgáló adatok idegen kezekbe kerülnek és ezáltal a gazdasági társaság jelentős anyagi kárt szenved, esetleg a konkurensek értékes adatokhoz juthatnak hozzá.

Ezekre megoldás lehet a fenti ajánlások követése illetve olyan PKI szolgáltató választása, amely megfelel a fenti követelményeknek.

11.3.3 Magyarországi szabályozások

Magyarországon az előző bekezdésben felsorolt lehetőségeket, üzleti gyakorlatot több különböző törvény illetve rendelet legitimálja, szabályozza. Mindezek alapja a 2001. május 29-én az Európai Unió ajánlásaival összhangban elfogadott, az elektronikus aláírásról szóló 2001. évi XXXV. törvény. A törvény, amellyel a 4.2 fejezetben részletesen foglalkoztunk, legitimálja az elektronikus gazdaságot, szabályozza az interneten már létező szerződéses viszonyokat és megteremti az online tranzakciók jogi keretét.

A törvény definiálja az elektronikus aláírások és az elektronikus dokumentumok kategóriáit. Az interneten lefolytatott tranzakciók és jogügyletek ezek alapján már beilleszthetőek a hagyományos jogi szabályozás rendszerébe, ezáltal rájuk is vonatkoztathatóak lesznek a hagyományos jogi biztosítékok és konstrukciók.

A jogszabály következményeképp minősített elektronikus aláírással ellátott bármely elektronikus okirat teljes bizonyító erejű magánokiratnak minősül, ha az alkalmazott tanúsítványt egy minősített hitelességszolgáltató adta ki.

Az elektronikus számláról szóló 20/2004. (IV. 21.) PM rendelet az elektronikus számlák előállításának és megőrzésének szabályairól rendelkezik. A rendelet szerint az elektronikus számlázáshoz szükséges fokozott biztonságú elektronikus aláírással és időbélyegzővel ellátni a kibocsátott számlát. A rendelet szerint biztosítani kell továbbá a számla olvashatóságát az őrzési időszak alatt. A számvitelről szóló 2000. évi C. törvény 169. § (5) bekezdése az elektronikus formában kiállított bizonylatok elektronikus formában való megőrzését követeli meg. A törvény (6) bekezdése lehetővé teszi az eredetileg papír alapon kiállított bizonylatok megőrzését elektronikus formában is.

Az Oktatási Minisztérium a 20/2004. számú rendeletének értelmében a szakmai vizsgák szervezői kötelesek, a szakmai vizsgák összesítő lapjait elektronikus formában, legalább fokozott biztonságú elektronikus aláírással ellátva, a vizsgát követő 30 napon belül elküldeni a Nemzeti Szakképzési Intézetnek.

A 2006. évi V. törvény alapján 2008. július 1-től a cégbejegyzési, változásbejegyzési

kérelem csakis elektronikusan nyújtható be, továbbá a céginformáció lekérése is kizárólag elektronikusan lehetséges valamennyi cégforma esetén. Az egyszerűsített cégeljáráshoz szükséges dokumentumok már 2007. szeptember 1. óta csak elektronikus formában nyújthatók be.

11.4 Az aláírások típusai

*Egyszerű elektronikus aláírás*on olyan, akár mindenféle technológia biztonságot is nélkülöző eljárást értünk, melynek során az aláíró, a személyazonosságára utaló információkat helyez el egy elektronikus dokumentumban. Az egyszerű elektronikus aláírás semmiféle biztosítékkal nem szolgál az aláíró személyére vonatkozóan, és az aláírt dokumentum integritása sincs biztosítva. Jogvita esetén, az aláírás elektronikus mivolta miatt nem zárható ki az eljárásból, mint bizonyíték. Mindazonáltal a bizonyító ereje néhány speciális esettől eltekintve elhanyagolható.

A *fokozott biztonságú elektronikus aláírás* létrehozása olyan eszközön történik, mely teljes egészében az aláíró felügyelete alatt áll. Az ilyen típusú elektronikus aláírás már nem csak az aláíró személyazonosságát igazolja, de az aláírt dokumentum integritását is igazolja. Ez a fajta aláírás az esetek többségében nyilvános kulcsú kriptográfia alkalmazását és PKI jelenlétét követeli meg.

A fokozott biztonságú aláírás további feltételeket szab meg az aláírás körülményeivel kapcsolatban: az aláíró kulcsot biztonságos, teljes egészében az aláíró jogi személy felügyelete alatt álló eszközön kell tárolni, továbbá minden olyan műveletet, amihez a kulcs szükséges a biztonságos eszközön kell elvégezni.

A *minősített elektronikus aláírás* biztonságos aláíró eszközön előállított kulccsal, minősített hitelességszolgáltató által kiállított tanúsítvánnyal készült aláírást jelent. A minősített elektronikus aláírással ellátott elektronikus dokumentum teljes bizonyító erejű magánokiratnak minősül. (A biztonságos aláíró eszköz egy az SSCD (Secure Signature Creation Device) követelményeinek megfelelő eszköz (az esetek többségében az eszköz egy speciális intelligens kártya)).

11.5 Bizalmi modellek

A nyilvános kulcsú infrastruktúrában kiemelt fontossággal bír a bizalom. Az infrastruktúra feladata, hogy a bizalmat, a megbízhatóságot továbbítsa és kezeskedjen az egyes szereplőkért illetve tanúsítványokért. A nyilvános kulcsú infrastruktúra kapcsán alapvetően kétféle bizalomról beszélhetünk. Egyrészt a bizalom jelentheti egy adott tanúsítvány megbízhatóságát, azt a tényt, hogy a vonatkozó titkos kulcs valóban annak a birtokában van, akinek az adatai a tanúsítványban szerepelnek. Másrészt egy teljesen általános értelemben vett bizalomról is szó lehet, ahol egy adott entitástól egy meghatározott

viselkedést várunk el. Az ebben az értelemben vett bizalom természetesen nem lehet teljes bizonyosságú, amikor emberi tényezőkről van szó. Azt hogy a bizalom milyen utat jár be és hogyan teszi lehetővé az egyes protokollok végrehajtását, alkalmazások használatát az alkalmazott bizalmi modell határozza meg. A körülményektől, a lehetőségektől, az elvárásoktól és a szervezeti felépítéstől függően az egyes helyzetekben más és más bizalmi modellek használata lehet indokolt.

Az egyes bizalmi modellek három központi kérdés körül forognak (refp2.131):

1. Hogyan határozzuk meg, hogy az egyes tanúsítványokat egy adott szereplő megbízhatónak találja-e vagy sem?

2. Hogyan alapozható meg a bizalom?

3. Milyen körülmények között korlátozható vagy szabályozható ez a bizalom egy adott környezetben?

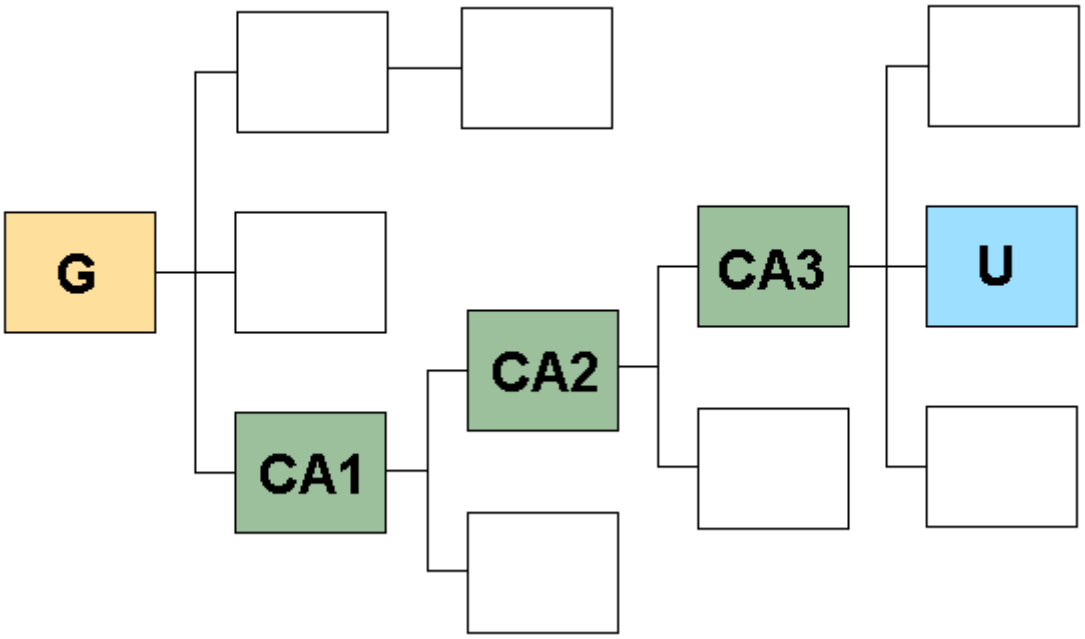
11.5.1 Szigorú hierarchia

A legalapvetőbb bizalmi modell. Minden modell ennek valamilyen változata, bővítése. A bizalom továbbadásának, megalapozásának módszere (a tanúsítvány lánc) is alapvetően megegyezik. A szigorú hierarchiánál az egyes hitelesítő szervezetek (CA - Certificate Authority) egy fa-struktúrába vannak rendezve. A fa gyökerében álló hitelesítő szervezet a gyökér CA, vagy más néven a bizalmi horgony. A modell alapja, hogy a rendszer minden szereplője megbízik a gyökérben. Ebből indul ki a bizalom továbbadása a tanúsítványláncon. A modell működésének előfeltétele, hogy a hierarchia minden résztvevője birtokában legyen a bizalmi horgony saját maga által aláírt tanúsítványának. Ezt a biztonság érdekében nem internetes csatornákon kell továbbítani vagy ellenőrizni. Tipikus megoldás a gyökér tanúsítványának valamilyen digitális ujjlenyomattal, ellenőrző összeggel való megerősítése. Ebben az esetben természetesen az ellenőrző összeget kell nem elektronikus csatornákon felülvizsgálni.

A bizalmi fa levélelemei a végfelhasználók, a köztes elemek pedig az alárendelt hitelesítő szervezetek. Szigorú hierarchia esetén a bizalom mindig a bizalmi horgonyból indul ki és a fában a kérdéses levélelemhez vezető úton az egyes alárendelt hitelesítő szervezeteken halad keresztül. Az úton a két elem közötti bizalom továbbadása digitális aláírás segítségével történik. Egy hitelesítő szervezet azzal szavatol a közvetlenül alatta lévő CA megbízhatóságáért, hogy annak tanúsítványát aláírja. Ennek megfelelően, amikor valamely végfelhasználó ellenőrizni akar egy másikat, akkor a bizalmi horgonyból kiindulva végigellenőrzi a hozzá vezető útvonalon a hitelesítő szervezetek tanúsítványait.

Tehát például, ha a gyökérből (G) egy U felhasználóig vezető úton rendre CA1, CA2 és CA3 alárendelt hitelesítő szervezetek szerepelnek, akkor az ellenőrző félnek U megbízhatóságának megítéléséhez első lépésben CA1 tanúsítványát kell ellenőriznie. Ha a gyökér tanúsítványának segítségével ellenőrizte és helyesnek találta a CA1 tanúsítványán lévő aláírást, akkor azt megbízhatónak ítéli. Következő lépésben az ellenőrző fél CA2 tanúsítványát fogja ellenőrizni az ekkorra már megbízhatónak ítélt CA1 tanúsítványával. Ha ennek alapján CA2 tanúsítványát megbízhatónak találja, akkor annak segítségével megalapozhatja CA3 megbízhatóságát. Ha sikeresen ellenőrizte CA3 tanúsítványát, akkor azt

felhasználva dönthet U megbízhatóságáról. Kölcsönös ellenőrzés után megkezdődhet a biztonságos kommunikáció a két fél között.



11.5 ábra

11.5.2 Laza hierarchia

A hitelesítési szervezetek itt is a szigorú hierarchiához teljesen hasonló fa-struktúrába szerveződnek. A fő különbség, hogy itt az egyes szereplők nem kizárólag a gyökér tanúsítványával rendelkeznek alaphelyzetben, hanem a saját tanúsítványukat kibocsátó hitelesítő szervezet tanúsítványával is. Ez lehetővé teszi a lokális hitelesítést, nevezetesen, ha a két fél ugyanazon hitelesítési szervezet hatáskörébe tartozik, akkor nem szükséges a tanúsítványláncot a gyökérig felépíteni és ellenőrizni, hanem elegendő a bizalmat a lokális CA -ig visszavezetni. Ebben az esetben a bizalom nem szigorúan véve a bizalmi horgonytól kerül levezetésre, hanem a tanúsítványt kibocsátó szervezettől.

11.5.3 Szabályzat alapú hierarchiák

Mind a szigorú mind a laza hierarchia esetén egy hitelesítési szervezetnek csak egy fölérendeltje lehetett. A szabályzat alapú hierarchia lehetővé teszi, hogy egy CA-nak tetszőleges számú fölérendeltje legyen. Ez virtuálisan több szigorú hierarchia párhuzamos használatát jelenti. Az egyes hierarchiák jelentik az egyes szabályzatokat, szabályozásokat. Az egyes szabályzatok az adott helyzetben elvárt szervezeti felépítés vagy szerkezet egyes ágait hivatottak képviselni és a bizalmi modell szintjén reprezentálni.

11.5.4 Elosztott bizalmi architektúra

Az elosztott bizalmi architektúra lehetővé teszi a szereplők hitelesítését több különböző bizalmi hierarchia esetén is. Több különböző hierarchia esetén több különböző bizalmi horgony van jelen és közvetlenül mindegyikük az architektúrának csak a saját fa-struktúrájába tartozó részét tudja hitelesíteni. Az egyes szereplők alaphelyzetben csak a saját gyökér CA -juk tanúsítványában bíznak meg. A teljes körű hitelesítés megvalósításához tehát szükség van valamilyen módszerre, amivel a bizalmat az egyes bizalmi horgonyok között továbbítani lehet. Erre a legelterjedtebb megoldás a kereszthitelesítés vagy más néven a "PKI networking".

Háló konfiguráció esetén a bizalmi horgonyok teljesen vagy majdnem teljesen kapcsoltak páronkénti kereszthitelesítéssel. Ez a megoldás akár tetszőleges két végfelhasználó kölcsönös hitelesítését is lehetővé teheti, viszont ehhez megközelítőleg n(n-1)/2 kereszthitelesítésre van szükség.

A hub konfiguráció (Hub-and-Spoke) egy kitűntetett hitelesítő szervezet jelenlétét feltételezi. Ez a kitűntetett hitelesítő szervezet a hub, ez áll kereszthitelesítéssel összeköttetésben az egyes bizalmi horgonyokkal. A kereszthitelesítés lehetővé teszi, hogy a hubon keresztül más hierarchiákba is eljusson az egyes gyökerek által képviselt bizalom. Ez a megoldás nem összesíti az architektúrát egyetlen hierarchiába, hiszen a kezdeti bizalom minden résztvevő esetén csak a saját bizalmi horgonyához kötődik, minden szereplőnél csak a saját gyökerének a tanúsítványa van, csak az abba helyezet bizalmat várjuk el az architektúra működéséhez. A hub architektúra előnye, hogy n bizalmi horgony esetén csupán n+1 kereszthitelesítésre van szükség.

Egyéb alternatív megoldások a kereszt-elismerés (nem technológiai jellegű megoldás, nem elektronikus eljárással hidalja át a problémát), a Tanúsítvány Bizalmi Lista (CTL - Certificate Trust List) amely a megbízható bizalmi horgonyok listáját jelenti, illetve az Akkreditációs Tanúsítvány, amivel egy megbízható és ismert tanúsítványkiadó szervezet felelősséget vállal egy kisebb CA -ért. Az Akkreditációs Tanúsítvány nem jelenti a kisebb CA integrálását a kiadó szervezet hierarchiájába, továbbra is független marad tőle.

11.5.5 A "Négy sarok" bizalmi modell

A négy sarok bizalmi modell egy tipikus feliratkozási illetve vásárlási szituációt vesz alapul. A szereplők itt egy feliratkozó vagy vásárló illetve egy szolgáltató, kereskedő. Mindkét félhez tartozik egy külön hitelesítő szervezet, amelyek kereszthitelesítéssel igazolják egymást. A vásárlás során felépülő biztonságos kommunikációban a bizalom megalapozása ezen a kétpólusú hitelesítési rendszeren keresztül történik.

11.5.6 A Webes modell

A modell a World Wide Web-ről kapta a nevét, az interneten alkalmazott, ott kialakult modellt jelent. Itt az egyes különálló hierarchiákat nem a bizalmi horgonyok közötti

kereszthitelesítések kötik össze, hanem az összes bizalmi horgony kiosztásra kerül az összes szereplőhöz. A gyakorlatban ez azt jelenti, hogy a szoftvergyártó a gyökér CA-k egy listájával szállítja a szoftvereit, így azok biztonságos és megbízható módon jutnak el a végfelhasználókhoz.

Maga a modell a gyakorlatban egyszerű és kényelmes, ugyanakkor feltételezi az összes bizalmi horgony teljes megbízhatóságát. Ha akár csak egy hitelesítő szervezet is megbízhatatlan vagy rosszindulatú befolyás alá kerül, akkor az egész rendszer kompromittálódik. Ebben a modellben a szoftver minden gyökérhitelesítő szervezetben egyformán és teljes mértékben megbízik, azaz, ha egy rosszindulatú befolyás alatt álló CA kiad egy olyan tanúsítványt, ami valamelyik szereplő személyazonosságához egy, a támadó birtokában lévő kulcsot köt, akkor sikerrel személyesítheti meg a kommunikációban azt a szereplőt.

Ez a probléma más modellekben is felléphet, de ott jellemzően (a használt modelltől függő mértékben) lokális problémáról van szó. Az ilyen jellegű gyengeség a Webes modell esetében sokkal súlyosabb: a teljes architektúra kompromittálódását jelentik.

A helyzetet súlyosbítja, hogy a modell nem ad választ arra a kérdésre, hogy egy gyökér CA kompromittálódása esetén hogyan távolítható el a listából. A biztonsági rés felfedezése után mihamarabb frissíteni kell az összes listát a megbízhatóság helyreállítása érdekében. Ez azonban egy több millió felhasználót érintő alkalmazás tekintetében akadályokba ütközik.

További problémát jelent, hogy ebben a modellben semmiféle jogi kapcsolat nincs a felhasználó és a CA között, a felelősség az esetleges károkért nem vezethető vissza az egyes hitelességszolgáltatókig, minden felelősség a szoftvergyártót terheli, amit viszont az ingyenesen letölthető programok (például böngészők) esetében a licencszerződésben szereplő használati feltételek és kikötések alapján minden valószínűség szerint nem fog vállalni.

11.5.7 Felhasználó központú bizalom

Ebben a modellben minden felhasználó egyben hitelesítő szervezetként működik. Mindenki kialakítja a saját bizalomhálóját azzal, hogy a megbízhatónak ítélt ismerőseinek a tanúsítványát aláírja (ha CA-nak tekintjük őket, akkor valamilyen típusú kereszthitelesítést végeznek). A bizalom, az egyes tanúsítványok ellenőrzése nem teljesen automatikus: a felhasználó dönt a kérdéses tanúsítványhoz vezető úton szereplő ismerős vagy ismerősök illetve az útvonal hossza alapján, hogy megbízik-e benne avagy sem. Ezt a modellt valósítja meg például a PGP (Pretty Good Privacy, [17] és refzimm) illetve a nyílt forráskódú verziója a GnuPG.

A rendszer fő problémáját az a feltételezés jelenti, hogy a felhasználók megértik és átlátják a publikus kulcsú infrastruktúra mibenlétét és az egyes esetekben erre a megértésre alapozva tudnak és akarnak is döntést hozni. Nevezetesen, hogy kinek a tanúsítványát írják alá és kiét ne, illetve hogy mely tanúsítási útvonalakat fogadjanak el és melyeket ne. Nagyon szimpatikus ez a feltételezés, de idealisztikus is. A felhasználók döntő többségéről ilyen felelősségteljes döntés nem várható el. Gondoljunk arra, hogy a leggyakrabban használt jelszavak még ma is az: abc123, qwertz, 12345678, stb..

11.5.8 Kereszthitelesítések

A kereszthitelesítés két hitelesítő szervezet közötti bizalomátadást tesz lehetővé. Alapvetően két fajtáját különböztetjük meg. Az egyik esetben a két CA ugyanahhoz a gyökérhez tartozik és ez a fajta kereszthitelesítés adja tovább a bizalmat például a hagyományos szigorú hierarchia esetében. Ezt nevezzük "intradomain" kereszthitelesítésnek. A másik változatban a két résztvevő CA két különböző bizalmi horgonyhoz tartozik. Ez az "interdomain" kereszthitelesítés. Az elosztott architektúrák esetén ez a fajta kereszthivatkozás az ami lehetővé teszi két különböző gyökérhez tartozó hierarchiák összekapcsolását. Osztályozhatjuk még a kereszthitelesítéseket annak iránya szerint. Ha csak az egyik CA hitelesíti a másikat, akkor egyirányú, ha mindketten hitelesítik a másikat, akkor kölcsönös kereszthivatkozásról beszélünk.

A kereszthitelesítés jelentősége túlmutat a különböző hierarchiák összekapcsolásán: lehetővé teszi a bizalom terjedésének korlátozását is. A tanúsítványok felépítését meghatározó X509-es szabvány meghatároz több a bizalom továbbadását szabályozó kiterjesztést is. Ezekkel többek között megadható például, hogy hány kereszthitelesítésen haladhat át a bizalom (Path length constraint), hogy milyen alkalmazások esetén használható a hitelesítési út felépítésében az adott kereszthitelesítés (Policy constraints) illetve, korlátozható az is hogy a céldomain mely szereplőire vonatkozzon az átadott bizalom (Name constraints). Ezek segítségével megadhatóak olyan jellegű korlátozások, mint például, hogy az adott kereszthitelesítés csak SSL azonosítás során továbbítja a bizalmat, vagy, hogy a Debreceni Egyetemen csak az Informatikai Karhoz tartozó résztvevőkben bízunk meg.

11.5.9 Elnevezések

Az X500 Directory egy szabványt kínál az univerzálisan egyedi nevek megadására. Ez a szabvány feltétez az elnevezendő objektumok közötti hierarchikus felépítést, és ezzel együtt azt, hogy az egyes névtereket valamilyen egyén vagy szervezet felügyeli és a saját névterén belül az egyediséget biztosítja. Az X509 tanúsítványokban szerepel a hozzá tartozó entitás az X500 Directory szerinti úgynevezett megkülönböztetett neve (DN - Distinguished Name, lásd bővebben a tanúsítványok felépítését tárgyaló fejezetet).

Ezzel a konvencióval kapcsolatban több dolog is sértheti a nevek egyediségét. Először is nem áll rendelkezésre egy szervezet vagy személy a hierarchia minden szintjén, ami az egyes szervezetek személyek és entitások egyedi elnevezését felügyeli.

Másrészt az egyes területeken már léteznek különböző elnevezési konvenciók, amelyeket az adott szereplők megnevezésére használnak. Ilyen lehet például egy szerver esetében a domain név vagy az ip cím, illetve személyek esetében az e-mail cím.

11.5.10 A tanúsítványlánc feldolgozása

Mint ahogy már szó volt róla, a publikus kulcsú infrastruktúra célja két résztvevő közötti bizalom megalapozása valamilyen elektronikus kapcsolattartás előkészítésének céljából. Ehhez szükség van egy tanúsítvány láncra, amely a résztvevőket valamilyen módon a bizalmi horgonyokkal összeköti. A bizalom megalapozásának első lépése a tanúsítványlánc

felépítése. A tanúsítványlánc felépítése az alkalmazott bizalmi modellnek megfelelően történik. A legegyszerűbb esetben, a hierarchikus modellnél a kérdéses tanúsítványtól visszafelé a gyökér CA-hoz az útvonalat az egyes tanúsítványokat kiállító szervezetek alapján egyértelműen fel lehet építeni. A hálós modellben viszont, mivel egy gyökérhitelesítő szervezet akárhány bizalmi horgonnyal lehet kereszthitelesítéses kapcsolatban, a tanúsítványlánc előállításához már komoly matematikai apparátus, illetve gráfkereső algoritmusok szükségesek.

Ha az útvonalat sikeresen előállítottuk, ellenőriznünk kell annak helyességét. Ez magában foglalja a tanúsítványlánc minden tanúsítványán az aláírás ellenőrzését, azt hogy meggyőződünk róla, hogy a tanúsítvány még nem járt le, hogy az éppen aktuális alkalmazásban használható, és hogy a tanúsítvány még nem lett visszahívva és úgy általában ellenőriznünk kell minden szabályozást, kulcshasználati megszorítást, amit a tanúsítvány kiterjesztések előírnak.

11.6 A tanúsítványkiadók felépítése

A PKI szolgáltatók zökkenőmentes üzemeltetéséhez három, mind funkcionális, mind biztonsági követelmények szempontjából elkülönülő egységre van szükség úgymint Hitelesítő szervezetek, Regisztráló szervezetek és Tanúsítványtár. A fejezetben ezek feladatait és működését tekintjük át.

11.6.1 A hitelesítő szervezet a rendszer központi eleme.

A hitelesítő szervezet nemzetközileg is elfogadott angol neve Cerification Authority – CA, így gyakran fogjuk ezt a rövidítést használni.

Feladatai:

- Tanúsítványkérések fogadása.
- Kulcspárok generálása a különböző implementációk esetén.
- A nyilvános kulcsú tanúsítványok kiállítása.
- A kiadott tanúsítványok közzététele a nyilvános tanúsítványtárban.
- Korábban kiadott tanúsítványok és szükség esetén kulcspárok megújítása.
- Tanúsítványok visszavonása.
- A visszavont tanúsítványok listájának közzététele (publikálása) a tanúsítványtárban.

A CA mint számítógép és mint szoftver megfelelő fizikai és logikai védelme alapvető fontosságú. Bármely rosszindulatú manipuláció, természeti katasztrófa, vagy az adatok bármely deformációja a teljes infrastruktúra működését, hitelességét veszélyezteti. A védelem kiépítése mind a fizikai mind szoftveres eszközök közül a legmodernebb technológiák alkalmazását, az eljárásrendben pedig a legszigorúbb szabályok életbe léptetését teszi szükségessé. Egy CA kiadhat tanúsítványokat felhasználók vagy más tanúsitványkiadók részére (vagy akár mindkettő). Ha felhasználó tanúsítványokat állít ki, akkor szavatol azért,

hogy a tanúsítványban szereplő publikus kulcshoz tartozó privát kulcs a tanúsítványban szereplő entitás birtokában van. Ha a CA egyéb információkat, mint például elérhetőségre, eljárásrendre vonatkozó vagy a tanúsítvány felhasználhatóságával kapcsolatos információkat is feltűntet, akkor azok helyességét is szavatolja. Ha a tanúsítványt a CA egy másik CA részére állította ki, akkor aláírásával szavatol minden, az adott CA által kiállított tanúsítványért. A CA minden tanúsítványban elhelyezi a saját nevét és aláírja azt, ezáltal, ha a CA irányában a bizalom megalapozható, akkor a tanúsítvány is megbízhatónak tekinthető. A CA titkos kulcsa az alapja az összes általa aláírt tanúsítványba vetett bizalomnak, ezért a CA első és legfontosabb feladata a saját titkos kulcsának a védelme.

11.6.2 **Regisztrációs hivatal**

A regisztrációs hivatal (Registration Authority - RA) az ügyfelek azonosítását végző szerv.

Feladata:

- Az ügyfelek megbízható azonosítása.
- A tanúsítványkérés összeállítása és továbbítása.
- Tanúsítvány visszavonási kérések fogadása.

A regisztrációs hivatal feladata a tanúsítvány biztonsági fokozatához tartozó eljárásrendnek megfelelően azonosítani a tanúsítvánnyal kapcsolatos műveletet kérvényező jogi személyt. A szabványos elektronikus formátumú kérvények összeállítását szintén egy célszoftver végzi, amelynek és az alapjául szolgáló hardverinfrastruktúrának a védelme szintén kritikus jelentőségű. Az alacsonyabb biztonsági fokozatú azonosítási eljárások esetén a személyes megjelenés sem feltétlenül szükséges, akár teljesen elektronikus úton is történhet. Ez utóbbi esetben, míg az infrastruktúra fizikai védelme jelentősen leegyszerűsödik, addig a kérvények előállítását végző szoftver biztonsága fokozott figyelmet igényel. Ugyanazon szolgáltató publikus kulcs infrastruktúrája az igényeknek megfelelően több földrajzilag elkülönülő regisztrációs egységet is tartalmazhat.

A CA felelősséget vállal azért, hogy az adott azonosító és a publikus kulcs párja összetartoznak, ezért azt is ellenőriznie kell, hogy a tanúsítványt igénylő fél valóban birtokában van a megadott publikus kulcshoz tartozó privát kulcsnak (proof of possession). Továbbá a CA csak a saját eljárásrendjének megfelelő tanúsítványokat adhat ki. Például meg kell tagadnia a tanúsítvány kiadását, ha az eljárásrendje alapján e-mail aláírásokhoz használatos tanúsítványok kiadására alkalmas és szerződés aláírására alkalmas tanúsítványt kérnek tőle, még akkor is ha egyébként az igénylő egyébként jogosult az adott művelet elvégzésére. A CA kötelessége, hogy biztosítsa, hogy a tanúsítványtárában és a visszavonási listáiban szereplő tanúsítványok mindegyike illik az eljárásrendjébe. A CA ezen feladatkörét rendszerint a regisztrációs hivatal veszi át.

Egy CA-hoz több RA is tartozhat. A CA nyilvántartja a megbízható regisztrációs hivatalokat a neveikkel és a hozzájuk tartozó tanúsítványokkal egyetemben. A CA le tudja ellenőrizni, hogy a kérvényező birtokában van a titkos és a publikus kulcsnak, de csak ennyit és semmi mást. A tanúsítványba foglalandó minden egyéb információ leellenőrzése a regisztrációs hivatal dolga. Az ellenőrzés elvégzésére alapvetően két modell van. Az egyik

során az RA előre leellenőrzi és hitelesíti a tanúsítványba foglalandó információkat és értesíti róla a CA-t. A CA elfogadja az adatok érvényességét illetve érvénytelenségét, mert megbízik a CA-ban. A másik modell szerint a CA megkapja a tanúsítványkérelmet és csak aztán kérdezi meg a regisztrációs hivatalt, hogy a feltűntetett adatok érvényesek-e.

Az első modellt tipikusan akkor használják, ha a felhasználó fizikailag meg tud jelenni a regisztrációs hivatalban és a megfelelő okmányokkal igazolni tudja a tanúsítványban feltüntetendő információkat. Alapvetően kívánatos, hogy a felhasználók generálják a saját kulcsukat és aztán bizonyítsák a saját személyazonosságuk, hatáskörük a regisztrációs hivatalban. Sajnos azonban a biztonságos kulcsgenerálás számítási kapacitást, kriptográfiai eszköztárat, olyan erőforrásokat igényel, amik rendszerint nem állnak a felhasználó rendelkezésére illetve jelentősen megnövelnék a kulcs tárolására alkalmazott kriptográfiai modul (rendszerint valamilyen célhardver) árát. Ezért gyakorta a regisztrációs hivatal végzi a kulcsgenerálást és helyezi a felhasználó modulára azt. Amennyiben a kulcsot a felhasználó generálja, a regisztrációs hivatal ellátja egy egyszer használatos jelszóval, amit a tanúsítványkérelméhez prezentálva igazolhatja az előzetes ellenőrzés tényét. A másik modellt tipikusan akkor alkalmazzák, ha nincs lehetőség vagy szükség a személyes megjelenésre. Ilyenkor a CA a kérvény kézhezvétele után intéz kérdést a regisztrációs hivatalhoz, az ellenőrzi a kérdéses információkat és az ellenőrzés eredményének megfelelően igennel vagy nemmel válaszol.

11.6.3 Tanúsítványtár

A tanúsítványtár egy olyan speciális adatbázis, amely tartalmazza a tanúsító eszköz (CA) által kibocsátott tanúsítványokat, a visszavont tanúsítványok listáját és egyéb, a tanúsítványra vonatkozó adatokat. A tanúsítványtár feladata, hogy bármely tanúsítvány állapotáról valós időben információt szolgáltasson. A tanúsítvány tár az alábbi szolgáltatásokat nyújtja a felhasználók (legyenek azok alkalmazások vagy természetes személyek):

- Biztosítja az ügyfeleket egy adott tanúsítvány hitelességéről.
- Biztosítja az ügyfeleket egy adott tanúsítvány érvényességéről.

A fenti két szolgáltatás igénybevételével a felhasználók és az alkalmazások meggyőződhetnek egy adott rendszerbeli identitás és egy publikus kulcs összetartozásáról.

A biztonságos működéshez a CA-nak helyes és teljes formában kell megőriznie a tanúsítványok státuszára vonatkozó információkat is. Azaz a visszavonási lista bejegyzéseinek korrektségét és a biztonságát is biztosítania kell. A visszavonási listából hiányzó bejegyzések érvénytelen tanúsítványok elfogadását eredményezhetik, míg a helytelen bejegyzések érvényes tanúsítványok elutasítására vezethetnek. A helytelen időpont a fenti hibák közül bármelyiket eredményezheti.

A nyilvános kulcsú infrastruktúra használatához szükséges, hogy a felek hozzáférjenek a tanúsítványokhoz illetve a visszavonási listákhoz. Illetéktelenek hozzáférése bármelyik listához szintén problémát jelenthet amennyiben a tanúsítványok személyes adatokat tartalmaznak a tulajdonosaikról. A CA feladata a tanúsítványok és a visszavonási listák elérhetővé tétele a felhasználók felé és mindemellett az adott környezetben megkövetelt hozzáférési korlátozások kikényszerítése.

A régi dokumentumok aláírásainak ellenőrzésekor szükség lehet arra az azóta már lejárt tanúsítványra, amivel azt a dokumentumot aláírták, és amely az aláírás időpontjában még érvényes volt. Ehhez szükséges a tanúsítványok és a visszavonási listák archiválása és a lejárat után is a rendelkezésre állás biztosítása.

A tanúsítványtár fogadja egy vagy több CA tanúsítványait és visszahívási listáit és elérhető formában közzé teszi őket. A tanúsítványtár nem feltétlenül megbízható entitás, a tanúsítványtárban szereplő adatok hitelességéért a kiállító CA az aláírásával szavatol. A tanúsítványtár ugyan adott esetben mindenki számára elérhetővé teheti a tárolt információkat, azonban azt mindenképpen korlátozni kell, hogy a tárolt adatokon ki tudjon módosítani, hiszen hamis rekordok feltöltésével a szolgáltatások elérhetetlenné tehetők, felfüggeszthetők (DoS - Denial of Service típusú támadás). Bizonyos esetekben, a hozzáférést is azonosításhoz kell kötni. Ilyen lehet például, ha a tanúsítványtár egy kiszervezett szolgáltatás és a számlázási információk miatt szükséges tudni, hogy ki hányszor fért hozzá a szolgáltatáshoz. Másik ilyen eset, amikor a tanúsítványok érzékeny személyes adatokat tartalmaznak, például a kutatás-fejlesztési osztály dolgozóiét, aminek alapján a versenytársak képet kaphatnak a cég kutatási irányvonaláról illetve átcsábíthatják az ott dolgozó munkatársakat.

A tanúsítványtárból az adatok az archívumba kerülnek. Az archívum felel a tanúsítványinformációk hosszú távú tárolásáért, azért hogy a letárolt tanúsítványok és visszavonási listák az archívumba érkezésükkor helyesek voltak és, hogy azóta ezek nem változtak meg. Az archívum szerepe, hogy a tanúsítvány lejárta után az azzal aláírt dokumentumokkal kapcsolatos viták kérdésében a döntéshez szükséges információt szolgáltassa.

11.6.4 A tanúsítványok életciklusa

A PKI működése során minden tanúsítványnak van egy meghatározott életciklusa. A tanúsítvány életciklusa három fő szakaszra bontható.

11.6.4.1 A tanúsítvány kiadása.

A tanúsítvány kiadásának lépései:

1. Az ügyfél a használni kívánt algoritmusnak megfelelő kulcspárt generál magának. A privát kulcsot biztonságba helyezi, a nyilvános kulccsal pedig a regisztrációs egységnél (RA), a személyazonosságának bizonyítása után elkészítteti a megfelelő tanúsítvány típushoz tartozó kérvényt. Ez a lépés a rendelkezésre álló szoftverek széles választéka ellenére is meghaladja egy átlagos felhasználó képességeit így ezt a lépést gyakran teljes egészében a szolgáltató regisztrációs egysége végzi el. Fokozott biztonságú tanúsítványok esetében a kulcspár generálása teljes egészében a biztonságos eszközön történik.

2. Az RA elvégzi a személyazonosság ellenőrzésére az igényelt tanúsítványhoz tartozó biztonsági szint eljárásrendjében szereplő lépéseket.

Magánszemély esetén ez többnyire valamely arcképes igazolvány felmutatását, illetve a legalacsonyabb biztonsági szintek esetén valamely személyes kérdés megválaszolását jelenti, intézmények esetén pedig többnyire cégbejegyzés, aláírási címpéldány illetve egyéb hivatalos dokumentumok szükségeltetnek az azonosításhoz. A tanúsítvány biztonsági

fokozatához tartozó eljárásrend megköveteli az azonosítás illetve kérvény generálás folyamatának valamilyen részletességű dokumentációját is.

3. Sikeres azonosítás esetén az RA az ügyfél nyilvános kulcsából valamint a tanúsítvány kiadásához szükséges egyéb adatokból szabványos elektronikus tanúsítványkérő dokumentumot állít elő, digitális aláírásával igazolja, majd pedig továbbküldi a hitelesítő szervezetnek.

4. A CA ellenőrzi a beérkező tanúsítványkérés digitális aláírását.

5. A CA elkészíti a kért tanúsítványt és közzéteszi a nyilvános tanúsítványtárban.

11.6.4.2 A tanúsítvány használata

A tanúsítvány használata rendszerint valamely alkalmazáshoz kötődik, ám a PKI, illetve a tanúsítványok szerepe minden esetben ugyanaz. Egy adott személy, szervezet vagy szerver és egy publikus kulcs közötti kapcsolat igazolásáról van szó.

1. Az alkalmazás működése közben olyan pontra ér ahol valamelyik fél publikus kulcsára van szükség. Ez tipikusan digitális aláírást, aszimmetrikus titkosítást illetve azonosítást alkalmazó protokollok implementációinál fordul elő.

2. Az alkalmazás ekkor alkalmazza a fentebb említett szabványok valamelyikét és elkéri a szolgáltató tanúsítványtárából a tanúsítványt. Ezek után a tanúsítványon található aláírás kerül ellenőrzésre, hogy az valóban a szolgáltatóhoz tartozik-e. Amennyiben az aláírás nem hiteles, vagy a tanúsítvány érvénytelen, az alkalmazás működése megakad és a megfelelő biztonsági intézkedések kerülnek végrehajtásra. Ezek a legkülönfélébb tevékenységek lehetnek a felhasználó vagy a rendszergazda értesítésén, logoláson keresztül akár a rendszer teljes leállításáig terjedhetnek.

3. Amennyiben a tanúsítvány hiteles és érvényes, az alkalmazás eldönti, hogy a kiállító hitelesítő szervezet megbízható-e avagy sem. Legtöbb PKI-t használó szoftver rendelkezik egy beépített egyes esetekben bővíthető listával azokról a hitelesítő szervezetekről, amelyeket megbízhatónak ítél meg. Ha a szolgáltató nincs benne ebben a listában, akkor az alkalmazás a hitelességszolgáltató aláíró tanúsítványának kiállítójához fordul.

4. Az alkalmazás mindaddig ismétli a 2. és 3. lépéseket, amíg megbízható hitelességszolgáltatóhoz nem jut, vagy pedig óvintézkedésekre nem kerül a sor.

Figyeljük meg, hogy ezen a módon az egyes hitelességszolgáltatók képesek akár közvetve, más, rendszerint kisebb hitelességszolgáltatókon keresztül is igazolni valamely személy vagy szerver identitását, a fent említett módszer segítségével.

11.6.4.3 Tanúsítvány visszavonása

A tanúsítványok mindenképpen visszavonásra kerülnek érvényességi idejük lejártakor. Lehetőség van továbbá a tanúsítvány visszavonására az érvényességi idő lejárta előtt. Ez utóbbit a felhasználó kezdeményezi. Erre tipikusan akkor lehet szükség, amikor tanúsítványhoz tartozó privátkulcs kompromittálódott elveszett, vagy valamilyen módon megsérült.

A visszavonás lépései felhasználói visszavonás esetén:

- Az ügyfél felismeri a tanúsítványhoz tartozó privát kulcs elvesztését, sérülését vagy arra gyanakszik, hogy az kompromittálódott.
- Az ügyfél értesíti a regisztrációs hivatalt.
- A regisztrációs egység ellenőrzi az ügyfél személyazonosságát, majd a tanúsítvány visszavonását a hitelesítő hivatalnál.
- A CA visszavonja a tanúsítványt. A művelet eredményét közzéteszi a tanúsítványtárban.

Amennyiben viszont a tanúsítvány érvényességi ideje járt le, és a CA úgy dönt, hogy a kulcs további használata az adott körülmények között biztonságos, meg is hosszabbíthatja a tanúsítvány érvényességi idejét. Ellenkező esetben a CA kezdeményezi a tanúsítvány visszavonását, és a felhasználó új kulcspár generálását követően új tanúsítványért folyamodik.

A visszavonás lépései a CA kezdeményezésére:

- A CA észreveszi, hogy az egyik kulcspárnak hamarosan lejár az érvényessége és jelzi a regisztrációs egységnek (RA).
- Az RA értesíti az ügyfelet, és felajánlja egy új kulcs elkészítésének lehetőségét vagy a kulcspár megújítását.
- A CA elkészíti az új kulcspárt a régieket pedig visszavonja. Megújítás esetén bejegyzi a kulcsok mellé az új lejárati időket.
- A CA értesíti az RA-t a művelet sikeres végrehajtásáról, és szükség esetén elküldi neki az új kulcsot.
- Az RA továbbküldi az értesítőt, vagy szükség esetén az új titkos kulcsot az ügyfélnek.

11.7 A tanúsítvány felépítése

A PKI a gyakorlatban sok különböző szolgáltató közreműködését jelenti, amit alkalmazások széles skálája vesz igénybe. Mindez szükségessé teszi az illeszkedő infrastruktúra komponensek, eszközök, protokollok és alkalmazások szabványosítását.

A felmerült igények nyomán, mivel a szabványosítás folyamatainak egészét nem felügyelte egy központi szervezet, több kezdeményezés született az eljárás különböző részeire, amik az idők folyamán az ipar de facto szabványaivá váltak. Az RSA cég, mely tevékenyen közreműködött a nyilvános kulcsú kriptográfián alapuló technológiák terjesztésében, ugyanezen technológiák szabványosításában is szerepet vállalt. Ezen törekvések eredményeképpen születtek meg a PKCS ajánlások (Public Key Cryptography Standard – Nyilvános Kulcsú Kriptográfiai Szabvány). Az International Telecommunication Unit (ITU) nemzetközi távközlési szervezet nevéhez fűződik az X.509 jelű ajánlás, amely a nyilvános kulcsú tanúsítványok formátumát határozza meg. Az Internet Engineering Task Force (IETF) pedig PKIX nevű szabványkészletével járult hozzá a folyamathoz. A PKIX készlet az RFC 2459 jelű dokumentum révén kapcsolódik az ITU ajánlásaihoz, nevezetesen az X.509 szabványnak megfelelő tanúsítvány leírását adja meg.

Ebben a részben az X509-es szabványnak megfelelő tanúsítványokról lesz szó. Első lépésben áttekintjük, hogy a tanúsítványainknak milyen feltételeknek kell eleget tennie, hogy a publikus kulcsú infrastruktúrában fel tudjuk őket használni, szó lesz az X509-es szabvány kialakulásáról. Végül a tanúsítvány egyes részeit és a felépítését vesszük górcső alá. A struktúra ismertetéséhez az ASN.1 jelölésrendszert fogjuk használni. Annak ellenére, hogy a jelölés erősen technikai jellegű, nagy előnye, hogy egyrészt lehetővé teszi a formális leírást és fogódzót ad az egyébként igen összetett szerkezet áttekintéséhez, megértéséhez, másrészt a jelenleg is érvényes RFC 2459-es ajánlás is ezt a jelölésrendszert használja, így ez a gyakorlat a továbbolvasáskor is segítséget jelenthet. A struktúra ismertetése előtt először kitérünk az X509-es tanúsítványok formátumában gyakran előforduló építőelemekre.

11.7.1 Az X509 szabvány áttekintése

Publikus kulcsú infrastruktúrában a tanúsítványok feladata elsősorban egy adott publikus kulcs és az ahhoz tartozó titkos kulcs összekapcsolása. Ugyanakkor a tanúsítványoktól elvárjuk, hogy rendelkezzen mindazokkal a tulajdonságokkal, amikkel mondjuk egy személyi igazolvány vagy egy bankkártya. Elvárjuk, hogy az egyes személyek szervezetek azonosítását, kezelését épp olyan, de inkább nagyobb biztonsággal el tudjuk végezni, mint ahogy azt az offline megfelelőikkel természetesen és magától értetődően tesszük. Elvárjuk tőlük mindazt a biztonságot és kényelmet és természetesen azt, hogy ezt a modern hálózati környezetben hajtsa végre. Mindezek alapján az elvárásainkat egy "ideális" elektronikus tanúsítványtól kilenc pontban foglaljuk össze (citep1.21):

1. Legyen tisztán digitális objektum, legyen automatikusan feldolgozható és a világhálón továbbítható.
2. Tartalmazza a titkos kulcs tulajdonosának nevét, a céget vagy szervezetet, amelyhez az egyén tartozik, és tartalmazzon kapcsolat-felvételi információkat.
3. Azonnal meg lehessen állapítani, hogy a tanúsítványt mikor adták ki.
4. Egy megbízható harmadik fél állítsa ki és ne pedig az, akinek az identitását a tanúsítvány igazolni hivatott.
5. Mivel egy tanúsítványkiadó több tanúsítványt is kiadhat, akár ugyanazon felhasználónak is, szükséges, hogy ezeket meg lehessen különböztetni.
6. Könnyen megállapítható legyen, hogy a szóban forgó tanúsítvány eredeti, vagy pedig hamisítvány.
7. Ellenállónak kell lennie módosításokkal szemben. Nem szabad, hogy bárki is meg tudja változtatni a tartalmát.
8. Ha a tanúsítvány már lejárt, azt azonnal meg lehessen állapítani.
9. A tanúsítványból azt is azonnal látni lehessen, hogy milyen alkalmazásoknál használható.

Az első pontnál említett automatikus feldolgozás megköveteli, hogy a tanúsítványnak legyen egy közös megegyezésen alapuló szabványos formátuma. Jelenleg az általánosan elfogadott tanúsítványformátum az X509 és a későbbiekben erről lesz szó.

Az X509-es tanúsítvány az 1988-ban kiadott CCITT X509-es ajánlásról kapta a nevét. Ebben az ajánlásban definiálták ugyanis a szabvány első verzióját. Ez a tanúsítvány még

kizárólag az X500 Directory-hoz készült, annak az azonosítását volt hivatott kezelni. Később a hangsúly eltolódott egy általánosabb megközelítés felé. Idővel a szabvány két további verziója látott napvilágot. A második verzió egyetlen újítást hozott, mégpedig a nevek újrafelhasználhatóságát. Napjainkban a legtöbb alkalmazás a harmadik verziót használja (X509v3). A harmadik verzió több kiterjesztést is definiál a formátumhoz, amelyek különböző plusz információk feltűntetését teszik lehetővé a tanúsítványban. Ezeknek a kiterjesztéseknek a használatát az egyes alkalmazások hivatottak kezelni és a vonatkozó megoldások alkalmazásról alkalmazásra változnak. Az Internetes Tanúsítványokat és a használatukat kiegészítő visszahívási listák formátumát szabványosító dokumentumot (RFC 2459) az IETF (Internet Engineering Task Force) 1999 márciusában adata ki.

11.7.2 ASN.1 építőelemek

Objektumazonosítók

Az objektumazonosítók (Object Identifier - OID) egészek egy sorozatát jelentik, amelyek egyértelműen azonosítanak valamilyen objektumot (szereplőt, szervezetet, algoritmust, bármit). Az IETF által szabványosított jelölésben az objektumazonosítót alkotó egészeket pontok választják el. Az azonosítókat alkotó egészek felépítése hierarchikus. A pontokkal elválasztott jelölésben ez a hierarchia teljesen hasonló például a domain nevek alkotta hierarchiához. Például: ha az 1.2.3.1 objektumazonosító az 1.2.3 OID alá tartozik a hierarchiában. Az objektumazonosító ASN.1 jelölése:

```
usdod ::= OBJECT IDENTIFIER { 1 3 6 }
```

Az X509-es szabványban nem csak az egyes személyek, entitások azonosítására használunk objektumazonosítókat, de az egyes algoritmusokhoz is OID -ket rendelünk. A hierarchikus felépítés miatt az egyes tanúsítványkiadóknak körültekintően kell eljárni az objektumazonosítók kiosztásakor, hogy a létrehozott entitások ne okozzanak ellentmondást, inkonzisztenciát a rendszerben. Minden entitás a saját maga alá tartozó objektum azonosítókra való tekintettel felelős a hierarchia megtartásáért.

Algoritmusazonosítók

Az algoritmusazonosító információt hordoz az alkalmazott kriptográfiai algoritmusról, tartalmazza az algoritmus objektumazonosítóját és az esetleg szükséges paramétereket. Az algoritmusazonosító ASN.1 jelölése:

```
AlgorithmIdentifier :: SEQUENCE {
    algorithm           OBJECT IDENTIFIER,
    parameters          ANY DEFINED BY algorithm OPTIONAL }
```

Az X509-es szabványban az algoritmus azonosítókat használhatják mind a tanúsítvány aláírására használt algoritmus meghatározására illetve annak a publikus kulcsú algoritmusnak

a megadására, amellyel a tanúsítványban szereplő kulcsot (illetve annak a párját) használni kívánjuk.

Könyvtársztring

A szabványban használatos sztringek megadására használt ASN.1 típus. Elsődleges célja, hogy karakterkódolástól és nyelvtől függetlenül lehessen vele szöveges információkat megadni. A Könyvtársztring ASN.1 jelölése:

```
DirectoryString ::= CHOICE {
    teletexString       TeletexString (SIZE (1..MAX)),
    printableString     PrintableString (SIZE (1..MAX)),
    universalString     UniversalString (SIZE (1..MAX)),
    utf8String          UTF8String (SIZE (1..MAX)),
    bmpString           BMPString (SIZE (1..MAX)) }
```

A fenti definíció azt jelenti, hogy ezen típus megadásakor választhatunk az öt alternatíva között.

printableString: a legtöbb nyomtatható ASCII karaktert tartalmazza.

TeletexString: a T.61 karakterkészletet használja. A legtöbb nyugat-európai nyelv abc-jét képes reprezentálni.

BMPString: 16 bites karakterkódolást alkalmaz, sok különböző nyelvet támogat.

UTF8String, UniversalString: több bájtos karakterkódolásokat használnak. Minden nyelvet támogatnak.

Alkalmazott típustól függetlenül két különböző Könyvtársztring lehet egyenlő. Az összehasonlítás úgy történik, hogy a sztringeket előbb UTF8String-gé konvertáljuk és összehasonlítjuk. Fontos megjegyezni, hogy a PrintableString-ek összehasonlításakor a kis és nagy betűk nem számítanak.

Megkülönböztetett nevek

A megkülönböztetett nevek célja, hogy egy általános, hierarchikus elnevezési konvenciót alkosson. Mint a többi hierarchiánál, a nevek kiosztásáért és egyediségéért a neveket kiosztó egység felel. A megkülönböztetett nevek attribútum név, érték párokból áll. A legelterjedtebb formátum az X500 elnevezési konvenció. Pl.:

c=HU; st=Hungary; l=Debrecen; o=Debreceni Egyetem; ou=Informatikai Kar; cn=Teszt Elek

Itt a c az országkódra (Country), az st az államra (State), az l a helységnévre (Location name), az ou a szervezeti egységre (Organisational Unit), a cn pedig az egyszerű névre (Common Name) utal. A név megadásakor lehetőség van ennél több vagy akár kevesebb attribútumot tartalmazó név rögzítésére is. Az internetes domainneveket követő formátum például:

dc=hu; dc=unideb; dc=inf; cn=Teszt Elek

Itt a dc a domain komponenst (Domain Component) jelenti.
A megkülönböztetett nevek ASN.1 jelölése (ref rfc):

```
Name ::= CHOICE {
      RDNSequence
      }

RDNSequence ::= SEQUENCE OF RelativeDistinguishedName

ReletiveDistinguishedName ::= SET OF AttributeTypeAndValue

AttributeTypeAndValue ::= SEQUENCE {
      type          AttributeType,
      value         AttributeValue }

AttributeType ::= OBJECT IDENTIFIER

AttributeValue ::= ANY DEFINED BY AttributeType
```

Általános nevek

Az Általános név választási lehetőséget ad hét elterjedt elnevezési struktúra között, illetve lehetővé teszi saját névstruktúra létrahozását és annak az alkalmazását. Az általános nevek ASN.1 jelölése (ref rfc):

```
GeneralName ::= CHOICE {
      otherName                  [0] OtherName,
      rfc822Name                 [1] IA5String,
      dNSName                    [2] IA5String,
      x400Address                [3] ORAddress,
      directoryName              [4] Name,
      ediPartyName               [5] EDIPartyName,
      uniformResourceIdentifier  [6] IA5String,
      iPAddress                  [7] OCTET STRING,
      registeredId               [8] OBJECT IDENTIFIER }

OtherName ::= SEQUENCE {
      type-id           OBJECT IDENTIFIER,
      value             [0] EXPLICIT ANY DEFINED BY type-id }

EDIPartyName ::= SEQUENCE {
      nameAssigner      [0] DirectoryString OPTIONAL,
```

```
    partyName          [1] DirectoryString }
```

Idő

Dátumok és idő megadására szolgál. ASN.1 jelölése (ref rfc):

```
Time ::= CHOICE {
    utcTime             UTCTime,            -- YYMMDDHHMMSSZ
    generalTime         GeneralizedTime     -- YYYYMMDDHHMMSSZ
    }
```

Az időformátumok végén a Z betű mindkét esetben "Zulu" szóra utal, nevezetesen hogy a greenwichi időt kell megadni.

A tanúsítvány felépítése

A szabvány szerint az X509 tanúsítvány három részből áll. A teljes tanúsítványt magában foglalja egy úgynevezett "módosításjelző boríték" (tamper-evident). Ez a gyakorlatban annyit jelent, hogy mindent, ami ezen a borítékon belül van, azt digitálisan aláírunk. A borítékon belül van a tanúsítvány tartalom és ez pedig opcionálisan tartalmazhat egy vagy több tanúsítvány kiegészítést.

Módosításjelző boríték

A borítékot a tanúsítvány tartalom, a tanúsítvány tartalom aláírása és az aláíró algoritmus azonosítója alkotja. A módosításjelző boríték a tanúsítvány legkülső rétege. Az X509 tanúsítvány ASN.1 jelölése (ref rfc):

```
Certificate ::= SEQUENCE {
    tbsCertificate      TBSCertificate,
    signatureAlgorithm  AlgorithmIdentifier,
    signatureValue      BIT STRING }
```

Tanúsítvány tartalom

Ez a tanúsítvány tényleges tartalmi része, tartalmazza a publikus kulcsot és a publikus kulcshoz tartozó privát kulcs tulajdonosának identitását. A boríték ASN.1 definíciójában a TBSCertificate utal erre a részre. Pontos szintaxisa (ref rfc):

```
TBSCertificate  ::=  SEQUENCE  {
    version         [0]  EXPLICIT Version DEFAULT v1,
    serialNumber         CertificateSerialNumber,
    signature            AlgorithmIdentifier,
    issuer               Name,
```

```
    validity            Validity,
    subject             Name,
    subjectPublicKeyInfo SubjectPublicKeyInfo,
    issuerUniqueID  [1]  IMPLICIT UniqueIdentifier OPTIONAL,
    subjectUniqueID [2]  IMPLICIT UniqueIdentifier OPTIONAL,
    extensions      [3]  EXPLICIT Extensions OPTIONAL
    }

Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }

CertificateSerialNumber  ::=  INTEGER

Validity ::= SEQUENCE {
    notBefore      Time,
    notAfter       Time }

UniqueIdentifier  ::=  BIT STRING

SubjectPublicKeyInfo  ::=  SEQUENCE  {
    algorithm            AlgorithmIdentifier,
    subjectPublicKey     BIT STRING  }

Extensions  ::=  SEQUENCE SIZE (1..MAX) OF Extension

Extension  ::=  SEQUENCE  {
    extnID      OBJECT IDENTIFIER,
    critical    BOOLEAN DEFAULT FALSE,
    extnValue   OCTET STRING  }
```

version

Ez a mező jelenti az alkalmazott X509 szabványverziót. A szabvány három verziója visszafelé kompatibilis, ezért ha nem adunk meg explicit verziót, akkor a formátumot a legalacsonyabb alkalmazható szabványverziónak kell tekinteni. A második verzióban vezették be az egyedi azonosítókat ezért ennek a mezőnek a jelenléte megköveteli a 2-es verziószámot. Ugyanígy mivel a harmadik verzióban vezették be a tanúsítvány kiterjesztések használatát, így ezen mezők jelenlétében alapértelmezetten 3-as verziószámúnak kell tekinteni a tanúsítványra vonatkozó szabványt. A manapság használatos tanúsítványok legtöbbje a 3-as verziójú X509 szabványt követi.

serialNumber

Sorozatszám. Egy egész érték, ami az adott hitelességszolgáltató által kiadott tanúsítványok között egyedileg azonosítja a tanúsítványt.

signature

Az aláíró algoritmus azonosítója, megegyezik a borítékban szereplő signatureAlgorithm értékével. Ennek a mezőnek a szerepe, hogy az aláíró algoritmus megváltoztatásának észlelését lehetővé tegye. Vegyük észre, hogy mivel a módosításjelző borítékon belül található ezért megváltoztatása esetén az aláírás ellenőrzése sikertelen lesz, míg a boríték signatureAlgorithm értékének megváltoztatását semmi sem jelzi.

issuer

Kiadó. A tanúsítványt kiadó hitelességszolgáltató megkülönböztetett neve. Habár a szabvány tetszőleges megkülönböztetett név használatát lehetővé teszi, a gyakorlatban a tanúsítványok nagy része a megkülönböztetett névről szóló bekezdésben hozott példákhoz hasonló jellegű neveket használ.

validity

Érvényesség. A mező értéke egy időintervallumot definiál, amelyen belül a tanúsítvány érvényes.

subject

Alany. A tanúsítványban szereplő publikus kulcshoz tartozó titkos kulcs birtokosának megkülönböztetett neve. Habár a szabvány tetszőleges megkülönböztetett név használatát lehetővé teszi, a gyakorlatban használt tanúsítványok nagy része a megkülönböztetett névről szóló bekezdésben hozott példákhoz hasonló jellegű neveket használnak.

subjectPublicKeyInfo

A publikus kulcsra vonatkozó információ. Ez a mező tartalmazza a publikus kulcsot és azon kriptográfiai algoritmus azonosítóját, amivel összefüggésben a kulcsot fel lehet használni.

issuerUniqueID, subjectUniqueID

Ezek a mezők az X509 szabvány második verziójában jelentek meg és a nevek újrafelhasználhatóságát hivatottak megoldani. A gyakorlatban azonban nem terjedt el ez a megoldás így ezek a mezők legtöbbször üresek.

extensions

Kiterjesztések. Ezek a mezők a szabvány harmadik verziójában jelentek meg és választ adnak egy sor olyan kérdésre, amelyek az alkalmazások során felmerülhetnek, de a tanúsítványtörzs nem tartalmaz rájuk vonatkozó információkat. Ezen túl lehetőséget adnak az adott tanúsítvány felhasználásának kifinomult szabályozására és az adott publikus kulcsú infrastruktúrát érintő részletes és hatékony szabályzatok megvalósítására.

Tanúsítvány kiterjesztések

A kiterjesztések három komponensből állnak: kiterjesztés azonosító, kritikusság és érték. A kiterjesztés azonosító egy objektumazonosító és egyedileg azonosítja az adott kiterjesztést. A kritikusság az adott kiterjesztés fontosságát jelzi: ha egy adott alkalmazás nem tudja értelmezni az adott kiterjesztést akkor, ha a kiterjesztés nem volt kritikus akkor figyelmen kívül hagyja azt, míg ha a szóban forgó kiterjesztés kritikus, akkor megtagadja a tanúsítvány felhasználását. A szabvány lehetővé teszi saját definiálású kiterjesztések használatát, amelyek segítségével tetszőleges bonyolultságú szabályozás illetve tetszőleges részletességű információ köthető egy tanúsítványhoz. A következőkben néhány fontosabb szabványos tanúsítvány kiterjesztésről lesz szó.

Basic Constraints

Ez a kiterjesztés megválaszolja azt a kézenfekvő kérdést, hogy a szóban forgó entitás amit a tanúsítvány a benne szereplő tanúsítvány által reprezentált kulcspárhoz köt a publikus kulcsú infrastruktúrában egy tanúsítványkiadó vagy pedig végfelhasználó szerepét játssza. Ebben a kiterjesztésben lehet korlátozni továbbá a hitelesítési útvonalban szereplő CA-k számát. Ez egyben az útvonalon szereplő kereszt hitelesítések számát is jelenti, így ez a kiterjesztés fontos szerepet játszik az egyes bizalmi modellek implementálásában.

A kiterjesztés ASN.1 jelölése (ref rfc):

```
id-ce-basicConstraints OBJECT IDENTIFIER ::=  { id-ce 19 }

BasicConstraints ::= SEQUENCE {
     cA                      BOOLEAN DEFAULT FALSE,
     pathLenConstraint       INTEGER (0..MAX) OPTIONAL }
```

Issuer Alternative Name, Subject Alternative Name

Mint ahogy az elnevezési konvencióknál arról szó volt, az X500 Directory elnevezési konvenciói nem terjedtek el általánosan, ezért az egyes publikus kulcsú infrastruktúrákat logikus lépés már létező elnevezési hierarchiák köré felépíteni. Ezen mezők segítségével adhatók meg a tulajdonos és a kiadó a kapcsolódó elnevezési konvenció(k)ban kapott neve(i). Ez a lehetőség végfelhasználó esetében különösen hasznos: ip címeket, URL-eket illetve e-mail címeket is rendelhetünk a segítségével a tanúsítványainkhoz.

A kiterjesztések ASN.1 jelölése (ref rfc):

id-ce-issuerAltName OBJECT IDENTIFIER ::= { id-ce 18 }

IssuerAltName ::= GeneralNames

id-ce-subjectAltName OBJECT IDENTIFIER ::= { id-ce 17 }

SubjectAltName ::= GeneralNames

GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName

Name Constrains

A különböző bizalmi modellekben rendszerint szükség van arra, hogy az egyes hierarchiákon belül szereplő gyökér CA-k és alárendelt hitelesítő szervezetek felosszák egymás között a rendelkezésre álló névteret. Ez a kiterjesztés lehetővé teszi, hogy ezen felosztásra irányuló szabályozásokat is kikényszerítsük. A kiterjesztés két lehetőséget biztosít egy CA adott név feletti illetékességének meghatározására. Megadhatunk engedélyezett illetve tiltott részfákat a névtérben. Egy CA akkor illetékes, ha a szóban forgó név benne van az engedélyezett részfában, de nincs benne a tiltott részfában.

A kiterjesztés ASN.1 jelölése (ref rfc):

```
NameConstraints ::= SEQUENCE {
    permittedSubtrees       [0]    GeneralSubtrees OPTIONAL,
    excludedSubtrees        [1]    GeneralSubtrees OPTIONAL }

GeneralSubtrees ::= SEQUENCE SIZE (1..MAX) OF GeneralSubtree

GeneralSubtree ::= SEQUENCE {
    base                    GeneralName,
    minimum         [0]     BaseDistance DEFAULT 0,
    maximum         [1]     BaseDistance OPTIONAL }

BaseDistance ::= INTEGER (0..MAX)
```

Kulcs attribútumok

A gyakorlati alkalmazások során gyakori, hogy az egyes szereplőknek több különböző kulcspárja van. Mindegyik kulcspár egy külön feladat ellátására szolgál. A tanúsítványtörzs csupán az algoritmust és az algoritmus paramétereket tartalmazza és nem ad választ arra a kérdésre, hogy hogyan különböztessük meg az ugyanazon szereplőhöz tartozó tanúsítványokat.

Key Usage Ezzel a kiterjesztéssel megadható, hogy a kulcs milyen jellegű felhasználásra alkalmas, milyen típusú biztonsági követelményeket képes kielégíteni:

keyCertSign: A kulccsal ellenőrizhető a kulcs párjával aláírt tanúsítványok hitelessége.

cRLSign: A kulccsal ellenőrizhető a kulcs párjával aláírt visszavonási listák hitelessége.

non-Repudiation: A kulccsal ellenőrizhető a párjával aláírt letagadhatatlan aláírások hitelessége.

digitalSignature: A kulccsal ellenőrizhető a párjával aláírt aláírások hitelessége (akkor kell használni, ha a fenti három közül egyik kategóriába sem esik a felhasználás).

keyEncipherment: A kulcs felhasználható titkos kulcs továbbítására kulcscsere protokollokban.

dataEncipherment: A kulcs felhasználható nyers adatok titkosítására. Ez az érték nem fedi a kulcs felhasználhatóságát kulcscsere vagy kulcsmegállapodás protokollokban.

keyAgreement: A kulcs felhasználható titkos kulcsok előállítására kulcsmegállapodás protokollokban.

encipherOnly: A titkos kulcs, ami egy olyan kulcsmegállapodás során született, amiben a tanúsítványban szereplő publikus kulcs felhasználásra került. Kizárólag adatok titkosítására lehet használni.

decipherOnly: A titkos kulcs, ami egy olyan kulcsmegállapodás során született amiben a tanúsítványban szereplő publikus kulcs felhasználásra került. Kizárólag titkosított adatok dekódolására lehet használni.

A szabvány szerint az egyes kulcsfelhasználási módok bármilyen kombinációját lehetővé teszi, ezért ezek megadásánál különösen nagy körültekintéssel kell eljárni, hiszen egyes kombinációk nyilvánvalóan ellentmondásosak vagy értelmezhetetlenek.

A kiterjesztés ASN.1 jelölése (ref rfc):

id-ce-keyUsage OBJECT IDENTIFIER ::= { id-ce 15 }

```
KeyUsage ::= BIT STRING {
    digitalSignature        (0),
    nonRepudiation          (1),
    keyEncipherment         (2),
    dataEncipherment        (3),
    keyAgreement            (4),
    keyCertSign             (5),
```

```
cRLSign          (6),
encipherOnly     (7),
decipherOnly     (8) }
```

Extended Key Usage

A kulcs használatára pontos meghatározást ad. Lehetőséget ad arra, hogy olyan felhasználási módokat is kikössünk a kulcs felhasználására, amik a szabványos Key Usage kiterjesztésben nem adhatóak meg. Megadhatjuk például, hogy a tanúsítványt TLS azonosítás során lehet használni és ott is csak valamely szerver azonosítására.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-extKeyUsage OBJECT IDENTIFIER ::= {id-ce 37}

ExtKeyUsageSyntax ::= SEQUENCE SIZE (1..MAX) OF KeyPurposeId

KeyPurposeId ::= OBJECT IDENTIFIER
```

Private Key Validity

A tanúsítványban megadott érvényességi időtartam alapvetően a tanúsítványra vonatkozik. Ennek a kiterjesztésnek a segítségével elválaszthatjuk a tanúsítvány érvényességét a titkos kulcsétól. Előfordulhat például, hogy azt szeretnénk, hogy a privát kulccsal csak egy korlátozott időtartamig lehessen érvényes aláírásokat osztani, viszont ezen időtartam leteltével is szeretnénk az időközben érvényesen kiosztott aláírásokat ellenőrizni. A mező értelmezéséhez természetesen szükség van egy az aláíráshoz csatolt időbélyegre is.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-privateKeyUsagePeriod OBJECT IDENTIFIER ::= { id-ce 16 }

PrivateKeyUsagePeriod ::= SEQUENCE {
          notBefore       [0]     GeneralizedTime OPTIONAL,
          notAfter        [1]     GeneralizedTime OPTIONAL }
```

SubjectKeyIdentifier

Több vagy nagyon specifikus nyilvános kulcsú infrastruktúra használata esetén könnyen előfordulhat, hogy egy alanynak több tanúsítványa is van, amelyek mindegyike más és más alkalmazási lehetőségekkel bír. Ezen kiterjesztés célja, hogy egy alany több kulcsa közül könnyebben ki lehessen választani az éppen szükségeset. Az egyszerűség és az azonosítók (nagy valószínűséggel) különbözősége kedvéért ennek az értéke általában a publikus kulcs egy hash értéke.

A kiterjesztés ASN.1 jelölése [31]:

```
KeyIdentifier ::= OCTET STRING
```

```
id-ce-subjectKeyIdentifier OBJECT IDENTIFIER ::= { id-ce 14 }

SubjectKeyIdentifier ::= KeyIdentifier
```

AuthorityKeyIdentifier

Ez a kiterjesztés a hitelesítési útvonal megkonstruálását segíti. Mint ahogyan arról már korábban szó volt, egy hitelesítő szervezet több aláíró kulccsal is rendelkezhet a biztonsági követelményeknek és a különböző céloknak megfelelően. Ezen kiterjesztés azonosítja az adott tanúsítvány aláírására szolgáló kulcshoz tartozó tanúsítványt a CA összes tanúsítványa között. Ezen mező hiányában a CA összes tanúsítványát ki kellene próbálni, amíg a megfelelőt meg nem találjuk. Az érték lehet a megfelelő CA tanúsítvány sorozatszáma (serialNumber) és a benne szereplő sorozatszám együttesen, illetve egy tetszőleges sztring, ha az a CA vonatkozó tanúsítványában SubjectKeyIdentifier-ként szerepel.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-authorityKeyIdentifier OBJECT IDENTIFIER ::= {id-ce 35}

AuthorityKeyIdentifier ::= SEQUENCE {
 keyIdentifier             [0] KeyIdentifier      OPTIONAL,
 authorityCertIssuer       [1] GeneralNames       OPTIONAL,
 authorityCertSerialNumber[2]  CertificateSerialNumber  OPTIONAL
}
```

Eljárásmódra vonatkozó információk

Az X509-es szabvány első két verziójában az alkalmazott eljárásmódot implicit értelmezték. A tanúsítványokban semmiféle erre vonatkozó információ nem szerepelt. A különböző eljárásmódokhoz más és más CA-kat léptettek üzembe és az eljárásmódok az egyes CA-khoz tartoztak, a tanúsítványkiadó megkülönböztetett neve azonosította azokat. A minden eljárásmódhoz külön CA létesítése illetve az egyes eljárásmódok többlépcsős azonosítása és kezelése meglehetősen körülményes és alacsony hatásfokú volt. Ezt hivatottak kiküszöbölni az eljárásmódra vonatkozó kiterjesztések.

CertificatePolicies

Az eljárásmódok kezelését szolgáló kiterjesztés. CA tanúsítványok esetében itt vannak feltűntetve a CA által megvalósított eljárásmódok, irányelvek, amelyeknek a betartásáért a CA szavatol. A felhasználó vagy a felhasználó alkalmazás hivatott eldönteni, hogy az adott esetben mely eljárásmódok számítanak elfogadhatónak.

Felhasználói tanúsítvány esetében ez a kiterjesztés a tanúsítvány által kielégített eljárásmódokat tartalmazza. Egy eljárásmód tartalmazhat arra vonatkozó megkötéseket, hogy a tanúsítványt milyen alkalmazásban lehet használni illetve esetlegesen az alkalmazott eljárásmód általános erejéről, biztonságáról is.

Ez a kiterjesztés erősen korlátozza a különböző nyilvános kulcsú infrastruktúrák közötti átjárhatóságot. Lehetőség van a teljes mértékben megbízható CA-k esetében az anyPolicy eljárásmód megadására is.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-certificatePolicies OBJECT IDENTIFIER ::= {id-ce 32}

CertificatePolicies ::= SEQUENCE SIZE (1..MAX) OF
PolicyInformation

PolicyInformation ::= SEQUENCE {
     policyIdentifier   CertPolicyId,
     policyQualifiers   SEQUENCE SIZE (1..MAX) OF
             PolicyQualifierInfo OPTIONAL }

CertPolicyId ::= OBJECT IDENTIFIER

PolicyQualifierInfo ::= SEQUENCE {
       policyQualifierId  PolicyQualifierId,
       qualifier          ANY DEFINED BY policyQualifierId }

PolicyQualifierId ::=
    OBJECT IDENTIFIER ( id-qt-cps | id-qt-unotice )

CPSuri ::= IA5String

UserNotice ::= SEQUENCE {
     noticeRef        NoticeReference OPTIONAL,
     explicitText     DisplayText OPTIONAL}

NoticeReference ::= SEQUENCE {
     organization     DisplayText,
     noticeNumbers    SEQUENCE OF INTEGER }

DisplayText ::= CHOICE {
     visibleString    VisibleString  (SIZE (1..200)),
     bmpString        BMPString      (SIZE (1..200)),
     utf8String       UTF8String     (SIZE (1..200)) }
```

PolicyMappings

Ez a kiterjesztés teszi lehetővé az együttműködés két különböző eljárásmódot megvalósító CA által lefedett PKI terület között. A felhasználók illetve azok alkalmazásai többnyire csak a saját CA-juk által megvalósított eljárásmódokat ismerik el. A különböző területek közötti együttműködéshez szükség van egy leképezésre a két CA által megvalósított eljárásmódok között. Ez a kiterjesztés tartalmazza a fordítótáblát a kiadó CA saját eljárásmódjai és azon eljárásmódok között, amik azokkal egyenértékűnek tekinthetők. Egy bejegyzéssel a kiadó CA szavatol azért, hogy ha a bejegyzésben szereplő saját eljárásmód

megfelel egy adott alkalmazásnak, akkor a feltűntetett másik eljárásmód is elég biztonságos az adott alkalmazás szempontjából. Ez utóbbi rendszerint egy másik CA valamely eljárásmódját jelenti.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-policyMappings OBJECT IDENTIFIER ::=  { id-ce 33 }

PolicyMappings ::= SEQUENCE SIZE (1..MAX) OF SEQUENCE {
     issuerDomainPolicy      CertPolicyId,
     subjectDomainPolicy     CertPolicyId }
```

PolicyConstraints

Ezzel a kiterjesztéssel tovább lehet szigorítani az eljárásmódokkal szemben támasztott követelményeket. A kiterjesztés két mezőből áll: requireExplicitPolicy és inhibitPolicyMapping. Mindkét mező egy megszorítást jelent a tanúsítványlánc felépítésére nézve. Az értékük egy-egy egész érték, amely azt jelzi, hogy a hitelesítési lánc felépítése során hányadik tanúsítványnál lépnek életbe. Ha tehát az egyik ilyen mező értéke n, akkor a tanúsítványláncban az első n tanúsítványra az adott megszorítás nem vonatkozik.

A requireExplicitPolicy egy megkövetelt eljárásmód explicit megadását írja elő a tanúsítványokban, míg az inhibitPolicyMapping a PolicyMappings kiterjesztést érvényteleníti.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-policyConstraints OBJECT IDENTIFIER ::= { id-ce 36 }

PolicyConstraints ::= SEQUENCE {
     requireExplicitPolicy         [0] SkipCerts OPTIONAL,
     inhibitPolicyMapping          [1] SkipCerts OPTIONAL }

SkipCerts ::= INTEGER (0..MAX)
```

CRLDistributionPoints

Ahhoz, hogy egy tanúsítvány érvényességéről meggyőződjünk a kibocsátó CA tanúsítványán kívül egyéb információkra is szükségünk van. Az eredeti elképzelés szerint minden szükséges információ elérhető egy globális X500 könyvtáron keresztül. Az X500 könyvtár technológia nem terjedt el széles körben így egy globális X500 könyvtár sem áll rendelkezésre, amin keresztül bármely megkülönböztetett névhez egyértelműen megtalálhatók a vonatkozó információk. Jelenleg több különböző protokoll és megoldás szolgáltatja a szükséges pluszinformációkat így a visszahívási listákat is (Certificate Revocation List - CRL). Könnyen előfordulhat, hogy egy tanúsítványlánc felépítésekor több különböző tanúsítványtárhoz több különböző protokollon keresztül kell hozzáférnünk.

A CRLDistributionPoints kiterjesztés lehetővé teszi, hogy információkat adjunk meg a vonatkozó tanúsítványtár elérhetőségére és elérési metódusára vonatkozólag. A kiterjesztésben több elosztási pont megadható, mindegyikhez megadható a neve az elérési

helye és módja illetve, hogy milyen indokkal visszavont tanúsítványokat szolgáltat. Ha az indok mező elmarad, akkor az elosztási pontnak minden visszavont tanúsítványt tartalmaznia kell a visszavonási indokra való tekintet nélkül.

A kiterjesztés ASN.1 jelölése [31]:

```
id-ce-cRLDistributionPoints OBJECT IDENTIFIER ::= { id-ce 31 }

cRLDistributionPoints ::= {
            CRLDistPointsSyntax }

CRLDistPointsSyntax    ::=    SEQUENCE    SIZE    (1..MAX)    OF
DistributionPoint

DistributionPoint ::= SEQUENCE {
   distributionPoint    [0]    DistributionPointName OPTIONAL,
   reasons                 [1]    ReasonFlags OPTIONAL,
   cRLIssuer               [2]    GeneralNames OPTIONAL }

DistributionPointName ::= CHOICE {
   fullName                [0]     GeneralNames,
   nameRelativeToCRLIssuer [1]     RelativeDistinguishedName }

ReasonFlags ::= BIT STRING {
   unused                  (0),
   keyCompromise           (1),
   cACompromise            (2),
   affiliationChanged      (3),
   superseded              (4),
   cessationOfOperation    (5),
   certificateHold         (6) }
```

FreshestCRL

A CA-k nem készítenek új visszavonási listát minden egyes tanúsítvány visszavonáskor, hanem csupán egy, a legutóbbihoz képest jelentkező módosításokat tartalmazó kisebb úgynevezett delta CRL-t tesznek közzé. A rendes visszavonási listákat csak megadott időközönként frissítik.

Ez a kiterjesztés szintaktikájában és értelmezésében megegyezik a CRLDistributionPoints kiterjesztésével. Azon elosztási pontokkal kapcsolatos információkat tartalmazza, amelyek a delta CRL-eket szolgáltatják.

AuthorityInfoAccess

Ez a kiterjesztés egyéb a hitelesítési szolgáltatóval kapcsolatos információk elérhetőségét tartalmazza. Jelenleg két, az útvonal felépítését szolgáló információkat közvetítő

szolgáltatás elérhetősége adható meg vele. Az egyik (id-ad-caIssuers) segítségével azon egyéb CA-k listája kérhető le, amelyek a kibocsátó CA számára tanúsítványt adtak ki. A hozzáférés módjánál ugyan lehetőség van HTTP, FTP, DAP segítségével való hozzáférés előírására is, a gyakorlatban azonban leginkább az LDAP segítségével való hozzáférés a legelterjedtebb. A másik szolgáltatás segítségével a CA által kiadott tanúsítványok státuszinformációi kérdezhetőek le (id-ad-ocsp).

```
id-pe-authorityInfoAccess OBJECT IDENTIFIER ::= {id-pe 1}

AuthorityInfoAccessSyntax  ::=
        SEQUENCE SIZE (1..MAX) OF AccessDescription

AccessDescription  ::=  SEQUENCE {
        accessMethod          OBJECT IDENTIFIER,
        accessLocation        GeneralName  }

id-ad-ocsp      OBJECT IDENTIFIER ::= { id-ad 1 }
id-ad-caIssuers OBJECT IDENTIFIER ::= { id-ad 2 }
```

SubjectDirectoryAttributes

Ez a kiterjesztés lehetővé teszi, hogy jogosultságokkal (authorization) kapcsolatos információkat kapcsoljunk a tanúsítványhoz. Habár a lehetőség fennáll, a mező alkalmazása a gyakorlatban mégis ellenjavallt. A jogosultságok élettartama rendszerint rövidebb, mint a tanúsítványoké ezért a kiterjesztés alkalmazása a tanúsítvány élettartamát csökkenti. Másik probléma, hogy a jogosultságokkal kapcsolatos információkért és folyamatokért rendszerint nem a CA a felelős, így a kiterjesztés alkalmazása további adminisztrációs terhet és felelősséget terhel a CA-ra.

11.8 PKI irányelvek és eljárásrendek

Mint ahogyan az korábban említésre került a kriptográfiai megoldásuk önmagukban nem elegendőek a biztonsághoz. A szoftveres és algoritmikus védelem az ügyviteli védelemmel együttesen alkalmazva válik teljessé. Ugyanez a nyilvános kulcsú infrastruktúrával kapcsolatban is érvényes, a szoftveres és algoritmikus megoldásokat a megfelelő eljárásokkal alkalmazva kell egy biztonsági irányelvet megvalósítani. A PKI -vel kapcsolatos biztonsági irányelveket két dokumentum foglalja össze. Az egyik a Tanúsítvány Irányelv (Certificate Policy - CP) a másik pedig a Tanúsítási Gyakorlat Nyilatkozat (Certification Practices Statement - CPS). Ezeken dokumentumok szerepe nem a közvetlen felhasználás, a végfelhasználók a megfelelő tanúsítvány kiterjesztésekből értesülnek az őket érintő irányelvekről.

A biztonsági irányelvek írják le a célokat, a felelősséget az elvárásokat valamely szervezet biztonságával kapcsolatban. A biztonsági eljárások pedig meghatározzák, hogy hogyan tudjuk az irányelvben meghatározott célokat elérni, hogy kinek mit kell ehhez tennie, tartalmazza azokat a lépéseket, amiket az adminisztrátoroknak és a felhasználóknak a biztonság fenntartása érdekében meg kell tennie.

A Tanúsítvány Irányelv tartalmazza egy vagy több CA biztonsági előírásait. A biztonsági előírások absztrakt módon írják le az elérendő biztonsági célokat. Csak a célokat írják le, nélkülöznek minden konkrétumot, hiszen a technológiák fejlődhetnek, változhatnak a biztonsági elvárásoktól függetlenül. A PKI bizalmi modelljétől függően előfordulhat, hogy több CA használ egy Tanúsítvány Irányelvet, míg a Tanúsítási Gyakorlatra vonatkozó Nyilatkozat mindegyiknél különböző lehet.

A Tanúsítási Gyakorlatra vonatkozó Nyilatkozat tartalmazza a Tanúsítási Irányelvben megfogalmazott biztonsági célok elérésének mikéntjét. Pontos és részletes leírást tartalmaz minden résztvevő és felhasználó a működés során tanúsított viselkedéséről, az alkalmazott szoftver és hardverelemek specifikációjáról. Meghatározza a biztonság elérésére használt eljárások és eszközök összességét. Meghatározza például, hogy hogyan azonosítja magát a felhasználó a regisztrációs hivatal felé (például megkövetelheti, hogy személyesen jelenjen meg és mutassa fel a személyi igazolványát). Azt, hogy a CPS-t megfelelően betartják, alkalmazzák-e illetve, hogy az alkalmas-e a CP -be meghatározott célok elérésére, szabályos időközönként vagy kereszttanúsítás esetén elvégzett revízió vagy akkreditáció során ellenőrzik.

A Tanúsítvány Irányelvvel ellentétben a Tanúsítási Gyakorlatra vonatkozó Nyilatkozat nem nyilvános. Bármely külső félnek, másik CA-nak szoftverfejlesztőnek el kell tudnia dönteni a CP és a vizsgálatok vagy akkreditáció eredményének ismeretében, hogy a szóban forgó CA megfelel-e a céljainak, avagy sem.

Az egyes CP-k a megfelelő tanúsítvány kiterjesztésben elhelyezett objektumazonosítóval (OID) kapcsolódnak a tanúsítványhoz. A szabvány tetszőleges számú CP kezelését lehetővé teszi, ám az egyes felhasználók többnyire csak a saját területük Tanúsítvány Irányelveivel vannak tisztában. A szabvány lehetőséget ad az egyes CP-k egymásnak való megfeleltetésére, így az egyes CA-k leképezéseket létesíthetnek az egyes CP-k között, hogy a felhasználók az ismeretlen CP-vel kiadott tanúsítványok biztonságáról is dönteni tudjanak, hogy az az ő esetükben megfelelő-e avagy sem. A CPS-ek csupán az implementált CP objektumazonosítójával kapcsolódnak a tanúsítványhoz. Egy CPS akár több CP-nek is megfelelhet.

A Tanúsítvány Irányelvek és Tanúsítási Gyakorlatra vonatkozó Nyilatkozatok felépítését és a formátumát az erre vonatkozó RFC 2527 ajánlás (Certificate Policies and Certification Practicies Framework) határozza meg. Ez egy keretrendszer, amely a PKI irányelvek és eljárásrendek tervezőit segítik, továbbá az egységes formátum és felépítés meghatározásával az egyes CP-k és CPS-ek összevetését is megkönnyítik. Fontos, hogy a különböző Tanúsítvány Irányelvek és Tanúsítási Gyakorlatra vonatkozó Nyilatkozatokat hatékonyan és átláthatóan össze lehessen hasonlítani, hogy az esetleges kereszthitelesítések esetén az egyes CP-k közötti összefüggéseket és leképezéseket meg lehessen határozni. A szabvány egy tartalomjegyzéket, egy vázat is definiál. Ez a tartalomjegyzék 8 fő szekciót és 185 fejezetet és alfejezetet ír elő. Az adott esetben nem releváns vagy nem jelentkező témák esetén az adott rész tartalma a "No Stipulation" azaz "Nincs megszorítás" szövegnek kell lennie, hogy egyértelmű legyen, hogy az valóban egy nem releváns téma és nem pedig arról van szó, hogy a tervezők egyszerűen elfelejtkeztek az adott fejezetről. Ugyanez a szabvány vonatkozik mind a CP-kre mind a CPS-ekre, és ugyanazt a kivonatot írja elő mindkettő számára. Ez logikus és kényelmes, hiszen a két dokumentum szoros kapcsolatban áll

egymással és ez az egységes felépítés mind tervezéskor mind ellenőrzéskor nagy segítséget jelent.

Az RFC 2527 felépítése

- Bevezetés
- Általános rendelkezések
- Azonosítás és hitelesítés
- Működési feltételek
- Fizikai, ügyviteli és személyi biztonsági ellenőrzés
- Technikai biztonsági ellenőrzés
- Tanúsítvány és CRL profilok
- Előírás adminisztráció

Bevezetés

Ez a rész tartalmazza az ezen eljárásrend alatt kiadott tanúsítványok áttekintését. Nevezetesen, hogy az ez alapján kiadott tanúsítványokat milyen típusú környezetben milyen alkalmazások esetén lehet használni, illetve, hogy a szóban forgó tanúsítványokat miről lehet megismerni. Ez az áttekintés nem elegendő ahhoz, hogy megerősítsük, hogy az ez alapján kiadott tanúsítványok megfelelnek a céljainknak, viszont alkalmas lehet arra, hogy előzetesen kiszűrjük azokat amelyek biztosan nem felelnek meg az elvárt biztonsági követelményeknek. Az egy adott Tanúsítvány Irányelv alapján kiadott tanúsítványokat rendszerint a CertificatePolicies tanúsítvány kiterjesztésben elhelyezett objektumazonosító alapján lehet felismerni. Az is bevett gyakorlat, hogy egyetlen CA-ra vagy részhierarchiára vonatkozó CP esetén a kiadó neve alapján is egyértelműen azonosítható a vonatkozó CP. Ebben a részben kap helyet a célközönség, a felhasználó szervezet vagy közösség leírása is. A rész végén a dokumentumokat előállító, karbantartó, jóváhagyó illetve a CA-t üzemeltető szervezetek elérhetősége is szerepel. Ezek CP-k esetén többnyire csak e-mail címek vagy telefonszámok, hogy személyi változások esetén ne kelljen a dokumentumot változtatni, míg CPS-ek esetén a felelős személyek neve is szerepelhet. A bevezetésben szerepel még a folyamatokban résztvevő entitások leírása, mint CA, RA, előfizetők vagy éppen szolgáltatók.

Általános rendelkezések

Ez a rész általános illetve jogi gyakorlati kérdésekkel foglalkozik. Kitér arra, hogy a működés során résztvevő szereplőknek milyen kötelezettsége és felelőssége van. A rendszer szereplői lehetnek a felhasználók, maga a CA, az RA, vagy éppen szolgáltatók. A PKI infrastruktúrát használó egyszerű felhasználóknak is vannak kötelezettségei, amiknek nem betartása esetén a hibákért való felelősség is rá hárul. Tipikus példa az ilyen kötelezettségre valamely tanúsítvány előtt az adott tanúsítvány hitelességének ellenőrzése az éppen aktuális visszavonási listák letöltésével.

Másik fontos témája ennek a fejezetnek a teljesítési audit. A dokumentum önmagában csupán egy papír, a benne meghatározott irányelveket és eljárásrendet a PKI rendszer működése során be is kell tartani, annak meg is kell felelni, ennek ellenőrzésére szolgál a

teljesítési audit. Ennek a körülményeit meg kell határozni. Ebben a fejezetben kerül rögzítésre az, hogy hogyan, mely szervezet által kerül elvégzésre az audit, az auditot végző szervezetnek milyen tanúsítványai, tapasztalatai és illetékessége van a területen és hogy milyen kapcsolatban áll a CA-val. Rögzítésre kerül, hogy milyen gyakorisággal kell elvégezni az auditot, illetve hogy annak eredménye milyen formában kerül összegzésre illetve rögzítésre, továbbá hogy milyen módon kell kezelni az esetlegesen feltárt gyengeségeket.

Ez a szekció írja le, hogy a kapcsolódó információkat hogyan kell kezelni, miként és miket kell közzétenni. Foglalkozik többek között a CP a CPS a tanúsítványok illetve a visszavonási listák közlésével. A CPS teljes formában nem szokás közzé tenni, mert sok olyan információt tartalmaz a rendszerről, amelyet a támadók kihasználhatnak. Ezért a CPS-nek csupán egy alaposan ellenőrzött kivonatát szokás közzétenni.

Előfordulhat, hogy a CA felé információszolgáltatási kéréssel fordulnak más szervezetek. Például elképzelhető, hogy valamely bűnténnyel kapcsolatban a rendőrségnek egy tanúsítvány tulajdonosának a valódi identitására kíváncsiak, vagy ha egy per folyamán egy aláírt vagy titkosított dokumentumot bizonyítékként akarnak felhasználni. Az ilyen esetekben követendő eljárásról is ez a rész rendelkezik.

Itt kap helyet a díjak, a garanciák, a visszafizetési garanciák, a felelősségvállalás mértéke illetve a szolgáltatások leírása is. Többek között ez arra is kitér, hogy csalás esetén milyen körülmények között illetve mekkora összegig vállal felelősséget illetve kártérítést a CA.

Azonosítás és hitelesítés

Az azonosítás és hitelesítés a rendszer legkritikusabb része. Az egész publikus kulcsú infrastruktúra szerepe, hogy egy tanúsítvány formájában valamilyen nevet, azonosságot kössön egy publikus kulcshoz. Az azonosítás illetve hitelesítés módja nagyban meghatározza ennek a köteléknek az erejét és a jellegét. Ez a rész írja le a felhasználók azonosításával kapcsolatos elvárásokat és eljárásokat és az azonosítás jellegét. Például, hogy a regisztrációs hivatalban a kérvényt benyújtónak személyesen meg kell-e jelennie illetve, hogy milyen formában kell igazolnia magát (személyi igazolvánnyal? E-mail címmel? Biometrikus azonosítóval?). Szerepel az is, hogy a CA mi alapján ítél egy visszavonási kérelmet jogosnak (dedikált telefonvonalon érkező hívás, céges levélpapíron érkező aláírt formanyomtatvány stb).

A tanúsítvány specifikációja nem követeli meg, hogy a tanúsítványban szereplő név valódi név legyen, egyetlen feltételt szab meg: az egyediséget. Ennek alapján az alany neve lehet bármilyen álnév vagy akár sorszám is. Ebben a fejezetben szerepel a tanúsítványban szereplő nevek jelentésével kapcsolatos megkötések leírása, hogy a név milyen jelentéssel bír, és hogy ha a nevek jelentése alapján kiválasztott név már foglalt, akkor hogyan kell feloldani a névütközést.

Működési követelmények

Az azonosításról és a hitelesítésről szóló fejezet csupán az egyes szereplők identitásának meghatározásával foglalkozik. Az alkalmazási területeken előfordulhat, hogy egy tanúsítvány birtoklása már önmagában olyan lehetőségeket nyit meg amelyeket csak egy bizonyos jogosultsággal vagy feladatkörrel rendelkező alkalmazottaknak lehet megadni. Ilyen

esetekben a személyazonosságon túl a jogköröket is ellenőrizni kell. Ez természetesen a informatikai rendszer szemszögéből nem feltétlenül jelenti valamely jogosultságkezelő rendszer alkalmazását, csupán jogi vagy hagyományos (nem elektronikus) értelemben vett jogosultságokról van szó. Hasonló helyzet állhat elő a tanúsítványok visszavonásánál. Gyakran előfordulhat, hogy nem csak a tanúsítvány birtokosa kérvényezheti a tanúsítvány visszavonását. Például ha az alkalmazott kilép, és valamely felettese kérvényezi a tanúsítványának a visszavonását, akkor a CA-nak valahogyan ellenőriznie kell, hogy a kérvényező jogosult visszavonatni a szóban forgó tanúsítványt. Ez a rész írja le, hogy az ilyen további ellenőrzéseket miként kell elvégezni.

Fizikai, ügyviteli és személyi biztonsági szabályozás

A fizikai szabályozások rész szól a nyilvános kulcsú infrastruktúra egyes komponenseinek fizikai védelméről, legyen az szándékos rongálás, emberi hanyagság vagy természeti katasztrófa. A fizikai szabályozásnak ki kell térnie

- A fizikai hozzáférés védelmére, a személyzet fizikai ki és beléptetésére.
- A környezeti hatásoktól való védelemre. (mint például áramszünet, csőtörés, szándékos vagy véletlen tűzeset).
- Másik helyen tárolt biztonsági mentésekre, hogy ha a többi védelmi eljárás kudarcot vall akkor is vissza lehessen állítani a rendszert.
- Hulladékfeldolgozási szabályozásokra, hogy a szeméttel se távozzon felhasználható információ.

Az ügyviteli szabályozásról szóló rész célja, hogy a PKI komponenseket üzemeltető személyzet a feladatát végezze és csakis azt. Senki se legyen képes kompromittálni a rendszer biztonságát. Ennek érdekében két fő elvet alkalmaznak: a legkisebb privilégium elvét és a kötelességek megosztásának elvét. A cél az, hogy mindenkinek éppen annyi jogosultsága legyen, annyi információ álljon a rendelkezésére, amennyi még a feladata elvégzéséhez elegendő. A feladatköröket pedig úgy kell szétválasztani, hogy semelyik feladatkörhöz se kelljen annyi jogosultság, amivel a rendszer biztonságát kompromittálni lehet. A kritikus pontokhoz a rendszerben n-m jogosultságokat is lehet rendelni, például, hogy az adott művelethez ötből három vezető beleegyezése kelljen, vagy, hogy egyszerre legalább két személy jóváhagyása legyen szükséges. Fontos a személyzet képesítése is, hogy tudatlanságból vagy hanyagságból ne tegyenek olyasmit, ami veszélyezteti a biztonságot. Azaz folyamatosan értesíteni kell őket az érvényes biztonsági szabályozásokról, azokat be kell tartatni, illetve a munkájukhoz szükséges PKI ismereteket is el kell sajátítaniuk.

Az alkalmazottak megbízhatósága, becsületessége is kiemelten fontos, hiszen a feladatkörök megosztása ellenére is sikeresen csalhatnak, ha többen is együttműködnek. Ez a rész az alkalmazottak háttérellenőrzésével megbízhatóságának megítélésével és az erre alkalmazott eljárásokkal, előírásokkal foglalkozik. Kitér az anyagi hátterük és korábbi munkahelyek ellenőrzésére és kiszűrni az olyanokat, akik személyes érzéseik, katasztrofális anyagi helyzetük vagy bármilyen oknál fogva hajlamosabbnak ítélhetők a szabályok rosszindulatú megsértésére.

Technikai biztonsági szabályozás

A technikai biztonsági szabályozás rész azokkal a biztonsági előírásokkal, óvintézkedésekkel foglalkozik, amelyek az informatikai rendszerbe is beépülnek akár hardver vagy szoftver szinten, illetve azokkal a szabályozásokkal, amelyek az informatikai rendszer komponenseit hivatottak védeni. A technikai biztonsági szabályozások olyan rendszerelemeket is előírnak, amelyek segítenek kikényszeríteni illetve betartatni a Működési követelményeket illetve az Ügyviteli szabályozásban foglaltakat, ezért a biztonsági célok elérése érdekében létfontosságú, hogy ez a három rész összhangban legyen egymással.

A legfontosabb technikai eszközök a CA privát kulcsát védik. A CA privát kulcsát rendszerint valamilyen, mind biztonság mind korrekt működés szempontjából egymástól élesen elkülönülő bevizsgálási folyamat során igatolt hardveres kriptográfiai modullal szokás védeni. A FIPS 140-1 szabvány rendelkezik a hardveres kriptográfiai modulokkal szemben támasztott követelményekről és a biztonsággal kapcsolatban három különböző szintet is meghatároz. A CA kulcsának védelmére tipikusan kettes vagy hármas szintű biztonsági követelményeknek megfelelő kriptográfiai modult kell használni, míg az RA-k kulcsait elegendő szoftveres, első vagy második szintű hardveres kulccsal védeni. A kliensek kulcsainak védelmi szintjének meghatározásakor elsősorban a kulcsot használó alkalmazás jellegzetességeit kell figyelembe venni. A hardveres modulok elsődleges előnye, hogy a privát kulcs sosem kerül a számítógép memóriájába és védelmére nem az operációs rendszer eszközeit használjuk.

A CA kritikus adatainak tárolásakor kénytelen az operációs rendszerre és alkalmanként felhasználói szoftverekre, mint például adatbázis kezelő rendszerekre hagyatkozni. Alapvető követelmény ezeknek a szoftvereknek a biztonságáról gondoskodni. Másik alapvető követelmény a CA elemeinek otthont adó számítógépes hálózat védelme. Hiába minden egyéb védelmi eljárás, ha a hálózaton keresztül könnyen támadható a rendszer. A rendszer elemeinek minden olyan részét amelynél ez funkcionálisan lehetséges, a offline üzemmódban kell üzemeltetni. A rendszer online üzemmódban működő részeit pedig a szükséges eszközökkel (mint például tűzfalak, behatolás jelző rendszerek) kell védenünk illetve elszeparálnunk a hálózat többi részétől.

Tanúsítvány és CRL profilok

Ez a rész szól a CA által kibocsátott tanúsítványokról és visszavonási listákról. Legfontosabb feladata, hogy rendelkezzen az aláírások és publikus kulcsok azonosítására használt algoritmus objektumazonosítókról. Itt kell feltűntetni azt, hogy a kiadott tanúsítványokban mely X.509 tanúsítvány kiterjesztéseket fognak használni.

A rész feladata a visszavonási listákkal kapcsolatos rendelkezések ismertetése is. Többek között rögzíteni kell, hogy a CA ad-e ki delta CRL-eket vagy használ-e több CRL elosztási pontot, közvetlenül adja-e ki a visszavonási listákat vagy pedig egy másik CA-n keresztül indirekt módon.

Ez a rész szól az irányelvekről szóló információk tanúsítványokban való megjelenési formájáról is. Ezt oly módon kell meghatározni, hogy egyrészt a helyi felhasználók dönteni tudjanak arról, hogy milyen tanúsítványra van szükségük illetve, hogy kódolás illetve az elnevezési konvenciók ne álljanak a más CA-kkal való esetleges együttműködés útjába.

Előírás adminisztráció

A CP-k és a CPS-ek adminisztratív információit tartalmazza. Többek között, hogy az aktuális dokumentum melyik verzió, ki tartja karban az egyes dokumentum verziókat, hol lehet elérni a dokumentum újabb verzióit illetve, hogy az adott CPS melyik CP-t valósítja meg és hogy az erre való alkalmasságot, az adott CP-hez való hűséget milyen módszerekkel kell ellenőrizni. További információkat tartalmaz a dokumentumok új verzióinak publikációs eljárásairól illetve az új verziókról való értesítési folyamatokról is.

Teljesítés vizsgálat és akkreditáció

Mint ahogy a korábbiakban arról már szó volt, az egyes Tanúsítvány Irányelvek és Tanúsítási Gyakorlatról szóló Nyilatkozatok megfelelőségét időről időre vizsgálni kell. Nevezetesen azt, hogy az adott CPS teljesíti a tanúsítványkiadó CP-jének követelményeit, illetve, hogy a gyakorlati folyamatok és eszközök megfelelnek a CPS-nek és hogy a CPS-ben előírt szabályzatokat és előírásokat a személyzet betartja a működés során. Az ilyen jellegű vizsgálatok megkönnyítik a kereszthitelesítéseket és megnövelik a CA-ba vetett bizalmat.

Az ellenőrzések és az akkreditáció elvégzését eleinte az SAS 70 auditoknak megfelelően végezték el. Ez azonban teljesen általános szolgáltatók bevizsgálására szolgál és nem segíti a PKI-specifikus rendszerelemek vizsgálatát. Mind az EU-ban mind Amerikában történtek törekvések a CA-k vizsgálatának egységesítésére és elősegítésére. Amerikában megszületett az e-célt szolgáló "WebTrust for CAs" folyamat, amely az ANSI X9.79-es PKI keretrendszer céljainak ellenőrzésére hivatott. Az EU-ban az egyes államok hivatottak az irányelveknek megfelelően a CA felügyelet kifejlesztésére és alkalmazására. Az EU-s irányelvek nem szabják meg az ellenőrzés üzleti modelljét, az állami és a piaci ellenőrző szervek működését is lehetővé teszik.

12 Irodalom

[1] Johannes Buchmann, Introduction to cryptography, Second edition. Undergraduate Texts in Mathematics. Springer-Verlag, New York, 2004.

[2] M. Bellare and P. Rogaway, The exact security of digital signatures - How to sign with RSA and Rabin. Berlin : Springer-Verlag, 1996., Eurocrypt '96, LNCS, 1070. 399-416.

[3] Budai Balázs Benjámin, E-Government, avagy kormányzati és önkormányzati kihívások az on-line demokrácia korában, Aula Kiadó, 2002.

[4] Buttyán Levente és Vajda István, Kriptográfia és alkalmazásai, Typotex, 2004.

[5] Catalano, Dario. Contemporary Cryptology. : Springer, 2005.

[6] Chaum, David. Blind Signatures for Untraceable Payments. New York : Plenum Press, 1983., Proceedings of Crypto '82, old.: 199-203.

[7] Joan Daemen és Vincent Rijmen, The Design of Rijndael: AES - The Advanced Encryption Standard, Springer Verlag, 2002.

[8] P.J. Denning, ed., *Computers under attack: intruders, worms, and viruses*, ACM Press, 1990.

[9] Whitfried Diffie and Martin Hellman, New direction in cryptography, IEEE Trans. onInformation Theory, 22 (1976), 644-654.

[10] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, IEEE Trans. Inform. Theory, 31 (1985), 469-472.

[11] ElGamal, T. *Cryptography and Logaritms over Finite Fields, PhD Thesis.* Stanford University : 1984.

[12] Gál Zoltán, A Kossuth Lajos Tudományegyetem elektronikus levelező rendszere, Informatika a Felsőoktatásban, Szerkesztők: Herdon Miklós és Pethő Attila, Debreceni Universitas, 1993, 684-690.

[13] J. Folláth, A.Huszti and A. Pethő, DESignIn asymmetric authentication system, ICAI 2007, 55-64.

[14] Dömölki Bálint, szerkesztő, Információs Társadalom Technológiai Távlatai, 2. kötet, Technológiai jelenségek részletes elemzése, NHIT, 2005.

[15] John M.D. Hunter, An information security handbook, Springer, 2001.

[16] Az informatikai hálózati infrastruktúra biztonsági kockázatai és kontrolljai, IHM-MTA SZTAKI, 2004.

[17] Ködmön József, kriptográfia: Az informatikai biztonság alapjai, a PGP kriptorendszer használata, ComputerBooks, 1999/2000.

[18] R.J. McEllice, A public-key cryptosytem based on algebraic coding theory, DSN progress report 42-44, Jet Propulsion Laboratory, Pasadena, 1978. 21.

[19] A.J. Menezes, P.C. van Oorshot and S.A. Vanstone, Handbook of applied cryptography, CRC, 1996.

[20] R. C. Merkle and Martin Hellman, Hiding information and signatures in trapdoor knapsacks, IEEE Trans. on Information Theory, 24 (1978), 525-530. 15.

[21] Kevin D. Mitnick és William L. Simon, A legendás hacker: A behatolás művészete, perfact kiadó, 2006.

[22] The National Election Committee: E-Voting System. Overview. http://www.vvk.ee/public/dok/Yldkirjeldus-eng.pdf., 2005.

[23] The Orange Book: Management of Risk - Principles and Concepts, 2004, http://www.hm-treasury.gov.uk/media/3/5/FE66035B-BCDC-D4B3-11057A7707D2521F.pdf

[24] RFC-1321: The MD5 Message-Digest Algorithm, http://www.faqs.org/rfcs/rfc1321.html 23.

[25] RFC-2251: Lightweight Directory Access Protocol v3, http://www.faqs.org/rfcs/rfc2251.html

[26] RFC 2459: Internet X.509 Public Key Infrastructure Certificate and CRL Profile, http://www.faqs.org/rfcs/rfc2459.html

[27] RFC 2559: Internet X.509 Public Key Infrastructure Operational Protocols - LDAPv2

[28] RFC-3174: US Secure Hash Algorithm (SHA1), http://www.faqs.org/rfcs/rfc3174.html

[29] Ronald Rivest, Adi Shamir, Leonard Adleman, A method for obtaining digital signature and public-key cryptosystems, Communications of the ACM, 21 (1978), 120-126.

[30] Z. Rjašková, Electronic Voting Schemes, diplomamunka, Comenius University, Bratislava, 2002.

[31] Rogaway, P. and T. Shrimpton, Cryptographic hash function basics: Definitions, implications, and separations for preimage resistance, and collision resistance. Berlin : Springer-Verlag, 2004., Fast Software Encryption, LNCS.

[32] Saliné Czinkóczy Anna, Az Internet lehetőségei és veszélyei a felsőoktatásban, Informatika a Felsőoktatásban' 99, Szerkesztők: Csirik János és Herdon Miklós, Debreceni Egyetemi Szövetség, 1999, 49-54.

[33] B. Schneier, Applied cryptography: protocols, algorithms and source code in C, 1996.

[34] Adi Shamir, A polynomial-time algorithm for breaking the basic Merkel-Hellman cryptosystem, IEEE Trans. on Information Theory, 30 (1984) 699-704.

[35] Stinson, Douglas R. Cryptography. Theory and practice. Third edition. Discrete Mathematics and its Applications (Boca Raton). Chapman & Hall/CRC, Boca Raton, FL, 2006.

[36] http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf.

[37] Tilborg, Henk C. A. van, [szerk.], Encyclopedia of Cryptography and Security. Berlin : Springer-Verlag, 2005.