

# Hogyan alakítsunk ki biztonságos hálózatot alacsony költségen?

**Kivonat:** A 80-20-as szabály az információbiztonságban  
**Verzió:** 1.0  
**Dátum:** 2005.07.04.



## A 80-20-as szabály az információbiztonságban

Miután az informatikai biztonság kézbentartása egyre összetettebb és költségesebb feladat, ajánlatos figyelembe venni, hogy a döntéshozók rendelkezésére álló eszközök hatékonysága, illetve hadrendbeállításuk erőforrásigénye igen széles skálán változik. Érdemes alaposan végiggondolni, mely intézkedésekből és eszközökből állítjuk össze biztonságtechnikai portfóliónkat. Nyilvánvaló, kifizetődő azokkal az alapvető intézkedésekkel kezdeni, melyek nem igényelnek hatalmas ráfordításokat, ám ennek ellenére jó eredményeket lehet velük elérni.

### A Pareto elv

Fontos és kevésbé fontos dolgok elkülönítése idegen feladat lehet egy mérnök számára, ezért a menedzserek eszköztárából kölcsönzünk egy nem éppen egzakt, ám igen hatékony eszközt. A Pareto elv általánosságban mondja ki, hogy tényezők egy kisebbik része (körülbelül 20 százalékuk) felelős a jelenségek nagyobbik részéért (körülbelül 80 százalékáért). Például, az ügyfelek 20 százaléka hozza a bevétel 80 százalékát, vagy egy projekt lépéseinek 20 százaléka viszi el az erőforrások 80 százalékát.

A Pareto elv érvényesülése az élet számtalan területén figyelhető meg, annak ellenére, hogy természettudományos alapon eddig nem sikerült bizonyítani érvényességét. A társadalomtudományok azonban elismerik, és előszeretettel alkalmazzák is, mint a csoportos emberi viselkedés leírásának egyik módját. Mivel az IT rendszerek biztonsági hiányosságai emberi tévedéseken alapulnak, illetve ezeket emberek használják ki különbözőféle károkozásokra, jó eséllyel az informatikai biztonság is modellezhető a Pareto elv segítségével.

Vizsgáljuk meg ebből a nézőpontból a problémát! Megfigyelhetjük, hogy mindig vannak olyan sebezhetőségek, melyek kihasználása divatos és népszerű a támadók (crackerek) között. A legtöbb esetben ezeket a biztonsági réseket használják belépési pontként a rendszerünkbe, miután ezek széles körben elterjedtek és könnyen kihasználhatók. Az olyan automata károkozó kódok például, mint a Code Red vagy a Nimda voltak, mind jól ismert biztonsági hiányosságokat használtak ki, melyhez már hónapok óta elérhetőek voltak a javítások is.

Jelentsük ki hát bátran, hogy a biztonsági hiányosságok körülbelül ötöde felelős a károk négyötödéért, és ezentúl kezeljük ennek tudatában a kérdést. Más szóval, vegyük figyelembe, hogy a biztonsági rések legalapvetőbb 20 százalékának befedésével megoldhatjuk a biztonsági problémáink 80 százalékát!

Ha ezt megtesszük, védelmet élvezhetünk az Internetet pásztázó automatikus károkozóktól és azoktól a kölyköktől, akik minden különösebb ok nélkül, pusztán szórakozásból hatolnak be mások rendszereibe, hogy ott kisebb-nagyobb galibát okozzanak. Ezenkívül, a profi bűnözőktől is jóval több időt és kifinomultabb módszereket követelünk meg a sikerhez, ami igen gyakran érdektelenné tehet bennünket a csekély haszon miatt. És közben ne feledjük, mindezt a biztonsági szintet igen alacsony költségen érhetjük el! Nem kell mást tennünk, mint követni a cikk következő három fejezetében foglalt alapvető intézkedéseket.

Fokozottan ki kell azonban jelentenünk, hogy azok a rendszerek, amelyeknek feltörése tetemes anyagi haszonnal jár, korántsem lesznek biztonságban a következőkben kifejtett egyszerű intézkedések által. Ha valaki direkt a mi rendszerünkbe akar behatolni, és erre a feladatra jelentős erőforrásokat képes mozgósítani; nos, ebben az esetben tovább kell haladnunk a „tökéletes biztonság” felé vezető költséges úton. Még hozzá olyan mértékben, hogy a támadónak a magas költségek és a jelentős kockázat miatt ne érje meg megpróbálni a betörés végrehajtását. Ebben az esetben is igaz azonban, hogy alulról felfelé kell építkezni, azaz a cikk további részeiben taglalt intézkedések végrehajtása nélkül nincs értelme komolyabb biztonsági beruházásokat tenni.



## Felesleges szolgáltatások eltávolítása

A modern operációs rendszerek igen kifinomult és összetett halmazai különböző funkciójú komponenseknek és önálló szoftvereknek. Ezek fejlesztését különálló munkacsoportok végzik, melyek gyakran különböző vállalatok szervezeteibe tagolódnak be, és így különböző vállalati kultúra szerint működnek. Egy ilyen komplex rendszer természetéből adódóan hordozza magában a hiba lehetőségét, minek a veszélyét csak fokozza, hogy mára a legtöbb szoftver hálózaton keresztül is kommunikál.

Legyen szó FTP-ről, HTTP-ről vagy bármely másik hálózati szolgáltatásról, ezek és az őket használó szoftverek hemzsegnek az ismert és még feltáratlan biztonsági résektől, melyek mindegyikének betömése tulajdonképp lehetetlen feladat, ha másért nem, hát a tetemes költségek miatt.

Mivel a legtöbb számítógépet korlátozott számú feladatra állítják munkába, a funkciók jelentős része kihasználatlan marad és csupán felesleges biztonsági kockázatot jelent az üzemeltető számára. Mindebből következik, hogy a legelső - mindent megalapozó - biztonsági intézkedésünk a felesleges szoftverkomponensek eltávolítása és a nem használt hálózati szolgáltatások letiltása kell legyen.

Az operációs rendszerek megtisztítása viszonylag egyszerű, egy képzett rendszergazda gond nélkül elvégzi. Ezzel szemben a hálózati szolgáltatások korlátozása nem evidens feladat, hiszen egyrészt az internetes infrastruktúra alapvetően nem tartalmaz olyan elemet, mely ellenőrzi a protokollok betartását, másrészt vannak olyan gyakran használt, többcélú protokollok, melyekre mindenképp szükség van, ezért – biztonsági hiányosságaik ellenére – nem lehet őket kikapcsolni.

A hálózati forgalom ellenőrzésére és korlátozására mindenképp tűzfalat kell alkalmaznunk. A tűzfal-technológia megválasztása alapjaiban határozza meg a rendszergazda mozgásterét. Míg egy csomagszűrő tűzfal csupán a csomagok feladóját és címzettjét (beleértve a portot is) képes ellenőrizni, addig egy proxy tűzfal már a kapcsolatok figyelésére is képes. A szolgáltatások precíz testreszabásához pedig már egy alkalmazásszintű proxy tűzfalra lesz szükségünk, mely a hálózati forgalmat utasítás szinten képes értelmezni és korlátozni. Utóbbi segítségével oldhatók meg a fentebb már említett, kevésbé biztonságos protokollok lehetőségeinek korlátozása.

## Naprakész hibajavítások

Sokan gondolják azt, ráérnek telepíteni egyes biztonsági rések javításait, hiszen eddig is nagyon jól megvoltak nélküle. Pedig egy javítás megjelenése azt is jelenti, hogy a hiba széles körben nyilvánosságra került. És nem csak a hiba leírása, de részletes ismertető is, melyek a kihasználás lehetőségeit taglalják. Valójában a folt (patch) publikálásától annak telepítéséig eltelt időszakban van a rendszerünk a legnagyobb veszélyben.

A cél tehát a kettő között eltelt idő a lehető legrövidebbre való csökkentése kellene, hogy legyen. Kapkodni azonban mégsem érdemes. Ugyanis, csakúgy, mint az eredeti szoftver, a hibajavító csomag is gyakran hibás. Sajnos a nem megfelelően működő foltok legalább akkora veszélyt jelentenek, mint maguk a hibák.

A fent említett okokból kifolyólag olyan jól megtervezett folyamatot kell kialakítani, mely állandóan monitorozza a gyártók weboldalait új javítócsomagok után, szabályozza az automatikus frissítés lehetőségeit, és tesztelést ír elő a kritikus rendszerösszetevők esetén.

A hibajavítások menedzselése talán a legnagyobb kihívás a cikkben tárgyalt intézkedések közül. Azonban, aki a felesleges komponensek eltávolítását maradéktalanul végrehajtotta, annak jóval könnyebb dolga lesz.



## Erős autentikáció

Ha egy jól megtervezett IT határvédelmen keresünk könnyen kihasználható rést, legegyszerűbb, ha megszerezünk az egyik alkalmazott jelszavát. Valószínűleg elég lesz megkérdeznünk tőle, és ő készségesen bediktálja nekünk a telefonba. Hihetetlen, ugye? Pedig a támadók többsége így jár el. Egyszerűen felhívnak valakit az informatikai osztály nevében, azzal a mesével, hogy ellenőrzik a jelszavakat. Egy-két bennfentesnek tűnő vicc és név említése után, az esetek elsőprő részében a támadók sikerrel járnak. Ha a vállalat túl kicsi ehhez a cselhez, vagy az alkalmazottakat kellően felkészítették, még mindig sikerrel járhatnak egy egyszerű teleobjektíves kamera segítségével, az irodával szemközti ablakból.

Sokáig sorolhatnánk még az egyszerűbbnél-egyszerűbb technikákat egy jelszó megszerzésére, de inkább nézzük, hogyan védekezhetünk!

Legelőször is követeljünk meg erős jelszavak használatát a felhasználóktól, hogy megnehezítsük azok kitalálását vagy próbálgatással való megfejtését. Egy erős jelszó legalább 10 karakter hosszú, és vegyesen tartalmaz kis- és nagybetűt, számokat és speciális karaktereket. Ennél is jobbak az algoritmusok által generált kódok, melyeket azonban gyakran felírják a felhasználók, például a billentyűzetük aljára.

Alapvető igazság, hogy minden jelszó feltörhető vagy megszerezhető, ha elég idő van rá; ezért a biztonsági szinttől függően 1-3 havonta cseréljük őket. Arra is ügyeljünk, hogy a felhasználók ne használhassák újból egy régi jelszavukat.

A legjobb megoldás, ha USB kulccsal egészítjük ki a jelszavas védelmet, melynek ára ma már minden vállalat számára elérhető.

Tovább fokozhatjuk a biztonságot, ha szegmentált hálózatot alakítunk ki, és az egyes jelszavakhoz külön-külön szabályozzuk a jogosultságokat. Így, még ha illetéktelen kezekbe kerül is egy jelszó, a támadók csak korlátozott jogokat szerezhetnek a hálózat egy jól elhatárolt részében.

## Végszó

Az információbiztonságnak nem feltétlen kell nyomasztó és költséges problémává válnia. Biztonságtechnikai portfóliónk összeállításakor alkalmazzuk a Pareto elvet, és kezdjük a munkát a leghatékonyabb intézkedésekkel. A cikk tanácsait követve rendkívül erős alapot hozhatunk létre, mely akár önmagában is elegendő biztonságot nyújthat számos vállalat számára. Vegyük figyelembe azt is, hogy a biztonságot nem elég kialakítani, de folyamatos munkával fenn is kell tartani.



Kérdéseivel, észrevételeivel keresse meg cégünket az alábbi címen: BalaBit IT Security, 1116 Budapest Csurgói út 20/B  
Telefon: 06 1 371 0540 Fax: 06 1 208 0875 E-mail: [info@balabit.hu](mailto:info@balabit.hu) Web: <http://www.balabit.hu>

© 2005 BalaBit IT Security. Minden jog fenntartva.

A dokumentumra vonatkozó jogi kitételeket a következő weboldal tartalmazza:  
[http://www.balabit.com/products/zorp/docs/legal\\_notice.bbq](http://www.balabit.com/products/zorp/docs/legal_notice.bbq)