UNIX RENDSZEREK BIZTONSÁGI KÉRDÉSEI

Dr. Krausz Tamás, *kuka@dragon.klte.hu*
KLTE ISZK

Abstract

This paper is about unix system security. The paper is not intended to be a hacker's guide, it is mainly for system administrators, who are not always alert enough. First, some well known cases were described, then the necessary measures that the system administrartor must consider. The main themes are the following: users and passwords, groups and the superuser, the filesystem, finding suid and sgid files, suspicious accounts, backups, system logging, network security, nfs , kerbereros, secure rpc, firewalls, encryption, phisical security.

Történeti áttekintés, néhány eddigi eset leírása.

A unix operációs rendszer nagy mértékben terjedt el, számos különbözõ verziója létezik. Szinte minden nagyobb gyártónak van saját verziója, ezenkívül létezik nagy mértékben elterjedt ingyenes verzió is. A legismertebb verziók a teljesség igénye nélkül: Sun Solaris, HPUX, AIX, DEC Unix, Linux.

A biztonságos mûködtetés kiemelkedõ fontossággal bír a mai világban, ahol a gépek nagyrésze hálózatra van kötve. Az egyetemi környezet különösen veszélyes, mert sok a ráérõ jóképességû diák. A másik nagyon veszélyeztetett helytípus, ahol az információ rendkívül értékes, inkább professzionális hackerek célpontjai. A unix, mint operációs rendszer megfelelõen biztonságos. A biztonságos mûködtetés két fõ tényezõje a megfelelõen képzett és résen levõ rendszergazda, valamint a rendszer iránt felelõsséget érzõ felhasználók. Elõször nézzük meg, mit értünk a biztonság alatt!

Azt mondhatjuk, hogy a számítógép biztonságosan mûködik, ha a rajta levõ szoftverek az elvárásnak megfelelõen mûködnek, a felvitt adataink bármikor elérhetõk maradnak számunkra és nem olvashatóak illetéktelenek számára. Ezen definíció alapján a biztonság több mint védekezés a betörések ellen. Valójában ha a felhasználó elveszíti értékes adatait , számára mindegy, hogy ez hibás rendszerszoftvernek, vírusnak, bosszúálló alkalmazottnak, hardware hibának vagy szoftveres feltörésnek az eredménye.

A máig is talán legismertebb eset a Morris féle Internet worm, ami szolgáltatásokat tett lehetetlenné, de nem volt destruktív olyan értelemben, hogy nem törölte le sem a rendszer, sem pedig a felhasználók fájlait. Nálunk nagyjából most érjük el azt a fejlettségi szintet, hogy érezzük mit is jelenthetett ez valójában annak idején.

Egy viszonylag frissebb eset, amikor Kevin Michnik nem elégedett meg több száz hitelkártya kódjának feltörésével, hanem feltörte az egyik legelismertebb biztonsági szakértõ Tsutomu Shimomura gépét és bizonyos állományait nyilvános ftp szerverre kirakta. Michnik rádiótelefont használt és több gépen keresztül jutott el a kívánt célpontokig. A hackert Tsutomu Shimomura kapta el az FBI segítségével, s Michnik végül gratulált Tsutomu Shimomura-nak.

Felhasználók szokásainak és jelszavainak szabályozása.

A unix - mint általában a többfelhasználós operációs rendszerek - a felhasználókat egy felhasználói névvel (valójában uid-del) azonosítja. A felhasználói névhez jelszó tartozik, melynek segítségével beléphetünk a rendszerbe. A kevésbé biztonságos rendszerekben a jelszó egy /etc/passwd nevû fájlban található kódolt formában. Sajnálatos módon ez a fájl minden felhasználó számára olvasható kell legyen más programok miatt. A biztonságosabb unixokban a jelszó kódolt formája egy /etc/shadow nevû állományban található, amihez csak

a 0 id-vel rendelkezõ felhasználónak van olvasási joga. Az egyik leggyakrabban kihasznált biztonsági lyuk a felhasználók triviális jelszavai. A rendszergazda kialakíthat minimális jelszó hosszat, idõnként lejárhatnak a jelszavak, de ami valójában fontos, az a felhasználók megfelelõ jelszó választása.

Tipikusan rossz jelszavak: nevünk, rokonaink neve, munkatársaink neve, számítógépünk neve, rendszámunk, bárki születési dátuma, bármilyen velünk kapcsolatban álló információ, értelmes angol szó, helynév, ugyanazon betûk, egymás melletti billentyûk (mint qwerty), az összes elõbbi fordított sorrendben, az összes elõzõ után egy számjegy. A jó jelszavak vegyesen tartalmaznak nagy és kis betût, van bennük számjegy, könnyû õket megjegyezni (ne kelljen leírni õket), hét-nyolc karakter hosszúak.

Amit a rendszeradminisztrátor tehet, hogy õ maga futtat le idõnként jelszófeltörõ programokat, és a könnyen kitalálható jelszóval rendelkezõ felhasználókra rákényszeríti a megfelelõ jelszó választását. Erre alkalmas többek között a nyilvánosan elérhetõ crack programcsomag.

Felhasználók csoportosítása, a superuseri hozzáférés szabályozása.

A 0-ás id-vel rendelkezõ felhasználó, melynek neve szokás miatt root, többek között megteheti a következõket: megváltoztathatja a futó processzek prioritásait, signal-t küldhet bármely processznek, megváltoztathatja a rendszerben levõ hard limit-eket, ki- és bekapcsolhatja az elszámolást, más felhasználóként mûködhet, lezárhatja a rendszert, átállíthatja a dátumot, írhatja és olvashatja a fizika memóriát, hálózati szolgáltatásokat indíthat tetszõleges porton, átkonfigurálhatja a hálózatot, bármely fájl fölött szabadon rendelkezhet, mountolhat és unmountolhat, felhasználókat hozhat létre és szüntethet meg. Amit a superuser se tehet meg: csak olvasható fájlrendszer módosítása, jelszavak megfejtése (bár átírhatja a login és su parancsokat saját verzióra), a kernelben várakozó állapotban levõ processzek megszüntetése (bár lezárhatja rendszert).

Mint láthatjuk, a superuser lényegében bármit tehet a rendszerrel, ezért biztonsági szempontból a legfontosabb ezen account védelme. Sok programhiba eredményezheti, hogy normál felhasználó superuseri privilégiumokhoz juthat. A másik lényeges dolog, hogy a rendszeradminisztrátor csak kimondottan rendszerrel kapcsolatos feladatokra, és ne saját személyes céljaira használja ezt az accountot. Célszerû root hozzáférést csak a konzolról engedélyezni, valamint a su parancsot log-olni.

A file rendszer biztosítása, megfelelõ jogosultságok beállítása, SUID, SGID programok ellenõrzése.

Néha szükség van arra, hogy normál felhasználók megváltoztassanak általuk nem szabadon hozzáférhetõ fájlokat, erre példa a jelszó megváltoztatása. A unixban erre a problémára az úgynevezett suid programokat használják, melyek effektív uid-je a futás idejére a program tulajdonosára változik a program futtatója helyett, ilyen program például a passwd parancs. Különösen veszélyes lehet ha valaki egy root tulajdonú suid fájlt tud megszerezni céljainak. Célszerû ezt néha a find parancs -perm opciójával végigkeresni.

Gyanús accountok kiszûrése.

Célszerû idõnként az /etc/passwd illetve a /etc/shadow fájlt áttanulmányozni, hogy nincs-e a rendszerben jelszó nélküli account, vagy 0-ás uid, esetleg nem megfelelõ csoportba tartozó felhasználó.

Backup fajtái, tervezése , esetleges visszaállítás módjai.

Rendszerösszeomlás esetén a bajt és a katasztrófát megkülönbözteti a rendszeres kimentés. A hiba oka lehet: felhasználói vagy rendszeradminisztrátori hiba, hardware vagy szoftver hiba, lopás ,szoftveres betörés, természeti katasztrófa.

Teljes backup-ot, inkrementális backup-ot rendszeresen kell végeznünk. Idõnként célszerû próba visszaállításokat végezni az esetleges szoftver vagy szalag hibák idõbeni észleléséhez. Ajánlatos a kimentett szalagok biztonságos és fizikailag más helyen való tárolása, valamint kb. száz kimentésenként a szalagok cseréje.

A rendszer logolás beállítása, és a log file-ok tanulmányozása.

A log fájlokat az esetleges betörési kísérletek felfedezésére, hibás mûködés észlelésére vagy esetleg már megtörtént esetek kinyomozására használhatjuk. A legfontosabb log fájlok helye és tartalma némileg változik a különbözõ unix verziókban. Általánosságban a következõ log fájlok a legfontosabbak:
lastlog: legutolsó bejelentkezés ideje
utmp, wtmp: rekordok keletkeznek be- illetve kijelentkezésnél, amelyek mezõi: usernév, terminal név, hoszt név, pid, stb.
acct: lejegyez minden felhasználók által futtatott programot
messages: különféle hibák

A hálózati szolgáltatások biztonsági kérdései.

Egyre inkább a unix rendszereket hálózaton keresztül használjuk, sok esetben csak a konzol terminál az egyetlen nem hálózati hozzáférés. A hálózaton a csomagok sok esetben kódolás nélkül haladnak, és a mai hardware eszközökkel megfelelõ gyorsaság érhetõ el a csomagok szûrésére. Az egy szegmensen levõ felhasználókat így egymás ellen nem tudjuk védeni. Fontos szempont, hogy a privilegizált felhasználók olyan router lábra kerüljenek melyhez nincs nyilvános hozzáférés.

Telnet, ftp, rexec, finger tftp, X window rendszer szabályozása.

A unixban alapvetõen kétfajta szerver szolgáltatás van: az egyik állandóan futó démonként jelenik meg, mint például az nfsd, másrészt olyan szolgáltatások, melyekre a rendszer szükség esetén indít el démont, például telnetd.
A /etc/inetd.conf fájl írja le a szükség esetén induló démonokat, a szolgáltatások pedig megtalálhatók /etc/services nevû állományban. A 0-1023 portokon csak a superuser szolgálhat ki. Az úgynevezett r parancsok (rlogin,rsh,rexec) elõnye, hogy a hálózaton nem megy át a jelszó, valamint kényelmes használni õket, de megbízható hosztok vagy felhasználók beállítását kívánja meg, ezért biztonsági szempontból veszélyesek. A megbízható gépre való bejutás után szabad bejárást kaphatunk a másik gépre, s mind a gépet mind a felhasználót lehet hamisítani. Célszerû ezért a rendszeradminisztrátornak idõnként a .rhosts fájlokat átvizsgálnia. Amennyiben anonymous ftp-t vagy tftp-t üzemeltetünk, csak egy erre a célra kijelölt könyvtártól lefelé engedélyezzünk hozzáférést. Az X window rendszerben az xhost paranccsal engedélyezzük a szerverünkön, hogy különbözõ kliensek megjelenhessenek. Elképzelhetõ olyan kliens, ami egy nem látható ablakot tesz ki és elmenti a billentyûzet leütéseinket, ezért csak a szükséges minimumra engedélyezzük az xhost parancsot.

NFS szolgáltatás, Kerberos, Secure RPC.

Az NFS a SUN által kifejlesztett szolgáltatás, mely lehetõvé teszi akár különbözõ operációs rendszert futtató gépekre a fájlrendszerek közös használatát. Mivel biztonsági szempontból sok veszélyt rejt magában, célszerû megfogadni az alábbiakat: csak a szükséges fájlrendszereket tegyük ily módon elérhetõvé, korlátozzuk esetleg a szolgáltatást egyes gépekre, lehetõleg csak olvashatóan exportáljuk a fájlrendszereinket.

A kerberos DES titkosítást használó rendszer melyet az MIT-n fejlesztettek ki. Az Athena projekt keretében megvalósították az NFS, rlogin passwd , email esetén. Hátrányai: szükséges egy biztonságos Kerberos szerver , a hálózati szolgáltatásokat egyenként kell módosítani hozzá.
Secure RPC -t a SUN fejlesztette ki, mind a szerver, mind a kliens rendelkezik nyilvános és titkos kulccsal, a nyilvános kulcsokat a NIS-en keresztül hirdetik és a saját titkos kulcsukkal létrehozzák ugyanazt a kommunikációs kulcsot, melyet a további információ cserénél alkalmaznak. Elõnyei között említhetõ, hogy a felhasználói jelszó sosem megy keresztül a hálózaton, a titkos kulcs csak egyszer, a jelszóval kódolva halad át a hálózaton. Hátrányok: csak installált NIS-sel használható, minden klienst módosítani kell, csökken a hatékonyság.

Tûzfalak és fajtáik.

Mind a secure RPC, mind a kereberos jelentõs szoftver módosításokat kíván. Egy másik alternatív megoldás, hogy tûzfal géppel védjük a belsõ hálózatunkat a külsõ világgal szemben. A tûzfal alapvetõen két részbõl áll: az egyik, mely a két hálózat között átadja csomagokat, a másik pedig, ami a tûzfalon belül levõ hálózatra nem engedi be a csomagokat, csak a tûzfalra jövõket, illetve a belsõ hálózatról csak a tûzfalra címzett csomagokat engedi ki. Legegyszerûbb és egyben legkevésbé költséges megvalósítás lehet egy unix operációs rendszert futtató gép két hálózati kártyával. Ne futtassuk ezen a gépen a unix routed démont.

Egyéb biztonságot növelõ eljárások fizikai védelem, kódolás titkosítás.

Célszerû a számítógépünket fizikailag elzárt helyen tartani, ez nem csak a lopást és a rongálást nehezíti, hanem az adatokhoz való illetéktelen hozzáférést is. A hálózat kialakításánál is el kell kerülni, hogy a vezetékek könnyen hozzáférhetõek legyenek, ahol lehet, a legjobb megoldás az üvegkábel alkalmazása. Az üvegkábel a legnehezebben megbontható, illetve lehallgatgatható.
Minden a számítógépen levõ érzékeny információnkat érdemes titkosítani. Az információ illetéktelen kezekbe sokféle módon kerülhet. Az elõbbieken kívül a rendszeradminisztrátor is el tudja olvasni fájlainkat. Alapvetõen kétfajta titkosítás létezik: a titkos kulcsú kódolás és a nyilvános kulcsú kódolás. A unixban a legismertebb titkos kulcsú kódolás a crypt(1) és a DES, a nyilvános kódú pedig az RSA. Feltörés ellen megfelelõ biztonságúnak csak a DES és az RSA számít. Levelezési gyakorlatban megfelelõ lehet egy tömörítés és egy crypt(1) használata. RSA alapú a nemrég nyilvánosságra hozott PGP, mely különbözõ ftp szerverekrõl szabadon letölthetõ.

Irodalomjegyzék:
Simson Garfinkel and Gene Spafford:Practical unix security
Rik Farrow:Unix system security
Grampp and Morris: Unix operating system security