Linuxos kiszolgálót mindenkinek! (6. rész)

A SuSE Linux, mint kiszolgáló, kisvállalati és otthoni környezetben.

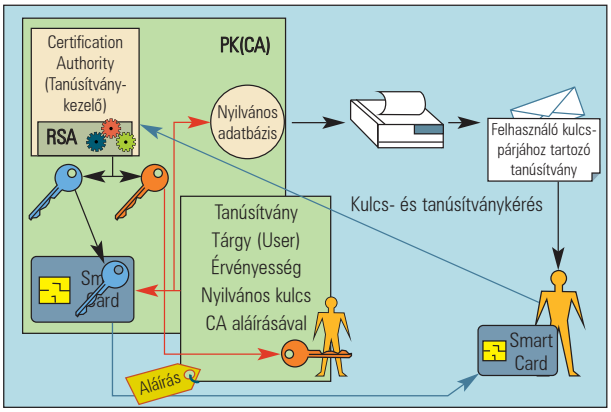
Cikksorozatunk előző részében egyszer már terítékre kerültek a tanúsítványok az SSL kiterjesztésű POP3 és IMAP protokollok kapcsán. Most kicsit általánosabban is foglalkozunk a tanúsítványokkal és a tanúsítványkezelő rendszerekkel. Saját tanúsítványkezelő rendszert telepítünk, elkészítjük az úgynevezett gyökértanúsítványt és saját tanúsítványokat is létrehozunk, amelyeket elhelyezünk a tanúsítványkezelő rendszerünkben. Ehhez szükségünk lesz az OpenSSL csomagra, amit a YaST-tal a már megszokott módon telepíthetünk. Az OpenSSL csomaghoz leírás is tartozik, amit felpakolhatunk. Ha mindezek végére értünk, tekintsük át a tanúsítványkezelő rendszerek felépítését és szerepét.

A tanúsítványkezelő-rendszerek szerepe

Az interneten tengernyi anyag található és rengeteg szolgáltatás vehető igénybe, ezt már mindannyian tapasztalhattuk. Az internet a szabad hozzáférésnek köszönhetően a jövő tudásbázisává és a kapcsolattartás csatornájává lehet. Ugyanakkor ez a szabad hozzáférhetőség kérdéseket is felvet, például a következőkön érdemes elgondolkodni. Mi szavatolhatja az interneten található adatok hitelességét, vagy az igénybe vett kapcsolati csatornák megbízhatóságát? A számítógépes hálózatok fejlődése lehetővé tette, többek között, az elektronikus kereskedelem és pénzforgalom kialakulását. Ezek a folyamatok azonban olyan adatbiztos környezetet igényelnek, amelyek könnyen kezelhetőek és törvényileg elfogadottak. A könnyű kezelhetőséget az elmúlt évek technikai fejlődése tette lehetővé, míg a törvényi szabályozás szerepét a Magyarországon 2001 szeptemberében életbe lépett 2001. évi XXXV. törvény hivatott betölteni.

A törvényi szabályozás létrejöttével elhárult minden akadály a PKI (Public Key Infrastucture, azaz a nyilvános kulcsú infrastruktúra) rendszerek hivatalos használata elől. A PKI rendszerek olyan környezetet teremtenek, amelyek lehetővé teszik a törvények által elfogadott kétkulcsú harmadik személyes hitelesítési és adatbiztosítási eljárások használatát számítástechnikai környezetben.

A PKI rendszerrel az elektronikus adatok kódolása és hitelesítése egységes módon történik. A rendszert használók mindegyike rendelkezik két kulccsal, egy nyilvános és egy titkos kulccsal. A nyilvános kulcs egy olyan kulcs, amit a felhasználó közzé tesz, ilyen módon az őt megszólító felek számára ismert. A titkos kulcsot a felhasználó a külvilágtól szigorúan elzárva tartja, amely egy állományban is lehet, amit CD-n vagy hajlékonylemezen őriz, de akár egy chipkártya is tartalmazhatja. (Például jelenleg ilyen chipkártyás megoldást használ hazánkban az APEH a kiemelt adózók által benyújtott bevallások digitális aláírásához.)



A két alapvető folyamat: a hitelesítés és a titkosítás (a módszer matematikai hátterét sorozatunk előző részében ismertettem röviden).

Titkosítás

A nyilvános kulcsú titkosítás nagy előnye több más titkosítással szemben, hogy a kódolás és a dekódolás nem ugyanazzal az eszközzel, kóddal történik, illetve az egyik kód a másikból jelenlegi matematikai tudásunk szerint nem állítható elő.

Így egy adott üzenet vagy adat kódolása úgy történik, hogy a küldő fél kódolja az adatot a címzett nyilvános kulcsával, így olyan adat jön létre, amely csak a címzett titkos kulcsával lesz visszafejthető. Mivel azonban a titkos kulcsot elvileg csak a címzett ismeri, ezért a titkosított adat más számára nem férhető hozzá.

Hitelesítés

Hitelesítés esetén arra van szükségünk, hogy megbizonyosodhassunk róla az adott üzenet küldője, vagy adott dokumentum szerzője valóban az a személy-e, akitől az adatot várjuk. Ennek módja, hogy az üzenet írója vagy a dokumentum szerzője az üzenetét a saját titkos kulcsával aláírja, majd a hitelesség ellenőrzéséhez a címzett oldalán ezt az aláírást kell dekódolni a küldő nyilvános kulcsával. Az alá-

írás gyakorlatilag az átküldött dokumentum leképzése egy olyan számmá – ellenőrző kóddá –, amely a dokumentum karaktereitől függ, illetve ez jellemzi az adott dokumentumot. Az adott dokumentumban bármilyen változtatás maga után vonja e számérték megváltozását, így a módszer arra is képes, hogy felfedezze azt, ha valaki illetéktelen módon az aláírás után módosította az üzenetet.
Amikor a címzett megkapja az üzenetet, a küldő nyilvános kulcsával dekódolja az aláírást, így meg tudja nézni, hogy valóban attól a feladótól kapta-e az üzenetet, akitől várta. Ha dekódolta az aláírást, a megérkezett üzenet ellenőrző kódját elkészítve megbizonyosodhat arról, hogy az átküldött dokumentumban történt-e változtatás a kézbesítés során.
Az eddig elmondottak logikusnak, fogalmazhatunk úgy is, egyszerűnek tűnnek. Felmerül azonban az a kérdés, honnan tudjuk, hogy a hozzánk eljutatott nyilvános kulcsok valóban ahhoz a személyhez tartoznak-e, aki hozzánk elküldte, illetve honnan tudjuk, hogy a kiosztásnál nekünk átadott titkos kulcs valóban csak a mi birtokunkban van, ahhoz más nem férhetett hozzá. E gond feloldására hozták létre a harmadik személyű hitelesítést, az úgynevezett tanúsítványkezelőket (CA – Certification Authority).
E tanúsítványkezelő rendszereknek feladata a kulcspárok kiosztása, a tanúsítványok kiállítása a kulcsok hitelességéről, valamint a megbízhatatlanná vált kulcsok visszavonása és nyilvántartásuk. A tanúsítványkezelőt akár nevezhetjük egy elektronikus közjegyzőnek is, akinek az a feladata, hogy a rajta keresztülmenő üzeneteket, esetünkben a kulcspárokat hitelesítse. Ebből természetesen következik az is, hogy egy tanúsítvány csak akkor tekinthető hitelesnek, ha megbízunk az adott hitelesítő szervezetben. E bizalom hiányában az általa kibocsátott tanúsítványok sem tekinthetőek megbízhatónak.

A rendszer felépítése

Kezdjük talán a legelején a tárgyalást, nézzük hol és miként lehet kulcspárokat előállítani. Alapvetően két kulcspár-előállítási mód létezik: az egyik a központosított, a másik a decentralizált kulcslétrehozás.
A központosított kulcs-előállításról akkor beszélünk, ha a kulcspárt a tanúsítványkezelő (CA) állítja elő és a titkos kulcsot kézbesíti a tulajdonosának, valamint a nyilvános kulcsot elhelyezi valamilyen hozzáférhető helyen, például valamilyen címtárban vagy névtárban.
Decentralizált kulcs-előállításról akkor beszélünk, ha a kulcspár nem a tanúsítványkezelő, hitelesítő szervezet oldalán készül, hanem ügyféloldalon. Ez történhet olyan módon, hogy a felhasználó saját maga előállít egy kulcspárt, azt hitelesítésre beadja valamelyik hitelesítő szervezethez, amely a bejegyzés (registration) után tanúsítványt bocsát ki a kulcspárhoz. A decentralizált kulcselőállítás másik módja, amikor egy chipkártyát használunk a kulcspár készítéséhez. Ilyenkor a chipkártya tartalmazza a titkos kulcsot és a chipkártya segítségével a nyilvános kulcs kerül előállításra és hitelesítésre a tanúsítványkezelő rendszer által. Ez utóbbi módszert nevezik fedélzeti (on-board) kulcslétrehozásnak. E módszer abból a szempontból célszerű, mert a titkos kulcs soha nem hagyja el a chipkártyát, így annak az esélye, hogy a titkos kulcs illetéktelen kezekbe kerül itt a legkisebb.
Természetesen a decentralizált kulcs-előállításnál a hitelesített kulcs beküldésére is vannak megbízható módszerek, ezek közül az egyik, ha a hitelesítésre szánt kulcsot a hitelesítő szervezet nyilvános kulcsával titkosítva küldjük el a hitelesítő szervezetnek, hiszen ez csak a hitelesítő szervezet titkos kulcsával lesz kibontható. A hitelesítés után a CA számunkra a titkos kulcsot a saját nyilvános kulcsunkkal aláírva küldi vissza, mivel ezt csak mi fogjuk tudni kinyitni a saját titkos kulcsunkkal. Ha még azt is szeretnénk, hogy más ne tudjon nekünk titkos kulcsot küldeni a hitelesítő szervezet nevében, akkor ez úgy oldható meg, ha az üzenetet a hitelesítő szervezet aláírja a saját titkos kulcsával, mivel így az aláírást kibontva a CA nyilvános kulcsával a küldő egyértelműen azonosításra került.
A fent leírtakból már látszik a rendszer működésének előnye, ez pedig az, hogy a két üzenetváltó félnek nem kell az üzenetváltást megelőzően kulcsot cserélni – például személyes találkozás, vagy megbízható telefonvonal útján – a biztonságos kapcsolattartáshoz, ez megoldható az első üzenetváltással.
Most nézzük a PKI rendszer vázlatos felépítését. A PKI rendszer áll egy tanúsítványkezelő rendszerből (CA), egy regisztrációs ügynökből (Registration Agent, RA) és kapcsolódik hozzá egy címtár, valamint egy ügyfél, aki a szolgáltatásokat igénybe veszi.

A CA

A tanúsítványkezelő rendszerek nemzedékeinek fejlődése jól nyomon követhető azon, hogy CA és az RA rész mennyire válik el egymástól. A kezdeti tanúsítványkezelők még nem különböztették meg a regisztrációs ügynököt, minden egy modulba volt összevonva. A fejlődés során az RA-k egyre több felügyeleti feladatot vettek át, míg a CA egyre inkább csak a tanúsítványok kiadásával foglalkozik.
A tanúsítványkezelő (CA) feladatai:

- RSA kulcspárok előállítása (központosított kulcselőállítás esetén).
- X.509 tanúsítványok kibocsátása.
- Nyilvános kulcsok személyhez kötése.
- Tanúsítvány-visszavonási listák karbantartása (Certification Revication List).
- Adatbázisok és névtárak karbantartása.
- Mester (Master) kulcsok kezelése.

A CRL (Certification Revication List – tanúsítvány-visszavonási lista) azon tanúsítványok gyűjteménye, amelyek valamilyen okból vissza lettek vonva. Ilyen ok lehet a titkos kulcs bizalmasságának sérülése. Itt jegyezném meg, hogy amikor tanúsítványkezelő rendszert készítünk, különös figyelmet kell fordítanunk az úgynevezett gyökértanúsítvány (Root Certification) védelmére, ugyanis ennek bizalmasságának sérülése az összes általunk hitelesített kulcs hitelességének elvesztésével jár. Ilyet komoly tanúsítványkezelő rendszer nem engedhet meg magának.
A CRL megfelelő időközökben való frissítése és közzé tétele az egyik legfontosabb kérdése a CA házirendjének. Szabályozni kell a CRL legnagyobb érvényességének idejét, a benne található tanúsítványok lejáratának idejét, valamint gondoskodni kell a CRL megfelelően sűrű időszakonkénti közzétételét egy mindenki által hozzáférhető helyen.

Az RA

A regisztrációs ügynök az újabb PKI rendszerekben gyakorlatilag teljesen átveszi a felügyeleti feladatokat a tanúsítványkezelőtől. Így az RA feladatai többek között az alábbiak:

- felhasználó-azonosító létrehozása;
- felhasználói kulcs kérése a CA-tól;
- felhasználói kulcs fogadása a CA-tól;
- felhasználói kulcs fájlban való tárolása;
- felhasználói kulcs kártyán való tárolása;
- tanúsítvány-visszavonás kérése a CA-tól;
- felhasználói házirend kezelése;
- felhasználói kulcs megújítás, illetve frissítés kérelmekre;
- kulcs-visszaállítás;
- felhasználó kulcs tárolására (smartkártyán vagy PKCS#12 kulcsfájlokban).

A PKI rendszer kezelése szempontjából rendkívül fontos a házirend megfelelő kialakítása, ami a kulcskezelésről és általában arról szól, hogy a rendszerben kinek, illetve melyik rendszerelemnek milyen szolgáltatási jogosultságai vannak.

Az ügyfelek

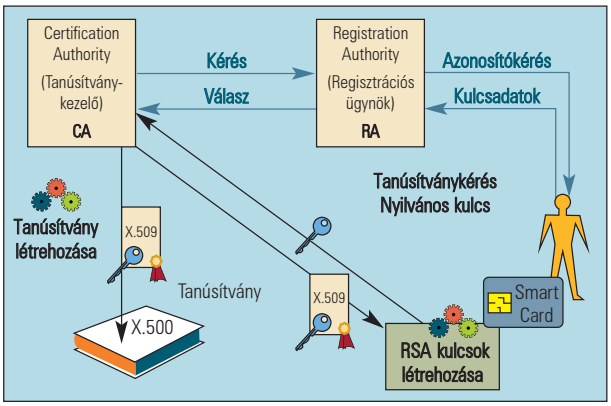
Az ügyfeleken szükségünk van olyan ügynökök telepítésére, amelyek képesek kezelni a tanúsítványokkal és chipkártyákkal kapcsolatos műveleteket.

A CA felépítéséről röviden

A tanúsítványkezelő rendszerben a különböző tanúsítványok egy faszerkezetben helyezkednek el. Ennek a faszerkezetnek a tetején az úgynevezett gyökértanúsítvány (Root Certification) áll. Ez egy különleges tanúsítvány a rendszerben, mivel ez az egyetlen olyan példány, ami önmagával van aláírva. Az összes többi kibocsátott tanúsítvány attól függően kerül elhelyezésre a fában, hogy melyik tanúsítvánnyal lett aláírva. Az aláírt tanúsítvány ugyanis az aláíró tanúsítvány egy gyerek csomópontjaként jelenik meg a fában. Ezzel a felépítéssel megoldható, hogy különböző különálló szervezeteket kezeljünk, illetve az általunk elismert szervezetek saját tanúsítványokat bocsássanak ki.

A gyakorlat

Az elmélet áttekintése után most nézzük miként is tudjuk létrehozni saját tanúsítványkezelő rendszerünket.
Miután telepítettük az OpenSSL csomagot, a */etc/ssl* könyvtárban, valamint a */usr/share/ssl* könyvtárban találjuk majd meg a szükséges, hivatkozott állományokat. Első lépésként a */usr/share/ssl/misc/CA.pl* állományról készítsünk egy hivatkozást a */usr/sbin* könyvtárban, mert erre a futtatható állományra a későbbiekben nagy szükségünk lesz. A *CA.pl* állomány egy olyan futtatható állomány, amely jelentősen leegyszerűsíti az OpenSSL használatát, így ennek a kiegészítésnek a használata célszerű lehet, ugyanakkor nem kötelező. Minden ismertetett parancs kivitelezhető a megfelelő `openssl` paranccsal is.
A */etc* könyvtáron belül találjuk az *openssl.conf* nevű állományt. Ebben az állományban tudjuk végrehajtani a legalapvetőbb beállításokat. Itt tudjuk megadni, hogy melyik könyvtárat használjuk alapértelmezett könyvtárként, azon



belül a különböző szerepű alkönyvtárak elhelyezkedése pontosan hogyan fest, hol találhatóak az alapvető tanúsítványok, továbbá a tanúsítványok érvényességére vonatkozó adatokat is itt tudjuk beállítani.
Érvényességi időnek ajánlatos olyan időtartamot megadni, ami nem túl rövid, de nem is lóg túl a tanúsítványok érvényességének előre becsülhető idején. Ha nincs tényleges feltétel, akkor használjunk 365 napos intervallumot az alábbi sor használatával:

```
default_days=365
```

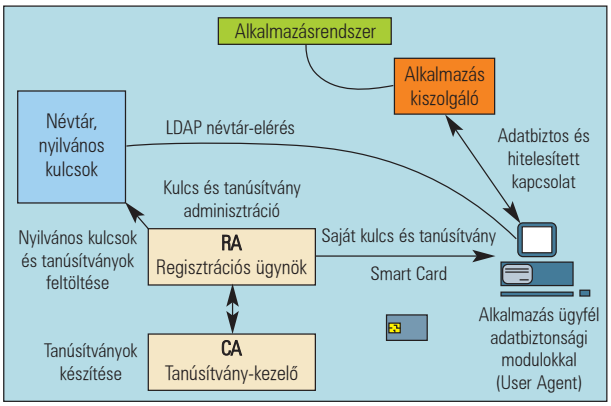
A `default_bits` változó értékének beállításával meg tudjuk adni, hogy a készített kulcsok hány bites titkosítást használjanak. Ajánlatos 1024 vagy 2048 bites titkosítást használni, mivel ez a jelenlegi számítási teljesítmény mellett gyakorlatilag feltörhetetlen.
Ha végeztünk az alapvető beállításokkal a `CA.pl -newca` paranccsal létrehozhatjuk az új tanúsítványkezelőnket.
A rendszer kérni fogja, hogy jelöljünk ki egy tanúsítványt, ami a jövőben a CA rendszer tanúsítványaként fog szolgálni. Ha nincs ilyen, akkor készíthetünk is egyet. Kövessük az utasításokat és készítsünk egyet, de sok időt nem érdemes ráfordítani, ugyanis mindjárt készítünk egy újat, mert ennek a tanúsítványnak 365 napnál hosszabb érvényességet szeretnénk adni.

Tanúsítvány készítése

Most hozzuk létre a CA első, egyedi tanúsítványát, a gyökértanúsítványt (Root Certification). Ez azért különleges tanúsítvány, mivel a rendszerben ez az egyetlen olyan tanúsítvány, ami önmagával van aláírva.
Adjuk ki a következő parancsot:

```
openssl req -config /etc/openssl.cnf -new -x509
➥-keyout private/cakey.pem \
-out cacert.pem -days 3650
```

Ezzel létrehozunk egy tíz évig érvényes tanúsítványt, amelynek a titkos kulcs részét elhelyezzük az *openssl.conf* állományban beállított gyökérkönyvtár *private* alkönyvtárban, míg a tanúsítvány részét a gyökérkönyvtárba tesszük.
A tanúsítvány készítésekor a Common Name (CN) értéknek adjuk meg, hogy ez melyik szervezet gyökértanúsít-



ványa lesz, például „AzÉnSzervezetem Root Certification". A tanúsítvány készítésekor meg kell adnunk egy jelszót, erre a jelszóra akkor lesz szükségünk, amikor a most készített gyökértanúsítvánnyal másik tanúsítványokat fogunk aláírni.
Ha végeztünk a gyökértanúsítvány készítésével, akkor ellenőrizzük, hogy az ***index.txt*** állomány üres legyen, valamint ellenőrizzük, hogy a ***serial*** állomány tartalma 01. Fontos, hogy a most készített tanúsítványt csak másik tanúsítványok aláírására használjuk, a hozzá tarozó titkos kulcsot és jelszót pedig olyan helyen őrizzük, ahol mások számára nem hozzáférhető.
Ha elkészültünk a gyökértanúsítvánnyal, akkor készítsünk belőle egy böngészők és egyéb alkalmazások által is feldolgozható tanúsítványt és telepítsük, mint megbízható gyökértanúsítványt. Innentől kezdve ugyanis az összes általunk aláírt tanúsítvány megbízható tanúsítványként lesz kezelve azokon a gépeken, ahová ezt a tanúsítványt telepítettük.

```
openssl x509 -in cacert.pem -out cacert.crt
```

Ha végeztünk a gyökértanúsítvány létrehozásával, akkor itt az idő, hogy elkészítsük az első olyan tanúsítványunkat, amit az imént készített gyökértanúsítvánnyal írunk alá. Ehhez először el kell készíteni egy kérelmet egy új tanúsítványhoz az alábbiak szerint:

```
CA.pl -newreq, vagy openssl req -config
➥/etc/openssl.cnf -new -keyout newreq.pem -out
newreq.pem -days 365
```

Ezzel a paranccsal létrehoztunk egy új kérelmet, amelyet a ***newreq.pem*** állományban helyeztünk el. A tanúsítvány létrehozásakor figyeljünk oda arra, hogy CN (Common Name) értéknek mit adunk meg. Amennyiben például a ***www.tartomany.hu*** weblap SSL tanúsítványát készítjük el, akkor a CN-nek a ***www.tartomany.hu*** értéket adjuk, ha azonban a ***cimzett@tartomany.hu*** elektronikus levélcímhez tartozó felhasználó számára készítünk digitális aláíráshoz, valamint titkosításhoz tanúsítványt, akkor a CN-nek adjuk a ***cimzett@tartomany.hu*** értéket.
Amennyiben az új kérelemmel elkészültünk, akkor nem marad más hátra, mint a kérelem aláírása a gyökértanúsítvánnyal (Root Certification):

```
CA.pl -sign, vagy openssl ca -config
➥/etc/openssl.cnf -policy policy_anything -out
newcert.pem -infiles newreq.pem
```

Az aláíráshoz szükségünk lesz a gyökértanúsítvány jelszavára, majd ha egyezik a jelszó, akkor a ***newcerts*** könyvtárban létrejön az ***XX.pem*** – ahol XX a ***serial*** állomány tartalma – tanúsítvány és ez megjelenik az ***index.txt*** állományban is.
Az elkészült tanúsítványból a már megismert módon készíthetünk ügyfélprogramok által feldolgozható tanúsítványt.

Tanúsítvány visszavonása

Tanúsítványt az `openssl -revoke newcert.pem` paranccsal vonhatunk vissza. A tanúsítvány visszavonása után ne feledjük, hogy frissíteni kell a visszavont tanúsítványokat tartalmazó adatbázist is. Ezt megtehetjük az `openssl ca -gencrl -config /etc/openssl.cnf -out crl/sopac-ca.crl` paranccsal. Ha frissítettük a listát, akkor gondoskodjuk a lista közzétételéről is, például a tanúsítványkezelő rendszerünkhöz tatozó weblapon.

Tanúsítvány megújítása

Ha egy tanúsítvány érvényessége lejár, akkor elképzelhető, hogy a tanúsítvány felhasználója meg szeretné újítani azt. Az ilyen kérelmek beérkezése esetén az eddig érvényben lévő tanúsítvány visszavonásáról is gondoskodnunk kell, mielőtt kiadjuk az új tanúsítványt. A régi tanúsítványt az ***index.txt*** állományban kikereshetjük, így nem fog nehézséget okozni a sorszám kiderítése, ami alapján a visszavonást el kell végezni.
Röviden talán ennyiben foglalható össze a tanúsítványkezelő rendszerek elméleti és gyakorlati működése. A gyakorlás kedvéért készítsünk tanúsítványokat, terjesszük ismerőseink, üzletfeleink körében, használjuk ki a digitális aláírás és a titkosítás nyújtotta biztonságot. Amennyiben olyan tanúsítványra volna szükségünk, amelyet valamelyik nagy tanúsítványkezelő hitelesít, úgy az interneten erre is találunk megoldást. A következőkben csak néhány példát említenék ezek közül.

Elérhető nyilvános CA szervezetek

Végül ejtsünk szót néhány nyilvános CA szervezetről, melynél mi is hitelesíttetni tudjuk az általunk használt kulcspárokat. Az EuroPKI egy olyan közös európai szervezet, amelynek feladata, hogy nyilvános kulcsú hitelesítést adjon magánszemélyek, szervezetek és különböző európai projektek számára. A szolgáltatás a ➲ http://www.europki.org címen érhető el.
Természetesen létezik magyar tanúsítványkezelő szolgáltatás is, példaként említeném a ➲ http://www.netlock.net oldal szolgáltatását.

Illés Viktor (viktor@ei.hu)
23 éves, a BME műszaki informatikus szakának hallgatója, mellette weblapokkal, linuxos és windowsos rendszerekkel foglalkozik. Szabadidejét legszívesebben a szabadban tölti, teniszezik és kerékpározik.