PPKE-ITK

Adatbiztonság és Kriptográfia

Mérési útmutató a

„PKI és SSL”

című méréshez

Tartalomjegyzék

1 Elméleti összefoglaló

A mérés két tárgya közeli kapcsolatban van egymással: az SSL használja a PKI infrastruktúráját, a PKI fogalma az SSL és hasonló protokollok által támasztott igények következtében jött létre. Minden rendszerben alapvető kérdés a másik fél azonosítása: személyi igazolvánnyal tudjuk igazolni kilétünket ismeretleneknek. Az elektronikus, internetes világban természetesen hasonló igények jelentek meg. Ezen cél megvalósulását segítik a PKI (Public Key Infrastructure) rendszerek. Az SSL célja a PKI által biztosított elemekkel történő szabványos kommunikáció, melynek segítségével nem biztonságos csatornán keresztül is tud két fél egymással kommunikálni.

1.1 PKI

A kriptográfia alapvetően négy alap biztonsági szolgáltatás támogatását biztosíthatja:

- adatok sértetlensége,
- adatok bizalmassága,
- a felhasználó vagy eszköz azonosságának hitelesítése, és
- letagadhatatlanság.

Az adat sértetlenséggel kapcsolatos szolgáltatások az adatok jogosulatlan vagy véletlen megváltoztatásának kérdését kezelik. A szolgáltatás célja, hogy a címzett meggyőződhessen arról, hogy az adatokban nem történt változtatás. Az adatok bizalmas kezelésével kapcsolatos szolgáltatás csak az adatok megtekintésére jogosultak számára teszi lehetővé az érzékeny adatokhoz való hozzáférést. A hitelesítésre vonatkozó szolgáltatások megállapítják az adatátvitel, az üzenet és a kezdeményező valódiságát. A szolgáltatások célja, hogy az adatok címzettje megállapíthassa azok származását. A letagadhatatlansági szolgáltatások meggátolják, hogy valaki letagadja korábbi cselekedetei megtételét a küldött adat és egy konkrétan meghatározható személy utólag is igazolható összekapcsolása révén.

Általánosságban elmondható, hogy a nyilvános kulcsú kriptográfiai eljárások minden biztonsági szolgáltatás részét képezik a ma általánosan használt protokollokban, alkalmazásokban, ahol a kriptográfia eszköztárára egyáltalán szükség van. A küldő és a címzett által a kriptográfiai eljárás során használt kulcsok kezelése (létrehozása, eljuttatása a jogosult felhasználóhoz, biztonságos tárolása, stb.) a kriptográfia egyik alap problémája. A szimmetrikus kulcsú rejtjelezésnél a kulcsok megosztását nyilvános kulcsú protokollok segíthetik, de ebben az esetben is, illetve a hitelesítés vagy letagadhatatlanság célú eljárásoknál is külön intézkedésekkel kell biztosítani a felhasználók és a kulcsok biztonságos összerendelését. Ez egy megbízható résztvevő beiktatásával oldható meg átfogó módon. A megbízható fél megnevezése hitelesítés-szolgáltató (Certification Authority - CA), melynek feladata az, hogy saját aláírásával tanúsítja egy adott személy (vagy eszköz) és egy nyilvános kulcs kapcsolódását. A hitelesítés-szolgáltató által kibocsátott tanúsítvány alapján fogadják el a kommunikáló felek egymás nyilvános kulcsát. Ez biztosítja a kulcscsere protokollok biztonságát és az aláírások letagadhatatlanságát. Ahhoz, hogy a biztonsági szolgáltatások egy szervezet határain kívülre is bővíthetők legyenek célszerű, hogy a hitelesítés-szolgáltatást a

kommunikáló felektől független harmadik fél biztosítsa. A világban működő hitelesítés-szolgáltatók egymással összekapcsolódva (egymást hitelesítve) egy olyan biztonsági infrastruktúrát alkotnak, amelyekre a felhasználók nyugodtan támaszkodhatnak biztonsági szolgáltatások igénybevétele céljából. Ezt az infrastruktúrát nevezik nyilvános kulcsú infrastruktúrának (PKI).

A nyilvános kulcsú infrastruktúra (PKI) a nyilvános kulcsokat kommunikáló felekhez (személyekhez vagy eszközökhöz) köti, és lehetővé teszi, hogy a kommunikáló felek, vagy harmadik felek ellenőrizzék a nyilvános kulcsok ezen hozzárendelését, és biztosítja a kulcsok elosztott rendszerben történő folyamatos gondozásához szükséges szolgáltatásokat. A PKI segítségével az elektronikus tranzakciók annak tudatában bonyolíthatók, hogy:

- A tranzakció küldőjeként megjelölt személy vagy eszköz/alkalmazás valóban a kezdeményező.
- A személy vagy eszköz/alkalmazás, ahova a tranzakció megérkezik, valóban a szándékolt címzett.
- Az adatok integritása nem sérült.

Ahhoz, hogy a kommunikáció nyilvános kulcsú kriptográfia segítségével biztonságossá váljon, a kommunikáló felek meg kell kapják egymás nyilvános kulcsait, és meg kell győződjenek a másik fél azonosságáról. Ha csak ketten akarnak egymással kommunikációt folytatni, ez megvalósítható más módon, ha azonban a kapcsolat a kommunikáló felek széles körére terjed ki, megbízható harmadik félre kell hagyatkozniuk a nyilvános kulcsok elosztásában és a megfelelő kulcspárhoz tartozó személy vagy eszköz azonosságának ellenőrzésében. A PKI lényegében a rejtjelezési és egyéb szoftvertechnológiák, illetve szolgáltatások olyan kombinációja, ami lehetővé teszi, hogy a szervezetek megvédjék kommunikációik és üzleti tranzakcióik biztonságát a hálózaton. A szervezet szempontjából a PKI a digitális tanúsítványokat, nyilvános kulcsú kriptográfiát használó alkalmazásokat és a hitelesítés-szolgáltatást egyetlen, teljes, az egész vállalkozásra kiterjedő hálózatbiztonsági architektúrává integrálja. Egy tipikus vállalati PKI-nak része a digitális tanúsítványok kiállítása egyéni felhasználók és szerverek részére, a végfelhasználói regisztrációt végző szoftver, a felhasználói adattárakkal történő integráció, a tanúsítványok nyilvántartására szolgáló eszközök, valamint a kapcsolódó adminisztráció és üzemeltetés. A fejezet további részeiben a PKI alkotóelemeit és a korábbiakban bemutatott alapfunkciókon túlmenő, egyes kiegészítő szolgáltatásait foglaljuk össze.

1.1.1 A PKI funkcionális elemei, szereplői

A PKI funkcionális elemei a tanúsítvány kibocsátó (CA), a regisztrációs szervezet (RA), tanúsítványtár és archívum. A hitelesítés-szolgáltató vagy csak a tanúsítvány kibocsátás feladatát látja el, vagy a többi PKI funkciót is. A PKI-t a végfelhasználók két típusa veszi igénybe: a tanúsítvány alanya és a tanúsítvány használója (érintett fél).

A **tanúsítvány kibocsátó** a PKI alap építőeleme. Két tulajdonság, attribútum jelöli: a neve és nyilvános kulcsa. Négy alap PKI funkciót végez:

- tanúsítványokat állít ki (azaz létrehozza és aláírja azokat);
- nyilvántartja a tanúsítványok állapotára vonatkozó információkat és tanúsítvány visszavonási listákat (CRL) készít;

- nyilvánosságra hozza az aktuális (azaz le nem járt) tanúsítványai és visszavont tanúsítványai listáját (CRL-jeit); és
- státuszinformációs archívumot tart fenn az általa kiadott, lejárt tanúsítványokról.

Amikor a tanúsítvány kibocsátó kiállít egy tanúsítványt, ezzel azt igazolja, hogy a tanúsítvány alanya (a tanúsítványban nevesített személy, szervezet, eszköz vagy alkalmazás) rendelkezik azzal a privát kulccsal, amely a tanúsítványban szereplő nyilvános kulcs párja. Ha a tanúsítvány alanya egy másik tanúsítvány kibocsátó CA, a kiállító azt igazolja, hogy a másik tanúsítvány kibocsátó által kiállított tanúsítványok megbízhatóak. A CA az általa generált minden tanúsítványra (és tanúsítvány visszavonási listára) ráteszi a nevét és aláírja azokat saját privát kulcsával. Ha a felhasználó (közvetlenül vagy egy tanúsítási útvonal segítségével) megbizonyosodott arról, hogy bízhat a tanúsítvány kibocsátóban, megbízhat az általa kiadott tanúsítványokban is. A felhasználók a nevek alapján egyszerűen azonosíthatják a kérdéses tanúsítvány kibocsátó által kiállított tanúsítványokat. A tanúsítvány valódiságának biztosítására a felhasználók a tanúsítvány kibocsátó nyilvános kulcsával ellenőrzik aláírását. Így fontos, hogy a tanúsítvány kibocsátó megfelelő módon védje saját privát kulcsát.

A **regisztrációs szervezet (RA)** feladata, hogy a tanúsítvány kibocsátó számára ellenőrizze a tanúsítvány tartalmát. A tanúsítvány tartalmában lehet olyan információ, amit a tanúsítványt kérő szolgáltat, pl. személyes adatok vagy elérhetőség. A regisztrációs szervezet összegyűjti és ellenőrzi ezeket az információkat, és bocsátja a tanúsítvány kibocsátó rendelkezésére. Minden tanúsítvány kibocsátónak van egy akkreditált regisztrációs szervezeti listája, azaz egy olyan lista, amely a megbízhatónak minősülő RA-kat tartalmazza. A tanúsítvány kibocsátó ellenőrzi az RA aláírását, hogy megbizonyosodjon a megadott adatok hitelességéről. Fontos, hogy a regisztrációs szervezet megfelelő módon védje saját privát kulcsát.

A **tanúsítványtár** biztosítja a tanúsítványok tárolásához és szétküldéséhez szükséges eszközöket, gondozza és karbantartja a tanúsítványokat. Leginkább az LDAP protokoll terjedt el a tanúsítványtárak (tanúsítványok és tanúsítvány visszavonási listák) lekérdezésére.

Az **archívum** feladata a tanúsítvány kibocsátó archív információinak hosszú távú tárolása a CA részére. Az archívum ellenőrzi, hogy az információ az érkezéskor korrekt volt és biztosítja, hogy az a tárolás során nem változhat meg. Amennyiben később vita alakulna ki, az archivált információ felhasználható annak ellenőrzésére, hogy a tanúsítványhoz tartozó privát kulcsot használták-e a dokumentum aláírására.

PKI **végfelhasználók** a PKI-t használó szervezetek, személyek, eszközök vagy alkalmazások, melyek azonban tanúsítványokat nem bocsátanak ki. A végfelhasználók a PKI más elemeire hagyatkoznak a tanúsítványok megszerzésében és ellenőrzésében. A végfelhasználók közé tartoznak az érintett felek, akik a tanúsítványra hagyatkoznak, hogy megbízhatóan megismerjék egy másik szervezet, személy, eszköz vagy alkalmazás nyilvános kulcsát; és a tanúsítvány alanyai, akik számára tanúsítványt állítanak ki. Egy végfelhasználó lehet egyszerre érintett fél és tanúsítvány alanya különböző alkalmazásokra vonatkozóan.

1.1.2 Fizikai architektúra

Számos lehetőség létezik a PKI fizikai kialakítására. Ajánlott azonban, hogy a PKI főbb elemei külön rendszerekben kerüljenek megvalósításra, azaz a CA legyen egy rendszerben, az RA egy másikban és a tanúsítványtár/címtár szerverek is másik rendszerekben legyenek. Tekintettel arra, hogy a rendszerekben kényes, illetve védendő adatok vannak, ezeket mindenképpen egy szervezet internetes tűzfal rendszere által védett módon kell elhelyezni. A

tanúsítvány kibocsátói (CA) rendszer különösen fontos, mert kompromittálódása a PKI teljes működését felboríthatja és az összes tanúsítvány visszavonását, és újbóli kibocsátását követelné meg. Ezért célszerű a CA rendszert egy további szervezeti tűzfal mögé elhelyezni azért, hogy mind az Internet felől, mind pedig a szervezet felől védve legyen. A szervezeti tűzfal természetesen ne akadályozza a CA és az RA valamint az egyéb szükséges rendszerek közötti kommunikációt. Ha elkülönült szervezetek akarnak egymás tanúsítványaihoz hozzáférni, elérhetővé kell tegyék egymás, és esetleg más szervezetek számára tanúsítványtáraikat az Interneten keresztül. Egyes szervezetek azonban a tanúsítvány- és címtárakat sokkal többre használják, mint csak a tanúsítványok tárolására. A címtár szerveren lehetnek más, a szervezet számára érzékeny adatok is, ezért maga a címtár is túl érzékeny lehet ahhoz, hogy közzétegyék. Egy tipikus megoldás lehet erre olyan címtárak létrehozása, amelyekben csak nyilvános kulcsok és tanúsítványok vannak, és ezt a címtárat a szervezet hálózatának határán helyezik el. Ezt a címtárat nevezik határcímtárnak. A címtár elhelyezkedhet a szervezeti tűzfalon kívül vagy esetleg a hálózat egy védett DMZ szegmensén, hogy a nyilvánosság számára rendelkezésre álljon, de viszonylag nagyobb védelmet élvezzen az esetleges támadások ellen.

A szervezet védett hálózatán belül elhelyezett fő címtár szerver rendszeresen frissíti a határcímtárat az új tanúsítványokkal és a létező tanúsítványokra vonatkozó új információkkal. A szervezeten belüli felhasználók a fő címtár szervert használhatják, míg más szervezetek és rendszerek csak a határcímtárhoz férnek hozzá.

Nagyon fontos, hogy pontosan meghatározott és dokumentált módon férjenek hozzá a PKI rendszerhez a különböző jogosultsággal rendelkező felhasználók, adminisztrátorok és üzemeltetők.

Az RA munkaállomások mellett - bizonyos konfigurációk esetén - szükség lehet regisztrációs szerverre is. Ez általában akkor jellemző, ha a végfelhasználók saját maguk, egy regisztrációs felületen keresztül végzik a tanúsítványok igénylését, megújítását, illetve egyéb adminisztratív folyamatok elindítását vagy akár teljes végrehajtását. (Pl.: Windows loginhez kapcsolódva az automatikus tanúsítvány kiadást Windows CA megoldás részeként.)

1.1.3 PKI adatstruktúrák

A PKI rendszerek két alap adatstruktúrát használnak: a nyilvános kulcs tanúsítványokat és a tanúsítvány visszavonási listákat.

Az X.509 nyilvános kulcs **tanúsítvány** formátuma rugalmas és hatékony módon az információk széles körének közlésére alkalmas. Az információk zöme opcionális, és a kötelező mezők tartalma is változhat. A PKI bevezetői számára fontos, hogy tisztában legyenek a választási lehetőségekkel és azok következményeivel. Egy esetleges helytelen döntés akadályozhatja a kritikus alkalmazások együttműködését vagy támogatását. Az X.509 nyilvános kulcs tanúsítványt a kibocsátó CA digitális aláírása védi. A felhasználók tudják, hogy a tartalmat nem másították meg, ha az aláírás ellenőrzése sikeres. A tanúsítványokban van számos kötelező mező, de tartalmazhatnak opcionális mezőket is. A kötelező mezők a következők: sorszám, a tanúsítvány aláírásának algoritmus azonosítója, a tanúsítvány kibocsátójának neve, a tanúsítvány érvényességi ideje, a nyilvános kulcs, a tanúsítvány alanyának adatai és a kulcs falhasználási területe.

A tanúsítvány alanya az, aki a megfelelő privát kulccsal rendelkezik. A tanúsítvány alanya lehet egy végfelhasználó, egy hálózati vagy kiszolgáló/szerver eszköz, vagy akár egy másik

CA. A felhasználók rendelkezhetnek több kulcspárral, vagy több tanúsítvánnyal ugyanarra a kulcspárra.

A kulcs felhasználási területe határozza meg, hogy a kulcspár milyen biztonsági szolgáltatások megvalósításához használható fel. Ezeknek általános szolgáltatásoknak (pl. letagadhatatlanság vagy adat titkosítás), vagy a PKI infrastruktúrához kapcsolódó specifikus szolgáltatásoknak kell lenni (pl. a tanúsítványok vagy tanúsítvány visszavonási listák aláírása).

Szervezetek az alkalmazások széles körét támogathatják PKI segítségével. A tanúsítványok kiállítására szolgáló eljárástól, illetve a felhasználói kriptográfiai modul (privát kulcsot tároló és kezelő hardver és szoftver környezet) típusától függően egyes tanúsítványok megbízhatóbbak lehetnek másoknál. A hitelesítési szabályzatok (Certification Policy - CP) kiterjesztés a tanúsítványban egy teljesen egyedi azonosítót tartalmaz, amely egy egyértelműen meghatározott szabályzatra utalva meghatározza a tanúsítványra vonatkozó hitelesítési irányelveket.

A tanúsítványok a lejárati dátumot is tartalmazzák. Előfordulhat azonban az is, hogy egy tanúsítvány adatai még a lejárati dátum előtt megbízhatatlanná válnak. A tanúsítvány kibocsátóinak szükségük van egy olyan mechanizmusra, amely alkalmas a tanúsítványok állapotának karbantartására. Az egyik ilyen mechanizmus az X.509 **tanúsítvány visszavonási lista** (CRL). A CRL-t a CRL kibocsátójának digitális aláírása védi. Ha az aláírás ellenőrizhető, a CRL felhasználói tudják, hogy az aláírás generálásának időpontjától az adatokat nem másították meg. A CRL-ek számos kötelező mezőt tartalmaznak, de tartozhat hozzájuk egy sor opcionális kiterjesztés is. A legfontosabbak: kibocsátó, kibocsátás dátuma, következő kibocsátás dátuma, visszavont tanúsítványok listája.

A visszavonási információkat általában CRL listában kötelező publikálni, de azonnali visszavonási információ is nyújtható a tanúsítványok érvényességének ellenőrzésére (OCSP – Online Certificate Status Protocol).

1.2 SSL

A Secure Sockets Layer (SSL) és Transport Layer Security (TLS) protokollok lehetővé teszik a kliens és a szerver hitelesítését, valamint a kommunikáció bizalmasságának megteremtését. Az SSL-t a Netscape Communications cég fejlesztette ki 1994-ben és a legutolsó verziója az SSL v3 (a korábbi verziók nem biztonságosak, és használatuk nem javasolt). Az 1996-ban az IETF (Internet Engineering Task Force) keretében megalakult TLS munkacsoport fejleszti és szabványosítja az TLS protokollt. A TLS v1.0 (RFC 2246) 1999-ben került kibocsátásra és biztonsági szempontból megfelel az SSL v3 változatának, jelen dokumentumban sem különböztetjük meg ezeket. A TLS aktuális verziója a v1.2.

Az SSL/TLS a TCP/IP-re épülő alkalmazások felé teremti meg a hálózat két pontja között a biztonságos kapcsolat lehetőségét. Leggyakoribb használata a biztonságos HTTP alapú kliens-szerver kommunikáció egy Web böngésző és egy webszerver között. Ma jóformán minden webszerver képes SSL/TLS kapcsolat kiépítésére és minden mai Web böngésző támogatja azt.

1.2.1 SSL/TLS funkciók

Az SSL/TLS az alábbi lehetőségeket biztosítja a HTTP-re vagy más alkalmazásrétegbeli protokollra épülő alkalmazások számára:

- szerver hitelesítés: Az SSL/TLS lehetővé teszi azt, hogy a webes kliens (illetve a felhasználó) meggyőződjön a webszerver kiléte felől. A kliens a szerver nevét és a nyilvános kulcsát tudja ellenőrizni, melyhez megszabhatja az általa megbízhatónak ítélt hitelesítés-szolgáltatók listáját. Így például, ha bizalmas adatot kell közölnie az ügyfél, akkor ez erősítheti meg, hogy az információ a megfelelő helyre fog eljutni.
- kliens hitelesítés: Az SSL/TLS lehetővé teszi a webszerver számára is, hogy meggyőződjön a felhasználó kilétéről a szerepek felcserélésével. A webszerver is az általa megbízhatónak ítélt hitelesítés-szolgáltatók által kibocsátott tanúsítványokat fogadja el. Így például, ha az elektronikus közigazgatási szolgáltatás bizalmas adatokat kíván közölni az ügyfelével, előzőleg ilyen módon ellenőrizheti az ügyfél kilétét.
- sértetlenség védelme: Az SSL/TLS kapcsolatban folytatott kommunikáció biztosítja azt, hogy az átküldött adatok véletlenül vagy szándékosan sem sérülhetnek meg.
- bizalmas kommunikáció: Az SSL/TLS lehetővé teszi azt, hogy az adatok titkosan közlekedjenek a Web böngésző és a webszerver (vagy két webszerver) között. Alkalmas titkosító algoritmus használata esetén ez a bizalmasság megfelelő szintű védelmét biztosítja.

1.2.2 Tanúsítványok

Az SSL/TLS protokoll tanúsítvány alapon végzi a felek azonosítását. Kulcskérdés ezeknek a tanúsítványoknak a felépítése, szerkezete és ellenőrizhetősége. További kérdés az, hogy megbízunk-e a bemutatott tanúsítványban.

Az SSL/TLS protokollok során a "kézfogás" alatt jelennek meg a tanúsítványok. Használatuk opcionális, azonban biztonsági okokból nem javasolható anonim eljárás.

A szerver tanúsítványa a „server hello” üzenet után, a „certificate” üzenetben jelenik meg. Ekkor a tanúsítvány a szerver publikus kulcsát tartalmazza. Az üzenet lista szerkezetű, a szerver tanúsítványával kell kezdődnie, majd a tanúsítványlánc elemeit kell felsorolni. A gyökérszintű HSZ tanúsítványa elhagyható abban az esetben, ha feltételezhető, hogy a kliens ismeri azt.

A szerver az első lépések lezárásaként kérheti a kliens azonosítását. Ilyenkor megadja, hogy milyen PKI részfákba tartozó tanúsítványokat fogad el a klienstől. Amennyiben a szerver kezdeményezte a kliens azonosítását, a kliensnek meg kell adni a tanúsítványát a kézfogási protokoll során. A kliens esetén is lehetséges a tanúsítványlánc elküldése.

A kézfogási protokoll következő lépésében a kliens a megosztandó információt a szerver nyilvános kulcsával kódolja, melyet a szervernek képesnek kell lennie visszafejteni. A kliens azonosításaként a kliens legvégül elkódolja az addigi üzeneteit a titkos kulcsával, így bizonyítva a magánkulcs birtoklását.

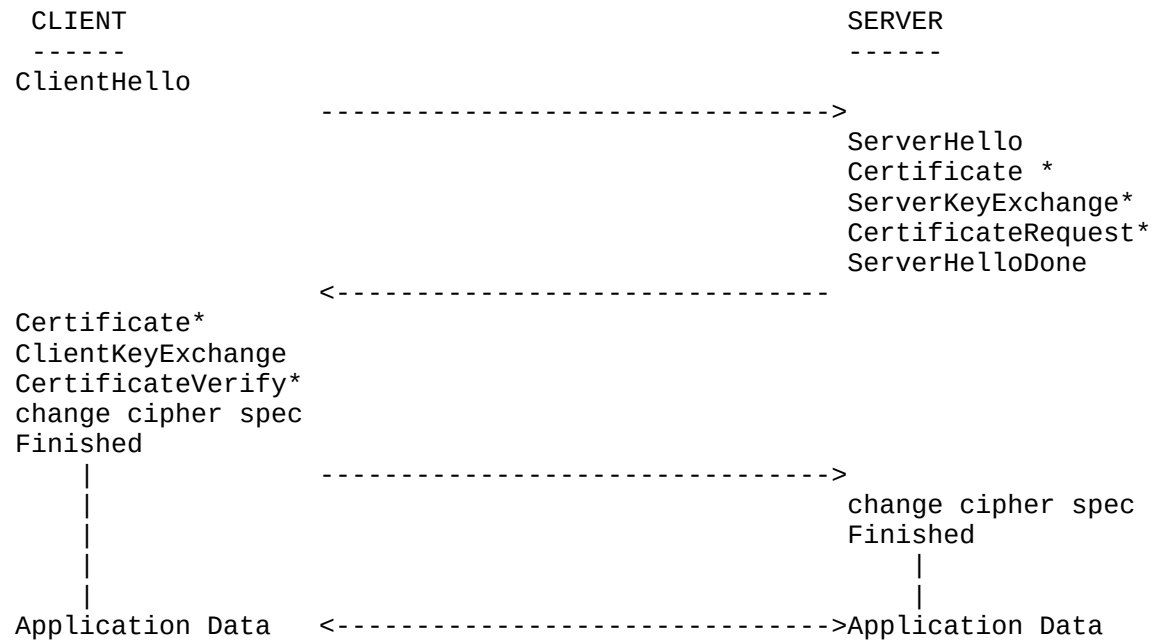
Amennyiben a tanúsítványláncok minden eleme megfelel a formai és egyéb követelményeknek, valamint a kliens és szerver tanúsítványai is felhasználhatóak a

szándékozott célnak, akkor a tanúsítványlánc minden elemét ellenőrizni kell az elérhető tanúsítvány-visszavonási eljárásoknak megfelelően. Ennek során figyelembe kell venni a tanúsítvány által hordozott elérési helyeket (CRL, OCSP).

1.2.3 Protokoll

A protokollokat a TLS 1.0 segítségével mutatjuk be. Az SSL főbb lépéseiben megegyezik ezzel.

A protokollok legérzékenyebb lépése a kézfogás protokoll.



A csillaggal megjelölt lépések opcionálisak, a kézfogás modelljétől függően.

A kézfogás protokoll általános lépései:

1. A kliens elküldi a szervernek az általa támogatott legmagasabb SSL/TLS verziószámát, az alkalmazható kulcs generáló és titkosító algoritmusok listáját és egy véletlen számot.
2. A szerver válaszában megjelöli az alkalmazandó (minimális) SSL/TLS verziót, a kiválasztott algoritmus azonosítóját és szintén egy véletlen számot. A szerver továbbá elküldi a tanúsítványát és (amennyiben szükséges) elkéri a kliens tanúsítványát.
3. A kliens ellenőrzi a tanúsítvány tartalmát és hitelességét, majd a szerver nyilvános kulcsával titkosított adatot küld a titkosító kulcs generálásához. Amennyiben szükséges, a tanúsítványát és hitelesítő adatot is küld a szervernek a kliens hitelesítéséhez.
4. Amennyiben szükséges, a szerver ellenőrzi a kliens hitelességét, majd megfejti a kulcs generálásához küldött tikosított adatot.

5. Mindkét fél összeállítja a master kulcsot és a tranzakciók titkosítására használt viszonykulcsokat, és egy-egy üzenettel kölcsönösen nyugtázzák az SSL/TLS kapcsolat létrejöttét.

A kézfogási protokollon belül lehetőség nyílik eltérő azonosítási rendek alkalmazására:

Szerver hitelesítése

Ekkor csak a szerver kerül hitelesítésre. A szerver a „server hello" üzenet után elküldi a saját tanúsítványát a kiválasztott profilnak megfelelő formátumban. A szerver publikus kulcsát arra használja a kliens, hogy a megosztott titok egyik felét a szerver tudomására juttassa.

Kliens hitelesítése

Ekkor nemcsak a szerver, hanem a kliens is azonosítja magát. A szerver kezdeményezheti a kliens azonosítását, de csak akkor, ha megadta a saját tanúsítványát (egyéb esetben a kliensnek „handshake failure" jelzést kell adnia). Ekkor a kliens a „server hello done" üzenet után a saját tanúsítványával válaszol. Ha nincs megfelelő tanúsítványa, akkor „no certificate" jelzést küld, amit a szerver „handshake failure" jelzéssel fogadhat, ami a kapcsolat végét jelenti. A megadott tanúsítvány hitelességét a „certificate verify" üzenettel bizonyítja a kliens.

Azonosítatlan kapcsolat

Bizonyos esetekben lehetőség van arra, hogy a szerver ne azonosítsa önmagát, s ekkor nem is kérheti a kliens azonosítási információit. A protokoll ekkor is képes kialakítani a biztonságos csatornát, azonban egyik fél sem bízhat meg a másikban.

Kapcsolat újrahasználata

A protokoll lehetőséget ad arra, hogy egymást már ismerő felek új kapcsolatot indítsanak egy korábbi kapcsolat alapján, illetve párhuzamos csatornákat nyissanak. Ekkor a felek a korábbi kapcsolat adatait alkalmazzák, nem történik meg a felek újraazonosítása, a kulcscsere, titokcsere.

1.2.4 Algoritmuskészletek (ciphersuites)

Az SSL/TLS protokollok a nyilvános és szimmetrikus kulcsú eljárások kombinációjából állnak. A szimmetrikus kulcsú eljárások számítási szempontból hatékonyabbak, míg a nyilvános kulcsú eljárások alkalmasabbak a (kölcsönös) hitelesítésre és titkosító (master) kulcsok generálására. Az SSL/TLS kapcsolat egy ún. SSL/TLS kézfogással (handshake) jön létre, melynek során a szerver nyilvános kulcsú eljárással hitelesíti magát a kliens felé, majd együttműködnek a titkosító kulcs generálásában. A kézfogás során kerülnek meghatározásra a titkosítás alkalmazandó algoritmusai is.

2 Mérési környezet

A mérés Linux operációs rendszer alatt kell elvégezni, melyen a méréshez szükséges szoftverek megtalálhatók. A mérés sikeres elvégzéséhez szükséges elméleti ismeretek:

- nyilvános és titkos kulcsú titkosítás: különbségek, előnyök, hátrányok, sajátosságok, alkalmazási területek
- lenyomat függvények: tulajdonságok, elvárások
- PKI: felépítés, szabványok, algoritmusok
- SSL protokoll
- Unix felhasználói alapismeretek (alapelvek, fájlok kezelése, bash)
- webes alapismeretek (HTTP protokoll, apache webkiszolgáló)
- TCP/IP hálózati alapismeretek

A feladatok elvégzéséhez adottak a szoftverek, melyek alapvető ismerete szintén szükséges:

- OpenSSL: a hitelesítő központ infrastruktúráját biztosítja a mérés során
- Apache HTTPd + mod-ssl: webkiszolgáló a https protokollhoz
- OpenSSH: távoli bejelentkezés a kiszolgálóra
- Debian Linux: a kiszolgálón használt operációs rendszer
- Mozilla vagy Firefox: HTTPS protokoll ügyfél oldali alkalmazása

Megjegyzés: az OpenSSL nem alkalmas "éles" hitelesítő központ funkciójának betöltésére (lásd man (1) ca), így csak a mérés labor keretein belül, demonstrációs célokra használatos. Minden csoportnak önálló környezet áll rendelkezésére, melyet szabadon állíthat be a szükségleteknek megfelelően.

A mérést a pki.itk.ppke.hu szerveren lehetséges elvégezni. (DNS bejegyzése folyamatban, addig is a /etc/hosts fájlba lehet felvenni). Ide a mérésvezetők által kiosztott login névvel lehet belépni.

2.1 OpenSSL környezet

Az OpenSSL környezet a $HOME/meres-pki-ssl/openssl könyvtárban található meg a szükséges fájlokkal. A beállításokban ezt az útvonalat meg kell adni alapútvonalnak ($HOME/merespki-ssl/openssl/openssl.cnf, [CA_default] dir opció). Azon OpenSSL parancsok esetén, melyek az openssl.cnf-ben tárolt adatokat igénylik, a -config kapcsolónak a $HOME/meres-pki-ssl/openssl/openssl.cnf útvonalat kell megadni paraméterül.

2.2 Apache HTTPd környezet

Az Apache HTTPd környezet a $HOME/meres-pki-ssl/apache könyvtárban található. Ezen belül a könyvtárak funkciója a következő:

- conf: beállítások
- data: a HTTP protokollal kiszolgált adatok
- log: napló fájlok
- run: aktuális futó folyamat állapota

Az apachectl segítségével lehet egyszerűen elindítani, leállítani a kiszolgálót, a beállításokat ellenőrizni.

A httpd.conf és apachectl fájlokban testre kell szabni a beállításokat:

- alapkönyvtár ($HOME/meres-pki-ssl/apache),
- HTTP és HTTPS portok, melyen a kiszolgáló fogadja a kéréseket.

Alapesetben a http port 8000 + a mérőhely száma. Tehát pl. aki pkimeres123 felhasználóval jelentkezett be, annak a 8123-as porton fog figyelni a http szerver. A HTTPS port a http port+1000, tehát a fenti példában a 9123-as.

3 Feladatok

A kitűzött feladatok célja, hogy az SSL alapú kommunikáció és az ahhoz szükséges PKI infrastruktúra technikai oldalát ismertessék a hallgatóval. A feladatok nem térnek ki a PKI rendszereknél szükséges adminisztratív teendőkre, mint például a tanúsítvány igénylőjének megfelelő azonosítására, a titkos kulcsuk tárolására és kezelésre vonatkozó előírásokra illetve azoknak a betartására, a jogi háttérre.

3.1 Hitelesítő központ felállítása

Hozzon létre OpenSSL segítségével egy hitelesítő központot, mely alkalmas kiszolgálók és böngészők számára tanúsítványt kibocsátani!

Milyen lépések szükségesek?

Ügyeljen a kulcs tárolására! Mire kell figyelni? Mekkora kulcsot célszerű használni?

A létrehozott hitelesítő központ miben nem felel meg a valós elvárásoknak?

3.2 Webkiszolgáló beüzemelése SSL motorral

3.2.1 Tanúsítvány kérelem készítése a webkiszolgálónak

Készítsen egy tanúsítvány kérelmet webkiszolgálónak! Mire kell a névválasztásnál figyelni?

Milyen egyéb szempontok vannak?

3.2.2 Tanúsítvány készítése a webkiszolgálónak a kérelem alapján

A kérelem alapján adjon ki tanúsítványt a hitelesítő központtal a webkiszolgáló számára! Milyen paraméterek lehetségesek a kiállításnál?

3.2.3 A webkiszolgáló beállítása

A kiadott tanúsítvány alapján állítsa be a webkiszolgálót, hogy HTTPS protokollal is ki tudja szolgálni a kéréseket. A nem titkosított kapcsolathoz a 80X0, a titkosítotthoz 80X1-es portot használja, ahol X a mérőhely számát jelenti. Vigyázzon arra, hogy ne fogadja el tartósan a böngésző számára a szerver által küldött tanúsítványt!

Milyen szempontokat kell figyelembe venni a kulcstárolási módjához? Tesztelje a beállításokat böngészővel! Mit tapasztal? Ellenőrizze az SSL kapcsolat létét a hálózati forgalom megfigyelésével!

3.2.4 Hitelesítő központ adatainak felvétele a böngészőkbe

Milyen haszna van a hitelesítő központ adatainak rögzítésének? Milyen előnyei és hátrányai vannak?

Milyen esetben célszerű rögzíteni az adatokat a böngészőben, melyekben nem?

3.2.5 SSL hálózati forgalom vizsgálata

Az SSL Handshake lehallgatásával állapítsa, hogy mely opcionális üzenetek kerülnek átvitelre és melyek nem. Magyarázza meg a tapasztaltakat!

Milyen rejtjelezési és integritás védelmi algoritmusban állapodik meg a webböngésző és a kiszolgáló?

3.2.6 Tanúsítvány igénylése a böngészőnek

Igényeljen és bocsásson ki tanúsítvány a böngészőjének! Állítsa be a webkiszolgálót, hogy az oldalakat csak érvényes böngésző tanúsítvány esetén szolgáltassa ki!

Milyen esetekben célszerű ügyfél oldali tanúsítványt használni? Soroljon fel néhányat!

Mennyiben változott az SSL Handshake folyamata?

3.2.7 Webkiszolgáló tanúsítványának visszavonása

Milyen esetben kerülhet visszavonásra egy tanúsítvány? Vonja vissza a tanúsítványt és frissítse a visszavonási listát!

3.2.8 Visszavonási lista frissítése a böngészőben

Frissítse a visszavonási listát a böngészőben! Mit tapasztalt előtte, mit utána? (Ügyeljen a helyes MIME típus beállításokra!)

3.2.9 Összegzés

Összegezze a feladatok során szerzett tapasztalatokat, milyen módszereket mely esetekben célszerű használni, milyen előnyökkel, hátrányokkal jár!

4 További információk

4.1 OpenSSL használata

Kulcspár létrehozása:

```
openssl genrsa -des3 -out mykey.key 512
```

Önaláírt tanúsítvány készítése:

```
openssl req -new -x509 -days 1 -key ca.key -out ca.crt
```

Tanúsítvány kérelem létrehozása:

```
openssl req -new -key server.key -out server.csr
```

Tanúsítvány kiadása:

```
openssl ca -out server.crt -in server.csr
```

Tanúsítvány és kulcs összevonása egy fájlba:

```
openssl pkcs12 -export -inkey client.key -in client.crt -out
client.p12
```

Tanúsítvány visszavonása:

```
openssl ca -revoke cert.crt
```

Visszavonási lista készítése:

```
openssl ca -gencrl -out crl.pem
```

Tanúsítvány konvertálása PEM-rol DER formátumba:

```
openssl crl -inform PEM -outform DER -in cert.pem -out cert.der
```

4.2 Apache webkiszolgáló

A kiszolgáló indítása:

```
apachectl start
```

A kiszolgáló leállítása:

```
apachectl stop
```

A kiszolgáló újraindítása:

```
apachectl restart
```

A kiszolgáló beállításainak tesztelése:

```
apachectl configtest
```

A beállítások a conf/httpd.conf fájlban találhatók.

```
SSLVerifyClient SSLCACertificateFile
```

4.3 Kapcsolódó anyagok, ajánlott olvasmányok

SSL - Secure Socket Layer

http://www.hit.bme.hu/~buttyan/courses/BMEVIHIM102/01_ssl-tls.pdf

The PKI page

http://www.pki-page.org

OpenSSL man oldalak

```
man openssl
```

OpenSSL honlap

http://www.openssl.org

Apache webkiszolgáló dokumentációja

http://httpd.apache.org/docs/

mod-ssl dokumentáció

http://www.modssl.org/

Mozilla: NSS and SSL Error Codes

http://www.mozilla.org/projects/security/pki/nss/ref/ssl/sslerr.html

Security Focus: An Introduction to OpenSSL, Part Three: PKI- Public Key Infrastructure

http://www.securityfocus.com/infocus/1466