Trusted Computing és a Trusted Platform Module

1 Trusted Computing

A Trusted Computing (TC) egy technológia, melynek célja annak biztosítása, hogy az ezt alkalmazó számítógépek minden esetben az elvárásoknak megfelelően működjenek. Az elvárás általában az, hogy a számítógépre telepített szoftverek módosítás nélkül fussanak, hogy a tárolt adatokhoz csak az arra kijelölt szoftverek férhessenek hozzá, valamint hogy a számítógéphez csatlakoztatva legyenek meghatározott hardver egységek. Az elvárásokat a számítógép felhasználója illetve egy külső fél is definiálhatja. A számítógép fogalma ebben az esetben tágan értelmezendő, a Trusted Computing a hagyományos PC-ken és kiszolgálókon túl beágyazott eszközökön – pl. set-top boxokon, mobiltelefonokon – is megvalósítható.

Mindezek teljesítéséhez a TC hardverben és szoftverben megvalósított kriptográfiai eljárásokat használ. Az ezeket összefogó hardver komponenst Trusted Platform Module-nak (TPM) nevezik. A TPM-et használó számítógépek és a rajtuk futó szoftverek összességét a TC terminológiájában platfomnak nevezzük.

A TPM és a kapcsolódó területek specifikációit az informatikai piac vezető szereplőit – többek között a Microsoftot, a Hewlet-Packardot és az Intelt - tömörítő Trusted Computing Group[1] adja ki és tartja karban, a TPM specifikáció 1.2-es verzióját az ISO/IEC szabványként beemelte[2].

1.1 Szolgáltatások

A Trusted Computing hat elv megvalósulását követeli meg:

Jóváhagyó kulcs

A megbízható platformok rendelkeznek egy úgynevezett jóváhagyó kulccsal. A jóváhagyó kulcs (Endorsement Key – EK) egy hardverbe gyártáskor beégetett, nem megváltoztatható és csak igen nehezen kiolvasható RSA kulcspár. Ennek segítségével a TC hardver képes azonosítani magát a megbízható platformon futó szoftverek, illetve ezeken keresztül egy távoli fél számára is.

Biztonságos be- és kimenet

A biztonságos be- és kimenet garantálja, hogy egy folyamat I/O műveletei csak az adott folyamat számára lesznek elérhetőek, így megakadályozható, hogy a hasznát csatornák lehallgatásával érzékeny információk váljanak hozzáférhetővé. A biztonságos I/O kihasználásával hatástalanná válnak a különböző billentyűzetfigyelő és képernyőrögzítő programok, de megakadályozható az is, hogy egy fájl a megbízható rendszeren kívülre kerüljön az USB-hubon, vagy akár a hangkimeneten keresztül.

Védett futtatás a memóriában

A megbízható szoftverfolyamatok memóriaterületei védettek az illetéktelen hozzáféréstől, így a futó programok működését külső folyamat nem módosíthatja, és a memóriában tárolt érzékeny információk hozzáférhetetlenné tehetők az illetéktelen processzek számára.

1 http://trustedcomputinggroup.org
2 http://www.trustedcomputinggroup.org/trusted_computing/standards_development

Kötött tárolás

A számítógép perzisztens tárában elhelyezett adatok hozzáférhetősége különböző hardver- és szoftverkomponensek meglétéhez köthető. A szolgáltatás használatával elérhető, hogy egy fájl vagy akár egy teljes lemezkötet csak egy adott számítógépen, csak egy adott típusú szoftverrel legyen olvasható.

Távoli tanúskodás

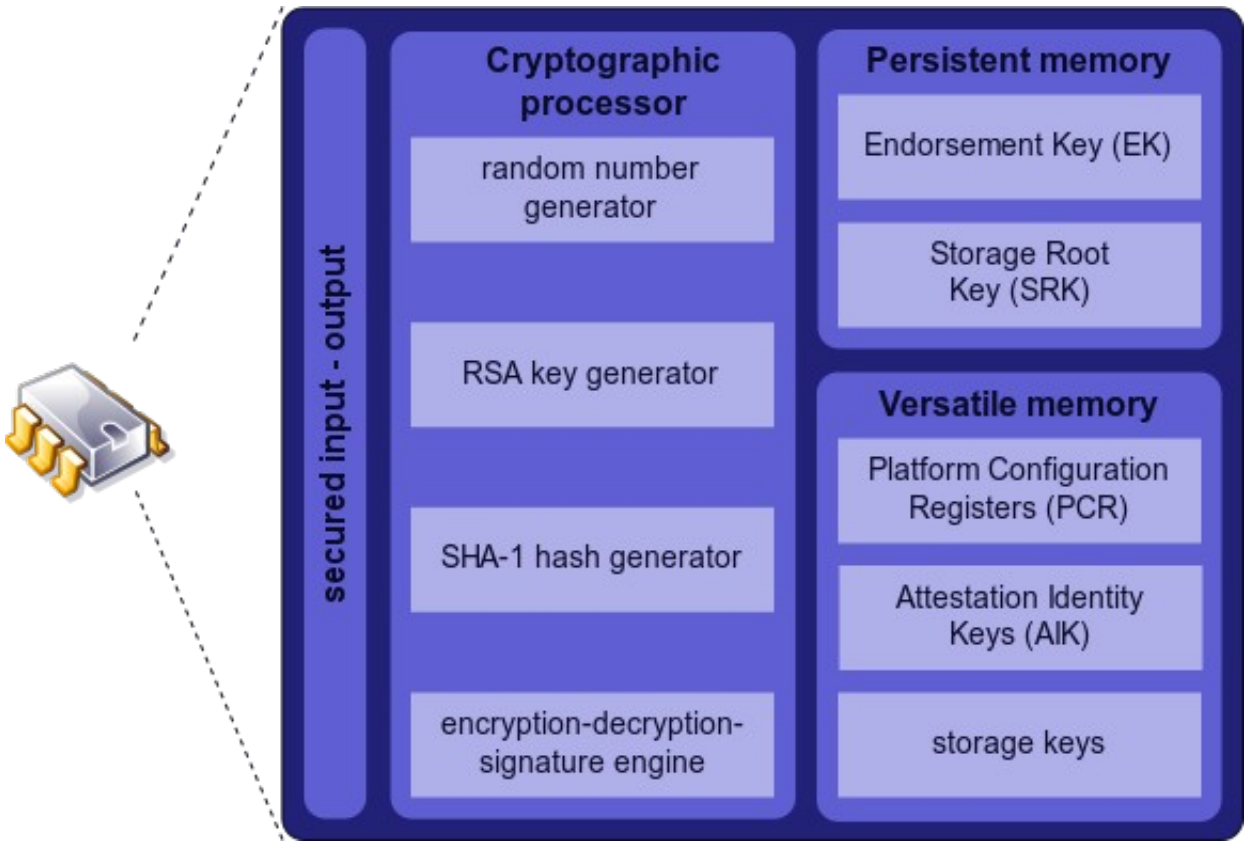
A távoli tanúskodás használatával egy megbízható fél képes detektálni a számítógép hardver- illetve szoftver-összeállításában történt változásokat. A TPM tanúsítványokat generál a futó szoftverekről, melyeket nyilvános kulcsú kriptográfiával védett csatornán képes eljuttatni az ellenőrzést végző félhez, például a szoftver gyártójához.

Megbízható harmadik fél

Annak érdekében, hogy egy távoli számítógépről eldönthető legyen, hogy az valóban eleget tesz-e a TC követelményeinek, ugyanakkor ne kelljen a számítógép identitását felfedni, egy megbízható harmadik félre (Trusted Third Party – TTP) van szükség. Amennyiben Alice meg szeretné győzni a megbízható platformot használó Bobot arról, hogy ő is megbízható platformot használ – azaz a számára küldött információk ugyanolyan biztonságban lesznek, mint Bobnál -, de nem szeretné felfedni identitását Bob előtt, Alice a TTP-hez fordul, aki felé hajlandó azonosítani magát, és aki egy azonosság-tanúsító kulcspár (Attestation Identity Key - AIK) hitelesítésével igazolja Bob számára Alice megbízható voltát.

2 Megvalósítás - Trusted Platform Module

A Trusted Platform Module egy kriptográfiai funkciókat és biztonságos tárolást megvalósító hardvereszköz, amely szorosan együttműködik a számítógép alapvető komponenseivel – a processzorral, a memóriával, illetve a különböző rendszerbuszokkal. A Trusted Platfom Module egy általános elnevezés a Trusted Computing Group TPM specifikációjának megvalósításaira, amely mögött több gyártó különböző implementációja állhat. Eddig több mint 100 millió személyi számítógépben helyeztek el ilyen eszközt[3].



A Trusted Platform Modul vázlatos belső felépítése

A TPM chipek önállóan képesek RSA kulcsgenerálást, titkosítást és aláírást végezni. A kulcsgenerálást a beépített álvéletlen-generátor, az aláírást a szintén az eszköz részeként működő (HMAC képzésre is képes) SHA-1 megvalósítás támogatja. A modul ezen kívül két különálló memóriaegységet is tartalmaz: egy nem-módosítható memória tárolja a jóváhagyó kulcsot és a perzisztens adatok titkosított tárolásához használt mesterkulcsot (Storage Root Key), egy általános célú memória pedig az AIK-k, az SRK-ból származtatott konkrét tároló kulcsok, valamint a rendszerkomponensek SHA-1 lenyomatait tartalmazó Platform Configuration Registerek (PCR) tárolására használható.

2.1 Kötött adattárolás

A TPM által nyújtott kétféle biztonságos adattárolási lehetőség a Data Sealing illetve a Data Binding. A perzisztens tárolóra kerülő adatok mindkét esetben titkosítódnak, a kulcsokat pedig a TPM tárolja, a különbség a megfejtési lehetőségekben van:

Data Binding esetén a titkosított adatok a TPM-hez kötötten fejthetők meg. A TPM a beégetett SRK-ból képes különböző adatok titkosítására alkalmas kulcsokat generálni, melyek szintén a TPM

3 http://www.wincert.net/news/security/1919-tpm-chips-dissolved-needled-prodded-and-hacked.html

hardverben tárolhatók. Így a titkosított adatok alapesetben csak a megfelelő TPM-mel rendelkező számítógéppel fejthetők meg, lehetőség van ugyanakkor úgynevezett migrálható kulcsok létrehozására is, melyek függetlenek a TPM beégetett adataitól és egy megbízható szoftverrel kinyerhetők a modulból, majd egy másik TPM-mel újra felhasználhatók.

Data Sealing esetén a TPM-hez kötöttség mellett megkövetelhető, hogy különböző hardver vagy szoftverelemek jelen legyenek a rendszerben a megfejtéshez. Ilyenkor a titkosítás paramétereként felhasználásra kerül a kiválasztott rendszerkomponensek egyedi lenyomatait tartalmazó PCR-ek tartalma. Migrálható kulcsok készítésére Data Sealing esetén nincs lehetőség.

2.2 Direkt Anonim Tanúskodás

Távoli platformok megbízhatóságának ellenőrzéséhez, amennyiben meg szeretnénk őrizni a felek anonimitását, szükség van egy megbízható harmadik félre, aki ismeri az összes érvényes EK-t és aki felé a felhasználók hajlandóak felfedni identitásukat. Az önmaga megbízhatóságát igazolni kívánó (Prover), a megbízhatóságot ellenőrző (Verifier) és a megbízható harmadik fél (TTP) közötti protokoll különbözőképpen működik a TPM 1.1-es és 1.2-es változataiban.

Az 1.1-es verzióban rögzített megoldás szinte magától értetődő: A TTP legyen egy, a Publik Key Infrastructure[4] területéről ismert Certificate Authority-hoz (CA) hasonló entitás, a Privacy CA! A Privacy CA ismeri a összes érvényes EK kulcsot, így bármely TPM képes vele biztonságosan kommunikálni. A Prover generál egy azonosság-tanúsító kulcspárt (AIK), melyet a Verifierrel történő kommunikációra fog használni. Ezt illetve saját EK-ját elküldi a Privacy CA-nak, aki ellenőrzi, hogy az EK érvényes-e, illetve hogy nem lett-e csalóként megjelölve. Amennyiben mindent rendben talált, az AIK-t a saját kulcspárjával hitelesíti. A Prover ezután elküldi a hitelesített kulcsot a Verifiernek, aki a Privacy CA aláírásának ellenőrzésével megbizonyosodhat arról, hogy megbízható platformmal van dolga, magát az AIK-t pedig a kommunikáció további részében használhatják a felek.

A protokoll ezen változatában a Privacy CA több szempontból is szűk keresztmetszetet jelent: Minden Verifier számára külön AIK-t kell generálni, ezek mindegyikének a hitelesítését a Privacy CA végzi. Ez maga után vonja, hogy a Privacy CA-nak nagy rendelkezésreállást kell biztosítania, ami részben ellentmond a szintén alapkövetelményként jelentkező biztonságos működésnek, hiszen online rendszereket nem, vagy csak nagyon nehezen lehet tesztelni, frissíteni vagy javítani.

Ezek mellett a Privacy CA-val történő összejátszás önmagában a teljes rendszer kompromittálódását jelenti: Ha a CA és a Verifier megosztják ismereteiket, a Prover anonimitása elveszik, míg a Prover és a CA összejátszása esetén a Verifiert verheti át egy nem megbízható platform, így a szerepek semmiképpen sem vonhatók össze, ráadásul a CA-k megbízhatóságát sem ellenőrzi senki!

Ezen problémák kivédésére a TPM 1.2-es verziójában egy új módszert, a Direkt Anonim Tanúskodást[5] (Direct Anonymous Attestation - DAA) dolgozták ki. Ebben a megbízható harmadik felet DAA Issuernek nevezik, és az 1.1-es verzióhoz hasonlóan az a feladata, hogy a TPM-ek identitását ellenőrizze, és az általuk szolgáltatott titkokat aláírja.

Az aláírt titok ebben az esetben azonban nem maga az AIK, hanem egy szintén DAA-nak nevezett titkos kulcs, amelyből elég ha csak egy létezik, vagyis nem kell minden Verifier számára külön hitelesíteni egy ilyet, ezzel csökkentve a DAA Issuer terhelését.

Mindezt az teszi lehetővé, hogy a Prover nem adja ki a tanúsítványt a Verifier felé, hanem csak egy kriptográfiai bizonyítékot szolgáltat arra, hogy rendelkezik ezzel a tanúsítvánnyal. A Prover emellett

4 http://archive.opengroup.org/public/tech/security/pki/apki_1-0.pdf
5 http://www.zurich.ibm.com/~jca/papers/brcach04.pdf

egy DAA-val aláírt üzenetet küld a Verifiernek, amely tartalmazza a Verifier nevét, az időt és a további kommunikációhoz használható AIK-t. Szintén bizonyításra kerül, hogy a DAA Issuer által hitelesített, és a legutóbbi üzenet aláírásához használt titok megegyezik.

A protokoll lényegi, nullaismeretű bizonyítást (zero-knowledge proof) megvalósító része vázlatosan a következőképpen írható le (A DAA protokoll a Camenisch-Lysyanskaya sémán[6] alapszik, melyben a DAA Issuer nyilvános kulcsa az (a,b,d,n) négyes, n pedig RSA modulus)[7]:

1) A Prover elküldi a DAA kulcsot és saját EK-ját a DAA Issuernek.
   $DAA = a^x \bmod n$
   ahol x a Prover TPM titka
2) A DAA Issuer ellenőrzi az EK-t, ha mindent rendben talál, kiállítja a DAA Camenisch-Lysyankaya aláírását, amelyet az alábbiak szerint előálló (c,e,s) hármas alkot:
   $c^e = DAAb^s d = a^x b^s d \bmod n$
3) A Prover elküldi a véletlen s', r1, r2, r3 értékekkel számított c' és t értéket a Verifiernek
   $c' = cb^{s'} = DAAb^{s''} d = a^x b^{s''} d \bmod n$
   $t = c'^{r1} a^{r2} b^{r3} \bmod n$
4) A Verifier egy véletlen R értéket küld a Provernek
5) A Prover kiszámítja, majd elküldi a Verifiernek a következő értékeket:
   $s1 = r1 - Re$
   $s2 = r2 - R(-x)$
   $s3 = r3 - R(-s'')$
6) A Verifier meggyőződhet róla, hogy a Prover rendelkezik a DAA Issuer által a DAA értékre kibocsátott aláírással, ha ellenőrzi a következő kongruencia teljesülését:
   $t = d^R c'^{s1} a^{s2} b^{s3} \bmod n$

Mivel a Verifier soha nem ismeri meg a DAA kulcsot illetve az arra vonatkozó aláírást, a Verifier és a DAA Issuer összejátszásával nem sérül a Prover anonimitása. A DAA Issuer és a Prover összejátszása esetén a Verifier számára nem garantált a Prover megbízhatósága.

Problémát jelent, ha egy DAA kulcs valahogyan kiszivárog, és nem megbízható entitások (csaló TPM-ek) is elkezdik használni azt. A csaló TPM-ek kiszűréséhez a Verifier feketelistát vagy gyakoriságelemzést használhat. Az ellenőrzés elvégzésének elősegítése érdekében a Prover egy

$$N = X^{DAA} \bmod p$$

értéket küld a Verifiernek, ahol X vagy egy véletlenszám, vagy a Verifier által meghatározott fix érték. A Verifier az általa ismert csaló DAA értékekből számított N-ek felhasználásával szűrheti ki a nemkívánatos TPM-jelölteket.

Véletlen X érték esetén nem lehetséges frekvenciaanalízist végezni, fix esetben viszont a Prover anonimitása sérül, mivel a Verifier nyomon tudja követni az azonos N értékekhez tartozó tevékenységeket. Ezt a problémát egy X generálására vonatkozó szabályrendszer megadásával, vagy a Verifier tevékenységeinek szétválasztásával lehet megoldani[8].

6 http://www.zurich.ibm.com/~gka/ePrivacy/identity-mixer.pdf
7 https://www.zisc.ethz.ch/events/ISC2004Slides/folien-jan-camenisch.pdf
8 https://www.zisc.ethz.ch/events/ISC2004Slides/folien-jan-camenisch.pdf

3 Alkalmazások

Elektronikus kereskedelem

Ez elektronikus kereskedelem (e-kereskedelem) a különböző javak elektronikus rendszereken, tipikusan interneten keresztüli értékesítését jelenti. Mivel a terület számottevő anyagi forrást vonz, megjelentek a különböző elektronikus támadási formák, melyek a nyilvános hálózatokat használó pénzügyi tranzakciók eltérítésére szakosodtak. Az ezek elleni védekezésül nyilvános-kulcsú infrastruktúrák épültek ki, de ezek egyelőre nem védenek meg az összes előforduló – általában klienseket célzó - támadástól.

A Trused Computing a meglévő PKI technológiák kiegészítőjeként szerepelhet. Bár a nyilvános kulcsú infrastruktúrákban megoldható a kliensek (vásárlók) azonosítása, az identitás-ellenőrzés komplexitása miatt (egy egyszerű felhasználó általában nem rendelkezik CA-által hitelesített tanúsítvánnyal) általában hagyományos, (egyszeri) jelszó alapú megvalósítások kerülnek alkalmazásra az e-kereskedelemben, amelyek adathalászattal, billentyűzetfigyeléssel és rokon módszerekkel támadhatók.

A Trusted Computing lehetőséget nyújt arra, hogy a vásárlók egy olyan, kereskedő által biztosított célszoftvert használjanak, melynek megbízható működése garantált, így soha nem csatlakozik a kereskedő kiszolgálója helyett más, nem megbízható hoszthoz, ami az adathalász támadások alapját adhatná. A memória- illetve tárolóvédelem pedig megakadályozza, hogy a felhasználó jelszavai valamilyen kártékony programon keresztül kiszivárogjanak.

Mindemellett a TPM egységek alkalmazásával a kereskedő egy hardver-alapú lehetőséget kap felhasználói azonosítására: a kereskedő rendszerében regisztrálhatók a vásárlók eszközeinek nyilvános TPM kulcsai, így pusztán a jelszavak megszerzésével egy idegen számítógépről nem indíthatna tranzakciókat egy rosszindulatú fél.

Szoftver-licensz kezelés

A szoftvergyártók már jó ideje igyekeznek olyan védelmi intézkedéseket alkalmazni termékeikben, amelyek megakadályozzák azok jogosulatlan használatát. Az ilyen intézkedések tipikus megjelenési formája az ún. termékkulcs, amely általában egy hosszú alfanumerikus karaktersorozat, melynek ismerete nélkül egy adott szoftvertermék nem, vagy csak korlátozott funkcionalitással telepíthető. Így, bár maga a szoftver tetszőleges példányban lemásolható, a termékkulcsok korlátozása gátat szab a jogosulatlan felhasználásnak. A termékkulcsok mellett a napjaink szoftverei egyre gyakrabban alkalmaznak termékaktiválást, melynek során a szoftver az interneten keresztül közli a gyártóval a telepítés tényét, és engedélyt kér a működésre.

A szoftveres megoldások azonban általában viszonylag könnyen visszafejthetők (reverse engineering) és módosíthatók, így szinte minden kereskedelmi szoftvernek elérhetők fizetés nélkül használható változatai. Ez egy globális probléma, amely különösen nagy károkat okoz az olyan piacokon mint például Kína vagy India, ahol a felhasználószám hatalmas, a szoftverért viszont csak igen kevesen fizetnek.

A TPM-ek felhasználásával garantálható a szoftverek érintetlensége, így megakadályozható, hogy valaki olyan programváltozatokat használjon, melyekből kiiktatták a termékkulcs-ellenőrzést, másrészt a szoftverpéldányok TPM-hez kötésével megakadályozható a másolatok készítése is. A termék használatára vonatkozó feltételeket ráadásul a gyártó bármikor ellenőrizheti akár az interneten keresztül is.

Digitális jogkezelés – DRM

A digitális formában terjesztett művek (pl. zenék, filmek, könyvek) analóg társaikkal ellentétben minőségvesztés nélkül másolhatók és terjeszthetők, ami szintén bevételkiesést eredményez a kiadótársaságok illetve az alkotók számára, hiszen a fogyasztók nem lesznek érdekeltek a tartalmak megvásárlásában, mikor azok azonos minőségű másolatait alternatív (illegális) forrásokból is megszerezhetik.

A digitális jogkezelés (Digital Rights Management - DRM) azon technológiák összessége, melyek a digitális szellemi javak előállítói illetve forgalmazói számára segítséget nyújtanak termékeik jogosulatlan felhasználásának megakadályozásában.

A TPM segítségével megoldható például, hogy a digitális műveket olyan formátumban terjesszék, melynek feldolgozására csak bizonyos, TPM-mel védett lejátszó alkalmazások legyenek képesek, a médiafájlokkal kapcsolatban pedig megszabható, hogy csak adott TPM-mel (és akár adott szoftverrel) rendelkező platformon legyenek hozzáférhetők.

Merevlemez-titkosítás

Merevlemez-titkosításról akkor beszélünk, amikor egy számítógép köteteinek teljes tartalmát titkosítva tároljuk. A fájlrendszer-szintű titkosítással ellentétben a titkosát fájlok illetve könyvtárak helyett teljes merevlemezblokkokra alkalmazzuk, így általában titkosításra kerülnek a fájlrendszer metaadatai (pl. inode bejegyzések) is.

A TPM – bár megléte nem követelmény a megvalósításához – jól alkalmazható a merevlemeztitkosítás során: A számítógép hardverével szorosan integrált modul már a rendszerindítás első fázisaiban is közreműködhet, így a merevlemezek Master Boot Recordja illetve boot partíciója is titkosíthatóvá válik, míg TPM (vagy más hasonló célhardver) használata nélkül ezeket a területeket nyíltan kell hagyni, mivel a titkosítás megfejtésére képes szoftverkomponensek csak azután töltődnek be, hogy az ezeken a speciális területeken tárolt adatok felhasználásra kerülnének. A nyíltan tárolt boot adatok lehetőséget kínálnának a betöltő olyan irányú módosítására, hogy az a titkosított adatok megfejtéséhez megadott kulcsokat rögzítse, majd egy támadóhoz továbbítsa.

Ezen túl a TPM használata feloldja azt merevlemez-titkosítással kapcsolatos dilemmát is, melynek lényege, hogy a perzisztensen tárolt adatok a kötetek csatolása (megfejtése) után hozzáférhetővé válnak a számítógépen futó rosszindulatú programok számára. TPM használata esetén az érzékeny adatokhoz csak megbízható szoftverek kapnak hozzáférést.

Kritpográfiai feladatok gyorsítása

A TPM-ben hardveresen megvalósított kriptográfiai funkciók jóval hatékonyabbak az általános CPU-ra tervezett szoftveres implementációknál, így a modult kriptográfiai koprocesszorként használva a nagy mennyiségű ilyen jellegű műveletet végző rendszerek működése jelentősen gyorsítható. Az ilyen jellegű alkalmazás valójában független a Trusted Computing elveitől.

Jelszómenedzsment

Optimális esetben egy felhasználó nem használja fel ugyanazt a jelszót két különböző helyen, így azonban a szolgáltatás-jelszó párok megjegyzése igen nehézzé válhat. A problémára a különböző jelszómenedzsment-szoftverek (pl. KeePass[9]) kínálnak megoldást, melyek a felhasználó jelszavait egy mesterjelszóval titkosítva tárolják, így csak egyetlen jelszót kell megjegyezni tetszőleges szolgáltatás igénybevételéhez.

9 http://keepass.info/

Ez a megoldás ugyanakkor azt is maga után vonja, hogy egy támadónak elegendő a mesterjelszót megszereznie ahhoz hogy az összes kezelt szolgáltatáshoz hozzáférést szerezzen, ez pedig egyszerűen megvalósítható egy billentyűzetfigyelő program, vagy egy, a jelszómenedzser memóriájához hozzáférő szoftver segítségével.

A TPM azonban képes megakadályozni az illetéktelen memória- és I/O hozzáférést, valamint a jelszóadatbázis számára is addicionális védelmet nyújthat a Data Sealing illetve Data Binding segítségével.

Rootkit védelem

A rootkitek olyan szoftverek, amelyek egy rendszerhez maximális jogosultsággal hozzáférő, rosszindulatú támadó számára lehetővé teszik a rendszer kompromittált voltának elfedését illetve a későbbi problémamentes, észrevétlen hozzáférést. A rootkitek átalában az operációs rendszer alapvető (kernel illetve felhasználói módú) elemeinek módosításával érik el, hogy a könyvtárlistázások során ne látszódjanak a támadó által létrehozott fájlok, a folyamatok listázásánál pedig a támadó által indított folyamatok.

Amennyiben az operációs rendszer komponenseinek integritását TPM segítségével tartjuk nyilván, az ilyen változások azonnal nyilvánvalóvá válnak.

A Trusted Computing Group által kiadott Platform Trust Services specifikációk[10] egységes interfészt definiálnak az szoftverek integritási adatainak lekérdezésére , valamint megadják a preferált jelentési formátumot is. A Platrom Trust Services nem követeli meg TPM jelenlétét, a modul alkalmazásával azonban biztonságosabb működés valósítható meg.

Adatszivárgás-védelem

Vállalati környezetben nagy jelentősége van annak, hogy az alkalmazottak se véletlenül, se szándékosan ne juttathassanak ki bizalmas üzleti dokumentumokat a vállalat által ellenőrzött informatikai infrastruktúrán kívülre.

Az adatszivárgás-védelem TPM segítségével a DRM-nél tárgyalt módon megvalósítható.

Csalás megakadályozása elosztott rendszerekben

Amennyiben egy feladat megoldását több távoli számítógép együttműködésével oldjuk meg, mindig számolni kell azzal, hogy néhány fél hamis számítási eredményeket tesz közzé, ezzel szabotálva a sikeres feladatmegoldást. Ilyen elosztott rendszerekre példa a BitTorrent fájlmegosztó hálózat[11], a prímszámkereső projektek[12] [13] vagy a grid rendszerek[14].

Mivel a TPM képes ellenőrizni a számítást végző szoftver sértetlenségét, valamint biztonságosan eljuttatni ezt az állapotinformációt a feladatot koordináló központ(ok)nak, kiszűrhetők azok az entitások, amelyek vélhetően nem a számítás elvégzése szempontjából előnyös eredményeket közölnek.

10 http://www.trustedcomputinggroup.org/resources/infrastructure_work_group_platform_trust_services_interface_specification_version_10
11 http://bittorrent.org/introduction.html
12 http://www.mersenne.org/
13 http://www.15k.org/
14 http://www.gridcomputing.com/

4 Támadások és Kritika

Szoftver sebezhetőségek

Egy átlagos személyi számítógép által kezel információkra a legnagyobb veszélyt napjainkban az számítógépen futó szoftverek biztonsági hibái jelentik. Ezek a problémák legtöbbször a nem elég körültekintő tervezésből illetve a hanyag programozói munkából adódnak, és rosszabb esetben lehetővé teszik egy támadó számára, hogy a hibás szoftver számára speciális bemenetet adva a saját parancsainak végrehajtására bírja rá a számítógépet. Ez a speciális bemenet lehet egy megfelelően összeállított, távoli hosztról érkező weboldal, egy pendrive-on hordozott elektronikus dokumentum, vagy akár a számítógép helyi felhasználója által közvetlenül megadott adat. Az ezeken keresztül elérhető kódfuttatási lehetőség alkalmat ad mindazon adatok kiszivárogtatására, amelyek a sebezhető szoftver folyamata által elérhetőek.

A Trusted Computingnak nem célja ezen problémák a kezelése, és valójában – bár a modern keretrendszerek, és a fejlesztési folyamatok megfelelő szabályozása sokat javítottak a helyzeten - irreális is lenne a szoftverhibák eltűnésében reménykedni. Ezek a problémák viszont lehetőséget adnak arra, hogy a szoftverek működését anélkül manipuláljuk, hogy magát a szoftvert megváltoztatnánk: a hibás program is minden esetben ugyanazt az előre kiszámítható viselkedést produkálja, amit a programozói elképzeltek és megvalósítottak, ám a szoftverhibák kihasználásával olyan működés is előcsalogatható, amelyet ezek a szakemberek sem láttak előre (különben nem követték volna el a hibákat). Ez viszont azt jelenti, hogy egy TPM által megbízhatónak titulált alkalmazás az elvárttól eltérő működést is megvalósíthat, ez pedig éppen a megbízhatóság követelményének mond ellent.

Ennek következtében első közelítésben elmondható, hogy a megbízható szoftverbe vetett hit legfeljebb olyan erős lehet mint a biztonsági hibától[15] mentes szoftverbe vetett hit. Ezt a képet árnyalja, hogy a biztonsági hibák kihasználhatóságát nagyban befolyásolhatja a szoftver belső felépítése, az operációs rendszer védelmi intézkedései, valamint a hardverarchitektúra is. A tapasztalat ugyanakkor azt mutatja, hogy általában nem építhetünk arra, hogy az általunk használt szoftverek biztonságosak lesznek.

Bontásellenállóság

A TPM szabvány nem követeli meg az eszközök gyártóitól, hogy a modulokat megvédjék a hardverszintű támadásoktól, egészen pontosan:

> *„A TPM-eket úgy tervezik, hogy ellenálljanak a szoftveres támadásoknak, és az olyan hardveres támadásoknak, melyek nem igényelnek különleges szakértelmet, kifinomult eszközöket, és hosszú időt.”*[16]

Ezek a paraméterek általában adottak rendelkezésére állnak egy ország hadserege, hírszerzése, és még jó néhány multinacionális cég számára is, így feltételezhető, hogy ezek a szervezetek hozzáférhetnek a TPM-ek adataihoz (különös tekintettel a jogosító és a tárolókat védő kulcsokra).

A probléma azonban nyugtalanítóbbá válik, ha figyelembe vesszük, hogy az idézett nyilatkozatot Christopher Tarnovsky egy 2010-es előadása kapcsán tették, ahol a kutató bemutatta[17], hogyan sikerült kinyernie titkosítatlan adatokat közvetlenül az egyik gyártó chipjéről. Ez persze nem jelenti azt, hogy a támadás könnyen kivitelezhető lenne:

15 Nem minden szoftverhiba biztonsági hiba
16 http://www.trustedcomputinggroup.org/media_room/faqs
17 http://redmondmag.com/articles/2010/02/03/black-hat-engineer-cracks-tpm-chip.aspx

A vizsgált chip belsejét egy sűrű szövésű fémháló és optikai érzékelők védték – előbbi megszakítása, vagy a burkolat felbontásával az érzékelőkre érkező fény működésképtelenné teszi a modult. Egy nagypontosságú elektronmikroszkóp és speciális, mikron pontosságú tűk szükségesek ahhoz, hogy a TPM processzorának adatbuszaira csatlakozva kiolvashatóvá váljanak az ott közlekedő adatok – köztük az EK és a tároló kulcsok.

Ezek a paraméterek természetesen még mindig kizárják a támadás általános elterjedését, de a kutatás bizonyítja, hogy nem csak az eleve különleges hatalommal rendelkező szervezetek férhetnek hozzá TPM-mel védett adatokhoz.

Hidegindítás

A hidegindításos támadás (cold-boot attack) egy tipikusan hordozható PC-kel szemben alkalmazható side-channel támadási fajta. Több fajtája létezik, melyek közös jellemzője, hogy a memóriában található érzékeny adatokat akkor olvassák ki, mikor az őket védő mechanizmusok – ide értve, hogy a memóriamodulok eltávolításával az operációs rendszer összeomlik - már nem működnek.

A hidegindításos támadások kivitelezéséhez fizikai hozzáférés szükséges a számítógéphez, melynek memóriájába már betöltésre került a megszerezni kívánt adat (tipikusan valamilyen kriptográfiai kulcs). A legdrasztikusabb módszer azt használja ki, hogy az elterjedt DRAM-ok és SRAM-ok bizonyos idővel az áramellátásuk megszűnése után is megtartják[18] a rajtuk tárolt adatokat, ez az idő pedig a modulok hűtésével növelhető. Megfelelő hűtés esetén elegendő idő áll rendelkezésre ahhoz, hogy a memóriamodult egy alkalmas eszközbe helyezve kiolvassák annak teljes tartalmát.

A módszer úgy finomítható, ha a memóriát nem távolítják el, hanem a számítógépet áramtalanítják (így a futó szoftverek nem képesek kiüríteni a memóriájukat) majd gyorsan visszakapcsolják úgy, hogy közben kikényszerítik egy speciális operációs-rendszer vagy boot-ROM betöltését, amely nem végzi el a memória nullázását, hanem helyette lementi annak tartalmát. Ez az eljárás is javítható a memória hűtésével.

Az ACPI-kompatibilis energiamenedzsmentet támogató számítógépek „alvó” illetve „hibernált” állapotai további segítségeket nyújtanak a hidegindításos támadások kivitelezéséhez. Hibernáláskor a memória tartalma a merevlemezre (vagy más perzisztens tárra) kerül, így az áram elvétele után is hosszú ideig megmaradnak az ideiglenes tárban felejtett érzékeny adatok. Hibernáláskor ugyanakkor a legtöbb érintett szoftver törli vagy titkosítja a tárolt kulcsait. Az alvó módba kerülés során ezzel szemben nem szünteti meg a memória áramellátását, hanem memóriába menti a processzor állapotát, „ébresztéskor” pedig az így tárolt állapot azonnal visszaállításra kerül, így további szoftveres védelemre nincs lehetőség.

Mindez komolyan érinti a TPM által védett tárolókat is: a hatékony hozzáférés érdekében az adathozzáféréshez közvetlenül használt kulcsok a memóriában titkosítás nélkül kerülnek tárolásra, és bár a TPM által nyújtott memóriavédelemnek köszönhetően normális futás esetén a fontos szoftverek memóriaterületeit nem olvashatják más processzek, a hidegindításos támadásokkal a kulcsok hozzáférhetővé válnak.

Több merevlemez titkosítást használó alkalmazás, - köztük a Microsoft BitLocker – feltételezi, hogy a TPM által szolgáltatott kulcsokhoz illetéktelenül nem lehet hozzáférni, így alapértelmezetten kizárólag ezt az adatot használják titokként a rejtjelezéshez. A memóriában tárolt kulcsok kiolvasásával ez a feltételezés természetesen megbukik, és a titkosítás hatástalanná válik.

18 Valójában a memóriamodulon tárolt adatok az áram elvételétől kezdve azonnal sérülnek, de a bitek „eltűnése” megjósolható módon zajlik le, ami lehetővé teszi azok visszaállítását, és a véglegesen elveszett információ próbálgatással történő kitalálása is lényegesen jobb eredményeket ad a titkosítási kulcsok nyers erővel történő támadásánál.

A problémára részleges megoldást jelent a Trusted Computing Group által kiadott Platform Reset Attack Mitigation Specification-ben a számítógépek BIOS-ával szemben támasztott követelmény, ami előírja a memória törlését gépindításkor. Ez természetesen nem szab gátat a memóriamodulok áthelyezésével kivitelezett támadásoknak.

Ki a jogos tulajdonos?

A TC-al kapcsolatos legtöbb kritika alapját az adja, hogy a technológia átadja a felhasználók számítógépe és adatai fölötti kontrollt külső – sokszor a Trusted Computing Grouppal szoros kapcsolatot ápoló – felek számára.

Ennek legszemléletesebb példája a DRM, melynek alapkoncepcióját ugyan az analóg alkotásokkal kapcsolatban alkalmazható korlátozások digitális világra történő kiterjesztése adja, ám ennél sokszor mégis sokkal szigorúbb feltételeket szab a művek fogyasztói számára. Például egy hagyományos magnókazettát le lehetett játszani egy walkmanben, egy autórádióban és természetesen egy asztali lejátszóban is, ezzel szemben a TPM-mel védett zeneművek felhasználása olyan mértékig korlátozható, hogy azokat a másolat jogos tulajdonosa is csak a saját számítógépén tudja lejátszani, egy MP3-lejátszón már nem. Sőt, olyan eset is elképzelhető, mikor új számítógép vásárlásakor a régi hardverhez kötött adatokat sem lehet továbbhordozni!

További következmény, hogy a védett műveket csak bizonyos, a mű kiadója által jóváhagyott szoftverek fogják tudni lejátszani, máskülönben egy speciális programmal DRM-mentes formátumra lehetne konvertálni a védett fájlokat. Emiatt a felhasználóknak nem, vagy csak korlátozottan lesz lehetőségük arra, hogy megválasszák a saját (elvileg tetszőleges alkalmazás futtatására alkalmas) számítógépükön futó szoftvereket, ami a felhasználó szabadsága mellet komolyan korlátozza a piaci versenyt is.

Aggodalomra adhat okot ezen kívül a tény, hogy a TPM-ek gyártói (a felhasználókkal ellentétben) elméletileg az összes EK és SRK kulcsot ismerhetik, vagyis ezek a vállalatok, illetve bárki, akivel ezek a vállalatok együttműködnek képesek lehetnek megfejteni a TPM-el védett adatokat, illetve megszemélyesíteni bármely felhasználót. De még ha feltételezzük, hogy a gyártók a lehető legnagyobb körültekintéssel járnak is el a kulcsok kezelésekor (pl. egyáltalán nem készítenek másolatot a titkos részekről), akkor is előfordulhat, hogy a hibás gyártás vagy tervezés következtében a TPM támadhatóvá válik. Az ilyen hibákkal kapcsolatban a gyártó jól felfogott érdeke a teljes titoktartás, tehát a felhasználók várhatóan nem fognak értesülni a problémákról.

Meghibásodás

Tipikusan a Data Sealing alkalmazásait érintő probléma (de a nem migrálható kulcsokat használó Data Bindigeket is érintheti), hogy az így kezelt adatok véglegesen hozzáférhetetlenné válnak, ha valamely kötéshez használt hardverelem, vagy maga a TPM meghibásodik. Az ilyen okból bekövetkező adatvesztés ellen biztonsági mentésekkel nem lehet védekezni, hiszen a technológia lényege veszne el akkor, ha idegen platformon hozzáférhető másolatokat készíthetnénk adatainkról.

5 Összefoglalás

A Trusted Computingban megfogalmazott törekvéseket kezdettől fogva heves kritika[19] [20] érte, első sorban a magánszférát védelmező aktivisták részéről. Bár a vita természetéből adódóan a bírálatok egy része túlzónak tűnhet, a megállapítások jelentős része fölött nem lehet elsiklani. A felmerülő problémákkal kapcsolatban a legnagyobb problémát talán az jelenti, hogy nem műszaki hibákról, hanem koncepcionális ellentmondásokról van szó, amelyeket nem lehet egy újratervezett hardverszériával vagy akár egy átdolgozott specifikációval orvosolni.

A hardver – vagyis a TPM – ugyanis önmagában egy átlagos felhasználó számára használható eszköznek tűnik a már meglévő, tisztán szoftveres információvédelmi technológiák kiegészítőjeként. Bár nagy biztonságot igénylő alkalmazásokat (pl. katonaság, államigazgatás) egyelőre korlátozza a modulok bonthatósága, egy egyszerű felhasználó vagy akár vállalat jó hasznát veheti a már megvalósult, vagy a jövőben elkészülő biztonsági megoldásoknak.

Ezzel együtt jogosnak tűnik a kérdés, hogy vajon a TC-ért összefogó gyártók valóban ennyi erőforrást fordítottak volna egy olyan technológia létrehozására, amely ugyan az illetéktelen felek számára megnehezítheti az érzékeny adatokhoz történő hozzáférést, de megakadályozni valójában nem tudja azt, ráadásul egy sor kellemetlenséget is okozhat a felhasználóknak?

A kritikusok válasza természetesen az, hogy a TC fő motivációja nem a felhasználók szempontjából biztonságosabb, hanem inkább a gyártók számára jobban kiaknázható platform létrehozása volt. Ezt az álláspontot látszik alátámasztani az a tény, hogy gyakorlatilag az összes megbízható, védett szoftver által kezelt adat elméletileg támadható marad a szoftver saját sebezhetőségein keresztül. Ezen sebezhetőségek felderítése és kihasználása viszont költséges folyamat, így az olyan „olcsó” adatok, mint a bármelyik (virtuális) áruház polcán megtalálható e-könyvek, zenealbumok, esetleg szoftverek megszerzéséért várhatóan inkább a legális utat fogják választani a felhasználók.

Mindemellett végül is egyelőre nem vagyunk alapvetően rákényszerítve sem a DRM-mel védett termékek, sem a különböző biztonsági szoftverek használatára, és amíg ezt elmondhatjuk, elég ha felismerjük, hogy a Trusted Computingnak is megvannak a maga előnyei és hátrányai, és csak rajtunk múlik, hogy hogyan használjuk ezt a technológiát. Azokban viszont, akik ezt a technológiát esetleg ránk szeretnék erőltetni, semmiképpen sem szabad megbíznunk.

19 http://lists.essential.org/pipermail/a2k/2005-February/000076.html
20 http://www.gnu.org/philosophy/can-you-trust.html