Tanúsítási jelentés

Hung-TJ-003-2003

a MultiSigno Developer

aláíró alkalmazás fejlesztő készletről

/Kopint-Datorg Rt./

/verzió: 1.2 /

Tartalom

1. A MultiSigno Developer legfontosabb tulajdonságainak összefoglalása

A MultiSigno Developer egy olyan fejlesztői készlet, mely Windows 32-bites operációs rendszerhez biztosít DLL felületet. Önmagában működésképtelen, megbízható aláíró alkalmazások fejlesztésére használható fel.
A MultiSigno Developer által nyújtott felület lehetőséget nyújt XML csomagok nyitására, dokumentumok és megjegyzések kezelésére, digitális aláírására és aláírások ellenőrzésére. Segítségével XML digitális aláírás szabványon alapuló aláíró (aláírás-létrehozó és aláírás-ellenőrző) alkalmazások fejleszthetők.
Az alábbi szabványos formátumokat és protokollokat támogatja:

- „XML Signature" szabványos csomagok kezelése, közte:
  - együttes aláírások kezelése (több dokumentumot összefogó csomag aláírása),
  - többszörös aláírások kezelése (több személy aláírása ugyanazon a csomagon),
- X.509 v3 tanúsítványok kezelése,
- tanúsítvány visszavonási listák lekérdezése a hitelesítés-szolgáltatóktól (HTTP, HTTPS, LDAP protokollokkal, a tanúsítványból kiolvasott elérési helyről),
- aláírás ellenőrzés (ahol mindig a rendszer idő az ellenőrzés alapja),
- időbélyegzés készíttetés és ellenőrzés (az RFC 3161 szabványt követő időbélyegző-szolgáltatókkal együttműködve).

A fenti aláírás létrehozó és ellenőrző funkciókon kívül a MultiSigno Developer támogatja a csomagok titkosítását és dekódolását is, de ez a funkcionalitás kívül esik jelen tanúsítvány jelentés hatókörén.
Ugyancsak kívül esnek jelen tanúsítvány jelentés hatókörén a MultiSigno Developer bázisán fejlesztett aláíró alkalmazások (bár a fejlesztő eszköz funkcionalitása, biztonságos és korrekt megvalósítása nyilván átöröklődik az ebből – szakszerűen és gondosan – fejlesztett alkalmazásokba).

A MultiSigno Developer fejlesztő készlet a Windows operációs rendszerek erőforrásaira, eszközeire támaszkodik. DLL elemei a Microsoft Crypto API függvényeit hívják meg, s ezen keresztül (tetszőleges szabványos CSP-t használva, valamint a CSP-vel kommunikáló driver-eken keresztül) magát az aláírás-létrehozó eszközt (intelligens kártyát) szólítják meg, mely szintén igen sokféle lehet. Az aláírandó/ellenőrizendő XML struktúrára az MS Crypto API-n keresztül történik az aláírás létrehozásának/ ellenőrzésének aktivizálása.

A MultiSigno Developer az alábbi algoritmusokat valósítja meg, illetve aktivizálja:

- a MultiSigno Developer által megvalósított (egy csomagon belül kezelt dokumentumok integritásának ellenőrzésére használt) lenyomatoló függvény: **SHA-1**
- a MultiSigno Developer által a CSP-n aktivizált (az XML struktúra digitális aláírására használt) lenyomatoló függvény: **MD5**
- a MultiSigno Developer által a CSP-n aktivizált, az intelligens kártyával végrehajtott (XML struktúra aláírására használt) digitális aláíró algoritmus: **RSA (1024 bit)**

A MultiSigno Developer képes együttműködni minden szabványos PC/SC kompatíbilis kártyaolvasóval és minden MD5-t, 1024 bites RSA algoritmust és szabványos MS Crypto API-t támogató kártyatípussal.

A MultiSigno Developer fejlesztő készlet (pack.dll) elemei lehetőséget nyújtanak szabványos XML csomagok nyitására, dokumentumok és megjegyzések kezelésére, aláírásra és az aláírások ellenőrzésére. Az alábbiak áttekintik a fejlesztő készlet által kínált lehetőségeket, egyúttal bemutatják (néhány programozási nyelvtől független lépésben) a felület alapvető funkcióinak használatát.

Első lépés, csomagnyitás

XML csomagot négy módon lehet nyitni:

- létező file-ból /a pack.dll által korábban létrehozott XML csomag megnyitása/,
- a memóriában /egy üres csomag létrehozása a megadott paraméterekkel/,
- létező, titkosított file-ból /a pack.dll által titkosítva mentett file-ból a dekódolás után a csomag megnyitása/,
- a memóriában tárolt forrásból /adatbázisból vagy más külső forrásból kapott csomagok megnyitása/.

Második lépés, csomag áttekintés

- csomag tulajdonságok lekérése /a csomag három fő tulajdonságának (a létrehozó neve, a létrehozás ideje és a csomag azonosítója) lekérdezése/,
- dokumentumok lekérése /lekérdezhető a dokumentumok száma, majd az egyes dokumentumok fejléce, s végül a tartalma is/,
- megjegyzések lekérése /lekérdezhető a megjegyzések száma, majd az egyes megjegyzések tartalma/.

Harmadik lépés, dokumentum-kezelés

Az alábbi funkciók a csomagban tárolt adatok karbantartására szolgálnak:

- dokumentum kezelés /dokumentum létrehozása file-ból vagy memóriában tárolt adatokból, dokumentum törlése a memóriából/,
- megjegyzés kezelés /megjegyzések létrehozása (memóriából) és törlése/.

Negyedik lépés, aláírás és ellenőrzés

Az aláírás létrehozása és ellenőrzése a pack.dll fő funkciói, ezekre számos függvény szolgál:

- dokumentummal kapcsolatos függvények
  - egy adott XML dokumentumon vagy megjegyzésen szereplő aláírások számának meghatározása,
  - annak meghatározása, hogy egy adott aláírás mely dokumentumokat hitelesíti (azonosítók felsorolásával és egy dokumentumszámmal),
  - egy aláírás ellenőrzése az aláírás azonosítója alapján,
  - digitális aláírások létrehozása,
- tanúsítvánnyal kapcsolatos függvények
  - tanúsítvány keresése adott e-mail cím alapján,
  - egy BASE64 kódolt tanúsítvány elmentése a Windows erre rendszeresített tárolójába,

- egy adott tanúsítvány megjelenítése a Windows natív tanúsítvány-megjelenítő ablakában,
- egy adott tárolóban található tanúsítványok számának megadása,
- egy tanúsítvány visszaadása a kért tárolóból (index alapján),
- egy tanúsítványhoz tartozó tanúsítványlánc érvényességének ellenőrzése egy adott időpontra nézve,
- egy adott aláíráshoz tartozó tanúsítványt visszaadása (az aláírás azonosítója alapján),
- egy tanúsítványból a tulajdonos nevének, e-mail címének és a tanúsítvány alanyának meghatározása,
- visszavonási lista letárolása,
- egy adott tanúsítványhoz tartozó visszavonási lista megkeresése a tárolt visszavonási listák között (a kiállító alapján), majd annak ellenőrzése, hogy a tanúsítvány szerepel-e a listán (visszavonták-e),
- a visszavonási lista elérési helyének kiolvasása a tanúsítványból, majd az ott definiáltak szerint HTTP-n, HTTPS-en vagy LDAP-on keresztül a lista letöltése és installálása.

Ötödik lépés, csomag lezárás

- csomag mentése /kódolatlanul (file-ba és memóriába) és kódolva (file-ba)/,
- csomag lezárása /a csomagkezelést megvalósító objektum lezárása, valamint a hozzá kapcsolódó memória felszabadítása/.

Egyéb függvények

- a pack.dll verziószámának megadása,
- bináris adatból BASE64 kódolt szöveg készítése, illetve ennek inverze,
- keresés végrehajtása egy LDAP könyvtárban,
- léptetés az LDAP keresés eredményei között (a keresett érték eléréséig, vagy a teljes találat kilistázásáig),
- az aktuális keresési rekord egy attribútumának lekérése,
- a keresés lezárása (és a lefoglalt memória felszabadítása).

2. A MultiSigno Developer értékelési követelményei a CEN/ISSS: 14170 és 14171 munkacsoport egyezményei szerint

Az alábbiakban áttekintjük azokat a (CEN/ISSS CWA 14170 és CEN/ISSS CWA 14171 követelményrendszereiből fakadó) funkcionális és biztonsági követelményeket, melyek minősített aláírások létrehozására és ellenőrzésére szolgáló alkalmazásokra vonatkoznak. A fokozott biztonságú aláírások számára fejlesztett MultiSigno Developer aláíró alkalmazás fejlesztő készletet ehhez a követelményrendszerhez hasonlítva jellemezzük. A fent említett nemzetközi (és nyilvánosan elérhető) dokumentumok által felállított funkcionális modellben szereplő összetevőket és egyéb fogalmakat ismertnek tételezzük fel. Ezek rövid összefoglalóját az „Aláíró alkalmazások funkcionális modellezése” című anyag is tartalmazza (készítette a HunGuard Kft.).

2.1 Funkcionális követelmények minősített elektronikus aláírásokat létrehozó/ellenőrző alkalmazások számára

F_SCA_1: Minden aláírás-létrehozó rendszer tartalmazzon egy (teljes) aláírás-ellenőrző rendszert is.

F_SDP_1: Minden aláírói dokumentumnak közvetett módon tartalmaznia kell egy tartalom-formátumot, amely meghatározza azokat a részleteket, ahogyan a dokumentumot az ellenőrző számára meg kell jeleníteni, vagy ahogyan fel kell használni.

F_SDP_2: Amennyiben az aláírói dokumentum szemantikája nem függ annak megjelenítésétől, akkor vagy az aláírói dokumentumban vagy egy aláírás tulajdonságban meg kell adni a tartalom egyértelműséghez szükséges információkat.

F_SDP_3: Amennyiben az aláírói dokumentum szemantikája függ annak megjelenítésétől, akkor az aláírónak elegendő információval kell ellátnia az aláírás ellenőrzőjét a dokumentum pontos megjelenítéséhez.

F_SDP_4: Az ellenőrzési folyamatok helyesen értelmezzék a F_SDP_1, F_SDP_2 és F_SDP_3 által megkövetelt információkat, illetve ezek alapján egyértelműen és helyesen jelenítsék azt meg az ellenőrző számára.

F_SAV_1: Mind az aláíró, mind az ellenőrző számára meg kell jeleníteni az aláírási tulajdonságokat, különös tekintettel a következőkre:

- az aláíró tanúsítványa,
- az aláíró dokumentumának tartalom-formátuma (ha szerepel),
- az aláírási szabályzat (ha szerepel),
- a kötelezettségvállalás típusa (ha szerepel).

F_SAV_2: Lehetőséget kell biztosítani az aláíró/ellenőrző számára ahhoz, hogy az aláíráshoz csatolandó/csatolt tanúsítványt átvizsgálja.

F_SIC_1: Egy aláírás létrehozása előtt meg kell győződni arról, hogy az aláíró valóban létre kíván hozni egy minősített elektronikus aláírást.

F_SIC_2: Az aláíró/ellenőrző számára vezérlő funkciók szükségesek, melyen keresztül irányíthatja az aláírási/ellenőrzési folyamatot és az aláírás-alkalmazás tevékenységét.

F_SIC_3: Egy elektronikus aláírás létrehozása előtt a biztonságos aláírás-létrehozó eszköznek és az aláírás-létrehozó alkalmazásnak is meg kell győződnie arról, hogy az aláíró a biztonságos aláírás-létrehozó eszköz tulajdonosa (vagy jogosult használója).

F_DTBSF_1: Ki kell alakítani a szabványos formattált aláírandó adatot az aláíró dokumentumából, az aláírási tulajdonságok felhasználásával.

F_DTBSF_2: Ha az aláírandó adatnak tartalmaznia kell az aláírói dokumentum lenyomatát, és ha ez még nem létezik, akkor a DTBSF összetevőnek kezdeményezni kell a lenyomatolási eljárást a formattált aláírandó adat kialakítása előtt.

F_DHC_1: Az aláírás-létrehozó folyamat kiváltása utáni első lépésként végre kell hajtani a lenyomatolást.

F_DHC_2: Második lépéseként végre kell hajtani a lenyomat formattálását (feltöltését).

F_SSC_1: Egy aláíró ellenőrzése alatti aláírás-létrehozó rendszer és a biztonságos aláírás-létrehozó eszköz között végre kell hajtani az 1. ábrán jelölt minden szükséges kommunikációt.

F_SSC_2[1]: Egy szolgáltató ellenőrzése alatti aláírás-létrehozó rendszer és a biztonságos aláírás-létrehozó eszköz között végre kell hajtani a 2. ábrán jelölt minden szükséges kommunikációt.

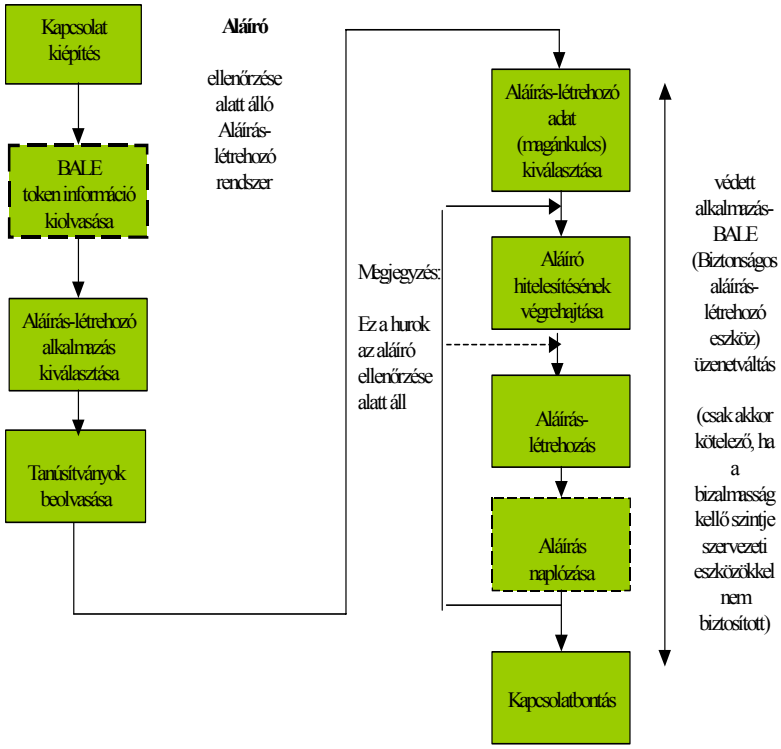
F_SSC_3: Az aláírás-létrehozó alkalmazásnak legalább egy fizikai interfésszel kell rendelkeznie, amely alkalmas a biztonságos aláírás-létrehozó eszközzel való kommunikációra.

F_SSC_4: A biztonságos aláírás-létrehozó eszköz funkcionalitása megvalósítható egy olyan platformon (pl. intelligens kártya), amely egy vagy több biztonságos aláírás-létrehozó eszköz funkciót (amelyeket gyakran biztonságos aláírás-létrehozó eszköz alkalmazásnak is neveznek) hordoz és, ezen felül esetleg más alkalmazásokat is. Ilyen több-alkalmazásos platform esetén az aláírás-létrehozó alkalmazásnak ki kell választania az egyiket.

F_SSC_5: Egy biztonságos aláírás-létrehozó eszköz hordozhat több tanúsítványt is. Ebben az esetben ki kell tudni választani az egyiket.

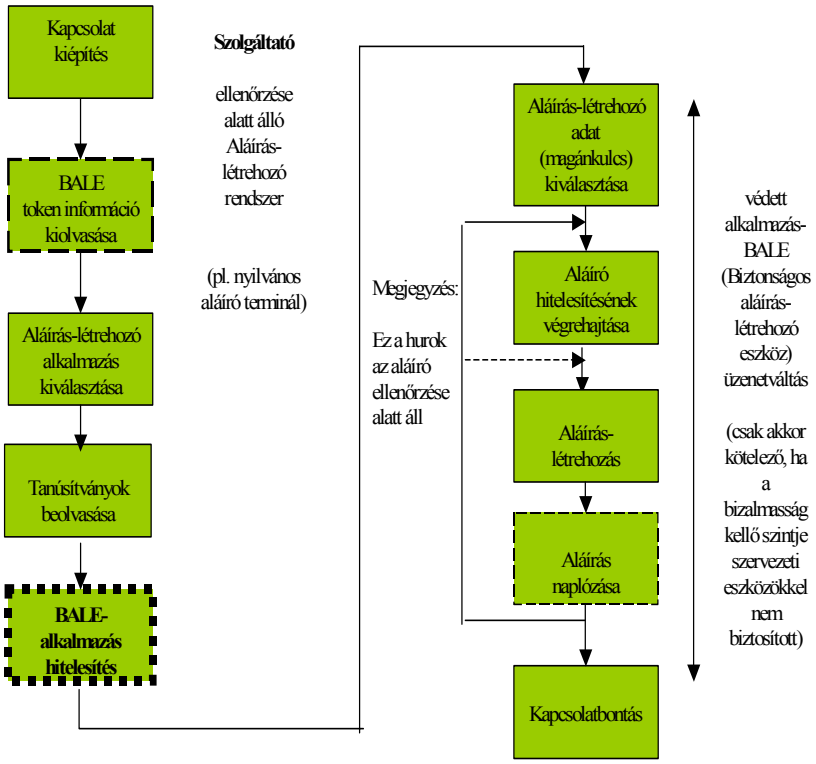
[1] Ez a funkcionális követelmény csak egy szolgáltató ellenőrzése alatt (is) működő aláíró rendszerre vonatkozik. A MultiSigno Developer fejlesztő készletet munkahelyi és otthoni felhasználásra szolgáló aláíró alkalmazásokhoz tervezték, ezért ez a követelmény nem vonatozik rá.

F_SSC_6: Ha egy biztonságos aláírás-létrehozó eszköz egynél több aláírás-létrehozó adatot (magánkulcsot) tartalmaz, akkor a megfelelőt ki kell tudni választani az aláíró szándéka szerint.



1. ábra

Az aláíró ellenőrzése alatt álló aláírás-létrehozó rendszer esetén megvalósítandó együttműködési sorozat az aláírás-létrehozó rendszer és a biztonságos aláírás-létrehozó eszköz között



2. ábra

Egy szolgáltató ellenőrzése alatt álló aláírás-létrehozó rendszer esetén megvalósítandó együttműködési sorozat az aláírás-létrehozó rendszer és a biztonságos aláírás-létrehozó eszköz között

F_SSC_7: A biztonságos aláírás-létrehozó eszköz és az aláírás-létrehozó alkalmazás közötti kommunikátor (SSC) összetevőnek át kell vennie az aláírót hitelesítő adatot az aláírót hitelesítő összetevőtől egy megbízható útvonalon keresztül, és el kell küldenie egy megfelelő parancs (utasítás) kíséretében a biztonságos aláírás-létrehozó eszköznek összehasonlításra.

F_SSC_8: Az aláírás létrehozó folyamat utolsó lépéseként ki kell számíttatni (a biztonságos aláírás-létrehozó eszköz által megvalósítva) magát az aláírást.

F_SSC_9: A befejezett aláírásokat naplózni kell.

F_SSA_1[2]: Ha az aláírás-létrehozás egy - szolgáltató ellenőrzése alatt álló – aláírás-létrehozó rendszeren történik meg, akkor az aláírónak képesnek kell lennie annak megállapítására, hogy feltételezhet-e ugyanolyan szintű bizalmasságot, mint amit a saját ellenőrzése alatt álló aláírás-létrehozó rendszer esetén elérhet.

F_SDC_1: Lehetővé kell tenni az aláíró számára az aláírói dokumentum létrehozását vagy kiválasztását.

F_SDOC_1: Össze kell kapcsolni a biztonságos aláírás-létrehozó eszköz kimeneti adatát (az elektronikus aláírást) a formattált aláírt adattal, a szabvány formátumnak megfelelően.

F_SLC_1: Minden létrehozott/ellenőrzött aláírásra egy napló bejegyzés tárolandó.

F_SCPC_1: A hitelesítés-szolgáltatóval való kapcsolat felvételén keresztül képesnek kell lennie az alábbiakra:

- az aláírói tanúsítványok megszerzése,
- az aláírói tanúsítványok állapotának lekérése.

F_I/O-1: Ha aláírás-létrehozásnál a biztonságos aláírás-létrehozó eszköz nem tartalmaz minden szükséges tanúsítványt az aláírási folyamathoz (mert csak a tanúsítvány azonosítókat tartalmazza), vagy ha aláírás-ellenőrzésnél az aláírótól nem érkezett meg a szükséges tanúsítvány (csak annak azonosítója), akkor az aláírás-alkalmazásnak képesnek kell lennie arra, hogy ezeket a tanúsítványokat megszerezze (lekérdezze a hitelesítés-szolgáltatótól).

F_I/O-2: Az aláírás-alkalmazásnak képesnek kell lennie arra, hogy a megszerzett tanúsítványok hitelességét ellenőrizze.

F_I/O-3[3]: Ha az aláírói dokumentumot, vagy annak egy részét, vagy az aláírási tulajdonságokat egy input/output interfészen keresztül adják meg, az aláírás-létrehozó alkalmazásnak biztosítania kell, hogy egyetlen rejtett rész se játszhasson szerepet, és hogy egyetlen aláírandó adat összetevőt se cserélhessenek ki.

F_ISV-1: Az érvényesítő adatokat az ellenőrzőnek be kell gyűjtenie, és az aláírási szabályzat minden követelményét teljesítenie kell.

F_ISV-2: Ha szükség lehet utólagos ellenőrzésre, az érvényesítő adatoknak tartalmazniuk kell annak bizonyítékát, hogy a felhasznált tanúsítvány lánc érvényes volt az aláírás létrehozásának időpontjában.

F_USV-1: A kezdeti ellenőrzés során begyűjtött érvényesítő adatok ellenőrzésénél az aláírási szabályzat minden követelményét teljesíteni kell.

---

[2] Ez a funkcionális követelmény csak egy szolgáltató ellenőrzése alatt (is) működő aláíró rendszerre vonatkozik. A MultiSigno Developer fejlesztő készletet munkahelyi és otthoni felhasználásra szolgáló aláíró alkalmazásokhoz tervezték, ezért ez a követelmény nem vonatkozik rá.

[3] A MultiSigno Developer nem céloz meg ilyen típusú alkalmazást (ahol az aláírandó adatok nem helyben készülnek, vagy választódnak ki), ezért ez a követelmény nem vonatkozik rá.

Ember által történő ellenőrzés esetén[4]:

F_human_1: Az aláírás-ellenőrző rendszernek eszközt kell biztosítania a felhasználó számára, amelyen keresztül az kommunikálni tud a rendszerrel. Ha az aláírt adatokhoz egynél több aláírás van hozzárendelve, akkor ennek a kommunikációnak azzal kell kezdődnie, hogy kijelzésre kerül az aláírások száma, amelyek feltehetően léteznek, és fel kell kínálni, hogy melyik legyen az ellenőrizendő.

F_human_2: Az aláírás-ellenőrző rendszernek interaktív eszközt kell biztosítania a felhasználó számára, hogy megnézhesse az aláírási szabályzat teljes egészét, vagy legalább az alkalmazási területre és feltételekre vonatkozó részeket.

F_human_3: A felhasználói felületnek megfelelő módon meg kell jelenítenie az aláíró dokumentumot, hogy az aláírást ellenőrző személy képes legyen az aláírói dokumentum tartalmának kellő meghatározására. Teljesíteni kell az "Ami megjelenik, azt írták alá." követelményt. Ha valamilyen okból az aláírói dokumentum nem jeleníthető meg pontosan a megfelelő módon, akkor ezt a felhasználói interfésznek világosan jeleznie kell.

F_human_4: Az aláíró azonosítójának, vagyis az állítólagos aláíró nevének vagy felvett nevének megjeleníthetőnek kell lenni. Az adott nevet az aláíró tanúsítványában szereplő, "megkülönböztető név" információjából kell venni. Ha az aláíró nem bocsátotta rendelkezésre a tanúsítványt, a hitelesítés-szolgáltató nevét kell megjeleníteni ehelyett, és ha ez a név elfogadható, akkor az adott tanúsítványt a hálózati interfész felhasználásával be kell szerezni. Az említett névnek csak a tanúsítványt kibocsátó hitelesítés-szolgáltató számára van jelentése, ezért a hitelesítés-szolgáltató nevét az aláíró nevével együtt ki kell jelezni.
A kezdeti aláírás-ellenőrzési folyamat kimenő állapota az alábbiak egyike legyen:

- „érvényes" állapot (sikeres ellenőrzés),
- „érvénytelen" állapot (sikertelen ellenőrzés),
- „befejezetlen ellenőrzés" állapot (befejezetlen ellenőrzés)

F_human_5: „Befejezetlen ellenőrzés" állapot esetén az aláírás-ellenőrző alkalmazásnak javasolnia kell a felhasználó számára, hogy szerezze be azt az információt, ami az aláírást érvényessé teszi hosszú távra.

F_human_6: A felhasználói interfészekre teljesüljenek az F_principles egyszerűségre és hibamentességre vonatkozó speciális elvárásai.

[4] A MultiSigno Developer fejlesztő készlet ember által végzett ellenőrzést nem támogat. A beérkező elektronikus aláírások automatikusan ellenőrzését képes megvalósítani (ennek eredményét a segítségével fejlesztett aláíró alkalmazásnak kell megjelenítenie a felhasználó számára). Következésképpen az F_human_x követelmények nem vonatkoznak rá.

Gépi (automatikus) ellenőrzés esetén:

Az automatizált feldolgozás esetében alkalmazói program interfészek (API-k) használhatók. Bár az ilyen interfészek kialakításának többféle módja van, ezeket két csoportba lehet sorolni:

- Az 1-es típusú API-k arra szolgálnak, hogy az elektronikus aláírásban tárolt adatokat kigyűjtsék.
- A 2-es típusú API-k az elektronikus aláírás ellenőrzésére és az érvényesítő adatok beszerzésére szolgálnak.

F_machine_1: Az 1-es típusú API-k alkalmasnak kell lenniük az elektronikus aláírásban tárolt információk kinyerésére és az elektronikus aláírás formátumának meghatározására.
Amennyiben az aláírási szabályzat explicit, akkor a szabályzat azonosítóját magából az elektronikus aláírásból kell kinyerni az 1-es típusú API-jainak felhasználásával[5].

F_machine_2: A 2-es típusú API-k az elektronikus aláírások hitelesítését és/vagy ellenőrzését kell lehetővé tenniük, illetve be kell szerezniük az aláírói információkat, az output állapotot és az érvényesítő adatokat.

F_general_1: A rendszer által megvalósított aláírás-ellenőrzési folyamatnak meg kell felelnie egy ember számára olvasható formájú leírásnak, feltételezve, hogy az aláírási szabályzat minden feldolgozási szabálya világosan meghatározott.

F_protocol: Mind az aláírás-létrehozó, mind az aláírás-ellenőrzési alkalmazásnak szabványos protokollt kell használnia a megbízható szolgáltatóval (szolgáltatókkal) történő kommunikáció során. Ez a következőket foglalja magában:

- tanúsítvány visszavonási állapot megszerzésekor;
- időbélyeg kérelem és válasz esetén;
- egyéb esetekben (pl. központi archiválási, időjelzési, naplózási szolgáltatások igénybe vétele esetén).

F_format: Mind az aláírás-létrehozó, mind az aláírás-ellenőrzési alkalmazásnak képesnek kell lennie szabványos formátumok kezelésére az alábbi területeken:

- szabványos aláírási formátumok;
- szabványos tanúsítvány formátumok.

F_principles: A felhasználói (aláírói, aláírás-ellenőrzői) felületek tervezésekor a következő elveket kell figyelembe venni:

- alkalmasnak kell lennie a feladatra;
- konzisztensnek kell lennie;
- felhasználóbarátnak (könnyen érthető, egyszerűen használható) kell lennie;
- ellenőrizhetőnek kell lennie;
- hibatűrőnek kell lennie;
- lehetővé kell tennie az egyedi beállításokat;
- egyenlőségen alapuló hozzáférést kell biztosítania;
- megfelelő állapotjelzéseket és hibaüzeneteket kell küldenie a felhasználó számára.

[5] A MultiSigno Developer nem támogat explicit aláírási szabályzat automatizált feldolgozását, ezért a követelmény második része nem vonatkozik rá.

A felhasználókkal (aláírók, ellenőrzők) párbeszédet folytató rendszer teljesítse az alábbiakat:

- félreérthetetlen felhasználói útmutatót kell szolgáltatnia arra nézve, hogy hogyan kell a rendszert installálni, konfigurálni és használni;
- ön-leírónak kell lennie abban az értelemben, hogy minden párbeszéd-lépésnek azonnal érthetőnek kell lennie vagy a rendszertől kapott visszajelzéseken keresztül, vagy úgy, hogy az ellenőrző kérésére a rendszer magyarázatot ad;
- meg kell felelnie a felhasználók szokásos elvárásainak, azaz tudásuknak, képzettségüknek, tapasztalatuknak és az általánosan elfogadott konvencióknak;
- adaptálhatónak kell lennie, azaz támogatnia kell a felhasználók egyéni igényeit és preferenciáit;
- hibatűrőnek kell lennie úgy, hogy a nyilvánvaló input hibák ellenére az eredményt el lehessen érni minimális javításokkal.
- tájékoztató hiba üzeneteket kell küldenie, a felhasználó továbbhaladása érdekében;
- visszajelzéseket kell szolgáltatnia, mely megerősíti a felhasználó által végrehajtott tevékenység helyességét (vagy helytelenségét);
- a hibaüzenetek legyenek kellően informatívak, adjanak eligazítást a hiba okáról, a szükséges teendőkről (pl. a “Hibakód: 213” hibaüzenet nem igazán segítőkész);
- szabatos és minden részletre kiterjedő terminológia helyett hétköznapi kifejezéseket kell használni (a technikai kifejezéseket ugyanis a legtöbb felhasználó nem érti, és nem is kell értenie);
- alkalmaznia kell a színek használatára vonatkozó konvenciókat (pl. piros = hiba, zöld = továbbhaladás/siker);
- minden időpontban képesnek kell lennie arra, hogy az éppen végrehajtás alatt álló műveletet félbeszakítsa és vagy visszatérjen a főmenübe, vagy teljesen kilépjen a rendszerből;
- a felhasználói egyének számára biztosítania kell a magántitok jelleget (pl. azáltal, hogy az információkat nem teszi mások számára hozzáférhetővé a felhasználói interfészen keresztül).

A műveletek helyes időzítésével elegendő időt kell biztosítani minden felhasználónak a folyamatok befejezéséhez (figyelembe véve azt a tényt is, hogy az emberek olvasási és reagálási és reagálási képességei különbözők).

2.2 Biztonsági követelmények minősített elektronikus aláírásokat létrehozó/ellenőrző alkalmazások számára

2.2.1 Követelmények az aláírás-létrehozó alkalmazás (SCA) egészére

Az aláírás-létrehozó alkalmazás és a biztonságos aláírás-létrehozó eszköz közötti megbízható útvonalra vonatkozó követelmények:

Bizt_köv1: Az aláírás-létrehozó alkalmazásnak meg kell őriznie a következők sértetlenségét:

- aláírandó adat (DTBS), formattált aláírandó adat (DTBSF), aláírandó adat reprezentáns (DTBSR) és minden egyéb, az aláíró által szolgáltatott információ,
- Az aláírás-létrehozó alkalmazás és a biztonságos aláírás-létrehozó eszköz között áramló valamennyi protokoll adat.

Bizt_köv2: Az aláírás-létrehozó alkalmazásnak meg kell őriznie az aláírandó adat komponensek, a formattált aláírandó adat és az aláírót hitelesítő adatok bizalmasságát.

Bizt_köv7[6]: Az aláírás-létrehozó alkalmazásnak biztosítania kell, hogy az aláírónak bemutatott aláírandó adat ugyanaz, mint amit az aláíró kiválasztott.

Bizt_köv8: Az aláírás-létrehozó alkalmazásnak biztosítania kell, hogy a formattált aláírandó adat és aláírandó adat reprezentáns előállításához felhasznált aláírandó adat komponensek ugyanazok, mint amelyeket az aláírónak bemutattak a bemutatási eljárás során, és amelyeket a felhasználó kiválasztott.

A nem megbízható folyamatokból/kommunikációs portokból adódó követelmény

Bizt_köv11[7]: Meg kell gátolni, hogy az aláírási folyamatba beavatkozhassanak olyan nem-megbízható rendszer és alkalmazási folyamatok, perifériák és kommunikációs csatornák, amelyek nem szükségesek az aláírás-létrehozás alkalmazás működéséhez.

Az aláírandó adatra vonatkozó követelmények

Bizt_köv12: Az aláírandó adatnak tartalmaznia kell egy aláírói dokumentumot. (Egy "üres" dokumentumhoz ne lehessen aláírást előállítani).

Bizt_köv13: Az aláírandó adatnak tartalmaznia kell az aláírónak azt a tanúsítványát, amely az elektronikus aláírás létrehozásánál a biztonságos aláírás-létrehozó eszköz által felhasznált aláírás-létrehozó adathoz kapcsolódik, s amely az aláíró szándékának megfelel.

Bizt_köv14: Az aláírandó adatnak tartalmaznia kell egy hivatkozást az aláírási szabályzatra.

[6] A 3.-6. követelmények csak a nyilvános aláíró alkalmazásokra vonatkoznak (melyek egy szolgáltató ellenőrzése alatt állnak /(pl. postahivatal, Internet-kávézó, stb./). A MultiSigno Developer fejlesztő készletet munkahelyi és otthoni felhasználásra szolgáló aláíró alkalmazásokhoz tervezték, ezért ezek a követelmények nem vonatoznak rá.

[7] A 9.-10. követelmények csak az osztott architektúrájú aláíró alkalmazásokra vonatkoznak. A MultiSigno Developert nem ilyen architektúrájú aláíró alkalmazásokhoz tervezték, ezért ezek a követelmények nem vonatoznak rá.

Bizt_köv15: Az aláírandó adatnak tartalmaznia kell a kötelezettségvállalás típus tulajdonságot, ha az aláírási szabályzat egynél több kötelezettségvállalás típust határoz meg.

Bizt_köv16: Ha az alkalmazás vagy az érvényben lévő biztonsági szabályzat egynél több aláírói dokumentum tartalom formátumot enged meg, az aláírandó adatnak tartalmaznia kell az aláírói dokumentum tartalom formátumot.

2.2.2 Követelmények az aláíró dokumentumát megjelenítő összetevőre (SDP)

A tartalom formátumra vonatkozó követelmények:

Bizt_köv17: Az aláíró dokumentumát megjelenítő összetevőnek lehetővé kell tennie az aláírói dokumentum tartalom-formátumának csatolását vagy közvetett módon az aláírási szabályzat részeként, vagy pedig egy közvetlen aláírás tulajdonságként.

Bizt_köv18: Az aláíró dokumentumát megjelenítő összetevőnek figyelmeztetnie kell az aláírót, ha a dokumentum nem felel meg a tartalom formátummal meghatározott szintaxisnak, és lehetővé kell tennie az aláíró számára, hogy félbeszakítsa az aláírási folyamatot.

Bizt_köv19: A használati útmutatóban jelezni kell, hogy milyen tartalom formátum helyes kezelésére alkalmas az aláíró dokumentumát megjelenítő összetevő.

Bizt_köv20: A használati útmutatóban jelezni kell, hogy milyen lehetséges következménnyel jár, ha az aláíró tévesen választja ki a tartalom formátumot.

Bizt_köv21: Az aláíró dokumentumát megjelenítő összetevőnek figyelmeztetnie kell az aláírót, ha olyan aláírói dokumentumot kíván aláírni, amelynek tartalom formátumát nem támogatja.

Bizt_köv22: Az aláíró dokumentumát megjelenítő összetevőnek biztosítania kell, hogy az aláírónak megmutatott aláírói dokumentum ugyanaz, mint amit az aláírási folyamat fog használni, és ugyanaz, mint amit az aláíró választott ki aláírásra.

Bizt_köv23: A megjelenítő folyamatnak tájékoztatnia kell az aláírót, hogy egyéb aláírt adatok vannak beágyazva az aláírói dokumentumba (az aláíró dokumentumát megjelenítő összetevőnek kapcsolódnia kell egy aláírás ellenőrző rendszerrel az ilyen aláírások ellenőrzésére).

Bizt_köv24: Az aláíró dokumentumát megjelenítő összetevőnek nem szabad lehetővé tennie az aláíró számára, hogy az aláírói dokumentum bármely részét megváltoztassa.

Bizt_köv25: Az aláíró dokumentumát megjelenítő összetevőnek figyelmeztetnie kell az aláírót, ha nem képes az aláírói dokumentum minden részének a helyes, tartalom formátumnak megfelelő megjelenítésére.

Az aláíró dokumentumára vonatkozó egyértelműségi követelmény:

Bizt_köv26: Az aláírás-létrehozó alkalmazásnak lehetővé kell tennie egy tartalom formátum tulajdonság csatolását az aláírandó adatokhoz annak biztosítására, hogy az aláíró dokumentumának megjelenítése egyértelmű legyen. Vagyis pontosan úgy lehessen azt a későbbiekben megjeleníteni, mint ahogyan az aláírónak a megjelenítési folyamat során.

A nem megjelenítés-érzékeny aláírói dokumentumokra vonatkozó követelmény:

Bizt_köv27: Az aláírás-létrehozó alkalmazásnak lehetővé kell tennie egy tartalom formátum tulajdonság csatolását az aláírandó adatokhoz annak biztosítására, hogy az aláíró dokumentum szemantikáját csak egyféleképpen lehessen értelmezni.

A rejtett szövegre és aktív kódra vonatkozó követelmény:

Bizt_köv28: Az aláírás-létrehozó alkalmazásnak figyelmeztetnie kell az aláírót a rejtett szövegek, makrók vagy aktív kódok jelenlétére. Amennyiben az aláírási szabályzat nem engedélyezi az ilyen aláírói dokumentumok aláírását, akkor az aláírás-létrehozó alkalmazásnak érvényre kell juttatnia ezt a tiltást.

2.2.3 Követelmények az aláírás tulajdonságokat megjelenítő összetevőre (SAV)

Bizt_köv29: Az aláírás tulajdonság megjelenítési folyamatának lehetővé kell tennie az aláíró számára az aláírás tulajdonságok megtekintését.

Bizt_köv30: Az aláírás tulajdonságokat megjelenítő folyamatnak biztosítania kell, hogy az aláírónak megjelenített aláírás tulajdonság ugyanaz, mint ami az aláírás folyamatában aláírásra kerül majd, és amit az aláíró kiválasztott az aláíráshoz.

Bizt_köv31: Az aláírás tulajdonságok sértetlenségét és hitelességét meg kell védeni.

Bizt_köv32: Az aláírót figyelmeztetni kell az aláírás tulajdonságokban jelenlévő bármilyen rejtett szövegről, makróról vagy aktív kódról. Amennyiben az aláírási szabályzat nem engedélyezi az ilyen aláírói dokumentumok aláírását, akkor az aláírás-létrehozó alkalmazásnak érvényre kell juttatnia ezt a tiltást.

Bizt_köv33: Az aláírás tulajdonság megjelenítő folyamatnak figyelmeztetnie kell az aláírót bármely, az aláírás tulajdonságokba beágyazott rejtett vagy aktív komponens (pl. word processzor makró) jelenlétére. Amennyiben az aláírási szabályzat nem engedélyezi az ilyen aláírói dokumentumok aláírását, akkor az aláírás-létrehozó alkalmazásnak érvényre kell juttatnia ezt a tiltást.

Bizt_köv34: Az aláírás tulajdonság megjelenítő összetevőnek lehetővé kell tennie az aláíró számára, hogy átvizsgálja a kiválasztott, aláírandó adatokhoz csatolandó tanúsítvány fő összetevőit.

2.2.4 Követelmények az aláíróval kölcsönható összetevőre (SIC)

Az aláírás kiváltására vonatkozó követelmények

Bizt_köv35: Az aláírási folyamat megkezdése előtt az aláíróval kölcsönható összetevőnek egy olyan nem nyilvánvaló, az aláírás-létrehozó alkalmazással folytatott, aláírás kiváltási cselekvést kell elvárnia az aláírótól, amely véletlenül valószínűleg nem következne be.

Az inaktivitási időkorlátra vonatkozó biztonsági követelmények

Bizt_köv36: Az aláírás-létrehozó alkalmazásnak egy korlátot kell megadnia arra az időtartamra, ami az aláírót hitelesítő adatok megadásától az aláírás kiváltásáig eltelhet.

Bizt_köv37: Ha az időkorlát letelik, az egész aláírási folyamatot félbe kell szakítani, az aláírótól az aláírási folyamat újraindítását követelve meg, hitelesítő adatainak újra megadását is beleértve. Az újraindítás szükségességéről tájékoztatni kell az aláírót.

2.2.5. Követelmények az aláírót hitelesítő összetevőre (SAC)

A tudáson alapuló aláírót hitelesítő adatokra vonatkozó követelmények[8]

Bizt_köv38: Az aláírás-létrehozó alkalmazásnak eszközt kell biztosítania a felhasználó számára ahhoz, hogy az megadhassa az aláírót hitelesítő adatot ezen keresztül a biztonságos aláírás-létrehozó eszköz számára.

Bizt_köv39: Az aláírás-létrehozó alkalmazásnak meg kell őriznie az aláírót hitelesítő adatok bizalmasságát, és biztonságosan törölnie kell azokat, amint azokra nincs már szükség.

Bizt_köv40: Ha az aláírni szándékozó egy korlátot meghaladó számban helytelen hitelesítő adatot ad meg, akkor az újrapróbálkozást le kell tiltani, egyúttal hibajelzést kell adni az aláíró részére, ha az aláíró hitelesítési módszert már nem blokkolta korábban a biztonságos aláírás-létrehozó eszköz. Az aláírni szándékozó részére egy megfelelő üzenetet kell küldeni.

Bizt_köv41: Ha az aláírni szándékozó ismételten helytelen hitelesítő adatot ad meg (pl. három egymást követő alkalommal), a biztonságos aláírás-létrehozó eszköznek félbe kell szakítania az aláíró hitelesítését, és erről informálnia kell az aláírás-létrehozó alkalmazást, amelynek az aláíró részére egy megfelelő üzenetet kell küldeni.

Bizt_köv42: Egy megbízható útvonalat kell biztosítani a PIN/jelszó továbbítására a PIN pad (vagy billentyűzet) és a biztonságos aláírás-létrehozó eszköz között az aláírás-létrehozó alkalmazáson keresztül.

Bizt_köv43: Biztosítani kell egy olyan funkciót, amellyel a tudáson alapuló hitelesítő adatok lecserélhetőek (hacsak ez nincs tiltva egy aláírás-létrehozó alkalmazás típus esetében az alkalmazás szolgáltatójának biztonsági szabályzatában).

[8] A 46.-47. követelmények csak a biometrikus (tehát a tudáson alapulóktól eltérő) aláírót hitelesítő adatokat használó alkalmazásokra vonatkoznak. A MultiSigno Developer nem támogatja ezt, ezért ezek a követelmények nem vonatoznak rá.

Bizt_köv44: A megadott PIN kódot (vagy jelszót) nem szabad kijelezni, bár egy számjegy vagy karakter begépelését vissza kell jelezni egy megfelelő jellel (pl. egy csillag karakterrel), amely nem fedi fel magát a PIN-t (vagy a jelszót).

Bizt_köv45: Az aláírás-létrehozó alkalmazásnak meg kell követelnie az új PIN kód (jelszó) kétszeri megadását, és ellenőriznie kell ezek azonosságát, mielőtt az új PIN kódot (jelszót) továbbítaná a biztonságos aláírás-létrehozó eszköznek.

2.2.6 Követelmények az aláírandó adat formattáló összetevőre (DTBSF)

Bizt_köv48: Az aláírás-létrehozó alkalmazásnak azt a helyes aláírandó adat formátumot kell előállítania, amelyet az aláíró által kiválasztott aláírási szabályzat határoz meg.

2.2.7 Követelmények az adat lenyomat készítő összetevőre (DHC)

Bizt_köv49: Az aláírás-létrehozó alkalmazásnak biztosítania kell egy „elfogadott” (szabványos és elterjedt) lenyomatoló algoritmus használatát lenyomatolásra.

Bizt_köv50: Az aláírás-létrehozó alkalmazásnak biztosítania kell egy „elfogadott” (szabványos és elterjedt) elektronikus aláírás input formátum (feltöltési módszer) használatát.

Bizt_köv51: Az aláírás-létrehozó alkalmazásnak biztosítania kell a helyes aláírandó adat reprezentáns előállítását az elektronikus aláíráshoz.

2.2.8 Követelmények a biztonságos aláírás-létrehozó eszköz és az aláírás-létrehozó alkalmazás közötti kommunikáció összetevőre (SSC)

Bizt_köv52: Az aláírás-létrehozó rendszernek támogatnia kell a fizikai interfész minden fontos részletét egy meghatározott tartományon belül, vagy egy meghatározott jellegzetességgel, az általa támogatott biztonságos aláírás-létrehozó eszköz típusok megfelelő működésének biztosítása érdekében.

Bizt_köv54[9]: Az SSC összetevőnek biztosítania kell a biztonságos aláírás-létrehozó eszköz helyes funkcionalitásának kiválasztását, amennyiben a biztonságos aláírás-létrehozó eszköz ilyen kiválasztást tesz szükségessé (mert pl. több alkalmazást támogat párhuzamosan). Biztosítania kell az aláíró választása szerinti, az aláírás tulajdonságoknak megfelelő aláírás-létrehozó adat (magánkulcs) használatát, amennyiben több magánkulcs van a biztonságos aláírás-létrehozó eszközön tárolva.

Bizt_köv55: Az SSC-t védeni kell a jogosulatlan módosításokkal szemben.

[9] Az 53. követelmény csak a vezeték nélküli összeköttetést használó alkalmazásokra vonatkozik. A MultiSigno Developer nem támogatja ezt, ezért ez a követelmény nem vonatozik rá.

2.2.9 Követelmények az SSCD/SCA hitelesítő összetevőre (SSA)

Bizt_köv56: Az SSA-nak támogatnia kell az aláírás-létrehozó alkalmazás és a biztonságos aláírás-létrehozó eszköz között az entitások hitelesítését, hogy megbízható jelzést adhasson az aláírónak egy sikeres hitelesítésről, és védenie kell az ezt követő kommunikációt egy biztonságos üzenetközvetítéssel.

2.2.10 Követelmények az Input/Output interfészre (I/O)

Bizt_köv57: Intézkedéseket kell tenni annak biztosítására, hogy vírusok ne ronthassák el az SCA összetevőket, és hogy az esetlegesen vírussal fertőzött SCA összetevők megfelelően helyre legyenek állítva.

Bizt_köv58: Az SCA-nak védenie kell funkcionális összetevőinek sértetlenségét, és meg kell akadályozni, hogy behatolók elrontsák ezeket.

Bizt_köv59: Intézkedéseket kell tenni az aláírás-létrehozó alkalmazásban arra, hogy importált aláírás-létrehozó alkalmazás komponenseket csak egy biztonságos letöltés felhasználásával lehessen telepíteni.

Bizt_köv60: Az aláírás-rendszer összes, az aláírás-létrehozás vagy aláírás-ellenőrzés folyamattal kölcsönhatásba lépő összetevőjét egy biztonságos területen[10] kell megvalósítani.

[10] A **biztonságos területet** egy olyan területet, melyen belül speciális ellenintézkedésekkel védekeznek a feldolgozott és tárolt adatok, illetve a folyamatok sikeres manipulálása ellen. Technikai módszerekkel (tehát nem adminisztratív úton) az alábbi három különböző módon lehet megvalósítani:

- Egy **szoftver modul**ban, melyben a biztonsági ellenintézkedések szoftverben vannak megvalósítva. Az így elérhető biztonság a működtető környezet biztonságától függ.
- Egy **módosítást-jelző modul**ban, ahol a biztonsági ellenintézkedéseket olyan módon valósítják meg, hogy a manipuláció ugyan nem akadályozható meg, de a felhasználó észlelheti azt. Ez azt jelenti, hogy a felhasználó védve van a biztonságos területen manipulált komponensek véletlen használatától.
- Egy **módosításnak ellenálló modul**ban, ahol a biztonsági ellenintézkedéseket olyan módon valósítják meg, hogy a manipuláció reális erőfeszítésekkel nem megvalósítható.

A MultiSigno Developer biztonságos területe szoftver modul. Az ezzel elérhető biztonság korlátozottságáról, illetve az ebből fakadó, a működtető környezetre vonatkozó feltételeket a 4. fejezet 3. és 5. feltételei tartalmazzák.

3. A MultiSigno Developer követelményeknek való megfelelése

Az alábbiakban összefoglaljuk az „Értékelési jelentés a Developer (1.2) aláíró alkalmazás fejlesztő készletről” című dokumentum eredményeit.

3.1 A MultiSigno Developer megfelelése a funkcionális követelményeknek

Funkcionális követelmény	Teljesülés (**I**gen/**N**em)	**Magyarázat**
F_SCA_1	**I**	A MultiSigno Developer az ellenőrzés mindhárom alábbi funkcióját támogatja: 1. aláírás ellenőrzése (az aláírt adat lenyomata megegyezik-e a digitális aláírás dekódolása után visszakapott értékkel) 2. aláíró tanúsítványának az ellenőrzése (jól van-e aláírva, érvényes-e, stb.) 3. a tanúsítvány érvényességének ellenőrzése (nincs-e visszavonva) (letárolt aktuális visszavonási listában való kereséssel, illetve ennek hiányában az aktuális CRL letöltésével a hitelesítés-szolgáltatóval kiépített szabványos – HTTP, HTTPS, LDAP - protokollal)
F_SDP_1	**N**	A MultiSigno Developer nem támogatja.
F_SDP_2	**N**	A MultiSigno Developer nem támogatja.
F_SDP_3	**N**	A MultiSigno Developer nem támogatja.
F_SDP_4	**N**	A MultiSigno Developer nem támogatja.
F_SAV_1	**I**	Az aláíró tanúsítványának és az időbélyegző megjelenítését támogatja /a dokumentum tartalom-formátumát, aláírási szabályzatot, kötelezettségvállalás típust nem kezel a felület/.
F_SAV_2:	**I**	A MultiSigno Developer támogatja az aláíráshoz csatolt tanúsítvány átvizsgálását.
F_SIC_1	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
F_SIC_2	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
F_SIC_3	**I**	A PIN kód bekérést a CSP támogatja.
F_DTBSF_1	**I**	A MultiSigno Developer támogatja. /A szabványos XML (RFC 3075) aláírási struktúra megvalósítása a MultiSigno Developer nagy erénye/.
F_DTBSF_2	**I**	A MultiSigno Developer támogatja, illetve jól felhívja a CSP megfelelő függvényét.
F_DHC_1	**I**	A MultiSigno Developer támogatja. A felhívás sorrendjét (közvetlenül az aláírás-létrehozás folyamat kiváltása után) a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
F_DHC_2	**I**	A MultiSigno Developer támogatja. A felhívás sorrendjét (közvetlenül a lenyomatolás végrehajtása után) a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
F_SSC_1	**I**	Támogatja. A felhívások sorrendjét a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.

F_SSC_3	**I**	A MultiSigno Developer támogatja a szabványos PC/SC kompatibilis olvasókat, melyek rendelkeznek ilyen fizikai interfésszel.
F_SSC_4	**I**	A MultiSigno Developer elvileg támogat több-alkalmazásos platformot is, de ilyenkor az aktivizálandó magánkulcshoz tartozó tanúsítványt be kell hozzá előzetesen állítani.
F_SSC_5	**I**	A Crypto API támogatja.
F_SSC_6	**I**	A Crypto API támogatja.
F_SSC_7	**N**	A MultiSigno Developer nem támogatja.
F_SSC_8	**I**	A MultiSigno Developer támogatja. A felhívás sorrendjét a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
F_SSC_9	**I**	A MultiSigno Developer támogatja.
F_SDC_1	**I**	A MultiSigno Developer támogatja.
F_SDOC_1	**I**	A MultiSigno Developer támogatja.
F_SLC_1	**I**	A MultiSigno Developer naplóz minden aláírás létrehozást (az ellenőrzéseket nem).
F_SCPC_1	**I**	A MultiSigno Developer képes CRL letöltésére. (Az OCSP-t nem támogatja.)
F_I/O-1	**I**	A MultiSigno Developer támogatja.
F_I/O-2	**I**	A MultiSigno Developer támogatja.
F_ISV-1	**N**	A MultiSigno Developer csak a rendszeróra szerinti aktuális aláírás ellenőrzést támogatja. /Nincs előzetes és utólagos aláírás ellenőrzés funkció./
F_ISV-2	**N**	A MultiSigno Developer csak a rendszeróra szerinti aktuális aláírás ellenőrzést támogatja. /Nincs előzetes és utólagos aláírás ellenőrzés funkció./
F_USV-1	**N**	A MultiSigno Developer csak a rendszeróra szerinti aktuális aláírás ellenőrzést támogatja. /Nincs előzetes és utólagos aláírás ellenőrzés funkció./
F_general_1	**N**	A fejlesztő készlet segítségével fejlesztett alkalmazásnak kell biztosítania, hogy a MultiSigno Developer által támogatott funkciók segítségével megvalósítható ellenőrzés világos feldolgozási szabályt kövessen.
F_protocol	**I**	A MultiSigno Developer szabványos protokollt használ a megbízható szolgáltatóval (szolgáltatókkal) történő alábbi kommunikáció során: ▪ tanúsítvány visszavonási állapot megszerzésekor (HTTP, HTTPS, LDAP); ▪ időbélyeg kérelem és válasz esetén (RFC 3161);
F_format	**I**	A MultiSigno Developer az alábbi szabványos formátumokat kezeli: ▪ aláírási formátum: /XML, RFC 3075/, ▪ tanúsítvány formátum /X509.v3, RFC 2459/, ▪ visszavonási lista /CRL, RFC 2459/.
F_principles	**N**	A felhasználói felületeket (s így annak logikus felépítését, könnyen kezelhetőségét, felhasználó barátságát) a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.

3.2 A MultiSigno Developer megfelelése a biztonsági követelményeknek

Az alábbiakban áttekintjük, hogy a minősített aláírásra vonatkozó biztonsági követelmények teljesítéséből melyeket támogat a MultiSigno Developer, vagy egy általa felhívott egyéb komponens (a CSP vagy a PC/SC szabványos kártyaolvasó).
A nem teljesített biztonsági követelményeket a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell megoldania, vagy biztonsági maradványkockázatként kell elfogadni.

Biztonsági követelmény	Teljesülés (**I**gen/**N**em/**-**)	**Magyarázat**
Bizt_köv1	**N**	A MultiSigno Developer nem támogatja.
Bizt_köv2	**I**	A MultiSigno Developer támogatja. A bizalmasság megőrzését a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak is támogatnia kell.
Bizt_köv3	**-**	Csak nyilvános[11] aláíró alkalmazásokra vonatkozik. /A MultiSigno Developer nem ilyen./
Bizt_köv4	**-**	Csak nyilvános aláíró alkalmazásokra vonatkozik.
Bizt_köv5	**-**	Csak nyilvános aláíró alkalmazásokra vonatkozik.
Bizt_köv6	**-**	Csak nyilvános aláíró alkalmazásokra vonatkozik.
Bizt_köv7	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv8	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv9	**-**	Csak osztott architektúrájú aláíró alkalmazásokra vonatkozik. /A MultiSigno Developer nem ilyen./
Bizt_köv10	**-**	Csak osztott architektúrájú aláíró alkalmazásokra vonatkozik.
Bizt_köv11	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv12	**I**	A MultiSigno Developer támogatja.
Bizt_köv13	**I**	A MultiSigno Developer támogatja.
Bizt_köv14	**N**	Az aláírási szabályzatok azonosítóval (OID-vel) történő ellátása még nem megoldott Magyarországon.
Bizt_köv15	**-**	A MultiSigno Developer által támogatott (implicit) aláírási szabályzat nem határoz meg egynél több kötelezettségvállalás típust.
Bizt_köv16	**I**	A MultiSigno Developer támogatja.
Bizt_köv17	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, vagy ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv18	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell majd kielégítenie.
Bizt_köv19	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazás használati útmutatójának kell majd kielégítenie.
Bizt_köv20	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazás használati útmutatójának kell majd kielégítenie.
Bizt_köv21	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell majd kielégítenie.
Bizt_köv22	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett

[11] A nyilvános aláíró alkalmazás egy szolgáltató ellenőrzése alatt áll (pl. postahivatal, Internet-kávézó, stb.), ahová a felhasználó betérhet, hogy aláírást létrehozzon, illetve ellenőrizzen.

		alkalmazásnak kell majd kielégítenie.
Bizt_köv23	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv24	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv25	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv26	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv27	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv28	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv29	**I**	A MultiSigno Developer támogatja az alábbi két aláírási tulajdonságot: ▪ az aláíró tanúsítványát (mindig), ▪ az időbélyegzőt (opcionálisan). Aláírás előtt ezek megtekinthetőségét (ellenőrizhetőségét) a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell megvalósítania.
Bizt_köv30	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv31	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv32	**I**	A MultiSigno Developer által támogatott aláírási tulajdonságok (tanúsítvány, időbélyegző) mindegyike szabványos formátumú, és nem tartalmazhat rejtett szöveget, makrót, vagy aktív kódot (erről a hitelesítés-szolgáltatónak kell gondoskodnia). Következésképpen a MultiSigno Developer által elvégzett ellenőrzések (érvényes és szabályos-e a tanúsítvány, illetve az időbélyeg) indirekt módon biztosítják azt is, hogy ilyen rejtett információ ne kerülhessen az aláírás tulajdonságok közé.
Bizt_köv33	**I**	A MultiSigno Developer által támogatott aláírás tulajdonságok (tanúsítvány, időbélyegző) egyike sem tartalmazhat beágyazott rejtett szöveget, makrót, vagy aktív kódot. Így ezek ellenőrzése formálisan a követelményt is kielégíti.
Bizt_köv34	**I**	A MultiSigno Developer támogatja az aláírás tulajdonságok (tanúsítvány, időbélyegző) megtekintését, átvizsgálását.
Bizt_köv35	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell majd megvalósítania.
Bizt_köv36	**N**	Sem a CSP, sem a MultiSigno Developer nem támogatja. A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell majd megvalósítania
Bizt_köv37	**N**	Sem a CSP, sem a MultiSigno Developer nem támogatja. A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell majd megvalósítania
Bizt_köv38	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak kell biztosítania.
Bizt_köv39	**I**	A CSP-k biztosítják.
Bizt_köv40	**I**	A CSP-k megvalósítják.
Bizt_köv41	**I**	A CSP-k megvalósítják.
Bizt_köv42	**N**	A MultiSigno Developer nem támogatja.
Bizt_köv43	**I**	A CSP-k támogatják.
Bizt_köv44	**I**	A CSP-k támogatják.
Bizt_köv45	**I**	A CSP-k támogatják.

Bizt_köv46	**-**	Csak a biometrikus aláírót hitelesítő adatokat használó aláíró alkalmazásokra vonatkozik. /A MultiSigno Developer nem ilyen./
Bizt_köv47	**-**	Csak a biometrikus aláírót hitelesítő adatokat használó aláíró alkalmazásokra vonatkozik.
Bizt_köv48	**I**	Bár a MultiSigno Developer tetszőleges formátumú dokumentumra lehetővé teszi az aláírást, a ténylegesen aláírt adat formátum mégis egyértelmű: a szabványos XML struktúra.
Bizt_köv49	**I**	A MultiSigno Developer az XML struktúrára **MD5** lenyomatot számoltat (a CSP-vel), saját maga pedig **SHA-1** lenyomatot készít valamennyi csomagba szervezett dokumentumról.
Bizt_köv50	**I**	A CSP-k támogatják.
Bizt_köv51	**I**	A CSP-k támogatják az MD5 lenyomat képzését.
Bizt_köv52	**I**	A PC/SC és az ISO 7816 1-4 automatikusan biztosítja.
Bizt_köv53	**-**	Csak a vezeték nélküli összeköttetést használó alkalmazásokra vonatkozik. A MultiSigno Developer nem támogat ilyen alkalmazásokat.
Bizt_köv54	**I**	A CSP-k támogatják.
Bizt_köv55	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv56	**N**	A MultiSigno Developer nem támogatja.
Bizt_köv57	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv58	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv59	**N**	A MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazásnak, illetve ezen alkalmazás működtetési környezetének kell majd kielégítenie.
Bizt_köv60	**N**	A biztonságos terület biztosítását csak a MultiSigno Developer-t és az ebből készült aláíró alkalmazást futtató számítógép informatika biztonsági alrendszere oldhatja meg. Ehhez kiegészítő támogatást (védelmi mechanizmusokat) nyújthat a MultiSigno Developer fejlesztő készlet segítségével fejlesztett alkalmazás is.

4. A Tanúsítási jelentés eredménye, érvényességi feltételei

Az alábbiakban összefoglaljuk azokat a feltételeket, melyek betartása hozzájárul a MultiSigno Developer segítségével fejlesztett aláíró alkalmazások fokozott biztonságához. Ezek a feltételek a fejlesztésekkel szembeni általános minőségbiztonsági (tervezési, tesztelési, dokumentálási stb.) követelményeken túlmutatóan az aláírás-specifikus elemek ellenőrzött védelmi szintjét szándékoznak garantálni. A feltételek között vannak kötelezően betartandó, a tanúsítvány érvényességére kiható feltételek, és vannak olyan feltételek, amelyek az aláírások biztonságára jelentős befolyással bírnak, ezért fokozott (nem minősített) aláíráshoz, ahol a lehetőség adott, ezen feltételek betartása erősen ajánlott.

4.1 Kötelezően betartandó feltételek

1. A MultiSigno Developer-t felhasználóihoz (akik aláíró alkalmazásokat fejlesztenek felhasználásukkal) CD-n szállítják. Használatba vétel előtt kötelező másolatot készíteni róla, hogy az eredetit mesterpéldányként lehessen felhasználni a későbbiekben végrehajtandó sértetlenség ellenőrzések során.

2. A MultiSigno Developer-rel fejlesztett aláíró alkalmazások elkészülésekor a fejlesztők felelőssége a felhasznált és a mesterpéldányként elmentett MultiSigno Developer függvényei sértetlenségének ellenőrzése /valóban a tanúsított fejlesztő készlet elemeit építették-e be/.

4.2 Ajánlottan betartandó feltételek

A MultiSigno Developer ”aláíró alkalmazást fejlesztő munkaállomásokon” alapvetően elszigetelt működtetési környezetben használandó, de kiegészítő feltételek garantálása esetén védett működtetési környezetben is lehet fejleszteni vele (sőt a teljes funkcionalitás végső tesztelése csak ilyen körülmények között valósítható meg).
Elszigetelt működtetési környezet (kisebb fejlesztéseknél ez a tipikus eset) esetén a fejlesztői készletet (és egyúttal a fejlesztés alatt álló aláíró alkalmazást) az védi, hogy nincs (sohasem) kapcsolódás kommunikációs hálózatokra (Internet, Intranet), és a működtetési környezetben olyan védelmi intézkedéseket valósítanak meg, melyek kivédik a jogosulatlan manuális hozzáférésen és adathordozóról történő adatbevitelen alapuló támadásokat is/. A MultiSigno Developer-rel alapvetően ilyen környezetben ajánlott fejleszteni, tesztelni. Ugyanakkor az ebből fejlesztett, teljes funkcionalitást biztosító végterméket nem lehet teljeskörűen ebben a működtetési környezetben tesztelni, mivel a MultiSigno Developer bázisán kifejlesztett aláíró alkalmazás:

- aláírás létrehozásánál nem képes tesztelni időbélyegzés opcióját (hisz nem kerülhet hálózati kapcsolatba egyetlen időbélyeg-szolgáltatóval sem),
- aláírás ellenőrzéséhez nem képes tesztelni azt a funkcióját, hogy amennyiben az adott munkaállomás nem rendelkezik érvényes visszavonási listával (amit elszigetelt környezetben adminisztratív úton, egy más munkaállomáson letöltött CRL adathordozóról történő betöltésével lehet biztosítani).

Védett működtetési környezet (kisebb fejlesztések esetén nem ez a tipikus eset) esetén a fejlesztői készletet (és egyúttal a fejlesztés alatt álló aláíró alkalmazást) a működtetési környezet nagy bizonyossággal megvédi a kommunikációs hálózatok (Internet, Intranet) irányából érkező, valamint a jogosulatlan manuális hozzáférésen és az adathordozóról történő adatbevitelen alapuló támadásoktól.

4.2.1 Általános működtetési feltételek

3. Eljárásrendi/szervezeti védelmi intézkedésekkel kell támogatni az aláíró alkalmazás fejlesztést megvalósító számítógép(ek)re irányuló olyan támadások kivédését, melyek manuális hozzáférésen, illetve adathordozóról történő adatbevitelen alapulnak. Garantálni kell, hogy a fejlesztés technikai környezete, valamint a fejlesztett programok és az ehhez felhasznált fejlesztő készlet funkcióit ne lehessen manipulálni, melyet különösen vírus és Trójai faló bejuttatása okozhat. Minden újonnan telepített szoftvernek manipulációtól mentesnek kell lennie.
   /A fenti intézkedések döntően ahhoz kellenek, hogy a fejlesztendő aláíró alkalmazás és az ennek bázisát képező fejlesztő készlet ne manipulálódjon./

4. A fejlesztő környezetben konfiguráció menedzselési eljárások kidolgozásával és betartásával kell garantálni a fejlesztett termék sértetlenségét azzal, hogy fegyelmet és ellenőrzést követelnek meg a fejlesztendő termék és más ezzel összefüggő információ pontosításában és módosításában.
   /A konfiguráció menedzselése akadályozza a fejlesztés alatt álló alkalmazások egyes verzióinak jogosulatlan módosítását, bővítését vagy törlését, illetve hozzájárul a mégis bekövetkező felhatalmazás nélküli változtatások észleléséhez (a verziónként elkülönítetten is letárolt példányok időszakos összehasonlításával)./

4.2.2 A védett működtetési környezetben történő felhasználás járulékos feltételei

5. Amennyiben a fejlesztő környezetnek hálózati kapcsolatai is vannak a 2. feltételben elvárt konfiguráció menedzselési eljárásokon kívül rendszeres időnként ellenőrizni kell a fejlesztő készlet és a fejlesztett aláíró alkalmazás verziók sértetlenségét (az elkülönítetten is letárolt (mester) példányok időszakos összehasonlításával).

4.2.3 Az elszigetelt működtetési környezetben történő felhasználás feltételei

Az elszigetelés számos fenyegetést eleve kizár (hálózati támadások), a fenyegetések más részét pedig az általános működtetési feltételek lefedik. (Nincs járulékos feltétel).

5. A követelményeknek való megfelelést ellenőrző független vizsgálat garancia szintje

A jelen Tanúsítási jelentéshez figyelembe vett, a fejlesztőktől független ellenőrző vizsgálat garancia szintje az ISO 14508 /Common Criteria/ **EAL 2**-es szintjéhez hasonló volt. /Az EAL2 a fejlesztőktől függetlenül garantált biztonság mérsékelt szintjét biztosítja, mely elegendő a fokozott biztonságú elektronikus aláíráshoz használható aláírási termékekre. /

Ez a fejlesztőktől függetlenül garantált biztonság az alábbiakkal jellemezhető:

- a követelményrendszerrel való egybevetést a fejlesztőktől független ellenőrző vizsgálat alapvetően a program tesztelésével végezte,
- a tesztelési eredményeket a fejlesztők által készített leírások, illetve az ezekben megnevezett szabványok egészítették ki,
- a vizsgálat a forrásszöveg ellenőrzésére nem terjedt ki,
- a vizsgálat feltételezte, hogy a Smart Card és CSP fejlesztői az általuk felvállalt funkcionalitást helyesen oldották meg.

Az ellenőrző vizsgálat a MultiSigno Developer biztonsági viselkedésének megértése érdekében elemezte a biztonsági funkciókat, ehhez felhasználta az alábbi fejlesztői dokumentációkat:

- a MultiSigno Developer konfigurációs tételei,
- a szállítás eljárásai,
- a fejlesztő készlet telepítésének, elindításának eljárásai,
- felső-szintű tervek,
- felhasználásra (fejlesztésre) vonatkozó útmutatók,

Ezeken kívül az ellenőrzés:

- funkcionális teszteket végzett,
- áttekintette a fejlesztők által végzett tesztelést, elemezte ennek teljességét,
- a fejlesztőktől független minta tesztelést végzett,
- értékelte a biztonsági funkciók erősségét, a termék sebezhetőségét.

6. A MultiSigno Developer biztonsági funkciók értékelt erőssége

Még ha az értékelés tárgyának (jelen esetben a MultiSigno Developer aláíró alkalmazás fejlesztő készletnek) a biztonsági funkcióit nem is lehet megkerülni, kiiktatni vagy tönkretenni, akkor is lehet lehetőség kijátszani azokat, ha a mögöttes biztonsági mechanizmusok sebezhetőek. E funkciók biztonsági viselkedése minősíthető a mechanizmusok biztonsági viselkedésének mennyiségi vagy statisztikai alapú elemzési eredményeinek felhasználásával és az ilyen mechanizmusok legyőzésére vonatkozó erőfeszítések segítségével.

A biztonsági funkciókat a biztonsági mechanizmusok valósítják meg. Például egy jelszókezelő mechanizmus az azonosítás és hitelesítés biztonsági funkciók megvalósításában használható fel.

A biztonsági funkciók erősségének elemzése a biztonsági mechanizmusok szintjén zajlott (az MD5 és az SHA-1 lenyomatoló függvények).
Eredménye a vizsgált biztonsági funkcióknak az azonosított veszélyek elleni fellépés képességére vonatkozó információt tartalmazza.

A biztonsági funkciók erőssége: **középszintű**

7. A tanúsításhoz figyelembe vett dokumentumok

7.1 Termékmegfelelőségi követelményeket tartalmazó dokumentumok

Az elektronikus aláírásról szóló 2001. évi XXXV. törvény

CEN/ISSS/E-Sign; Area G1, 14170 munkacsoport egyezmény: Security Requirements for Signature Creation Systems

CEN/ISSS/E-Sign; Area G2, 14171 munkacsoport egyezmény: Procedures for Electronic Signature Verification

CEN/ISSS/E-Sign; Area V, 14172-4 munkacsoport egyezmény: Signature Creation Application and Procedures for Electronic Signature Verification

ETSI TS 101 733 Electronic Signature Formats

ETSI TS 101 903 XML Advanced Electronic Signatures (XadES)

ISO/IEC 14508-3 Information Technology. Security technique. Evaluation criteria for IT security. Part 3: Security assurance requirements

7.2 A tanúsítási jelentéshez figyelembe vett egyéb dokumentumok

7.2.1 A tanúsításhoz figyelembe vett egyéb fejlesztői dokumentumok

Kérelem a tanúsítás elvégzésére

Kérdőív a tanúsítás kérelmezéséhez

Pack.dll interface /felső-szintű fejlesztői leírás a DLL könyvtárról/

Pack.h /a függvények részletesebb leírása, paraméterezése és a paraméterek leírása/

Pack.dll /maga a függvény könyvtár/

7.2.2 A tanúsításhoz figyelembe vett, fejlesztőktől független dokumentumok

Értékelési jelentés a MultiSigno Developer (1.2) aláíró alkalmazás fejlesztő készletről /Bizalmas dokumentum, készítette a HunGuard Kft./

Aláíró alkalmazások funkcionális modellezése
/Nyilvános dokumentum, készítette a HunGuard Kft./

CAPI	Microsoft Cryptographic Application Programming Interface
PKCS #1	RSA Cryptography Standard /RFC 2313/
RSA	Rivest-Shamir-Adleman (public key cryptosystem) /ANSI X9.31/
SHA-1	Secure Hash Algorithm /FIPS PUB 180-1/

8. Rövidítések

API	Application Programming Interface
CEN	European Comittee for Standardization
CSP	Cryptograpic Service Provider /kriptográfiai szolgáltató/
CRL	Certification Revocation List /tanúsítvány visszavonási lista/
DHC	Data hashing component /adatlenyomat-készítő összetevő/
DTBS	Data to be Signed /aláírandó adat/
DTBSF	DTBS formatter /aláírandó adat formattáló/
DTBSR	Data to be Signed Representation /aláírandó adat reprezentáns/
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol sECURE
I/O	Input/Output
ISO	International Organization for Standardization
ISSS	Information Society Standardization System
ISV	Initial Signature Verification /kezdeti aláírás ellenőrzés/
LDAP	Lightweight Directory Access Protocol
MD5	Rivest: “The MD5 Message Digest Algorithm”
PC/SC	Personal Computer Smart Card
PIN	Personal Identification Number
PKCS	Public Key Cryptographic Standards
PKCS #1	RSA Cryptography Standard
RFC	Request for Comment
RSA	Rivest-Shamir-Adleman (public key cryptosystem)
SAC	(Signer's authentication component) aláírót hitelesítő összetevő
SAV	Signature attribute viewer /aláírási tulajdonság megjelenítő/
SCA	Signature creation application /aláírás-létrehozó alkalmazás/
SDC	Signer's document composer /aláírói dokumentum szerkesztő/
SDOC	Signed data object composer /aláírt adat objektum szerkesztő/
SDP	Signer's document presenter /aláírói dokumentumot megjelenítő/
SHA-1	Secure Hash Algorithm
SHI	SSCD holder indicator /Az SSCD tulajdonos jelző/
SIC	Signer's interaction component /aláíróval kölcsönható összetevő/
SLC	Signature logging component /aláírás-naplózási összetevő/
SP	Signature Policy /aláírási szabályzat/
SSA	SSCD/SCA Communicator authenticator /az SSCD/SCA közötti kommunikációt hitelesítő összetevő /
SSC	SSCD/SCA Communicator /az SSCD és SCA közötti kommunikáció összetevője/
SSCD	Secure signature creation device /biztonságos aláírás-létrehozó eszköz, BALE/
USV	(Usual Signature Verification /utólagos aláírás ellenőrzés/
XML	eXtensible Markup Language