University of Debrecen
Faculty of Health
Nyíregyháza

ACTA MEDICINAE ET SOCIOLOGICA 1990

ACTA
MedSoc
Volume 1.
2010

Többcsatornás kriptográfiai protokollok formális vizsgálata - MANA III -

Takács Péter

Debreceni Egyetem, Egészségügyi Kar,
Egészségügyi Informatika Tanszék
e-mail: vtp@de-efk.hu

Abstract. Formal verification of multi-channel cryptographic protocols - MANA III. The topic of this paper is to examine cryptographic protocols based on formal methods. As the original sources of the CSN-logic do not reflect the expected exactitude of the mathamatical logic, we present a remodelled CSN-logic. A multi-sorted modal logic and its notation is given. We modify the axiom system in a lesser degree. We apply the extended logic to verify validity of MANA III protocol.

Keywords: cryptographic protocols, formal verification, CSN-logic, MANA III protocol

DOI: 10.19055/ams.2010.1/1/8

Lektor: Dr. Vályi Sándor Ph.D, főiskolai docens, Nyíregyházi Főiskola

1. Bevezetés

Kommunikáló partnerek sok esetben több információs csatornát is képesek használni. A dolgozat célja többcsatornás rendszerek matematikai logikai eszközökkel történő vizsgálatának bemutatása. Munkánk során a Coffey-Saidha-Newe-féle (a továbbiakban: CSN) logikai rendszert bővítjük a többcsatornás kommunikáció leírásának és kezelésének lehetőségeivel. Az új rendszert már ismert és gyakorlatban is alkalmazott többcsatornás kriptográfiai protokollok biztonsági elemzésére használjuk fel. Jelen írásban a MANA III protokoll elemzését mutatjuk be.

A CSN-logikai rendszer két közleményben jelent meg. Elször 1997-ben T. Coffey és P. Saidha tette közzé az alaprendszert. [1] Ebben a cikkben a nyilvános kulcsú titkosítást használó protokollok számára kidolgozott elmélet látott napvilágot. Ezt követően 2003-ban jelent meg a Newe és T. Coffey írása, amely kibővítette a modellt a titkos kulcsú titkosítást alkalmazó rendszerek körére. Mindezek a [11] és [12] dolgozatokban érhetők el.

Mivel az eredeti források nem tükrözik teljes mértékben a matematikai logika elvárt precizitását, itt saját átdolgozott rendszerünket mutatjuk be. Munkánk során tovább pontosítjuk az alkalmazott logikai nyelvet, a jelölésrendszert, a következtetési szabályokat. Kisebb módosításokat eszközlünk az axiómák körében is. A teljes rendszer a közlemény mellékletében kapott helyet - a továbbiakban az ott lévő jelöléseket és számozási rendszert alkalmazzuk. A szerzőkre utalva továbbra is a CSN-logika nevet használjuk.

A bővített CSN-logika alkalmazásaként vizsgált MANA III protokollról [4]-ben olvashatunk.

További kapcsolódási pontként kell megemlítenünk Wong F-L. és Stajano F. 2005-ben megjelent cikkét, amely a többcsatornás kriptográfiai protokollok körében a MANA protokollcsaládot vizsgálja. [16] Írásukban a MANA I, II és III protokollok kerülnek elemzésre valószínűségszámítási eszközök felhasználásával. A cikk végén a szerzők jelzik, hogy aktuális feladat egy olyan rendszer kidolgozása, amely lehetővé tenné a többcsatornás protokollok logikai alapú vizsgálatát. Munkánk során ezen az úton indultunk el, és alkalmaztuk eredményeinket a már említett MANA protokollcsalád egyes elemeire. Hasznos összefoglalását találja az érdeklődő még a többcsatornás kriptográfiai protokollok körének Wong és Stajano 2007-ben megjelent cikkében, amely a tudományterület aktuális helyzetét foglalja össze. [17]

2. A kriptográfiai- és a többcsatornás protokollok

A kriptográfiai protokollok olyan kommunikációs protokollok, amelyek célja a partnerek biztonságos, védett és ellenőrzött kommunikációjának biztosítása. A protokollok elmélete a támadások detektálásától kezdve a protokollok különböző eszközökkel történő vizsgálatára vállalkozik. Napjainkban erősödött meg az az irányzat, amely külön vizsgálja a protokollokban szereplő csatornák szerepét és jelentőségét. Ez a vonulat kapcsolódik a „mindenütt jelen lévő, láthatatlan számítás-

technika" (*ubicomp - ubiquitous computing*, vagy *pervasive computing*) fogalmához. Az újonnan megjelenő számítástechnikai és informatikai eszközök már az egymáshoz kapcsolódás képességével, a vezetékes, vagy vezeték nélküli kommunikációval vannak felruházva. Kialakult a PAN (*Personal Area Network* - Személyes hálózat) fogalomköre és technológiája, amelynek kommunikációs protokolljai eltérnek az Internet és a mobilhálózatok eddig felépített protokolljaitól. [6][15]

Részletesen és behatóan tanulmányozva a tradicionális (titkos kulcsú) kriptográfiai rendszereket, könnyen megtaláljuk többcsatornás protokollok alapjait. Például a protokollokban szereplő titkos kulcsú kommunikáció kulcsát egy védett csatornán kell eljutatni a parnerhez, ami jelentheti egy futár alkalmazását, vagy személyes találkozáskor lebonyolított kulcscserét. Hasonló megoldások találhatók különböző elektronikus pénzkezelési megoldásokban, eBank rendszerekben is.

Elmondhatjuk tehát, hogy több csatorna használata nem jelent lényegesen új eszközt a kritptográfiában, viszont az ilyen megoldások megalapozott tudományos vizsgálata még csak napjainkban zajlik. Ahhoz, hogy részletesebben ismertessük elért eredményeinket, be kell mutatnunk a protokollok formális vizsgálatát.

3. A kriptográfiai protokollok formális vizsgálata

A kriptográfiai kutatások napjainkra két fő irányzatot alakítottak ki. Az egyik a számításelméleti megközelítés (főleg valószínűségelméleti és komplexitáselméleti eszközök alkalmazása), a másik pedig a formális megközelítés (főleg modális logikai eszközök). [5] A formális módszerek kriptográfiai protokollok tervezésének és ellenőrzésének különböző szakaszaiban használhatók. A leginkább kutatott az ellenőrzés (verifikáció) szakasza. A bővített CSN logika a modális logika alkalmazását jelenti a verifikáció során. A modális logika alkalmazásának sémája a kriptográfiai protokollok ellenőrzése során a következő: 1. lépésben a vizsgált protokollt kell formalizálni. 2. lépés a kezdeti feltételek meghatározása. 3. lépésként a protokoll céljait kell megfogalmazni. 4. lépés a logikai posztulátumok alkalmazását jelenti. 5. lépés a negyedik lépés esedményeinek és a protokoll céljainak (harmadik lépés) összevetését jelenti.

4. A CSN-logika és bővítése

A CSN-logika egy többtípusú, multi-modális elsőrendű levezetési rendszer. A többtípusú logika akkor használatos, ha a vizsgált objektumok nem alkotnak homogén halmazt. Többtípusú logikák lefordíthatók egytípusú, hagyományos elsőrendű logikává. A modális operátorok alkalmazásával az állítások eredeti jelentése módosítható. Egy modális logika eredetileg egy klasszikus logikai rendszer bővítése a *„szükségszerű"* és a *„lehetséges"* kifejezésekkel.[1] A CSN-rendszer többek között a

[1] Jelekben: a *szükségszerűség* operátora: $\Box$ a *lehetségesség* operátora: $\Diamond$. $\Box$ és $\Diamond$ egymásból kifejezhetők: $\Box\alpha \leftrightarrow \neg\Diamond\neg\alpha$ és $\Diamond\alpha \leftrightarrow \neg\Box\neg\alpha$ (α formula). A $\Box$ operátornak többféle értelmezése

K-val (*knowledge* - tud, ismer) és a B-vel (*belief* - hisz, bízik) jelölt operátorokat vezeti be, amivel egy multi-modális logikát hoz létre.

A levezetési rendszer egy klasszikus elsőrendű levezetési rendszerből indul ki, bővítve azt többek között az új operátorokra vonatkozó levezetési szabályokkal (például: $R2(a)$, $R2(b)$) és axiómákkal (például: $A1(a)$, $A1(b)$, $A2(a)$).

Egy másik osztályozás szerint a CSN-rendszer episztemikus-doxatikus[2] rendszer. [9] E szemléletmód szerint a rendszer kidolgozói abból indultak ki, hogy két tendencia figyelhető meg a kriptográfiai protokollok logikai vizsgálatában. Az egyik a bizalom/megbízhatóság fejlődésének vizsgálata a protokoll lépései során (*logics of belief*), a másik a protokollok működésére alapozott tudás (protokoll szereplőinek ismerete) elemzése (*logics of knowledge*). A CSN-logika célja a kétféle megközelítés ötvözése, lehetővé téve ezáltal a kriptográfiai protokollok szélesebb körű és mélyebb vizsgálatát.

A logikai modellünk egyik kiinduló célja a partnerek közötti védett kommunikáció leírása (formalizálás). Ennek során legelőször a modell típusait kell megadnunk. [10]

A legegyszerűbb kommunikációs kapcsolat során a küldő fél üzenetet küld a fogadó fél felé. Ez alapján külön típusnak kell tekintenünk a szereplő partnereket (EGYED típus) és az átküldött üzenetet (ÜZENET típus). A védett kommunikáció a küldött üzenet titkos voltát jelenti, amelyet kriptográfiai algoritmusok alkalmazásával érünk el. Az algoritmusok titkosító és visszafejtő kulcsokat használnak a működés során (KULCS típus). Le kell írnunk a vizsgálandó protokollok időbeli viselkedését, ami külön típust jelent (IDŐ típus). Szintén le kell írnunk a partnerek által használt csatornákat és azok tulajdonságait is (CSATORNA típus).

A típusok megadása után a *Melléklet*ben megadjuk az alkalmazható nyelvi elemeket, a következtetési szabályokat, az axiómarendszert, és azokat a megjegyzéseket, amelyek a rendszer pontosabb értelmezését szolgálják.

5. A bővített CSN-logika alkalmazása - A MANA protokollcsalád

A MANuális Authentikáció (MANA protokollcsalád) főleg vezeték-nélküli (wireless) eszközök hitelesítésére lett kialakítva. Ez a hitelesítés egy nem biztonságos vezeték-nélküli csatornát egészít ki manuális adatátvitellel (mint második csatorna), így biztosítva a megfelelő szintű védelmet. Négy protokoll tartozik jelenleg a MANA protokollcsaládba. Ezek között az alapvető különbség a protokoll során felhasznált eszközök tulajdonságaiban van (alkalmazható az eszközön billentyűzet, LED, kijelző képernyő, beviteli gomb, nyomógomb, stb.). A nyilvános csatorna általában gyors és szélessávú; míg a nem nyilvános csatorna általában a manuális

lehetséges. Ezek közül néhány: $\Box\alpha$ igaz, ha ... α *szükségszerű*en igaz; *tud*om, hogy α igaz; *ismer*etes, hogy α igaz; *hisz*em, hogy α igaz; α igaz *most, és a jövőben mindig* igaz lesz; stb. [3]

[2] angolul: *epistemic-doxastic*

csatorna kis kapacitással - a felhasználók olvassák és/vagy írják a csatornajeleket. [17]

A továbbiakban csak a MANA III protokollal foglalkozunk. A MANA I és II protokollok vizsgálatának eredményei a [13] és [14] közleményekben jelentek meg.

5.1. MANA III

A protokollban két eszköz (A és B) és az eszközöket kezelő felhasználó (U - user) vesz részt. Mindkét eszköz rendelkezik egy-egy input egységgel (billentyűzet), és egy-egy output egységgel (világító dióda - LED). A protokoll célja az, hogy mindkét eszköz bizonyítottan rendelkezzen ugyanazzal a kezdeti paraméterrel (n_A), amelyet a későbbi védett kommunikáció során használhat fel.

A MANA III protokoll lépései

1. Az A eszköz generál egy n_A számot. Ezt és azonosítóját (A) átküldi a B eszköznek a ch_1 csatornán. B egy n_B számot és Σ azonosítót kap a ch_1 csatornán (a ch_1 csatorna nem védett, feltételezzük, hogy egy támadó képes megváltoztatni az üzeneteket (fennállhat $n_A \neq n_B$ és $\Sigma \neq A$)).
2. A B eszköz a ch_1 csatornán elküldi az B azonosítóját. Az A eszköz Ψ számot (azonosítót) fogad a ch_1 csatornán (hasonlóan fennállhat, hogy $\Psi \neq B$).
3. Az U user egy r_U véletlenszámot generál, és ezt a védett ch_2 és ch_3 csatornákon eljuttatja az A és a B félhez.
4. A egy k_A véletlenszámot (kulcs) generál és kiszámítja az $m_1 = h(\{A, n_A, r_U\}, k_A)$ értéket.
5. Az A eszköz elküldi m_1-t B-nek a ch_1 csatornán. B m_{11}-t kap üzenetként (fennállhat, hogy $m_1 \neq m_{11}$).
6. B egy k_B véletlenszámot (kulcs) generál és kiszámítja az $m_2 = h(\{B, n_B, r_U\}, k_B)$ értéket.
7. B elküldi m_2-t A-nak a ch_1 csatornán. A m_{22}-t kap üzenetként (fennállhat, hogy $m_2 \neq m_{22}$).
8. Miután A fogadja m_{22}-t B-től (és nem előbb), A átküldi a k_A kulcsot B-nek a ch_1 csatornán (B k_Σ-et kap, fennállhat, hogy $k_\Sigma \neq k_A$).
9. Amikor B megkapja az m_{11} értéket A-tól (és nem előbb), B átküldi a k_B számot A-nak a ch_1 csatornán (A k_Ψ-t kap, fennállhat, hogy $k_\Psi \neq k_B$).
10. A újraszámítja m_2-t, m_{222}-t kap eredményül. Amennyiben ez megegyezik a B-től kapott m_{22} értékkel, úgy A erről egy jelet küld (világító LED) U-nak a ch_2 csatornán. $m_{222} = h(\{\Psi, n_A, r_U\}, k_\Psi)$ a kapott üzenetek alapján.
11. B újraszámítja m_1-t, m_{111}-t kap eredményül.

Amennyiben ez megegyezik az A-tól kapott m_{11} értékkel, úgy B erről egy jelet küld (világító LED) U-nak a ch_3 csatornán. $m_{111} = h(\{\Sigma, n_B, r_U\}, k_\Sigma)$ a kapott üzenetek alapján.

12. Amennyiben mindkét eszköz sikeres számítást jelez (és csak ekkor), U visszajelzi ezt mindkét eszköznek. □

Kezdeti feltételek rögzítése

I301. $CH(ch_1, pub)$; $CH(ch_2, sec)$; $CH(ch_3, sec)$.

I302. $ENT_{ch_2} = \{A, U\}$; $ENT_{ch_3} = \{B, U\}$.

I303. $K_{A,t_{17}}(m_{22} = m_{222}) \rightarrow S(ch_2, A, t_{17}, \text{"1"})$.

I304. $K_{A,t_{19}}(m_{11} = m_{111}) \rightarrow S(ch_3, B, t_{19}, \text{"1"})$.

I305. $K_{A,t_{17}}(m_{22} \neq m_{222}) \rightarrow S(ch_2, A, t_{17}, \text{"0"})$.

I306. $K_{A,t_{19}}(m_{11} \neq m_{111}) \rightarrow S(ch_3, B, t_{19}, \text{"0"})$.

I307. $K_{U,t_{21}}(R(ch_2, U, t_{18}, \text{"1"}) \wedge R(ch_3, U, t_{20}, \text{"1"}))$
$\rightarrow S(ch_2, U, t_{21}, \text{"1"}) \wedge S(ch_3, U, t_{23}, \text{"1"})$

I308. $K_{U,t_{21}}(R(ch_2, U, t_{18}, \text{"0"}) \vee R(ch_3, U, t_{20}, \text{"0"}))$
$\rightarrow S(ch_2, U, t_{21}, \text{"0"}) \wedge S(ch_3, U, t_{23}, \text{"0"})$

I309. $R(ch_2, A, t_{22}, \text{"1"}) \rightarrow K_{A,t_{22}}(n_A = n_B)$.

I310. $R(ch_3, B, t_{24}, \text{"1"}) \rightarrow K_{B,t_{24}}(n_A = n_B)$.

I311. $R(ch_2, A, t_{22}, \text{"0"}) \rightarrow K_{A,t_{22}}(n_A \neq n_B)$.

I312. $R(ch_3, B, t_{24}, \text{"0"}) \rightarrow K_{A,t_{24}}(n_A \neq n_B)$.

A MANA III protokoll formális alakja

1. $S(ch_1, A, t_1, \{n_A, A\})$; $R(ch_1, B, t_2, \{n_B, \Sigma\})$
2. $S(ch_1, B, t_3, B)$; $R(ch_1, A, t_4, \Psi)$
3. $S(ch_2, U, t_5, r_U)$; $R(ch_2, A, t_6, r_U)$
4. $S(ch_3, U, t_7, r_U)$; $R(ch_3, B, t_8, r_U)$
5. $S(ch_1, A, t_9, m_1)$; $R(ch_1, B, t_{10}, m_{11})$
6. $S(ch_1, B, t_{11}, m_2)$; $R(ch_1, A, t_{12}, m_{22})$
7. $S(ch_1, A, t_{13}, k_A)$; $R(ch_1, B, t_{14}, k_\Sigma)$

8. $S(ch_1, B, t_{15}, k_B)$; $R(ch_1, A, t_{16}, k_\Psi)$
9. $S(ch_2, A, t_{17}, x)$; $R(ch_2, U, t_{18}, x)$
10. $S(ch_3, B, t_{19}, y)$; $R(ch_3, U, t_{20}, y)$
11. $S(ch_2, U, t_{21}, z)$; $R(ch_2, A, t_{22}, z)$
12. $S(ch_3, U, t_{23}, z)$; $R(ch_3, B, t_{24}, z)$

A protokoll céljai - tételek és bizonyítások

Tétel 5.1. *Tegyük fel, hogy a n_A és n_B paraméterek nem egyenlők a protokoll végrehajtása során (egy illetéktelen felhasználó módosítja a kommunikációt). Ekkor a MANA III protokoll lefutásának a végén az A és B partnerek (eszközök) mindketten tudják azt, hogy $n_A \neq n_B$. Formálisan:*

$$n_A \neq n_B \rightarrow K_{A,t_{22}}(n_A \neq n_B) \wedge K_{B,t_{24}}(n_A \neq n_B).$$

Bizonyítás Az első lépés és az $A5(a)$ és $A6(a)$ axióma alapján

$$L_{A,t_1}\{n_A, A\}\ , \tag{1}$$
$$L_{B,t_2}\{n_B, \Sigma\}\ . \tag{2}$$

(M3) alapján

$$L_{A,t_1} n_A\ , \tag{3}$$
$$L_{A,t_1} A\ , \tag{4}$$
$$L_{B,t_2} n_B\ , \tag{5}$$
$$L_{B,t_2} \Sigma\ . \tag{6}$$

A második lépésben $A5(a)$ és $A6(a)$ alapján

$$L_{B,t_3} B\ , \tag{7}$$
$$L_{A,t_4} \Psi\ . \tag{8}$$

A harmadik és negyedik lépésben $A5(a)$ és $A6(a)$ alapján

$$L_{U,t_5} r_U\ , \tag{9}$$
$$L_{A,t_6} r_U\ , \tag{10}$$
$$L_{B,t_8} r_U\ . \tag{11}$$

Az ötödik lépésben $A5(a)$ és $A6(a)$ alapján

$$L_{A,t_9} m_1\ , \tag{12}$$
$$L_{B,t_{10}} m_{11}\ . \tag{13}$$

$(m_1 = h(\{A, n_A, r_U\}, k_A))$
A hatodik lépésben $A5(a)$ és $A6(a)$ alapján

$$L_{B,t_{11}} m_2 \ , \tag{14}$$
$$L_{A,t_{12}} m_{22} \ . \tag{15}$$

$(m_2 = h(\{B, n_B, r_U\}, k_B))$
A hetedik lépésben $A5(a)$ és $A6(a)$ alapján

$$L_{A,t_{13}} k_A \ , \tag{16}$$
$$L_{B,t_{14}} k_\Sigma \ . \tag{17}$$

A nyolcadik lépésben $A5(a)$ és $A6(a)$ alapján

$$L_{B,t_{15}} k_B \ , \tag{18}$$
$$L_{A,t_{16}} k_\Psi \ . \tag{19}$$

A protokoll szerint t_{16} után A újraszámítja m_2-t és m_{222}-t kap eredményül. m_{222} formailag, az A rendelkezésére álló, fogadott üzenetek alapján: $m_{222} = h(\{\Psi, n_A, r_U\}, k_\Psi)$. Eredetileg $m_2 = h(\{B, n_B, r_U\}, k_B)$. Ha $m_2 = m_{22}$, $\Psi = B$ és $k_\Psi = k_B$ ($A17(a)$ alapján), még akkor sem lehet $m_{22} = m_{222}$, mivel a tétel feltételei szerint $n_A \neq n_B$. Így $K_{A,t_{17}}(m_{22} \neq m_{222})$, amivel $I305.$ alapján $S(ch_2, A, t_{17}, \text{"0"})$. A védett csatornák miatt (M5) U az $R(ch_2, U, t_{18}, \text{"0"})$ üzenetet kapja, ami alapján az $I308.$ kezdeti feltétel érvényes. Ez azt jelenti, hogy $S(ch_2, U, t_{21}, \text{"0"}) \wedge S(ch_3, U, t_{23}, \text{"0"})$. Védett csatornák miatt (M5), $I311.$ és $I312.$ érvényes: $R(ch_2, A, t_{22}, \text{"0"})$ és $R(ch_3, B, t_{24}, \text{"0"})$.
Tehát $K_{A,t_{22}}(n_A \neq n_B)$ és $K_{B,t_{24}}(n_A \neq n_B)$, ami a keresett állítás egyik fele. Hasonló gondolatmenettel belátható, hogy a B egyed által elvégzett $m_{11} = m_{111}$ összehasonlítás is a $K_{B,t_{24}}(n_A \neq n_b)$-t adja. A két rész a keresett állításhoz vezet. □

Tétel 5.2. *Tegyük fel, hogy n_A és n_B paraméterek egyenlők a protokoll végrehajtása során. Ekkor a MANA III protokoll nem garantálja, hogy lefutásának a végén az A és B partnerek mindketten tudják azt, hogy $n_A = n_B$.*
Formálisan: $n_A = n_B \rightarrow K_{A,t_{22}}(n_A \neq n_B) \wedge K_{B,t_{24}}(n_A \neq n_B)$ nem levezethető.

Bizonyítás Az előző tétel bizonyításához hasonlóan az A és B felhasználók eljutnak a fogadott m_{22} és számított m_{222} (m_{11} és m_{111}) összehasonlításához. Ez abban az esetben, ha a támadó aktívan nem avatkozik a protokollba, a keresett állításhoz vezet. Amennyiben a protokollt támadás éri, az $n_A = n_B$ feltételt megtartva, de az A vagy k_A értékek közül bármelyiket változtatva, a protokoll az $m_{22} = m_{222}$ vagy $m_{11} = m_{111}$ állítások bármelyikének elvetéséhez vezet. Így $K_{A,t_{22}}(n_A \neq n_B)$ és $K_{B,t_{24}}(n_A \neq n_B)$. Ez azt jelenti, hogy hiába lett helyesen átküldve az n_A üzenet, mégsem fogadják el a felhasználók ezt. □

Ebben az esetben elmondható, hogy a felek hiába rendelkeznek a helyes n_A értékkel, annak közös elfogadhatóságát a protokoll nem tudja garantálni. Mivel az A, B egyednevek és a k_A, k_B kulcsok nyilvános csatornán kerülnek továbbításra, egy támadó fél módosítani tudja értéküket, megzavarva így a protokollt. A fenti két tétel elemzése megmutatja, hogy a javítás a MANA II protokolléhoz hasonló módon nem oldható meg (lásd [13][14]). Kulcshasználat nélküli $H(m)$ egyirányú függvények alkalmazása az összetett m üzenet miatt nem jelent megoldást. n_A, illetve n_B részeket tartalmaznia kell az átküldött üzeneteknek, de r_U elhagyása már megszemélyesítő támadást tesz lehetővé. A protokoll ezen hiányosságainak javítása új protokoll kidolgozását igényli.

5.2. Összefoglalás

Összefoglalva elmondhatjuk, hogy CSN-logika pontosítása és bővítése alkalmas a többcsatornás protokollok vizsgálatára. Nyilvánvaló, hogy nem ez lehet az egyetlen lehetséges megoldás a problémkörben. A bemutatott példa evidensnek és egyszerűnek tűnhet, viszont megnyithat olyan vizsgálati utakat, amelyek komolyabb protokoll-hibákra is rámutathatnak.

Többcsatornás protokollokat a vizsgált három protokollon (MANA család) kívül más körben is alkalmaznak. Szaporodik azoknak az alkalmazásoknak a köre, amelyek összekapcsolják az Internetet és a személyes kommunikációs eszközöket. A mobiltelefonra küldött SMS, amely a banki szolgáltatások elérését teszi védettebbé; a mobiltelefonokkal történő hang- és képátvitel (2D-s BAR-kódok, biometriai azonosítás, stb.) összekapcsolva más kommunikációs csatornákkal (Internet, fax, stb.), mind azt erősítik, hogy a többcsatornás protokolloknak jogosultsága van a kommunikációs fejlődésben.

Hivatkozások

[1] T. Coffey and P. Saidha. Logic for verifying public-key cryptographic protocols. *IEEE Proceedings Computers and Digital Techniques*, 144(1):28–32, 1997.

[2] D. Dolev and A. Yao. On the security of public key protocols. *IEEE Transactions on Information Theory*, IT-29(2):198–208, March 1983.

[3] M. Ferenczi. *Matematikai logika*. Műszaki Könyvkiadó, 2002.

[4] C. Gehrmann, C. J. Mitchell, and K. Nyberg. Manual authentication for wireless devices. *Cryptobytes*, 7(1):29–37, 2004.

[5] Á. Gergely. Biztonságos útvonalválasztás ad hoc hálózatokban. Master's thesis, Budapesti Műszaki és Gazdaságtudományi Egyetem, Villamosmérnöki és Informatikai Kar, Híradástechnikai Tanszék, 2005.

[6] S. Goeman. Specification of prototypes - D11, IST - 2000 - 25350 - SHAMAN, Public Report. http://www.isrc.rhul.ac.uk/shaman/docs, March 2003. D11v2.pdf.

[7] J. Hintikka. *Knowledge and Belief. An Introduction to the Logic of the Two Notions*. Cornell University Press, 1962.

[8] J. Hintikka. *Részletek Jaakko Hintikka Tudás és Hit (Bevezetés a két fogalom logikájába) című művéből. A hiedelmek természete, szerveződése és szerepe a mindennapi tudatban című munkaértekezlet segédanyaga 7.* Fordítási részek. Fordította: Berkes Ildikó, 1975.

[9] S. Kramer. Logical concepts in cryptography. http://citeseer.ist.psu.edu/759062.html.

[10] T. Mihálydeák. *Az informatika logikai alapjai*. University of Debrecen, 2007. Egyetemi jegyzet.

[11] T. Newe and T. Coffey. Formal verification logic for hybrid security protocols. *International Journal of Computer Systems Science & Engineeing*, pages 17–25, 2003.

[12] P. Takács. The additional examination of the Kudo-Mathuria time-release protocol. *Journal of Universal Computer Science*, 12(9):1373–1384, 2006.

[13] P. Takács. The extension of CSN-logic for multi-channel protocols. In *Proceedings of the 7th ICAI Conference, Eger*, pages 147–154, 2007.

[14] P. Takács and S. Vályi. An extension of protocol verification modal logic to multi-channel protocols. *Tatra Mountains Mathematical Publications*, 41:153–166, 2008.

[15] P. Windirsch. Security for mobile systems beyond 3G - Presentations and posters of the IST - 2000 - 25350 - SHAMAN Worshop, 2002. http://www.isrc.rhul.ac.uk/shaman/docs, 2002.

[16] F-L. Wong and F. Stajano. Multi-channel protocols. In *Proceeding of Security Protocols, 13th International Workshop, Cambridge, UK*, volume 4631 of *Lecture Notes in Computer Science*. Springer-Verlag, April,20-22 2005.

[17] F. L. Wong and F. Stajano. Multichannel security protocols. *IEEE Pervasive computing*, VI.:31–39, October-December 2007.

Melléklet
A bővített CSN-logika[3]

A CSN-logikai rendszerhez tartozó nyelv a következő

$$L^{(CSN)} = \langle Sort, LC, Var, Con, Term, Form \rangle$$

rendezett hatos, ahol
Sort $= \{U, E, K, T, C\}$
A típusok (fajták) halmaza: U üzenet-típus; E egyed-típus; K kulcs-típus; T idő-típus; C csatorna-típus.
LC $= \{\neg, \rightarrow, \leftrightarrow, \wedge, \vee, \equiv, =, \forall, \exists, (,)\}$
A nyelv logikai konstansainak halmaza, amelyeket az elsőrendű logikában megszokott módon használunk.
Var $= Var_U \cup Var_E \cup Var_K \cup Var_T \cup Var_C$
A nyelv változóinak megszámlálhatóan végtelen halmaza. Minden változónak meghatározott típusa van. Var_δ a δ típusú változók halmazát jelöli.
Con $= Con_U \cup Con_E \cup Con_K \cup Con_T \cup Con_C$
A nyelv nemlogikai konstansainak legfeljebb megszámlálhatóan végtelen halmaza. Minden nemlogikai konstansnak meghatározott típusa van. Con_δ a δ típusú nemlogikai konstansok halmazát jelöli, egyes típusok esetén üres is lehet a halmaz. $F(0)_\delta$ a névkonstansok, $F(n)_\delta$ az n argumentumú függvényjelek halmaza. Az argumentumban szereplő szám a paraméterek számát jelöli. Függvényjelek esetén szokás megadni egy véges $\langle \delta_1, \delta_2, \ldots, \delta_n, \delta \rangle$ indexsorozatot is, amely rendre megadja a konkrét függvényjel n darab argumentumának típusát ($\delta_i \in Sort$) és a függvényjel típusát ($\delta \in Sort$). $P(0)$ az állításkonstansok, $P(n)$ az n argumentumú predikátumkonstansok halmaza. Itt szintén szokás megadni az egyes predikátumkonstansok argumentumában szereplő $\langle \delta_1, \delta_2, \ldots, \delta_n \rangle$ ($\delta_i \in Sort$) indexsorozatot.
Term $= Term_U \cup Term_E \cup Term_K \cup Term_T \cup Term_C$
A nyelv terminusainak, termjeinek halmaza, típusonként induktív definícióval megadva. $Term_\delta$ a δ típusú Termek halmazát jelöli, egyes típusok esetén üres is lehet a halmaz.
Az induktív definíció általános formája minden δ típus esetén:
(a) $\quad Var_\delta \cup F(0)_\delta \subseteq Term_\delta$.

[3] Az eredeti CSN-logika két cikkben jelent meg: T. Coffey, P. Saidha. Logic for verifying public-key cryptographic protocols. IEEE Proceedings Computers and Digital Techniques, 144(1):28-32, 1997. és T. Newe, T. Coffey. Formal verification logic for hybrid security protocols. International Journal of Computer Systems Science and Engineering, 17-25, 2003. Ezt a rendszert egészítette ki M. Kudo, A. Mathuria. Az entended logic for analyzing timed-release public-key protocols. In Proceedings Information and Communication Security, Second International Conference, ICICS'99, Sydney, 1999. A többcsatornás jelölésrendszert első alakját Takács P. és Vályi S. dolgozta ki. An extension of protocol verification modal logic to multi-channel-protocols. Tatra Mountains Mathematical Publications - TATRACRYPT 2007. Vol.41. 2008. A melléklet ezen rendszer lényegesen átdolgozott változatát tartalmazza.

(*b*) Ha $f \in F(n)_\delta$, $(n = 1, 2, \ldots)$ és $s_1, s_2, \ldots, s_n \in Term$, akkor $f(s_1, s_2, \ldots, s_n) \in Term_\delta$.

Form A nyelv formuláinak halmaza, induktív definícióval megadva:

(*a*) $P(0) \subseteq Form$.

(*b*) Ha $s_1, s_2 \in Term_\delta$, akkor $(s_1 = s_2) \in Form$.

(*c*) Ha $P \in P(n)$, $(n = 1, 2, \ldots)$, és $s_1, s_2, \ldots, s_n \in Term$, akkor $P(s_1, s_2, \ldots, s_n) \in Form$.

(*d*) Ha $A \in Form$, akkor $\neg A \in Form$.

(*e*) Ha $A, B \in Form$, akkor $(A \to B), (A \wedge B), (A \vee B), (A \equiv B) \in Form$.

(*f*) Ha $x \in Var$, $A \in Form$, akkor $\forall xA, \exists xA \in Form$.

Kiegészítő részletek és az egyes típusokhoz tartozó sajátosságok a következők:

• i, j általános indexváltozókat jelölnek, a természetes számokon futnak.

• x, y, z általános változók, több típusra vonatkozó esetekben használjuk, megadva, hogy milyen típusú változókat képviselnek.

• A leírás során az egyértelműség érdekében sok esetben alkalmaznunk zárójeleket. Ezeket más matematikai tárgyak keretében megszokott módon kell olvasni, értelmezni.

• A következtetési szabályokban és az axiómákban a szabad változók univerzálisan kötöttek.

U - üzenet típus

Jellemzés: a kommunikációban szereplő üzenetek leírása; MSG az összes üzenetek halmaza, amely megszámlálhatóan végtelen halmaz.

• Var_U:

m, n, r, m_1, m_2, ..., m_i, m_j, ... általános üzenet-változók.

n_A, n_B, ..., n_Σ, ... speciális üzenet-változók - egyedi üzenetrészek jelölésére (általában az üzenet friss voltát hivatottak biztosítani, a visszajátszásos támadások kivédése érdekében).

r_A, r_B, ..., r_Σ, ... speciális üzenet-változók - általában véletlen számok jelölésére. Az indexekben szereplő latin nagybetűk az üzenetet generáló egyedet jelöli.

• Con_U:

$F(0)_U$:

(a) A kommunikáció során átküldött bitsorozatok (1, 2, ... bájt - ASCI vagy Unicode kódolással értelmezett bájtok) által reprezentált jelek, karakterek üzenet-konstansok.

(b) A kommunikáció során használt rögzített jelentésű karaktersorozatok (parancsok, utasítások): "enc", "dec", "0", "1", ... üzenet-konstansok. Ezek mindig dupla idézőjelek között szerepelnek, jelentésüket minden esetben megadjuk.

$F(n)_U$:

$\{m_1, m_2\}$ Konkatenáció: kapcsos zárójelbe írt, vesszővel elválasztott, egymás utáni üzenetekből újabb üzenetek származtathatók. $\{\} \in F(2)$; $\langle U, U, U \rangle$.

$E(m,k)$ Titkosító függvény (*encryption function*) - szimmetrikus kulcsú tikosítás esete. $E(m, ks_{(\Sigma,\Psi)})$ jelentése: az m nyílt szöveg (üzenet) titkosítása a Σ és Ψ által használt osztott titkos $ks_{(\Sigma,\Psi)} \in KS_{(\Sigma,\Psi)}$ kulcs segítségével. A függvény kimenete üzenet fajta. $E \in F(2)$; $\langle U, K, U\rangle$.

$D(m,k)$ Visszafejtő fügvény (*decryption function*) - szimmetrikus kulcsú tikosítás esete. $D(x, ks_{(\Sigma,\Psi)})$ jelentése: az m titkosított üzenet visszafejtése a Σ és Ψ által használt osztott titkos $ks_{(\Sigma,\Psi)} \in KS_{(\Sigma,\Psi)}$ kulcs segítségével. A függvény kimenete üzenet fajta. $D \in F(2)$; $\langle U, K, U\rangle$.

$e(m,k)$ Titkosító függvény (*encryption function*) - nyilvános kulcsú titkosítás során $e(m, k_\Sigma)$ jelentése: m üzenet titkosítása a k_Σ kulcs segítségével. $e(m, k_\Sigma^{-1})$ jelentése: az m üzenet digitális aláírása. A függvény kimenete üzenet fajta. $e \in F(2)$; $\langle U, K, U\rangle$.

$d(m,k)$ Visszafejtő fügvény (*decryption function*) - nyilvános kulcsú titkosítás során $d(m, k_\Sigma^{-1})$ jelentése: az m üzenet visszafejtése a k_Σ^{-1} kulcs segítségével. $d(m, k_\Sigma)$ jelentése: digitális aláírás ellenőrzése, ha m digitálisan aláírt üzenet. A függvény kimenete üzenet fajta. $d \in F(2)$; $\langle U, K, U\rangle$.

$h(m,k)$ Kulcsolt üzenetkivonat (hash) fügvény. $h(m,k)$ jelöli az m üzenet k kulcs segítségével előállított üzenetkivonat értékét. A függvény kimenete üzenet fajta. $h(m,k) \in F(2)$; $\langle U, K, U\rangle$.

$H(m)$ Üzenetkivonat (hash) függvény - MD sorozat, SHA sorozat, HAVAL, RIPEM, stb. A függvény kimenete üzenet fajta. $H(m) \in F(1)$; $\langle U, U\rangle$.

Megjegyzések:

1. Az irodalmi források $ss_{(\Sigma,\Psi)}$ -vel jelölik a Σ és Ψ egyedek között megosztott (friss) titkot (*shared secret*), ami üzenet, vagy kulcs fajta lehet. $SS_{(\Sigma,\Psi)}$ jelöli Σ és Ψ egyedek között megosztott titkok halmazát.
2. Az EGYED, KULCS, IDŐ és CSATORNA fajtájú változóknál értelmezünk két-két olyan függvényjelet ($E2U(\Sigma), U2E(m)$; $K2U(k), U2K(m)$; $T2U(t)$, $U2T(m)$; $C2U(ch), U2C(m)$), amely az egyedek, kulcsok,időpontok és csatornanevek üzenetbe ágyazását (mint karaktersorozatok) és az onnan való kiemelését teszik lehetővé (típus-konverzió).

E - egyed típus

Jellemzés: a kommunikáció szereplőinek leírása. ENT az összes lehetséges egyedek halmaza. ENT számossága véges.

• Var_E: Σ, Ψ, Γ, Λ, ...

• Con_E:
$F(0)_E$:

A, B, C, D, U, ... Az egyedek jelölése során alkalmazzuk a dolgozat 2.1.1. fejezetének *Jelölések* részében leírtakat. Ezek szerint *A Aliz*, *B Botond*, stb. szereplőket jelöli. Az egyedek elnevezése lehetőség szerint követi a hagyományos szerepköröket: kommunikáló felek A, B; passzív támadó E; abszolút megbízható fél T, stb. - az említett fejezetben leírtak szerint.

$F(n)_E$:

$E2U(\Sigma)$	Az egyed típusú változó átalakítása üzenet típusú változóvá. Ez a függvényjel lehetővé teszi a protokollok üzenetrészeiben egyedek szerepeltetését és továbbítását. A függvényjel input paramétere: egyed fajta, a függvényjel-output: üzenet fajta. A függvényjel helyettesítésekor adódó eredményt egyszeres idézőjelek közé írjuk: $E2U(\Sigma)='\Sigma'$. $E2U \in F(1)$; $\langle E, U\rangle$.
$U2E(m)$	A megfelelő üzenet fajta változó átalakítása egyed fajta változóvá. Ez a függvényjel lehetővé teszi egy protokollban az üzenetként átküldött egyednevek értelmezését. A függvényjel input paramétere: üzenet fajta, függvényjel-output: egyed fajta. $U2E('\Sigma')=\Sigma$. $U2E \in F(1)$; $\langle U, E\rangle$.

Megjegyzések:

1. A különböző csatornák esetén értelmezzük a csatornát használni képes egyedek halmazát. Nyilvános csatorna esetén ez az ENT halmaz, jelölésben: $ENT_{ch_i} = ENT$. Védett csatorna esetén pedig vesszővel elválasztva felsoroljuk azokat egyedeket, akik a csatornához hozzáférnek: $ENT_{ch_i} = \{\ldots\}$. Természetesen $ENT_{ch_i} \subseteq ENT$.

K - kulcs típus

Jellemzés: titkosító és visszafejtő kulcsok leírása. KEY jelöli az összes lehetséges kulcsok halmazát; KS jelöli az egyedek közötti szimmetrikus kulcsú kommunikációt lehetővé tevő kulcsok halmazát. Mindkét halmaz megszámlálhatóan végtelen számosságú.

• Var_K: k általános kulcsváltozó.
• Con_K:
$F(n)_K$:

$ks_{(\Sigma,\Psi)}$	(Ki)osztott titkos kulcs - (*shared secret key*). $ks_{(\Sigma,\Psi)}$ jelenti Σ és Ψ egyedek számára kiosztott, általuk ismert közös titkos kulcsot - szimmetrikus kulcsú tikosítás esete. $ks_{(\Sigma,\Psi)} \in F(2)$; $\langle E, E, K\rangle$.
k_Σ	Nyilvános kulcs (*public key*) - nyilvános kulcsú titkosítás során k_Σ a Σ egyed nyilvános kulcsa. $k_\Sigma \in F(1)$; $\langle E, K\rangle$.
k_Σ^{-1}	Titkos kulcs - (*secret key*) - nyilvános kulcsú titkosítás során k_Σ^{-1} a Σ egyed titkos kulcsa. $k_\Sigma^{-1} \in F(1)$; $\langle E, K\rangle$.
$k_{t_i}, k_{t_i}^{-1}$	Idő-kulcs - az indexben megadott t_i időponthoz kötött nyilvános és titkos kulcs. $k_{t_i} \in F(1)$; $\langle T, K\rangle$. $k_{t_i}^{-1} \in F(1)$; $\langle T, K\rangle$.

$K2U(k)$ A kulcs fajta változó átalakítása üzenet fajta változóvá. Ez a függvényjel lehetővé teszi a protokollok üzenetrészeiben kulcsok szerepeltetését és továbbítását. A függvényjel input paramétere: kulcs fajta, a függvényjel-output: üzenet fajta.

A függvényjel helyettesítésekor adódó eredményt egyszeres idézőjelek közé írjuk: $K2U(k_\Sigma) =' k_\Sigma{}'$. $K2U \in F(1)$; $\langle K, U\rangle$.

$U2K(m)$ A megfelelő üzenet fajta változó átalakítása kulcs fajta változóvá. Ez a függvényjel lehetővé teszi egy protokollban az üzenetként átküldött kulcsok kulcsként való értelmezését. A függvényjel input paramétere: üzenet fajta, függvényjel-output: kulcs fajta. $U2K('k_\Sigma{}') = k_\Sigma$. $U2K \in F(1)$; $\langle U, K\rangle$.

Megjegyzések:

1. A szakirodalmi források definiálják a $KS_{(\Sigma,\Psi)}$ halmazt, amely Σ és Ψ egyedek számára megfelelő, jó kulcsok halmazát (*set of good shared keys* - szimmetrikus kulcsú titkosítás) jelenti.

T - idő típus

Jellemzés: a protokollok időbeli leírása; $TIME$ jelöli az összes lehetséges időpontok halmazát. Ez a halmaz véges.

• Var_T: t, t_1, t_2, ... t_i, t_j, ..., t', t'', ...

• Con_T:

$F(0)_T$:

(a) t_0 a vizsgált protokoll kezdetének időpontja.

(b) t_g egy protokollban előforduló kulcsgenerálás időpontja.

(c) τ időfeloldó protokollokban rögzített feloldási időpont.

$F(n)_T$:

$T2U(t)$ Az idő fajta változó átalakítása üzenet fajta változóvá. Ez a függvényjel lehetővé teszi egy protokoll üzenetrészében időadatok szerepeltetését. A függvényjel input paramétere: idő fajta, a függvényjel-output: üzenet fajta.

A függvényjel helyettesítésekor adódó eredményt egyszeres idézőjelek közé írjuk: $T2U(t_i) =' t_i{}'$. $T2U \in F(1)$; $\langle T, U\rangle$.

$U2T(m)$ A megfelelő üzenet fajta változó átalakítása idő fajta változóvá. Ez a függvényjel lehetővé teszi egy protokollban az üzenetként átküldött időadatok értelmezését. A függvényjel input paramétere: üzenet fajta, függvényjel-output: idő fajta. $U2T('t_i{}') = t_i$. $U2T \in F(1)$; $\langle U, T\rangle$.

• $Form$:

(a) Ha $t_1, t_2 \in Term_T$, akkor $(t_1 < t_2) \in Form$.

Megjegyzések:

1. A protokoll-leírás során használt összes lehetséges időpontok $TIME$ halmaza lineárisan rendezett halmazt alkot. Ezt az $A20(a)$ axióma rögzíti.
2. Értelmezettek a $t_i \leqslant t_j$, $t_i > t_j$, $t_i \geqslant t_j$ formulák is.

C - csatorna típus

Jellemzés: a protokollokban alkalmazott kommunikációs csatornák leírása; CH jelöli az összes lehetséges csatornák halmazát. Ez a halmaz véges.

• Var_C: ch, ch_1, ch_2, ... ch_i, ch_j csatorna-változók.

• Con_C:

$F(n)_T$:

$C2U(t)$ Az csatorna fajta változó átalakítása üzenet fajta változóvá. Ez a függvényjel lehetővé teszi egy protokoll üzenetrészében csatorna adatok szerepeltetését. A függvényjel input paramétere: csatorna fajta, a függvényjel-output: üzenet fajta.

A függvényjel helyettesítésekor adódó eredményt egyszeres idézőjelek közé írjuk: $C2U(ch_i) =' ch_i'$. $C2U \in F(1)$; $\langle C, U \rangle$.

$U2C(m)$ A megfelelő üzenet változó átalakítása csatorna fajta változóvá. Ez a függvényjel lehetővé teszi egy protokollban az üzenetként átküldött csatorna adatok értelmezését. A függvényjel input paramétere: üzenet fajta, függvényjel-output: csatorna fajta. $U2C('ch_i') = ch_i$. $U2C \in F(1)$; $\langle U, C \rangle$.

Megjegyzések:

1. Szükséges a csatornák tulajdonságainak leírása a rendszerben. Az egyszerűség kedvéért csak kétféle csatornát (védett és nyilvános) különböztettünk meg. Jelölje $CH(ch_i, sec)$ azt, hogy a ch_i csatorna védett. Hasonlóan jelölje $CH(ch_i, pub)$ azt, hogy a ch_i csatorna nyilvános. A nyilvános csatorna alapfelfogása követi a Dolev-Yao-féle támadási modellt. [2] Amennyiben egy csatorna védett, úgy meg kell határozni azoknak az egyedeknek a körét, akik használhatják azt. Erre az egyed-típusnál említett $ENT_{ch_i} = \{...\}$ jelölést alkalmazzuk.

Operátorok és predikátumjelek:

A CSN-logika operátorai által az állítások eredeti jelentése módosul:

$K_{\Sigma,t}\Phi$ Hintikka-féle tudás, ismeret operátor - (*knowledge operator of Hintikka*). $K_{\Sigma,t}\Phi$ jelentése: Σ egyed ismeri (*knows*) a Φ állítást a t időpontban (részletesebben: [7][8]).

$B_{\Sigma,t}\Phi$ Hit operátor - (*belief operator*). $B_{\Sigma,t}\Phi$ jelentése: Σ egyed elhiszi, elfogadja a t időpontban, hogy a Φ állítás igaz.

$L_{\Sigma,t}x$ Tudás predikátum - (*knowledge predicate*). $L_{\Sigma,t}x$ jelentése: Σ egyed ismeri és elő tudja állítani (*knows and can reproduce*) az x objektumot (üzenet vagy kulcs) a t időpillanatban.

$S(ch_i, \Sigma, t, m)$ Kibocsátó predikátum - (*emission operator*). $S(ch_i, \Sigma, t, m)$ jelentése: Σ egyed az m üzenetet bocsájtja ki, küldi a t időpontban a ch_i csatornán.

$R(ch_i, \Sigma, t, m)$ Fogadó predikátum - (*reception operator*). $R(ch_i, \Sigma, t, m)$ jelentése: Σ egyed az m üzenetet fogadja a t időpontban a ch_i csatornán.

$C(x, y)$	Tartalmazás predikátum - (*'contains' operator*). $C(x, y)$ jelentése: az x üzenet tartalmazza az y üzenetet.
$A(\Sigma, t, \Psi)$	Hitelesítési predikátum (*authentication operator*). $A(\Sigma, t, \Psi)$ jelentése: Σ egyed hitelesíti a Ψ egyedet a t időpontban.
$O_{\Sigma,t}(x, y)$	(*'obtain' predikátum*). $O_{\Sigma,t}(x, y)$ jelentése: az Σ egyed képes kinyerni, megkapni az y objektumot (üzenet vagy kulcs) az x objektumból (üzenet vagy kulcs) a t időpillanatban. Ezt az operátort M. Kudo és A. Mathuria vezette be. A szerzők az operátort σ betűvel jelölik. Az áttekinthetőbb jelölés érdekében mi a O betűt használjuk.

Következtetési szabályok[4]

Legyenek α, β a nyelv tetszőleges formulái, p, q tetszőleges állításai. A logikai rendszer következtetési szabályai a következők:

R1	Az α és az $\alpha \to \beta$ formulák bizonyíthatóságából következik a β formula bizonyíthatósága: $\alpha \wedge (\alpha \to \beta) \Rightarrow \beta$ *(modus ponens).*
R2(a)	Az α formula bizonyíthatóságából következik a $K_{\Sigma,t}\alpha$ formula bizonyíthatósága: $\alpha \Rightarrow K_{\Sigma,t}\alpha$ *(generelisation rule I).*
R2(b)	Az α formula bizonyíthatóságából következik $B_{\Sigma,t}\alpha$ formula bizonyíthatósága: $\alpha \Rightarrow B_{\Sigma,t}\alpha$ *(generalisation rule II).*
R3	Az $(\alpha \wedge \beta)$ formula bizonyíthatóságából következik az α formula bizonyíthatósága: $(\alpha \wedge \beta) \Rightarrow \alpha$ *(simplification).*
R4	Az α és β formulák bizonyíthatóságából következik $\alpha \wedge \beta$ formula bizonyíthatósága: (α) , $(\beta) \Rightarrow (\alpha \wedge \beta)$ *(conjunction).*
R5	Az α formula bizonyíthatóságából következik az $\alpha \vee \beta$ formula bizonyíthatósága: $\alpha \Rightarrow (\alpha \vee \beta)$ *(addition).*
R6	A $\neg\neg\alpha$ formula bizonyíthatóságából következik az α formula bizonyíthatósága: $\neg\neg\alpha \Rightarrow \alpha$ *(double negation).*
K1(a)	A $K_{\Sigma,t}(p \wedge q)$ formula bizonyíthatóságából következik a $K_{\Sigma,t}p$ és $K_{\Sigma,t}q$ formulák bizonyíthatósága: $K_{\Sigma,t}(p \wedge q) \Rightarrow K_{\Sigma,t}p \wedge K_{\Sigma,t}q$.

[4] A következtetési szabályok és az axiómák az új jelölési rendszer felhasználásával vannak megadva. Lényegi változtatásokat nincsnek bevezetve, kiseb kiegészítésekkel bővült a rendszer - az idő- és a hash függvények leírása. Az eredeti források: [1] és [11]

K2(a) A $K_{\Sigma,t}p$ és $K_{\Sigma,t}q$ formulák bizonyíthatóságából következik a $K_{\Sigma,t}(p \wedge q)$ formula bizonyíthatósága:
$K_{\Sigma,t}p \wedge K_{\Sigma,t}q \Rightarrow K_{\Sigma,t}(p \wedge q)$.

Axiómák:

A1(a) $K_{\Sigma,t}p \wedge K_{\Sigma,t}(p \rightarrow q) \rightarrow K_{\Sigma,t}q$
A 'modus ponens' szabály alkalmazása a K tudás operátorra.

A1(b) $B_{\Sigma,t}p \wedge B_{\Sigma,t}(p \rightarrow q) \rightarrow B_{\Sigma,t}q$
A 'modus ponens' szabály alkalmazása a B hit, elfogadás operátorra.

A2(a) $K_{\Sigma,t}p \rightarrow p$
Ha valami ismert, akkor az igaz (az axióma a tudás (K operátor) és a hit (B operátor) közötti különbséget fejezi ki, hasonló axióma B-re nincs).

A3(a) $L_{\Sigma,t}x \rightarrow \forall t_i \geqslant t \;\; L_{\Sigma,t_i}x$
A tudás predikátum monotonitása: amennyiben a tudás egyszer már birtokolt, akkor azt nem lehet elveszíteni. x kulcs vagy üzenet fajta változót jelöl.

A3(b) $K_{\Sigma,t}p \rightarrow \forall t_i \geqslant t \;\; K_{\Sigma,t_i}p$
A tudás operátor monotonitása: amennyiben a tudás egyszer már birtokolt, akkor azt nem lehet elveszíteni.

A3(c) $B_{\Sigma,t}p \rightarrow \forall t_i \geqslant t \;\; B_{\Sigma,t_i}p$
A hit operátor monotonitása: amennyiben a hit egyszer már birtokolt, akkor azt nem lehet elveszíteni.

A4(a) $L_{\Sigma,t}y \wedge C(y,x) \rightarrow \exists \Psi \in ENT \;\; L_{\Psi,t}x$
Ha egy üzenetrész egy másik üzenetrészből származik, akkor minden üzenetdarab, ami a konstrukcióban szerepel, ismert kell legyen valamely egyed által. x és y üzenet, vagy kulcs fajta.

A4(b) $C(x,x)$
A C operátor reflexív. x üzenet, vagy kulcs fajta.

A4(c) $C(x,y) \wedge C(y,z) \rightarrow C(x,z)$
A C operátor tranzitív. x, y és z üzenet, vagy kulcs fajta.

A4(d) $C(e(m,k_\Sigma),m) \wedge C(d(m,k_\Sigma^{-1}),m)$
Az m üzenetet tartalmazza minden olyan üzenet, amely az üzenet k_Σ kulccsal történő titkosításával és k_Σ^{-1} kulccsal történő visszafejtésével kapcsolatos.

A5(a) $S(ch_i,\Sigma,t,m)$
$\rightarrow L_{\Sigma,t}m \;\wedge\; \exists \Psi \in ENT_{ch_i} \backslash \{\Sigma\} \; \exists t_i > t \; R(ch_i,\Psi,t_i,m)$
Kibocsátási axióma. Amennyiben a Σ egyed egy m üzenetet küld a t időpontban a ch_i csatornán, akkor Σ ismeri az m üzenetet a t időpontban, valamint valamely Ψ egyed (Σ-n kívül) fogadja majd az m üzenetet a ch_i csatornán egy t utáni t_i időpontban.

A6(a) $R(ch_i, \Sigma, t, m)$
$\rightarrow L_{\Sigma,t}m \ \wedge \ \exists\Psi \in ENT_{ch_i}\backslash\{\Sigma\} \ \exists t_i < t \ S(ch_i, \Psi, t_i, m)$
Befogadási axióma. Ha a Σ egyed fogad egy m üzenetet a ch_i csatornán a t időpontban, akkor Σ ismeri az m üzenetet t időpontban, és valamely Ψ egyednek (Σ-n kívül) el kellett küldeni az m üzenetet a t-t megelőző t_i időpontban a ch_i csatornán.

A6(b) $R(ch_i, \Sigma, t, m_1) \wedge C(m_1, m_2) \wedge O_{\Sigma,t}(m_1, m_2) \rightarrow \exists\Psi \in ENT \ \exists t_i < t \ \exists m_3 \ (S(ch_i, \Psi, t_i, m_3) \wedge C(m_3, m_2) \wedge L_{\Psi,t_i}m_2 \wedge O_{\Sigma,t}(m_1, m_3) \wedge O_{\Sigma,t}(m_3, m_2))$
Ez az axióma A6(a) axióma általánosítása. Amennyiben Σ egy olyan m_1 üzenetet kap a ch_i csatornán, amelynek része a már általa ismert m_2 üzenet, akkor (mivel Σ nem küldhetett üzenetet magának, a rendszer ilyen üzenetek küldését nem teszi lehetővé) kell lennie egy korábbi m_3 üzenetnek (küldővel, küldési időponttal, stb. együtt), amely tartalmazta az m_2 üzenetet.

A7(a) $L_{\Sigma,t}m \wedge L_{\Sigma,t}k_\Psi \rightarrow L_{\Sigma,t}e(m, k_\Psi)$
Egy egyed képessége, hogy titkosítani tud egy üzenetet amennyiben ismeri a partner nyilvános kulcsát.

A7(b) $L_{\Sigma,t}m \wedge L_{\Sigma,t}k_\Sigma^{-1} \rightarrow L_{\Sigma,t}d(m, k_\Sigma^{-1})$
Egy egyed képessége, hogy vissza tud fejteni egy titkosított üzenetet, ha ismeri a (saját) titkos kulcsát.

A8(a) $\neg L_{\Psi,t}k_\Sigma \ \wedge \ \forall t_i < t \ \neg L_{\Psi,t_i}(e(m, k_\Sigma)) \ \wedge \ \neg(\exists n(R(ch_i\Psi, t_i, n) \wedge C(n, e(m, k_\Sigma)))) \rightarrow \neg L_{\Psi,t}(e(m, k_\Sigma))$
Egy üzenet titkosításának lehetetlensége helyes titkosító kulcs nélkül. Ha egy egyed nem ismeri a k_Σ kulcsot a t időpontban, és ha nem ismeri $e(m, k_\Sigma)$ titkosított üzenetet t időpont előtt, valamint üzenetet sem kap $e(m, k_\Sigma)$ tartalommal t_i időpontban a ch_i csatornán, akkor az egyed nem ismeri $e(m, k_\Sigma)$ titkosított üzenetet a t időpontban.
Kudo-Mathuria-féle módosítás az O operátor segítségével:
$\neg L_{\Psi,t}k_\Sigma \ \wedge \ \forall t_i \leqslant t \ \neg L_{\Psi,t_i}(e(m, k_\Sigma)) \ \wedge \ \neg(\exists n \ (R(ch_i, \Psi, t_i, n) \wedge C(n, e(m, k_\Sigma)) \wedge \ O_{\Psi,t_i}(n, e(m, k_\Sigma)))) \rightarrow \neg L_{\Psi,t}(e(m, k_\Sigma))$

A8(b) $\neg L_{\Psi,t}k_\Sigma^{-1} \ \wedge \ \forall t_i < t \ \neg L_{\Psi,t_i}(d(m, k_\Sigma^{-1})) \wedge \ \neg(\exists n(R(ch_i, \Psi, t_i, n) \ \wedge C(n, d(m, k_\Sigma^{-1})))) \rightarrow \neg L_{\Psi,t}(d(m, k_\Sigma^{-1}))$
Egy titkosított üzenet visszafejtésének lehetetlensége helyes visszafejtő kulcs nélkül. Ha egy egyed nem ismeri a k_Σ^{-1} titkos kulcsot a t időpontban, és ha nem ismeri t időpontot megelőzően a $d(m, k_\Sigma^{-1})$ visszafejtett üzenetet, valamint nem fogad üzenetet $d(m, k_\Sigma^{-1})$ tartalommal a t időpontban, vagy előtte a ch_i csatornán, akkor az egyed nem ismeri a $d(m, k_\Sigma^{-1})$ visszafejtett üzenetet a t időpontban.

Kudo-Mathuria-féle módosítás az O operátor segítségével:
$\neg L_{\Psi,t}k_\Sigma^{-1} \ \wedge \ \forall t_i \leqslant t \ \neg L_{\Psi,t_i}(d(m, k_\Sigma^{-1})) \wedge \ \neg(\exists n \ (R(ch_i, \Psi, t_i, n) \ \wedge C(n, d(m, k_\Sigma^{-1})) \wedge \ O_{\Psi,t_i}(n, d(m, k_\Sigma^{-1})))) \ \rightarrow \neg L_{\Psi,t}(d(m, k_\Sigma^{-1}))$

A9(a) $L_{\Sigma,t}k_{\Sigma}^{-1} \ \wedge \ \forall \Psi \in ENT \backslash \{\Sigma\} \ \neg L_{\Psi,t}k_{\Sigma}^{-1}$
Kulcs titkossági axióma. A privát kulcsok használata a rendszerben csak a tulajdonosaik által lehetséges.

A10(a) $L_{\Sigma,t}(d(m, k_{\Sigma}^{-1})) \rightarrow L_{\Sigma,t}m$
Egy titkos kulcs tulajdonosa tudja használni a kulcsát, képes visszafejteni a titkosított üzeneteket.

A11(a) $L_{\Gamma,t}m \wedge L_{\Gamma,t}ks_{(\Sigma,\Psi)} \rightarrow L_{\Gamma,t}(E(m, ks_{(\Sigma,\Psi)}))$
Egy egyed képes titkosított üzenetet létrehozni a szimmetrikus kulcsú rendszerben, használva az általa ismert titkos kulcsot.

A11(b) $L_{\Gamma,t}m \ \wedge \ L_{\Gamma,t}ks_{\{\Sigma,\Psi\}} \rightarrow L_{\Gamma,t}(D(m, ks_{\{\Sigma,\Psi\}}))$
Egy egyed képes visszafejteni titkos üzenetet a rendelkezésére álló szimmetrikus kulcs segítségvel.

A11(c) $L_{\Sigma,t}m \wedge O_{\Sigma,t}(m, n) \rightarrow L_{\Sigma,t}n$
Az O és az L operátor kapcsolata. Üzenetek felbontása.

A11(d) $L_{\Sigma,t}m \wedge L_{\Sigma,t}n \leftrightarrow L_{\Sigma,t}\{m, n\}$
Üzenetek konjatenációja és felbontása.

A12(a) $(\neg L_{\Gamma,t}ks_{(\Sigma,\Psi)} \wedge \forall t_i \leqslant t \ \neg L_{\Gamma,t_i}(E(m, ks_{(\Sigma,\Psi)})) \wedge$
$\neg(\exists n(R(ch_i, \Gamma, t_i, n) \ \wedge \ C(n, E(m, ks_{(\Sigma,\Psi)}))))$
$\rightarrow \neg L_{\Gamma,t}(E(m, ks_{(\Sigma,\Psi)})))$
Ha valamely Γ egyed nem ismeri a $ks_{(\Sigma,\Psi)}$ kulcsot a t időpontban és nem ismeri t előtti időpontban a $E(m, ks_{(\Sigma,\Psi)})$ titkosított üzenetet és nem fogad $E(m, ks_{(\Sigma,\Psi)})$ üzenetrészt tartalmazó üzenetet t időpontban a ch_i csatornán, akkor Γ nem ismeri a $E(m, ks_{(\Sigma,\Psi)})$ üzenetet t időpontban.

A12(b) $(\neg L_{\Gamma,t}ks_{(\Sigma,\Psi)} \wedge \forall t_i \leqslant t \ \neg L_{\Gamma,t_i}(D(m, ks_{(\Sigma,\Psi)})) \wedge$
$\neg(\exists n(R(ch_i, \Gamma, t_i, n) \ \wedge \ C(n, D(m, ks_{(\Sigma,\Psi)}))))$
$\rightarrow \neg L_{\Gamma,t}(D(m, ks_{(\Sigma,\Psi)})))$
Amennyiben valamely Γ egyed nem ismeri a $ks_{(\Sigma,\Psi)}$ kulcsot a t időpontban és a t időpont előtt nem ismeri a $D(m, ks_{(\Sigma,\Psi)})$ visszafejtett üzenetet, valamint nem kap $D(m, ks_{(\Sigma,\Psi)})$ üzenetrészt tartalmazó üzenetet a t időpontban a ch_i csatornán, akkor Γ nem ismeri a $D(m, ks_{(\Sigma,\Psi)})$ üzenetet a t időpontban.

A13(a) $\forall \Gamma \ \in \ ENT \backslash \{\Sigma, \Psi\} \ \neg L_{\Gamma,t}ks_{(\Sigma,\Psi)} \wedge \exists \Lambda \ \in \ \{\Sigma, \Psi\} \ L_{\Lambda,t}ks_{(\Sigma,\Psi)} \ \rightarrow$
$ks_{(\Sigma,\Psi)} \in \{KS_{(\Sigma,\Psi)}\}$
Csak az osztott titkos kulcs valódi tulajdonosai ismerik a kulcsot, és csak ők tudják, hogy a kulcsuk helyes kulcs.

A14(a) $\forall \Gamma \ \in \ ENT \backslash \{\Sigma, \Psi\} \ \neg L_{\Gamma,t}ss_{(\Sigma,\Psi)} \wedge \exists \Lambda \ \in \ \{\Sigma, \Psi\} L_{\Lambda,t}ss(\Sigma, \Psi) \ \rightarrow$
$ss_{(\Sigma,\Psi)} \in \{SS_{\{\Sigma,\Psi\}}\}$
Csak az osztott titkos kulcs tulajdonosai ismerik a megosztott titkot, és csak ők tudják, hogy a megosztott titok „helyes” titok (’good secret’). Az axióma vonatkozik a titok frissességére is.

A15(a) $[A(\Sigma,t,\Psi) \rightarrow (L_{\Sigma,t}ss_{(\Sigma,\Psi)} \wedge ss_{(\Sigma,\Psi)} \in \{SS_{\{\Sigma,\Psi\}}\} \wedge R(\Sigma,t,m)) \wedge C(m,ss_{(\Sigma,\Psi)}) \wedge \forall t_i \leqslant t\ \neg S(\Sigma,t_i,m)] \rightarrow K_{\Sigma,t}(S(\Psi,t_i,m))$
Hitelesítési axióma - szimmetrikus forma. Ha Σ egyed hiteles partnernek fogadja el Ψ egyedet, akkor ha Σ ismeri a $ss_{(\Sigma,\Psi)}$ titkot, amit megoszt a Ψ egyeddel (a titok friss), és ez a titok „jó titok" ('good secret'), valamint Σ üzenetet kap (amit nem ő küldött) a t időpontban, amely tartalmazza a $ss_{(\Sigma,\Psi)}$ üzenetet, akkor Σ tudja azt, hogy Ψ küldte az üzenetet a t időpont előtt.

A15(b) $[A(\Sigma,t,\Psi) \rightarrow (L_{\Sigma,t}k_\Psi \wedge L_{\Sigma,t}m \wedge R(\Sigma,t,n) \wedge C(n,e(m,k_\Psi^{-1}))\ ] \rightarrow \forall t_i \leqslant t\ K_{\Sigma,t}(S(\Psi,t_i,n))$
Hitelesítési axióma - asszimetrikus forma. Ha Σ egyed hiteles partnernek fogadja el Ψ egyedet, akkor ha Σ ismeri a Ψ egyed k_Ψ nyilvános kulcsát és az m üzenetet, és Σ az n üzenetet fogadja, amely tartalmazza az $e(m,k_{\Psi^{-1}})$ üzenetdarabot, akkor Σ tudja azt, hogy Ψ küldte az n üzenetet a t időpontot megelőző időpontban.

A16(a) $L_{\Sigma,t}m \wedge L_{\Sigma,t}k \rightarrow L_{\Sigma,t}h(m,k)$.
A Σ egyed képes elkészíteni a t időpontban a kulcsolt üzenetkivonatot (kulcsolt hash függvény), ha rendelkezésre áll az m üzenet és a k kulcs. Az eredmény $h(m,k)$ üzenet típusú.

A17(a) $h(n_A,k_A) = h(n_B,k_B) \leftrightarrow n_A = n_B \ \wedge\ k_A = k_B$.
A kulcsolt hash függvény alaptulajdonságainak rögzítése.

A18(a) $L_{\Sigma,t}m \rightarrow L_{\Sigma,t}h(m)$.
A Σ egyed képes elkészíteni a t időpontban az üzenetkivonatot (hash függvény), ha rendelkezésre áll az m üzenet. Az eredmény $h(m)$ üzenet típusú.

A19(a) $h(n_A) = h(n_B) \leftrightarrow n_A = n_B$.
A hash függvény alaptulajdonságainak rögzítése.

A20(a) $\forall t \in TIME\ (t \leqslant t) \ \wedge\ \forall t,s \in TIME\ (t \leqslant s \wedge s \leqslant t \rightarrow t = s) \ \wedge\ \forall t,s,r \in TIME\ (s \leqslant t \wedge t \leqslant r \rightarrow s \leqslant r)$
Az időtípus alaptulajdonságai.

Kudo-Mathuria-féle kiegészítő axiómák.

TA1(a) $\forall t < \tau\ L_{T,t}k_\tau^{-1} \wedge \forall \Sigma \in ENT\backslash\{T\}\ \neg L_{\Sigma,t}k_\tau^{-1}$
Az A9(a) axióma konkretizált változata. A nyilvános kulcsú rendszerekben a megfelelő időpontokban a titkos kulcsot csak a kulcs tulajdonosa (itt a T generáló megbízható fél, a kulcskezelő szerver) ismerheti. k_τ, k_τ^{-1} nyilvános és titkos kulcsok, τ az időfeloldó titkosítás feloldásának rögzített időpontja (amikor a titkos kulcs elküldhető az üzenet fogadójának).

TA2(a) $L_{\Sigma,t}x \wedge L_{\Sigma,t}k_\tau \rightarrow L_{\Sigma,t}(e(x,k_\tau))$
Az A7(a) axióma konkretizált változata. x üzenet fajta.

TA2(b) $L_{\Sigma,t}x \wedge L_{\Sigma,t}k_\tau^{-1} \rightarrow L_{\Sigma,t}(d(x,k_\tau^{-1}))$
Az A7(b) axióma konkretizált változata. x üzenet fajta.

TA3(a) $\neg L_{\Sigma,t} k_\tau \wedge \forall t_i \leqslant t \ \neg L_{\Sigma,t_i}(e(m,k_\tau)) \wedge \neg(\exists n(R(ch_i,\Sigma,t_i,n) \wedge C(n,e(m,k_\tau)) \wedge O_{\Sigma,t}(n,e(m,k_\tau)))) \rightarrow \neg L_{\Sigma,t}(e(m,k_\tau))$
Az A8(a) axióma módosított változata.

TA3(b) $\neg L_{\Sigma,t} k_\tau^{-1} \wedge \forall t_i \leqslant t \ \neg L_{\Sigma,t_i}(d(m,k_\tau^{-1})) \wedge \neg(\exists n(R(ch_i,\Sigma,t_i,n) \wedge C(n,d(m,k_\tau^{-1})) \wedge O_{\Sigma,t}(n,d(m,k_\tau^{-1})))) \rightarrow \neg L_{\Sigma,t}(d(m,k_\tau^{-1}))$
Az A8(b) axióma módosított változata.

TA4(a) $L_{\Sigma,t}(e(m,k_\tau)) \rightarrow L_{T,t} k_\tau$
T ismeri az idő-kulcsokat. .

TA5(a) $\forall \Sigma \in ENT \backslash \{T\} \ \forall t < \tau \ L_{\Sigma,t} m \wedge m = e(n,k_\tau) \wedge C(n,z) \rightarrow \neg O_{\Sigma,t}(m,z)$
Az időbizalmas adatokat csak T ismerheti a megadott τ időpont előtt. x, y, z üzenet fajta.

KM1(a) $O_{i,t}(x,y) \wedge O_{i,t}(y,z) \rightarrow O_{i,t}(x,z)$
Az O operátor tranzitivitása. x, y, z üzenet vagy kulcs fajta.

Megjegyzések:

(M1) A típuskonverziót megvalósító függvényjelek ($E2U(\Sigma)$, $U2E(m)$; $K2U(k)$, $U2K(m)$; $T2U(t)$, $U2T(m)$) az egyes típusok üzenetbe ágyazását és az onnan történő kinyerését teszik lehetővé.

(M2) Az $A7(a)$ (konkretizált változat $TA2(a)$) és $A7(b)$ (konkretizált változat $TA2(b)$) axi-ómák írják le az üzenetek titkosítását és visszafejtését nyilvános kulcsú rendszrekben. Az e és d függvényjelek definiálása során szerepel az $e(m,k_\Sigma^{-1})$ függvényjel a digitális aláírás elkészítésére és a $d(m,k_\Sigma)$ függvényjel az aláírás ellenőrzésére. Az axiómák nem tartalmaznak közvetlenül utalást a digitális aláírás kezelésére. A következőkben feltesszük, hogy az egyedek képesek a digitális aláírás elkészítésére és annak visszafejtésére:
$L_{\Sigma,t} m \wedge L_{\Sigma,t} k_\Sigma^{-1} \rightarrow L_{\Sigma,t}\, e(m,k_\Sigma^{-1})$,
$L_{\Sigma,t}\, e(m,k_\Psi^{-1}) \wedge L_{\Sigma,t} k_\Psi \rightarrow L_{\Sigma,t}\, d(e(m,k_\Psi^{-1}),k_\Psi) = L_{\Sigma,t}\, m$.

(M3) A konkatenációval összekapcsolt üzenetdarabok szétbontásának lehetőségét tartalmazza az $A11(c)$ axióma. Az axiómában $O_{\Sigma,t}(m,n)$ hordozza magában a kapcsolódás jelölését. Az üzenetdarabok összeillesztését és felbontását az $A11(d)$ axióma írja le.

(M4) A protokollokban szereplő üzenetforgalom leírása során az áttekinthetőség és a könnyebb olvashatóság érdekében egyszerüsítjük a jelölést. Minden olyan esetben, ahol nem okoz félreértést, elhagyjuk a típuskonverziót jelölő függvényjelet, csak az argumentum elemeit tüntetjük fel. Például egy A egyedet, k_Σ kulcsot és t időpontot tartalmazó üzenetet $\{E2U(A), K2U(k_\Sigma), T2U(t)\}$ helyett $\{A, k_\Sigma, t\}$-vel jelölünk. Ez a megoldás igazodik szakirodalomban alkalmazott jelölési szokásokhoz.

(M5) A protokollok formális alakban történő rögzítese az eddigiekben leírt formális rendszer alkalmazását jelenti konkrét protokollok esetében. Többcsatornás protokollok esetén rögzítjük azt a kódolási előírást, hogy nyilvános csatorna esetén a küldött és a fogadott üzenetekről feltesszük, hogy azok nem egyeznek meg. Ez a Dolev-Yao támadási modell alkalmazását jelenti, ami azt mondja ki, hogy a kommunikációs hálózatot úgy kell tekintenünk, hogy azt a támadó teljes mértékben lehallgathatja, megváltoztathatja az üzeneteket, új üzeneteket generálhat.
Így, ha $CH(ch_i, pub)$ és $S(ch_i, A, t_1, n_A)$, akkor $R(ch_i, B, t_2, n_B)$.
Védett csatornák esetén, ha $CH(ch_i, sec)$ és $S(ch_i, A, t_1, n_A)$, akkor $R(ch_i, B, t_2, n_A)$.

$\cdots \bullet \cdots$

Takács Péter főiskolai adjunktus
Debreceni Egyetem, Egészségügyi Kar, Nyíregyháza, 4400, Sóstói út 2-4.