Special Topics in Logic and Security I - Exam Recap

Bogdan Macovei

February 10, 2023

Structura examenului

- Pentru jumatatea de materie de **analiza protocoalelor de securitate**, examenul va acoperi urmatoarele doua formalisme:
  - **BAN Logic** - un exercitiu de constructie de demonstratie, si un exercitiu de analiza a unui protocol;
  - **Semantica operationala** - exercitii cascadate pentru analiza unui protocol.

BAN Logic

- **Primul exercitiu**: fiind date doua formule φ si ψ din logica BAN, sa se determine o multime de ipoteze Γ, astfel incat din $\Gamma \cup \{\varphi\}$ sa putem deduce ψ. Sa se scrie si demonstratia completa.

BAN Logic

- **Primul exercitiu**: fiind date doua formule φ si ψ din logica BAN, sa se determine o multime de ipoteze Γ, astfel incat din $\Gamma \cup \{\varphi\}$ sa putem deduce ψ. Sa se scrie si demonstratia completa.
- **Exemplu**: Consideram P, Q doi agenti, iar X un mesaj.
  - Fie $\varphi := P \mid\equiv Q \mid\equiv X$. Sa se determine o multime de ipoteze, nu neaparat minimala, Γ, astfel incat din $\Gamma \cup \{\varphi\}$ sa se deduca $\psi := P \mid\equiv X$.

BAN Logic

- **Primul exercitiu**: fiind date doua formule φ si ψ din logica BAN, sa se determine o multime de ipoteze Γ, astfel incat din $\Gamma \cup \{\varphi\}$ sa putem deduce ψ. Sa se scrie si demonstratia completa.
- **Exemplu**: Consideram P, Q doi agenti, iar X un mesaj.
  - Fie $\varphi := P \mid\equiv Q \mid\equiv X$. Sa se determine o multime de ipoteze, nu neaparat minimala, Γ, astfel incat din $\Gamma \cup \{\varphi\}$ sa se deduca $\psi := P \mid\equiv X$.
  - **Solutie**: Din regula jurisdictiei, e suficient ca formula $(P \mid\equiv Q \Rightarrow X) \in \Gamma$, deci e suficient $\Gamma := \{P \mid\equiv Q \Rightarrow X\}$.

Analiza de protocol

- Urmatorul exercitiu **BAN** consta intr-o analiza de protocol, si acelasi protocol va fi utilizat si pentru partea de semantica operationala.
- Pentru BAN Logic, se cer urmatoarele:
  - idealizarea protocolului - care sunt toate asumptiile pe care le putem face referitoare la protocol? **Discutia asumptiilor daca este necesar!** Unele sunt de bun simt, dar unele sunt neintuitive sau gresite. Le marcam pe cele din categoria a doua, pentru ca scopul nostru este...
  - demonstrarea autentificarilor mutuale intre doi agenti modulo un anumit mesaj sau un *nonce*. Daca agentii sunt P si Q, se va cere demonstrarea modulo X (mesaj sau *nonce*), adica $P \mid\equiv Q \mid\equiv X$ si $Q \mid\equiv P \mid\equiv X$. Cel mai probabil, nu putem ajunge la aceste doua concluzii fara asumptii suplimentare care trebuie marcate si comentate (pe scurt). Daca sunt mai multi agenti participanti, se va cere demonstrarea doar pentru doi dintre ei.

Analiza de protocol in logica BAN - exemplu

- Fie I, R si S agentii participanti la protocol, iar n_R un *nonce* generat de agentul R. Schimbul de mesaje este, dupa cum urmeaza:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$

Analiza de protocol in logica BAN - exemplu

- Fie I, R si S agentii participanti la protocol, iar n_R un *nonce* generat de agentul R. Schimbul de mesaje este, dupa cum urmeaza:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Se cer: idealizarea (=formalizarea) in BAN logic si autentificarea mutuala a agentilor I si R modulo n_R.

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Se formalizeaza schimbul de mesaje in formule care au o reprezentare si o semnificatie in BAN.

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Se formalizeaza schimbul de mesaje in formule care au o reprezentare si o semnificatie in BAN.
  - primele doua mesaje nu au nicio semnificatie in BAN. Nu este relevant sa avem o formula de forma $R \triangleleft I$, e presupus ca agentii se cunosc de dinainte sa joace protocolul si nu au nevoie de alta initializare de context. **Identitatea agentilor mereu va lipsi din formulele BAN!**
  - este suficient sa formalizam mesajele 3, 4, 5.

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Se formalizeaza schimbul de mesaje in formule care au o reprezentare si o semnificatie in BAN.
  - primele doua mesaje nu au nicio semnificatie in BAN. Nu este relevant sa avem o formula de forma $R \triangleleft I$, e presupus ca agentii se cunosc de dinainte sa joace protocolul si nu au nevoie de alta initializare de context. **Identitatea agentilor mereu va lipsi din formulele BAN!**
  - este suficient sa formalizam mesajele 3, 4, 5.
  - **Intotdeauna, o actiune $P \to Q : X$ se va formaliza $Q \triangleleft X$ si NU $P \mid\sim X$! Diferenta este data de semnificatia operatorilor, $\triangleleft$ este in sesiunea curenta, $\mid\sim$ poate fi orice alta sesiune anterioara.**

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Idealizarea mesajelor 3, 4, 5 va fi:
  - $P_1 : R \triangleleft \{\xmapsto{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Idealizarea mesajelor 3, 4, 5 va fi:
  - $P_1 : R \triangleleft \{\stackrel{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
- Asumptiile pe care le putem face:
  - $A_1 : R \mid\equiv \stackrel{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \stackrel{k_R}{\longmapsto} R$

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Idealizarea mesajelor 3, 4, 5 va fi:
  - $P_1 : R \triangleleft \{\stackrel{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
- Asumptiile pe care le putem face:
  - $A_1 : R \mid\equiv \stackrel{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \stackrel{k_R}{\longmapsto} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Idealizarea mesajelor 3, 4, 5 va fi:
  - $P_1 : R \triangleleft \{\stackrel{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
- Asumptiile pe care le putem face:
  - $A_1 : R \mid\equiv \stackrel{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \stackrel{k_R}{\longmapsto} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \stackrel{k_I}{\longmapsto} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$

Idealizarea BAN Logic

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
- Idealizarea mesajelor 3, 4, 5 va fi:
  - $P_1 : R \triangleleft \{\overset{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
- Asumptiile pe care le putem face:
  - $A_1 : R \mid\equiv \overset{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \overset{k_R}{\longmapsto} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \overset{k_I}{\longmapsto} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\overset{k_I}{\longmapsto} I)$

BAN Logic - Demonstratie: $R \mid\equiv I \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\xmapsto{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \xmapsto{k_S} S$
  - $A_2 : I \mid\equiv \xmapsto{k_R} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \xmapsto{k_I} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\xmapsto{k_I} I)$
- Demonstratie:

BAN Logic - Demonstratie: $R \mid\equiv I \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\xrightarrow{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \xrightarrow{k_S} S$
  - $A_2 : I \mid\equiv \xrightarrow{k_R} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \xrightarrow{k_I} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\xrightarrow{k_I} I)$

- Demonstratie:

  1. $R \mid\equiv S \mid\sim \xrightarrow{k_I} I$ din $MM - PK(P_1, A_1)$

BAN Logic - Demonstratie: $R \mid\equiv I \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\stackrel{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \stackrel{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \stackrel{k_R}{\longmapsto} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \stackrel{k_I}{\longmapsto} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\stackrel{k_I}{\longmapsto} I)$

- Demonstratie:

  1. $R \mid\equiv S \mid\sim \stackrel{k_I}{\longmapsto} I$ din $MM - PK(P_1, A_1)$
  2. $R \mid\equiv S \mid\equiv \stackrel{k_I}{\longmapsto} I$ din $NV(1, A_7^*)$

BAN Logic - Demonstratie: $R \mid\equiv I \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\stackrel{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \stackrel{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \stackrel{k_R}{\longmapsto} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \stackrel{k_I}{\longmapsto} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\stackrel{k_I}{\longmapsto} I)$

- Demonstratie:
  1. $R \mid\equiv S \mid\sim \stackrel{k_I}{\longmapsto} I$ din $MM - PK(P_1, A_1)$
  2. $R \mid\equiv S \mid\equiv \stackrel{k_I}{\longmapsto} I$ din $NV(1, A_7^*)$
  3. $R \mid\equiv \stackrel{k_I}{\longmapsto} I$ din $JR(2, A_5)$

BAN Logic - Demonstratie: $R \mid\equiv I \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\xrightarrow{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \xrightarrow{k_S} S$
  - $A_2 : I \mid\equiv \xrightarrow{k_R} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \xrightarrow{k_I} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\xrightarrow{k_I} I)$

- Demonstratie:

  1. $R \mid\equiv S \mid\sim \xrightarrow{k_I} I$ din $MM - PK(P_1, A_1)$
  2. $R \mid\equiv S \mid\equiv \xrightarrow{k_I} I$ din $NV(1, A_7^*)$
  3. $R \mid\equiv \xrightarrow{k_I} I$ din $JR(2, A_5)$
  4. $R \mid\equiv I \mid\sim n_R$ din $MM - PK(P_3, 3)$

BAN Logic - Demonstratie: $R \mid\equiv I \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\xmapsto{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \xmapsto{k_S} S$
  - $A_2 : I \mid\equiv \xmapsto{k_R} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \xmapsto{k_I} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\xmapsto{k_I} I)$

- Demonstratie:
  1. $R \mid\equiv S \mid\sim \xmapsto{k_I} I$ din $MM - PK(P_1, A_1)$
  2. $R \mid\equiv S \mid\equiv \xmapsto{k_I} I$ din $NV(1, A_7^*)$
  3. $R \mid\equiv \xmapsto{k_I} I$ din $JR(2, A_5)$
  4. $R \mid\equiv I \mid\sim n_R$ din $MM - PK(P_3, 3)$
  5. $R \mid\equiv I \mid\equiv n_R$ din $NV(4, A_4)$

BAN Logic - Demonstratie: $I \mid\equiv R \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\xrightarrow{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \xrightarrow{k_S} S$
  - $A_2 : I \mid\equiv \xrightarrow{k_R} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \xrightarrow{k_I} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\xrightarrow{k_I} I)$
- Demonstratie:

BAN Logic - Demonstratie: $I \mid\equiv R \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\stackrel{k_I}{\longmapsto} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \stackrel{k_S}{\longmapsto} S$
  - $A_2 : I \mid\equiv \stackrel{k_R}{\longmapsto} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \stackrel{k_I}{\longmapsto} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\stackrel{k_I}{\longmapsto} I)$

- Demonstratie:
  1. $I \mid\equiv R \mid\sim n_R$ din $MM - PK(P_2, A_2)$

BAN Logic - Demonstratie: $I \mid\equiv R \mid\equiv n_R$

- Idealizare:
  - $P_1 : R \triangleleft \{\xrightarrow{k_I} I\}_{k_S^{-1}}$
  - $P_2 : I \triangleleft \{n_R\}_{k_R^{-1}}$
  - $P_3 : R \triangleleft \{n_R\}_{k_I^{-1}}$
  - $A_1 : R \mid\equiv \xrightarrow{k_S} S$
  - $A_2 : I \mid\equiv \xrightarrow{k_R} R$
  - $A_3 : I \mid\equiv \#(n_R)$
  - $A_4 : R \mid\equiv \#(n_R)$
  - $A_5 : R \mid\equiv S \Rightarrow \xrightarrow{k_I} I$
  - $A_6 : I \mid\equiv R \Rightarrow n_R$
  - $A_7^* : R \mid\equiv \#(\xrightarrow{k_I} I)$

- Demonstratie:
  1. $I \mid\equiv R \mid\sim n_R$ din $MM - PK(P_2, A_2)$
  2. $I \mid\equiv R \mid\equiv n_R$ din $NV(1, A_3)$

Semantica operationala

- Plecand de la acelasi protocol - pe care il numim **P** - deja analizat in logica BAN, se vor rezolva urmatoarele cerinte:
  - se vor descrie rolurile agentilor in protocol;
  - se va scrie un *trace* onest al protocolului;
  - se vor executa **trei** pasi in semantica operationala;
  - se va verifica o proprietate (*claim*) de securitate.

Semantica operationala - specificarea rolurilor

- Protocolul (caruia i se adauga si *claim*-ul de verificat):
  - $I \rightarrow R : I, R$
  - $R \rightarrow S : I, R$
  - $S \rightarrow R : \{I, pk(I)\}_{sk(S)}$
  - $R \rightarrow I : \{R, n_R\}_{sk(R)}$
  - $I \rightarrow R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$

Semantica operationala - specificarea rolurilor

- Protocolul (caruia i se adauga si *claim*-ul de verificat):
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Pentru specificarea rolurilor, folosim ca pentru un protocol P si pentru un rol R, specificarea rolului R este

$$P(R) = \{(IK(R), events(R))\}$$

unde in $events(R)$ includem toate *send*-urile, *receive*-urile si *claim*-urile asociate. **Important!** Orice actiune $P \to Q : X$ inseamna, de fapt, doua actiuni: P trimite un mesaj, iar Q receptioneaza un mesaj.

Semantica operationala - specificarea rolurilor

- Protocolul (caruia i se adauga si *claim*-ul de verificat):
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Pentru specificarea rolurilor, folosim ca pentru un protocol P si pentru un rol R, specificarea rolului R este

$$P(R) = \{(IK(R), events(R))\}$$

unde in $events(R)$ includem toate *send*-urile, *receive*-urile si *claim*-urile asociate. **Important!** Orice actiune $P \to Q : X$ inseamna, de fapt, doua actiuni: P trimite un mesaj, iar Q receptioneaza un mesaj.

- Actiunile sunt *etichetate*, si *send*-urile si *receive*-urile sunt perechi: $send_1$ cu $recv_1$, $send_2$ cu $recv_2$ etc.

Semantica operationala - specificarea rolurilor

- Protocolul:
  - $I \rightarrow R : I, R$
  - $R \rightarrow S : I, R$
  - $S \rightarrow R : \{I, pk(I)\}_{sk(S)}$
  - $R \rightarrow I : \{R, n_R\}_{sk(R)}$
  - $I \rightarrow R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$

Semantica operationala - specificarea rolurilor

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$

Semantica operationala - specificarea rolurilor

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$

Semantica operationala - specificarea rolurilor

- Protocolul:
  - $I \rightarrow R : I, R$
  - $R \rightarrow S : I, R$
  - $S \rightarrow R : \{I, pk(I)\}_{sk(S)}$
  - $R \rightarrow I : \{R, n_R\}_{sk(R)}$
  - $I \rightarrow R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$
  - $send_4(R, I, \{R, n_R\}_{sk(R)})$
  - $recv_4(I, R, \{R, V\}_{sk(R)})$

Semantica operationala - specificarea rolurilor

- Protocolul:
  - $I \to R : I, R$
  - $R \to S : I, R$
  - $S \to R : \{I, pk(I)\}_{sk(S)}$
  - $R \to I : \{R, n_R\}_{sk(R)}$
  - $I \to R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S)})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$
  - $send_4(R, I, \{R, n_R\}_{sk(R)})$
  - $recv_4(I, R, \{R, V\}_{sk(R)})$
  - $send_5(I, R, \{V, I\}_{sk(I)})$
  - $recv_5(R, I, \{n_R, I\}_{sk(I)})$

Semantica operationala - specificarea rolurilor

- Protocolul:
  - $I \rightarrow R : I, R$
  - $R \rightarrow S : I, R$
  - $S \rightarrow R : \{I, pk(I)\}_{sk(S)}$
  - $R \rightarrow I : \{R, n_R\}_{sk(R)}$
  - $I \rightarrow R : \{n_R, I\}_{sk(I)}$
  - $claim(I, R, recent - alive)$
- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S)})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$
  - $send_4(R, I, \{R, n_R\}_{sk(R)})$
  - $recv_4(I, R, \{R, V\}_{sk(R)})$
  - $send_5(I, R, \{V, I\}_{sk(I)})$
  - $recv_5(R, I, \{n_R, I\}_{sk(I)})$
  - $claim_6(I, R, recent - alive)$

Semantica operationala - specificarea rolurilor

- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S)})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$
  - $send_4(R, I, \{R, n_R\}_{sk(R)})$
  - $recv_4(I, R, \{R, V\}_{sk(R)})$
  - $send_5(I, R, \{V, I\}_{sk(I)})$
  - $recv_5(R, I, \{n_R, I\}_{sk(I)})$
  - $claim_6(I, R, recent - alive)$

- Specificam $P(I)$: cunostintele initiale pe care le are I, respectiv secventa de instructiuni care i se asociaza (toate actiunile cu I pe prima pozitie).

$$\begin{aligned} P(I) = \{(&\{pk(I), sk(I), pk(R), pk(S)\}, \\ &[send_1(I, R, (I, R)); \\ &recv_4(I, R, \{R, V\}_{sk(R)}); \\ &send_5(I, R, \{V, I\}_{sk(I)}; \\ &claim_6(I, R, recent - alive)])\} \end{aligned}$$

Semantica operationala - specificarea rolurilor

- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$
  - $send_4(R, I, \{R, n_R\}_{sk(R)})$
  - $recv_4(I, R, \{R, V\}_{sk(R)})$
  - $send_5(I, R, \{V, I\}_{sk(I)})$
  - $recv_5(R, I, \{n_R, I\}_{sk(I)})$
  - $claim_6(I, R, recent - alive)$

$$\begin{aligned} P(I) = \{(\{&pk(I), sk(I), pk(R), pk(S)\}, \\ &[send_1(I, R, (I, R)); \\ &recv_4(I, R, \{R, V\}_{sk(R)}); \\ &send_5(I, R, \{V, I\}_{sk(I)}; \\ &claim_6(I, R, recent - alive)])\} \end{aligned}$$

$$\begin{aligned} P(R) = \{(\{&pk(R), sk(R), pk(S)\}, \\ &[recv_1(R, I, (I, R)); \\ &send_2(R, S, (I, R)); \\ &recv_3(R, S, \{I, pk(I)\}_{sk(S)}); \\ &send_4(R, I, \{R, n_R\}_{sk(R)}); \\ &recv_5(R, I, \{n_R, I\}_{sk(I)})])\} \end{aligned}$$

Semantica operationala - un *trace* onest

- *Trace*-urile sunt secvente de perechi de forma (instantiere, eveniment)
- Intr-un *trace* vrem sa simulam executia unui protocol. Forma de mai sus este abstracta, vorbim despre roluri si variabile, dar intr-un *trace* vrem sa vorbim despre agenti si mesaje (mai corect, RunTerms).
  - Agentii sunt instantieri ale rolurilor
  - Mesajele (RunTerms) sunt instantieri ale variabilelor

Semantica operationala - un *trace* onest

- *Trace*-urile sunt secvente de perechi de forma (instantiere, eveniment)
- Intr-un *trace* vrem sa simulam executia unui protocol. Forma de mai sus este abstracta, vorbim despre roluri si variabile, dar intr-un *trace* vrem sa vorbim despre agenti si mesaje (mai corect, RunTerms).
  - Agentii sunt instantieri ale rolurilor
  - Mesajele (RunTerms) sunt instantieri ale variabilelor
- O instantiere este definita printr-un triplet (θ, ρ, σ), unde θ este un identificator al rundei (este unic pentru fiecare agent), $\rho : Role \to Agent$, iar $\sigma : Var \to RunTerm$

Semantica operationala - un *trace* onest

- *Trace*-urile sunt secvente de perechi de forma (instantiere, eveniment)
- Intr-un *trace* vrem sa simulam executia unui protocol. Forma de mai sus este abstracta, vorbim despre roluri si variabile, dar intr-un *trace* vrem sa vorbim despre agenti si mesaje (mai corect, RunTerms).
  - Agentii sunt instantieri ale rolurilor
  - Mesajele (RunTerms) sunt instantieri ale variabilelor
- O instantiere este definita printr-un triplet (θ, ρ, σ), unde θ este un identificator al rundei (este unic pentru fiecare agent), $\rho : Role \to Agent$, iar $\sigma : Var \to RunTerm$
- Un *trace* onest este un *trace* in care sunt implicati doar agenti onesti, adica $Im\rho$ nu contine atacatori.

Semantica operationala - un *trace* onest

- Detalierea actiunilor:
  - $send_1(I, R, (I, R))$
  - $recv_1(R, I, (I, R))$
  - $send_2(R, S, (I, R))$
  - $recv_2(S, R, (I, R))$
  - $send_3(S, R, \{I, pk(I)\}_{sk(S)})$
  - $recv_3(R, S, \{I, pk(I)\}_{sk(S)})$
  - $send_4(R, I, \{R, n_R\}_{sk(R)})$
  - $recv_4(I, R, \{R, V\}_{sk(R)})$
  - $send_5(I, R, \{V, I\}_{sk(I)})$
  - $recv_5(R, I, \{n_R, I\}_{sk(I)})$
  - $claim_6(I, R, recent - alive)$

- $\rho := \{I \rightarrow A, R \rightarrow B, S \rightarrow Srv\}$

$$
\begin{aligned}
&[((1, \rho, \emptyset), create(I)),\\
&((1, \rho, \emptyset), send_1(A, B, (A, B))),\\
&((2, \rho, \emptyset), create(R)),\\
&((2, \rho, \emptyset), recv_1(B, A, (A, B))),\\
&((2, \rho, \emptyset), send_2(B, Srv, (A, B))),\\
&((3, \rho, \emptyset), create(S)),\\
&((3, \rho, \emptyset), recv_2(Srv, B, (A, B))),\\
&((3, \rho, \emptyset), send_3(Srv, B, \{A, pk(A)\}_{sk(Srv)})),\\
&((2, \rho, \emptyset), recv_3(B, S, \{A, pk(A)\}_{sk(Srv)})),\\
&((2, \rho, \emptyset), send_4(B, A, \{B, n_B^{\#2}\}_{sk(B)})),\\
&((1, \rho, \{V \rightarrow n_B^{\#2}\}), recv_4(A, B, \{B, V\})_{sk(B)}),\\
&((1, \rho, \{V \rightarrow n_B^{\#2}\}), send_5(A, B, \{V, A\}_{sk(A)})),\\
&((2, \rho, \emptyset), recv_5(B, A, \{n_B^{\#2}, A\}_{sk(A)})),\\
&((1, \rho, \emptyset), claim_6(A, B, recent - alive))]
\end{aligned}
$$

Semantica operationala - rulare protocol

- vom executa pasi in semantica operationala - lucram intr-un sistem etichetat de tranzitii;
- se pleaca dintr-o stare initiala, $s_0(P)$;
- starea se schimba pe masura ce se executa actiunile - tranzitia este in functie de instantierea curenta si de actiunea curenta.

Semantica operationala - rulare protocol

- starile sunt formate dintr-o pereche care contine cunostinta curenta a adversarului, respectiv ce mesaje mai sunt de executat;
- starea initiala este $s_0(P) =<< AKN_0(P), \emptyset >>$, cunostinta intiala a adversarului, si multimea vida de evenimente;
- in cazul *trace*-urilor oneste, cunostinta initiala a adversarului este formata din informatiile publice: identitatile agentilor, respectiv cheile publice stiute intre acestia.

$$AKN_0(P) := \{A, B, Srv, pk(B), pk(Srv)\}$$

Semantica operationala - rulare protocol

- starile sunt formate dintr-o pereche care contine cunostinta curenta a adversarului, respectiv ce mesaje mai sunt de executat;
- starea initiala este $s_0(P) = << AKN_0(P), \emptyset >>$, cunostinta intiala a adversarului, si multimea vida de evenimente;
- in cazul *trace*-urilor oneste, cunostinta initiala a adversarului este formata din informatiile publice: identitatile agentilor, respectiv cheile publice stiute intre acestia.

$$AKN_0(P) := \{A, B, Srv, pk(B), pk(Srv)\}$$

- trebuie sa executam pasii din *trace*-ul deja descris, conform regulilor de tranzitie.

Semantica operationala - rulare protocol

- Consideram doar primii trei pasi:

$$[((1, \rho, \emptyset), create(I)),$$
$$((1, \rho, \emptyset), send_1(A, B, (A, B))),$$
$$((2, \rho, \emptyset), create(R))]$$

- suntem in starea $s_0(P) =<< \{A, B, Srv, pk(B), pk(Srv)\}, F = \emptyset >>$
- prima actiune este un $create(I)$. Regula este

$$\frac{I \in dom(P) \quad ((\theta, \rho, \emptyset), s) \in runsof(P, I) \quad \theta \notin runIDs(F)}{<< AKN, F >> \rightarrow << AKN, F \cup \{((\theta, \rho, \emptyset), s)\} >>}$$

- unde $runsof(P, I)$ returneaza instantierea $(1, \rho, \emptyset)$ si secventa de instructiuni deja specificata in $P(I)$, iar conditia $\theta \notin runIDs(F)$ impune ca identificatorul rundei pentru I sa nu fi aparut deja si pentru alt rol.

Semantica operationala - rulare protocol

- Consideram doar primii trei pasi:

$$\begin{aligned}[&((1,\rho,\emptyset), create(I)),\\ &((1,\rho,\emptyset), send_1(A,B,(A,B))),\\ &((2,\rho,\emptyset), create(R))]\end{aligned}$$

- suntem in starea $s_0(P) =<< \{A, B, Srv, pk(B), pk(Srv)\}, F = \emptyset >>$
- prima actiune este un *create*(I). Regula este

$$\frac{I \in dom(P) \quad ((\theta,\rho,\emptyset), s) \in runsof(P, I) \quad \theta \notin runIDs(F)}{<< AKN, F >> \rightarrow << AKN, F \cup \{((\theta,\rho,\emptyset), s)\} >>}$$

- se executa tranzitia, si obtinem

$$<< AKN_0(P), \emptyset >> \rightarrow << AKN_0(P), \{((1,\rho,\emptyset), [send_1; recv_4; send_5; claim_6])\} >>$$

Semantica operationala - rulare protocol

- Ne-au ramas urmatorii doi pasi:

$$[((1, \rho, \emptyset), send_1(A, B, (A, B))),$$
$$((2, \rho, \emptyset), create(R))]$$

- suntem in starea
$s_1(P) = << AKN_0(P), F = \{((1, \rho, \emptyset), [send_1; recv_4; send_5; claim_6])\} >>$
- urmatoarea actiune este *send*. Regula este

$$[send]\frac{e = send_l(R_1, R_2, m) \quad (inst, [e] \cdot s) \in F}{<< AKN, F >> \to << AKN \cup \{inst(m)\}, (F - \{(inst, [e] \cdot s\})\}) \cup \{(inst, s)\} >>}$$

Semantica operationala - rulare protocol

- Ne-au ramas urmatorii doi pasi:

$$[((1, \rho, \emptyset), send_1(A, B, (A, B))),$$
$$((2, \rho, \emptyset), create(R))]$$

- suntem in starea
  $s_1(P) = << AKN_0(P), F = \{((1, \rho, \emptyset), [send_1; recv_4; send_5; claim_6])\} >>$
- urmatoarea actiune este *send*. Regula este

$$[send]\frac{e = send_l(R_1, R_2, m) \quad (inst, [e] \cdot s) \in F}{<< AKN, F >> \rightarrow << AKN \cup \{inst(m)\}, (F - \{(inst, [e] \cdot s\})\}) \cup \{(inst, s)\} >>}$$

- cum urmatoarea actiune este $send_1(A, B, (A, B))$, se adauga cunostintei adversarului instantierea mesajului, adica tuplul (A, B), care nu schimba, de fapt, cu nimic ceea ce adversarul stia deja. In acest caz, executia [*send*] conduce, de fapt, la tranzitia:

$$<< AKN_0(P), \{((1, \rho, \emptyset), [send_1; recv_4; send_5; claim_6])\} \rightarrow$$
$$<< AKN_0(P), \{((1, \rho, \emptyset), [recv_4; send_5; claim_6])\} >>$$

Semantica operationala - rulare protocol

- Ne-au ramas ultimul pas:

$$[((2, \rho, \emptyset), create(R))]$$

- suntem in starea
  $s_2(P) =<< AKN_0(P), F = \{((1, \rho, \emptyset), [recv_4; send_5; claim_6])\} >>$
- urmatoarea actiune este *create*, aplicam direct regula.

$$<< AKN_0(P), \{((1, \rho, \emptyset), [recv_4; send_5; claim_6])\} >> \rightarrow$$
$$<< AKN_0(P), \{((1, \rho, \emptyset), [recv_4; send_5; claim_6])\} \cup$$
$$\{((2, \rho, \emptyset), [recv_1; send_2; recv_3; send_4; recv_5])\} >>$$

Semantica operationala - verificare *claim*

- trebuie sa verificam daca $\gamma := claim_6(I, R, recent - alive)$ este valid;
- din definitie, γ este valid daca si numai daca:

$$
\begin{aligned}
&\forall t \in traces(P) \;\; \forall((\theta, \rho, \sigma), \gamma) \in t, \;\; honest((\theta, \rho, \sigma)) \rightarrow \\
&\qquad \exists ev, ev' \in t \\
&\qquad\qquad actor(ev) = \rho(R) \\
&\qquad\qquad runidof(ev') = runidof(inst) \\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

Semantica operationala - verificare *claim*

- trebuie sa verificam daca $\gamma := \mathit{claim}_6(I, R, \mathit{recent} - \mathit{alive})$ este valid;
- din definitie, γ este valid daca si numai daca:

$$\begin{aligned}
&\forall t \in \mathit{traces}(P)\ \ \forall((\theta, \rho, \sigma), \gamma) \in t,\ \ \mathit{honest}((\theta, \rho, \sigma)) \rightarrow \\
&\qquad \exists \mathit{ev}, \mathit{ev}' \in t \\
&\qquad\qquad \mathit{actor}(\mathit{ev}) = \rho(R) \\
&\qquad\qquad \mathit{runidof}(\mathit{ev}') = \mathit{runidof}(\mathit{inst}) \\
&\qquad\qquad \mathit{ev}' <_t \mathit{ev} <_t (\mathit{inst}, \gamma)
\end{aligned}$$

- pentru a demonstra, $\forall \rightarrow$ fie... iar $\exists \rightarrow$ martor

Semantica operationala - verificare *claim*

- trebuie sa verificam daca $\gamma := claim_6(I, R, recent - alive)$ este valid;
- din definitie, γ este valid daca si numai daca:

$$
\begin{aligned}
&\forall t \in traces(P) \ \ \forall((\theta, \rho, \sigma), \gamma) \in t, \ \ honest((\theta, \rho, \sigma)) \rightarrow \\
&\qquad \exists ev, ev' \in t \\
&\qquad\qquad actor(ev) = \rho(R) \\
&\qquad\qquad runidof(ev') = runidof(inst) \\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

- pentru a demonstra, $\forall \rightarrow$ fie... iar $\exists \rightarrow$ martor
- pentru a demonstra $p \rightarrow q$, presupunem p adevarat

Semantica operationala - verificare *claim*

- Ce avem de demonstrat

$$
\begin{aligned}
&\gamma := claim_6(I, R, recent - alive)\\
&\forall t \in traces(P) \;\; \forall((\theta, \rho, \sigma), \gamma) \in t, \;\; honest((\theta, \rho, \sigma)) \rightarrow\\
&\qquad \exists ev, ev' \in t\\
&\qquad\qquad actor(ev) = \rho(R)\\
&\qquad\qquad runidof(ev') = runidof(inst)\\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

Semantica operationala - verificare *claim*

- Ce avem de demonstrat

$$
\begin{aligned}
&\gamma := claim_6(I, R, recent - alive)\\
&\forall t \in traces(P) \;\; \forall((\theta, \rho, \sigma), \gamma) \in t, \;\; honest((\theta, \rho, \sigma)) \rightarrow\\
&\qquad \exists ev, ev' \in t\\
&\qquad\qquad actor(ev) = \rho(R)\\
&\qquad\qquad runidof(ev') = runidof(inst)\\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

- Cum demonstram
  - Fie $t \in traces(P)$, $((\theta, \rho, \sigma), \gamma)$ un element al lui t, cu instantierea (θ, ρ, σ) onesta

Semantica operationala - verificare *claim*

- Ce avem de demonstrat

$$
\begin{aligned}
&\gamma := claim_6(I, R, recent - alive) \\
&\forall t \in traces(P) \;\; \forall((\theta, \rho, \sigma), \gamma) \in t, \;\; honest((\theta, \rho, \sigma)) \rightarrow \\
&\qquad \exists ev, ev' \in t \\
&\qquad\qquad actor(ev) = \rho(R) \\
&\qquad\qquad runidof(ev') = runidof(inst) \\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

- Cum demonstram
    - Fie $t \in traces(P)$, $((\theta, \rho, \sigma), \gamma)$ un element al lui t, cu instantierea (θ, ρ, σ) onesta
    - De acum trebuie sa demonstram constructiv - trebuie sa verificam daca putem alege doua elemente din *trace* incat sa respecte conditiile de mai sus

Semantica operationala - verificare *claim*

- Ce avem de demonstrat

$$
\begin{aligned}
&\gamma := claim_6(I, R, recent - alive)\\
&\forall t \in traces(P)\ \ \forall((\theta, \rho, \sigma), \gamma) \in t,\ \ honest((\theta, \rho, \sigma)) \rightarrow\\
&\qquad \exists ev, ev' \in t\\
&\qquad\qquad actor(ev) = \rho(R)\\
&\qquad\qquad runidof(ev') = runidof(inst)\\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

- Cum demonstram
  - Fie $t \in traces(P)$, $((\theta, \rho, \sigma), \gamma)$ un element al lui t, cu instantierea (θ, ρ, σ) onesta
  - De acum trebuie sa demonstram constructiv - trebuie sa verificam daca putem alege doua elemente din *trace* incat sa respecte conditiile de mai sus
  - Conditiile sunt: ev trebuie sa fie asociat rolului R, iar ev' sa fie asociat rolului care verifica proprietatea (au acelasi *run identifier*).

Semantica operationala - verificare *claim*

- Ce avem de demonstrat

$$
\begin{aligned}
&\gamma := claim_6(I, R, recent - alive)\\
&\forall t \in traces(P)\ \ \forall((\theta,\rho,\sigma),\gamma) \in t,\ \ honest((\theta,\rho,\sigma)) \rightarrow\\
&\qquad \exists ev, ev' \in t\\
&\qquad\qquad actor(ev) = \rho(R)\\
&\qquad\qquad runidof(ev') = runidof(inst)\\
&\qquad\qquad ev' <_t ev <_t (inst, \gamma)
\end{aligned}
$$

- Cum demonstram
  - este suficient sa gasim un eveniment al rolului R care sa fie cuprins intre alte doua evenimente ale rolului I;
  - e corecta orice alegere din *trace*, astfel incat sa avem un eveniment asociat pe rolul I, unul pe rolul R, si apoi *claim*-ul de pe I.

Exam Preview

Special Topics in Logic and Security I

Exam Paper

February 12, 2023

1 BAN Logic

Subject 1 (7 points) *Consider that S is an agent, X is a message and k a symmetric encryption key. Having $\varphi :=$* [redacted] *and $\psi :=$* [redacted] *give an example of a set of BAN-formulas Γ such that ψ can be inferred from $\Gamma \cup \{\varphi\}$. Write the proof.*

Consider the following security protocol **P**:

[redacted]

Subject 2 (7 points) *Formalize this protocol in BAN Logic. You have to formalize the exchange of messages and then add all the assumptions you consider relevant in the analysis, bearing in mind that the goal is to prove a mutual authentication between the two agents.*

Subject 3 (7 points) *Prove that agents I and R are mutual authenticated through* [redacted] *$I \mid\equiv R \mid\equiv$* [redacted] *and $R \mid\equiv I \mid\equiv$* [redacted]

2 Operational Semantics

Subject 4 (6 points) *Describe the roles in this security protocol* **P***, $P(I)$ and $P(R)$, by specifying the initial knowledge of each agent and the associated sequence of events.*

Subject 5 (6 points) *Give an example of an honest trace in this protocol.*

Subject 6 (6 points) *Perform three steps in the operational semantics, starting with the initial state.*

Subject 7 (6 points) *Prove that the claim* [redacted] *holds.*

Succes la examen!