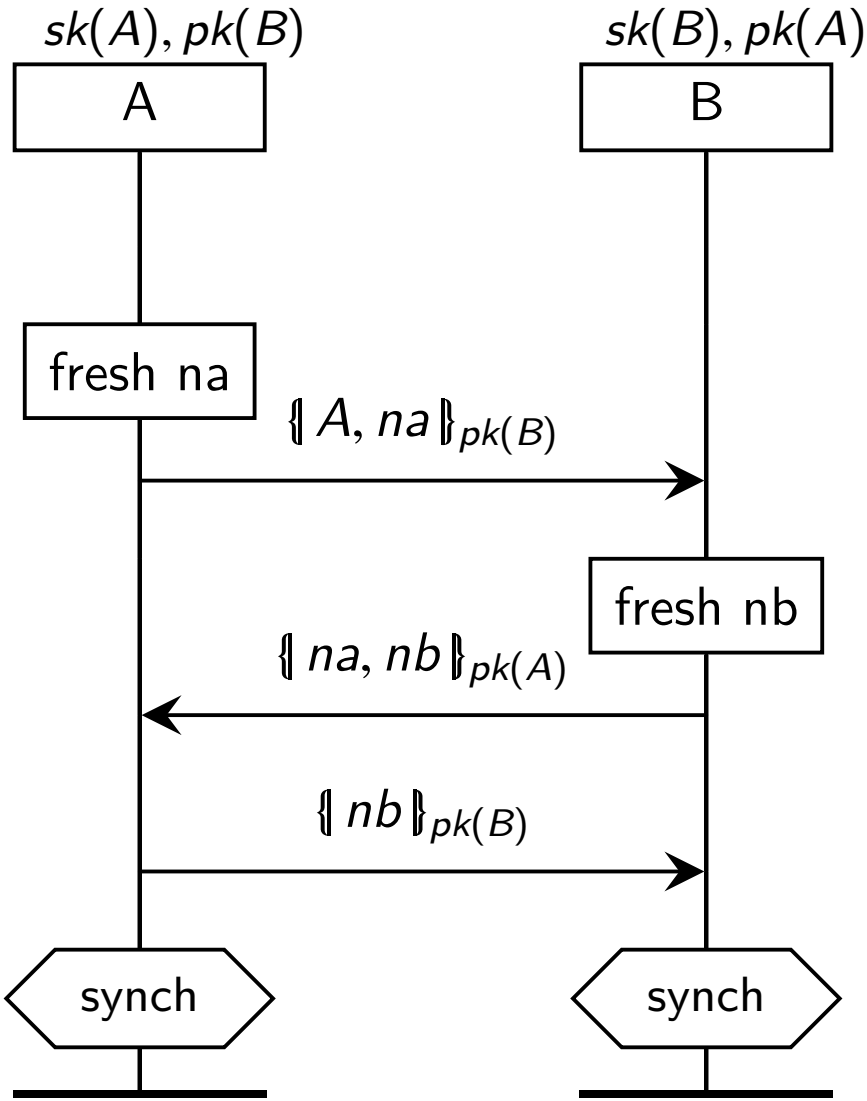
Topici speciale în logică și securitate I

Protocoale de securitate II

Ioana Leuștean

Master anul II, Sem. I, 2019-2020

Protocolul Needham-Schroeder cu cheie publică (1978)



Protocolul Needham-Schroeder

Needham-Scroeder Public Key Protocol (NSPK)

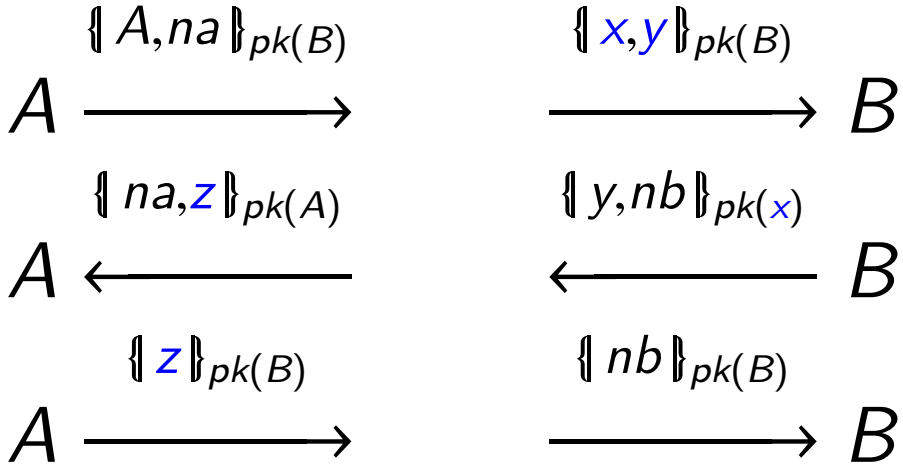
$$
\begin{aligned}
A \longrightarrow B: &\quad \{\!|\, A, na \,|\!\}_{pk(B)} \\
B \longrightarrow A: &\quad \{\!|\, na, nb \,|\!\}_{pk(A)} \\
A \longrightarrow B: &\quad \{\!|\, nb \,|\!\}_{pk(B)}
\end{aligned}
$$

Scopul este autentificarea mutuală a celor doi participanți. După execuția cu succes a protocolului:

- Alice și Bob sunt siguri că au comunicat unul cu celălalt
- toate mesajele primite au fost trimise de partener și toate mesajele trimise au fost primite de partener
- *na* și *nb* sunt cunoscute numai de Alice și Bob

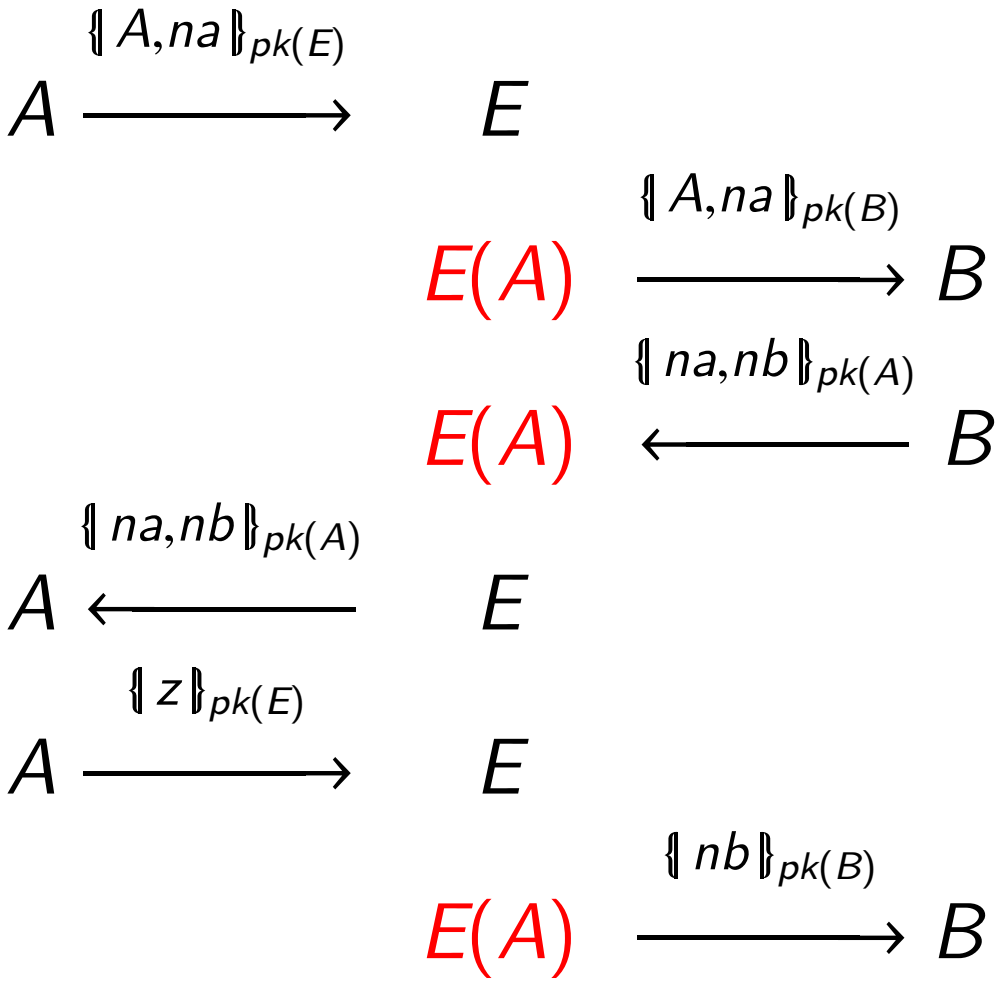
Protocolul Needham-Schroeder

Needham-Scroeder Public Key Protocol (NSPK)



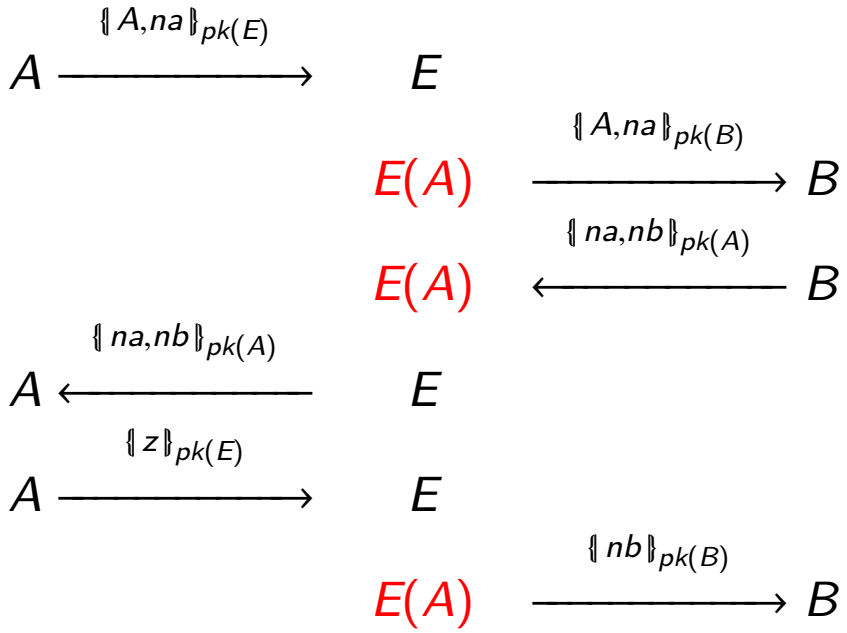
Protocolul Needham-Schroeder

Atacul "man in the middle" asupra NSPK



Protocolul Needham-Schroeder

Atacul "man in the middle" asupra NSPK



- A inițiază protocolul cu E
- E este un agent corupt, care îl impersonează pe A în comunicarea cu B
- este violată proprietatea de sincronizare *din punctul de vedere al lui B*: B crede că stabilit o sesiune de comunicare sigură cu A, când de fapt comunică cu E

Protocolul Needham-Schroeder-Lowe

Prevenirea atacului "man in the middle" [Lowe 1996]

$$
\begin{aligned}
A \longrightarrow B: &\quad \{\!|\, A, na \,|\!\}_{pk(B)} \\
B \longrightarrow A: &\quad \{\!|\, na, nb, B \,|\!\}_{pk(A)} \\
A \longrightarrow B: &\quad \{\!|\, nb \,|\!\}_{pk(B)}
\end{aligned}
$$

- A se așteaptă să primească $\{\!|\, na, nb, E \,|\!\}_{pk(A)}$, dar E nu are cum să afle nb.
- E poate trimite $\{\!|\, na, nb, B \,|\!\}_{pk(A)}$, dar acest mesaj nu corepunde ci forma pe care așteaptă A să o primească.

Analiza formală a protocoalelor

- Am văzut cum analizăm *informal* protocoalele de securitate.
- Scopul *analizei formale* este acela de a defini un model al protocolului și de a-i analiza proprietățile într-o teorie matematică consistentă.
- Protocoalele reale sunt abstractizate, obținându-se modele mai simple. De exemplu protocolul (real) Kerberos are la baza protocolul (academic) Needham-Scroeder.
- Tipuri de modele formale
  - bazate pe logică epistemică, de exemplu BAN logic
  - bazate pe model-checking (tool-uri: Proverif, AVISPA, Scyther, Tamarin, ...)
  - ...

  Noi vom prezenta abordarea din [OSVSP]:
  Cas Cremers and Sjouke Mauw. Operational Semantics and Verification of Security Protocols. Springer, 2012.

Analiza formală protocoalelor

- Componentele analizei formale
  - specificarea protocolului,
  - modelarea agenților,
  - modelarea comunicării,
  - modelarea adversarilor,
  - modelarea proprietăților de securitate.

- Limbajul formal va fi cel al unei *logici multi-sortate de ordinul I*
  - rolurile și mesajele sunt reprezentate prin termeni,
  - specificarea unui protocol este o mulțime de roluri,
  - cunoștințele adversarului sunt determinate printr-un sistem de deducție,
  - execuția protocolului se definește prin mulțimea urmelor (trace) unui sistem de tranziții etichetat,
  - proprietățile de securitate pot fi formalizate și demonstrate.

Variabile și funcții

- Variabilele sunt de mai multe feluri
  - $Var : V, X, Y, Z, \ldots$ (folosite pentru mesaje)
  - $Fresh : ni, nr, sessionkey, \ldots$ (folosite pentru nonce-uri si alte valori care sunt unice pe sesiune)
  - $Role : i, r, s, \ldots$ (folosite pentru roluri: inițiator, respondent, server, etc)

Variabile și funcții

- Variabilele sunt de mai multe feluri
  - $Var : V, X, Y, Z, \ldots$ (folosite pentru mesaje)
  - $Fresh : ni, nr, sessionkey, \ldots$ (folosite pentru nonce-uri si alte valori care sunt unice pe sesiune)
  - $Role : i, r, s, \ldots$ (folosite pentru roluri: inițiator, respondent, server, etc)

- *Func* mulțime de simboluri de funcții
  - fiecare simbol are o aritate (nr. de argumente) fixată
  - funcțiile de aritate 0 desemnează constantele; de exemplu numerele naturale sunt văzute ca funții constante
  - exemplu de funție: h funcție hash

Termeni: *RoleTerm*

$$
\begin{aligned}
RoleTerm \quad ::= \quad & Var \mid Fresh \mid Role \\
& \mid Func(RoleTerm^*) \\
& \mid (RoleTerm, RoleTerm) \\
& \mid \{\!|\, RoleTerm \,|\!\}_{RoleTerm} \\
& \mid sk(RoleTerm) \mid pk(RoleTerm) \mid k(RoleTerm, RoleTerm)
\end{aligned}
$$

Termenii din *RoleTerm* sunt folosiți pentru a specifica:

mesaje, nonce-uri, roluri, chei pe termen lung

Vom nota $\{\!|\, t_1, t_2 \,|\!\}$ în loc de $\{\!|\, (t_1, t_2) \,|\!\}$.

Funcție inversă. Criptare

$$^{-1} : RoleTerm \to RoleTerm$$

- pentru orice $rt \in RoleTerm$ definim *termenul invers* $rt^{-1} \in RoleTerm$ astfel încât:
  - $pk(rt)^{-1} = sk(rt)$
  - $sk(rt)^{-1} = pk(rt)$

- $rt^{-1} = rt$ pentru orice termen $rt \in RoleTerm$ care nu este de forma $sk(t)$ sau $pk(t)$ cu $t \in RoleTerm$

Vom presupune că sk și pk definesc chei asimetrice, iar k definește o cheie simetrică.

Exemplu: criptare și semnare

Presupunem că $m \in RoleTerm$ reprezintă un mesaj și că $R \in Role$

- $\{\!|\, m \,|\!\}_{pk(R)}$ reprezintă criptarea lui m cu cheia publică a lui R

Exemplu: criptare și semnare

Presupunem că $m \in RoleTerm$ reprezintă un mesaj și că $R \in Role$

- $\{\!|\, m \,|\!\}_{pk(R)}$ reprezintă criptarea lui m cu cheia publică a lui R
- pentru a semna mesajul m
  - se poate folosi cheia secretă $\{\!|\, m \,|\!\}_{sk(R)}$

Exemplu: criptare și semnare

Presupunem că $m \in RoleTerm$ reprezintă un mesaj și că $R \in Role$

- $\{\!|\, m \,|\!\}_{pk(R)}$ reprezintă criptarea lui m cu cheia publică a lui R
- pentru a semna mesajul m
  - se poate folosi cheia secretă $\{\!|\, m \,|\!\}_{sk(R)}$
  - se poate folosi o funcție hash $h \in Func$, mesajul semnat fiind $(m, \{\!|\, h(m) \,|\!\}_{sk(R)})$

Sistem de deducție pe termeni

$$\vdash \subseteq \mathcal{P}(RoleTerm) \times RoleTerm$$

$M \vdash t$ modelează ceea ce se pate deduce știind M

Sistem de deducție pe termeni

$$\vdash \subseteq \mathcal{P}(RoleTerm) \times RoleTerm$$

$M \vdash t$ modelează ceea ce se pate deduce știind M

$\vdash$ este cea mai mică relație care satisface următoarele proprietăți:

dacă	$t \in M$	atunci	$M \vdash t$
dacă	$M \vdash t_1$ și $M \vdash t_2$	atunci	$M \vdash (t_1, t_2)$
dacă	$M \vdash (t_1, t_2)$	atunci	$M \vdash t1$ și $M \vdash t_2$
dacă	$M \vdash t$ și $M \vdash k$	atunci	$M \vdash \{\!\!\{ t \}\!\!\}_k$
dacă	$M \vdash \{\!\!\{ t \}\!\!\}_k$ și $M \vdash k^{-1}$	atunci	$M \vdash t$
dacă	$M \vdash t_1$ și ... și $M \vdash t_n$	atunci	$M \vdash f(t_1, \ldots, t_n)$

unde $f \in Func$ are aritatea n.

Notație: $Cons(M) = \{t \in RoleTerm \mid M \vdash t\}$

Sistemul de deducție

Observăm că acest sistem respectă regulile criptografiei perfecte.

- Presupunem că $m, k \in RoleTerm$ și că $\{\!|\, m \,|\!\}_k \not\vdash k^{-1}$.
  In aceste condiții, folosind sistemul de mai sus, din $\{\!|\, m \,|\!\}_k$ nu se pot deduce m sau k.

- Presupunem că $m \in RoleTerm$ și $f \in Func$. Folosind sistemul de mai sus, din $f(m)$ nu se poate deduce m.

Exercițiu: Determinați

$Cons(\{\!|\, m \,|\!\}_k) = \{t \in RoleTerm \mid \{\!|\, m \,|\!\}_k \vdash t\}$

Atenție! se poate folosi teorema de punct fix.

Exemplu: sistemul de deducție

Determinați $Cons(M)$ pentru

$$M = \{\!|\, m \,|\!\}_k, \{\!|\, k^{-1} \,|\!\}_{pk(b)}, \{\!|\, h(m) \,|\!\}_m, sk(b)\}$$

Termeni: *RoleEvent*

- Considerăm două mulțimi disjuncte:
  - $Label : 1, 2, 3, \ldots$ (sunt etichete)
  - $Claim : secret, alive, \ldots$ (descriu proprietățile de securitate)

$$
\begin{aligned}
RoleEvent_R \quad ::= \quad & send_{Label}(R, Role, RoleTerm) \\
& | \, recv_{Label}(Role, R, RoleTerm) \\
& | \, claim_{Label}(R, Claim[, RoleTerm])
\end{aligned}
$$

Termeni: *RoleEvent*

- Considerăm două mulțimi disjuncte:
  - $Label$: $1, 2, 3, \ldots$ (sunt etichete)
  - $Claim$: $secret, alive, \ldots$ (descriu proprietățile de securitate)

$$
\begin{aligned}
RoleEvent_R \quad ::= \quad & send_{Label}(R, Role, RoleTerm) \\
& | \, recv_{Label}(Role, R, RoleTerm) \\
& | \, claim_{Label}(R, Claim[, RoleTerm])
\end{aligned}
$$

$$
RoleEvent = \bigcup_{R \in Role} RoleEvent_R
$$

Termeni: *RoleEvent*

$$
\begin{aligned}
RoleEvent_R \quad ::= \quad & send_{Label}(R, Role, RoleTerm) \\
& | \, recv_{Label}(Role, R, RoleTerm) \\
& | \, claim_{Label}(R, Claim[, RoleTerm])
\end{aligned}
$$

- $send_l(R, R', rt)$ trimiterea de către R a mesajului rt cu destinația R',

Termeni: *RoleEvent*

$$
\begin{aligned}
RoleEvent_R \quad ::= \quad & send_{Label}(R, Role, RoleTerm) \\
& | \, recv_{Label}(Role, R, RoleTerm) \\
& | \, claim_{Label}(R, Claim[, RoleTerm])
\end{aligned}
$$

- $send_l(R, R', rt)$ trimiterea de către R a mesajului rt cu destinația R',
- $recv_l(R', R, rt)$ primierea mesajului rt de către R, aparent de la R',

Termeni: *RoleEvent*

$$
\begin{aligned}
RoleEvent_R \quad ::= \quad & send_{Label}(R, Role, RoleTerm) \\
& | \, recv_{Label}(Role, R, RoleTerm) \\
& | \, claim_{Label}(R, Claim[, RoleTerm])
\end{aligned}
$$

- $send_l(R, R', rt)$ trimiterea de către R a mesajului rt cu destinația R',
- $recv_l(R', R, rt)$ primierea mesajului rt de către R, aparent de la R',
- $claim_l(R, c, rt)$ proprietatea de securitate care se dorește a fi satisfăcută după execuția rolului R.

Termeni: *RoleEvent*

$$
\begin{aligned}
RoleEvent_R \quad ::= \quad & send_{Label}(R, Role, RoleTerm) \\
& | \, recv_{Label}(Role, R, RoleTerm) \\
& | \, claim_{Label}(R, Claim[, RoleTerm])
\end{aligned}
$$

- $send_l(R, R', rt)$ trimiterea de către R a mesajului rt cu destinația R',
- $recv_l(R', R, rt)$ primierea mesajului rt de către R, aparent de la R',
- $claim_l(R, c, rt)$ proprietatea de securitate care se dorește a fi satisfăcută după execuția rolului R.
- Etichetele identifică fiecare eveniment și stabilesc corespondența între evenimente *send* și *recv*.

Specificarea unui protocol

Informal, pentru a descrie un protocol trebuie să precizăm comportamentul fiecărui rol în parte.

Specificarea unui protocol

Informal, pentru a descrie un protocol trebuie să precizăm comportamentul fiecărui rol în parte.

Fie P un protocol si R un rol în P.

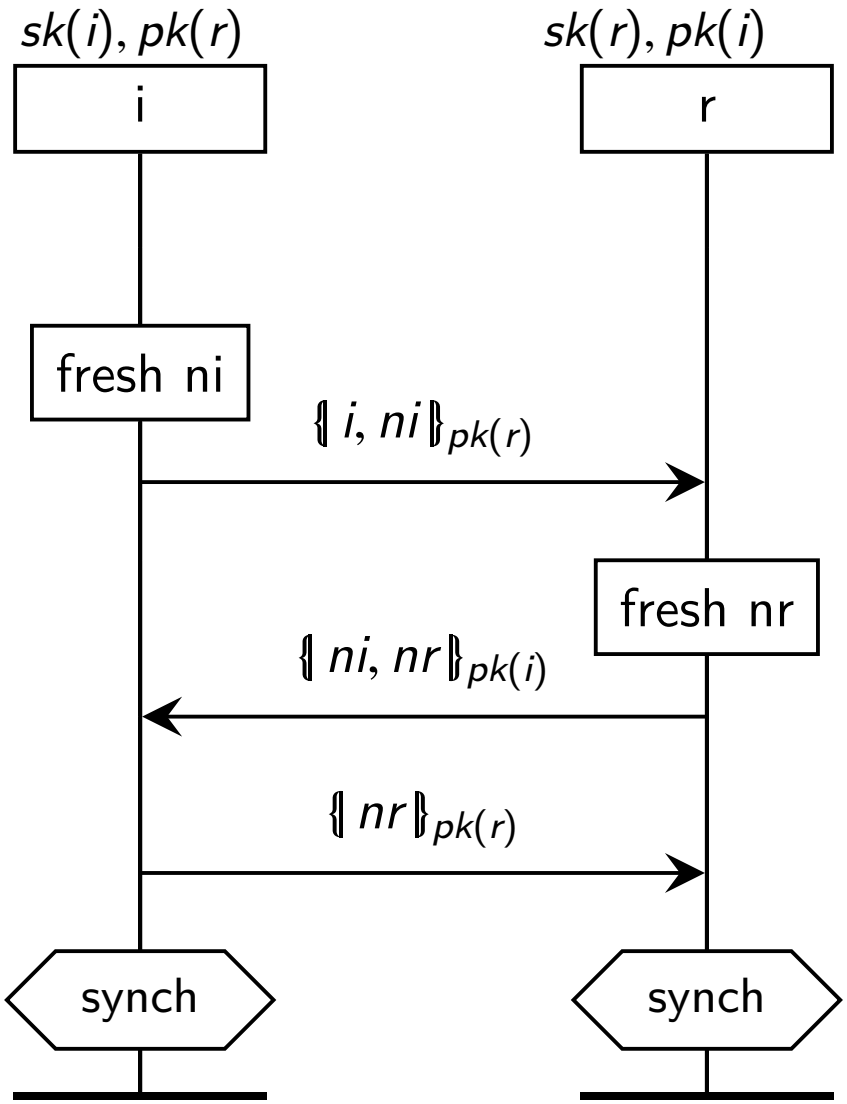
Specificarea lui $P(R)$ este o pereche din $\mathcal{P}(RoleTerm) \times RoleEvent_R^*$.

Exemplul: Specificarea rolului inițiator în NSPK

$$NS(i) = (\{i, r, ni, sk(i), pk(i), pk(r)\},$$
$$[send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}),$$
$$recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}),$$
$$send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}),$$
$$claim_4(i, synch)])$$

Protocolul Needham-Schroeder

$$
\begin{aligned}
NS(i) = \quad & (\{i, r, ni, sk(i), pk(i), pk(r)\}, \\
& [send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}), \\
& recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}), \\
& send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}), \\
& claim_4(i, synch)])
\end{aligned}
$$



Specificarea unui rol

$$NS(i) = (\{i, r, ni, sk(i), pk(i), pk(r)\},$$
$$[send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}),$$
$$recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}),$$
$$send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}),$$
$$claim_4(i, synch)])$$

Specificarea unui rol

$$NS(i) = (\{i, r, ni, sk(i), pk(i), pk(r)\},$$
$$[send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}),$$
$$recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}),$$
$$send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}),$$
$$claim_4(i, synch)])$$

$P(R) = (KN_0(R), s)$ unde $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ și $s \in RoleEvent_R^*$

Specificarea unui rol

$$NS(i) = (\{i, r, ni, sk(i), pk(i), pk(r)\},$$
$$[send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}),$$
$$recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}),$$
$$send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}),$$
$$claim_4(i, synch)])$$

$P(R) = (KN_0(R), s)$ unde $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ și $s \in RoleEvent_R^*$

- $KN_0(i) = \{i, r, ni, sk(i), pk(i), sk(r)\}$ reprezintă cunoștințele inițiale ale inițiatorului

Specificarea unui rol

$$NS(i) = (\{i, r, ni, sk(i), pk(i), pk(r)\},$$
$$[send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}),$$
$$recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}),$$
$$send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}),$$
$$claim_4(i, synch)])$$

$P(R) = (KN_0(R), s)$ unde $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ și $s \in RoleEvent_R^*$

- $KN_0(i) = \{i, r, ni, sk(i), pk(i), sk(r)\}$ reprezintă cunoștințele inițiale ale inițiatorului
- $s = [send_1(\cdots), \ldots, claim_4(\cdots)]$ reprezintă secvența de evenimente pe care trebuie să o execute inițiatorul

Specificarea unui rol

$P(R) = (KN_0(R), s)$

- $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ reprezintă cunoștințele inițiale ale lui R
- $s \in RoleEvent_R^*$ reprezintă secvența de evenimente pe care o execută R

Se observă că:

- cunoașterea inițială conține termeni fără valori din Var (variabilele folosite pentru mesaje); va fi utilizată ulterior pentru a deriva cunoașterea adversarului;

Specificarea unui rol

$P(R) = (KN_0(R), s)$

- $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ reprezintă cunoștințele inițiale ale lui R
- $s \in RoleEvent_R^*$ reprezintă secvența de evenimente pe care o execută R

Se observă că:

- cunoașterea inițială conține termeni fără valori din *Var* (variabilele folosite pentru mesaje); va fi utilizată ulterior pentru a deriva cunoașterea adversarului;
- datorită *etichetării*, fiecare termen eveniment (din *RoleEvent*) e unic într-o specificare a unui protocol;

Specificarea unui rol

$P(R) = (KN_0(R), s)$

- $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ reprezintă cunoștințele inițiale ale lui R
- $s \in RoleEvent_R^*$ reprezintă secveța de evenimente pe care o execută R

Se observă că:

- cunoașterea inițială conține termeni fără valori din *Var* (variabilele folosite pentru mesaje); va fi utilizată ulterior pentru a deriva cunoașterea adversarului;
- datorită *etichetării*, fiecare termen eveniment (din *RoleEvent*) e unic într-o specificare a unui protocol;
- secvența de evenimente conține varaibile pentru mesaje din *Var*, acestea vor fi instanțiate cu mesaje concrete;

Specificarea unui rol

$P(R) = (KN_0(R), s)$

- $KN_0(R) \subseteq \mathcal{P}(RoleTerm)$ reprezintă cunoștințele inițiale ale lui R
- $s \in RoleEvent_R^*$ reprezintă secveța de evenimente pe care o execută R

Se observă că:

- cunoașterea inițială conține termeni fără valori din *Var* (variabilele folosite pentru mesaje); va fi utilizată ulterior pentru a deriva cunoașterea adversarului;
- datorită *etichetării*, fiecare termen eveniment (din *RoleEvent*) e unic într-o specificare a unui protocol;
- secvența de evenimente conține varaibile pentru mesaje din *Var*, acestea vor fi instanțiate cu mesaje concrete;în specificarea unui rol, aceste variabile trebuie să apară prima dată într-un mesaj de tip *receive* și într-o *poziție accesibilă*

Accesibilitate și secvențe bine formate

- Relația de accesibilitate $\sqsubseteq_{acc} \subseteq \mathit{RoleTerm}^2$ este închiderea reflexivă și tranzitivă a relației definite de:

$$t_1 \sqsubseteq_{acc} (t_1, t_2),\ t_1 \sqsubseteq_{acc} (t_1, t_2),\ t_1 \sqsubseteq_{acc} \{\!|\, t_1 \,|\!\}_{t_2}$$

Accesibilitate și secvențe bine formate

- Relația de accesibilitate $\sqsubseteq_{acc} \subseteq RoleTerm^2$ este închiderea reflexivă și tranzitivă a relației definite de:

$$t_1 \sqsubseteq_{acc} (t_1, t_2),\ t_1 \sqsubseteq_{acc} (t_1, t_2),\ t_1 \sqsubseteq_{acc} \{\!|\, t_1 \,|\!\}_{t_2}$$

- Pentru $\rho \in RoleTerm^*$ spunem că este *bine formată* ddacă

$$\forall\, V \in vars(\rho)\ \exists\, \rho', l, R, R', rt, \rho''$$
$$(\rho = \rho' \cdot [recv_l(R, R', rt)] \cdot \rho'') \wedge V \notin vars(\rho') \wedge V \sqsubseteq_{acc} rt$$

unde $vars(\rho) \subseteq Var$ este mulțimea de variabile care apare în secvența de evenimente ρ.

Definim predicatul $wellformed(\rho) =$ "ρ este bine formată"

Specificarea unui protocol

- Specificarea rolurilor

$$RoleSpec = \{(kn, s) \mid kn \in \mathcal{P}(RoleTerm) \ \wedge \ \forall rt \in kn\, vars(rt) = \emptyset \\ \wedge\, s \in RoleEvent^* \ \wedge \ \ wellformed(s)\}$$

- Specificarea unui protocol

$$Protocol = Role \rightharpoonup RoleSpec$$

$P(R)$ este specificarea rolului R pentru orice $P \in Protocol$ și $R \in Role$.

Protocolul Needham-Schroeder

$$
\begin{aligned}
NS(i) = \quad & (\{i, r, ni, sk(i), pk(i), pk(r)\}, \\
& [send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}), \\
& recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}), \\
& send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}), \\
& claim_4(i, synch)])
\end{aligned}
\qquad
\begin{aligned}
NS(r) = \quad & (\{i, r, nr, sk(r), pk(r), pk(i)\}, \\
& [recv_1(i, r, \{\!|\, W, i \,|\!\}_{pk(r)}), \\
& send_2(r, i, \{\!|\, W, nr \,|\!\}_{pk(i)}), \\
& recv_3(i, r, \{\!|\, nr \,|\!\}_{pk(r)}), \\
& claim_5(r, synch)])
\end{aligned}
$$

Protocolul Needham-Schroeder

$$NS(i) = (\{i, r, ni, sk(i), pk(i), pk(r)\},$$
$$[send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}),$$
$$recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}),$$
$$send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}),$$
$$claim_4(i, synch)])$$

$$NS(r) = (\{i, r, nr, sk(r), pk(r), pk(i)\},$$
$$[recv_1(i, r, \{\!|\, W, i \,|\!\}_{pk(r)}),$$
$$send_2(r, i, \{\!|\, W, nr \,|\!\}_{pk(i)}),$$
$$recv_3(i, r, \{\!|\, nr \,|\!\}_{pk(r)}),$$
$$claim_5(r, synch)])$$

- Limbajul asociat protocolului Needham-Schroeder este

  $Role = \{i, r\}$, $Fresh = \{ni, nr\}$, $Func = \emptyset$, $Label = \{1, 2, 3, 4, 5\}$, $Var = \{V, W\}$

Protocolul Needham-Schroeder

$$
\begin{aligned}
NS(i) = \quad & (\{i, r, ni, sk(i), pk(i), pk(r)\}, \\
& [send_1(i, r, \{\!|\, ni, i \,|\!\}_{pk(r)}), \\
& recv_2(r, i, \{\!|\, ni, V \,|\!\}_{pk(i)}), \\
& send_3(i, r, \{\!|\, V \,|\!\}_{pk(r)}), \\
& claim_4(i, synch)])
\end{aligned}
\qquad
\begin{aligned}
NS(r) = \quad & (\{i, r, nr, sk(r), pk(r), pk(i)\}, \\
& [recv_1(i, r, \{\!|\, W, i \,|\!\}_{pk(r)}), \\
& send_2(r, i, \{\!|\, W, nr \,|\!\}_{pk(i)}), \\
& recv_3(i, r, \{\!|\, nr \,|\!\}_{pk(r)}), \\
& claim_5(r, synch)])
\end{aligned}
$$

- Limbajul asociat protocolului Needham-Schroeder este

  $Role = \{i, r\}$, $Fresh = \{ni, nr\}$, $Func = \emptyset$, $Label = \{1, 2, 3, 4, 5\}$, $Var = \{V, W\}$

Exercițiu: modelați celelalte protocoale prezentate la curs!