Weryfikacja protokołów bezpieczeństwa z wykorzystaniem automatów probabilistycznych

Olga Siedlecka-Lamch

Instytut Informatyki Teoretycznej i Stosowanej
Politechniki Częstochowskiej

30 marca 2015 r.

Znaczenie protokołów bezpieczeństwa

- Protokoły bezpieczeństwa - kluczowa rola dla współłczesnych systemów informatycznych
- Wykorzystywane w wielu dziedzinach
- Błędy struktury, działania, zabezpieczeń
- Znaczenie specyfikacji i weryfikacji
- Potrzeba pełnego formalnego opisu
- Tempo rozwoju rynku IT stawia nowe wymagania

Veni, vidi, ... I

- Zbudowano pełen formalizm
- Zbudowano strukturę obliczeniową
- Wykorzystano do weryfikacji sieć automatów
- Wykorzystano do weryfikacji łańcuchy stanów
- Publikacje:


Kurkowski, M., Srebrny, M.: A Quantifier-free First-order Knowledge Logic of Authentication, Fund. Inform., vol. 72, pp. 263-282, IOS Press 2006


Kurkowski, M., Penczek, W.: Verifying Security Protocols Modeled by Networks of Automata, Fund. Inform., Vol. 79 (3-4), pp. 453-471, IOS Press 2007


Kurkowski, M., Penczek, W.: Verifying Timed Security Protocols via Translation to Timed Automata, Fund. Inform., vol. 93 (1-3), pp. 245-259, IOS Press 2009


Kurkowski M., Penczek W.: Applying Timed Automata to Model Checking of Security Protocols, in ed. J. Wang, Handbook of Finite State Based Models and Applications, pp. 223–254, CRC Press, Boca Raton, USA, 2012

Veni, vidi, ... II


Kurkowski M., Siedlecka-Lamch O., Piech H., A New Effective Approach for Modelling and Verification of Security Protocols, in Proceedings of 21th international Workshop on Concurrency, Specification and Programming (CS&P 2012) Humboldt University Press, Berlin, Germany, 191-202, 2012


Kurkowski M., Siedlecka-Lamch O., Dudek P., Using Backward Induction Techniques in (Timed) Security Protocols Verification, in Proceedings of 12th International Conference CISIM 2013, Krakow, Poland, Lecture Notes in Computer Science vol. 8104, 265 - 276, 2013


Kurkowski M., Formalne metody weryfikacji własności protokołów zabezpieczających w sieciach komputerowych, s. 208, wyd. Exit, Warszawa, 2013


Kurkowski M., Siedlecka-Lamch O., Szymoniak S., Piech H., Parallel Bounded Model Checking of Security Protocols, in Proc. of PPAM'13, vol. 8384 of LNCS, Springer Verlag, 224 - 234, 2014


Kurkowski M., Grosser A., Piątkowski J., Szymoniak S., ProToc - an universal language for security protocols specification, Advances in Intelligent Systems and Computing, vol. 342, pp 237-248, Springer Verlag, 2015

- Weryfikacja odbyła się dla określonego założenia...

Nowe wyzwania

- Brak założenia bezpiecznego szyfrowania
- Bezpieczeństwo "szyte na miarę"
- Myślenie probabilistyczne

Światowi liderzy

- PRISM
- UPPAAL
- AVISPA

PRISM

PRISM jest probabilistycznym narzędziem do formalnego modelowania i analizy systemów wykazujących losowe zachowanie. Jest wykorzystywany do badania protokołów komunikacyjnych, bezpieczeństwa, losowych algorytmów rozproszonych, systemów biologicznych, gier itd.

PRISM wykorzystuje różne modele probabilistyczne:

- czasowe łańcuchy Markova
- procesy decyzyjne Markova
- automaty probabilistyczne
- czasowe automaty probabilistyczne

PRISM wykorzystuje własny język do opisu modelu.

Kwiatkowska M., Norman G. and Parker D.. PRISM 4.0: Verification of Probabilistic Real-time Systems. In Proc. 23rd International Conference on Computer Aided Verification (CAV'11), volume 6806 of LNCS, pages 585-591, Springer, 2011

UPPAAL

UPPAAL

to zintegrowane środowisko do modelowania, walidacji i weryfikacji systemów czasu rzeczywistego z wykorzystaniem sieci automatów czasowych, rozszerzone z typów danych (ograniczonych liczb całkowitych, macierze, itp). Jest odpowiedni dla systemów, które mogą być modelowane jako zbiór procesów niedeterministycznych ze skończoną strukturą, o rzeczywistych wartościach zegarów, komunikujących się za pośrednictwem kanałów lub wspólnych zmiennych.

Uppaal składa się z:

- języka specyfikacji problemu
- symulatora
- weryfikatora

Behrmann G., David A., Larsen K. G., A Tutorial on Uppaal, In proceedings of the 4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM-RT'04). LNCS 3185

AVISPA

(Automated Validation of Internet Security Protocols and Applications) narzędzia do analizy dużych protokołów i technologii zabezpieczających stosowanych w Internecie.

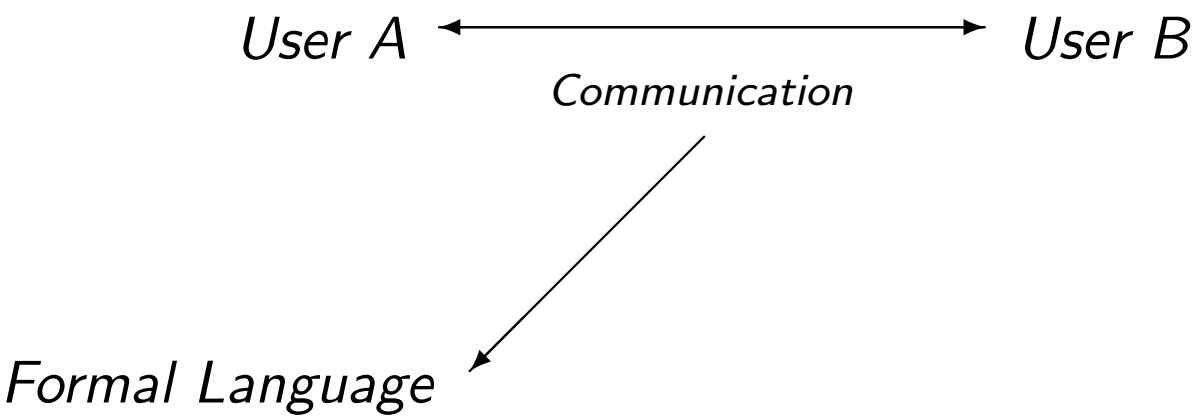
Avispa udostępnia:

- język specyfikacji HLPSL
- cztery moduły szukający ataku na cztery różne sposoby

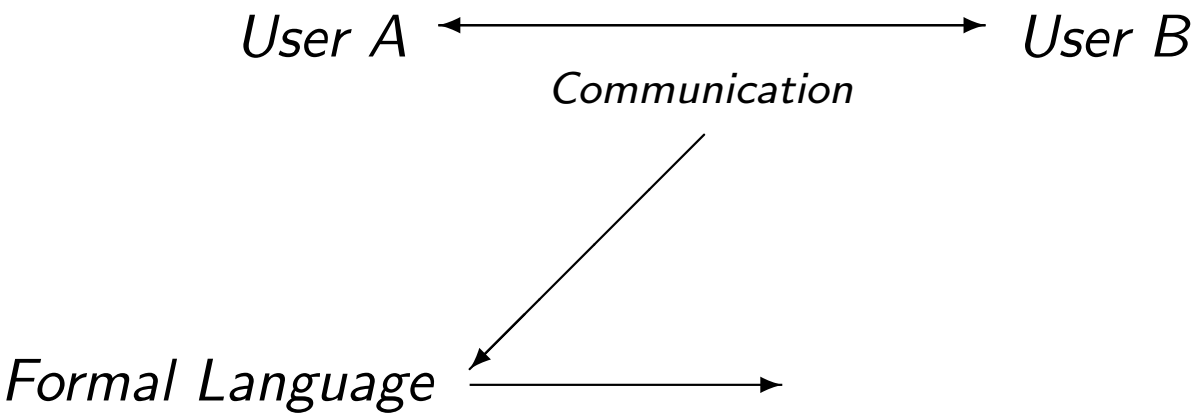
A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, P.C. Heám, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Vigano, L. Vigneron, The Avispa Tool for the automated validation of internet security protocols and applications, in proceedings of CAV 2005, Computer Aided Verification, LNCS 3576, Springer Verlag, 2005

Nasze dotychczasowe podejście

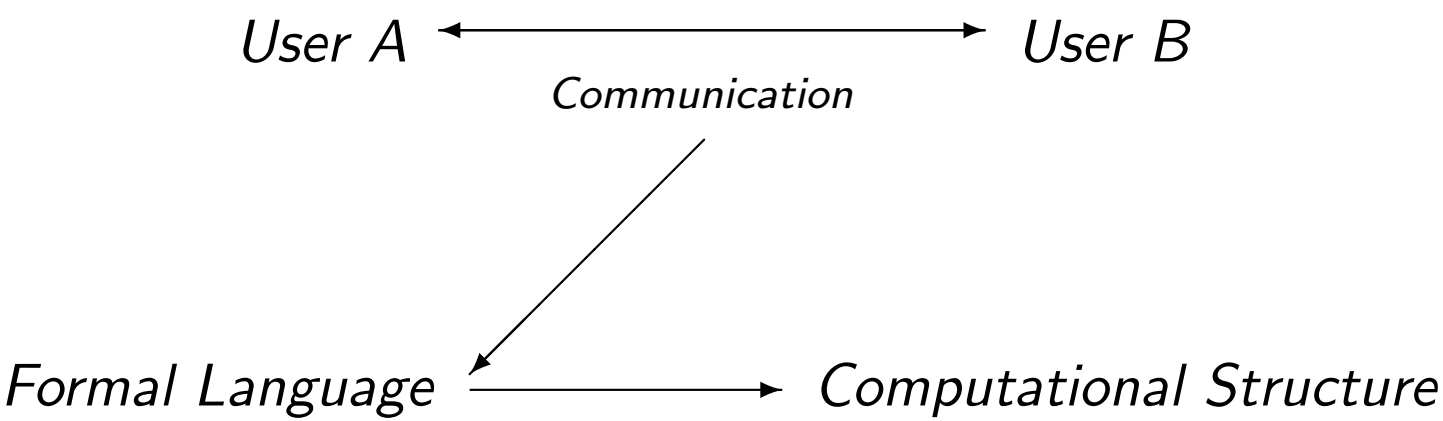
Nasze dotychczasowe podejście



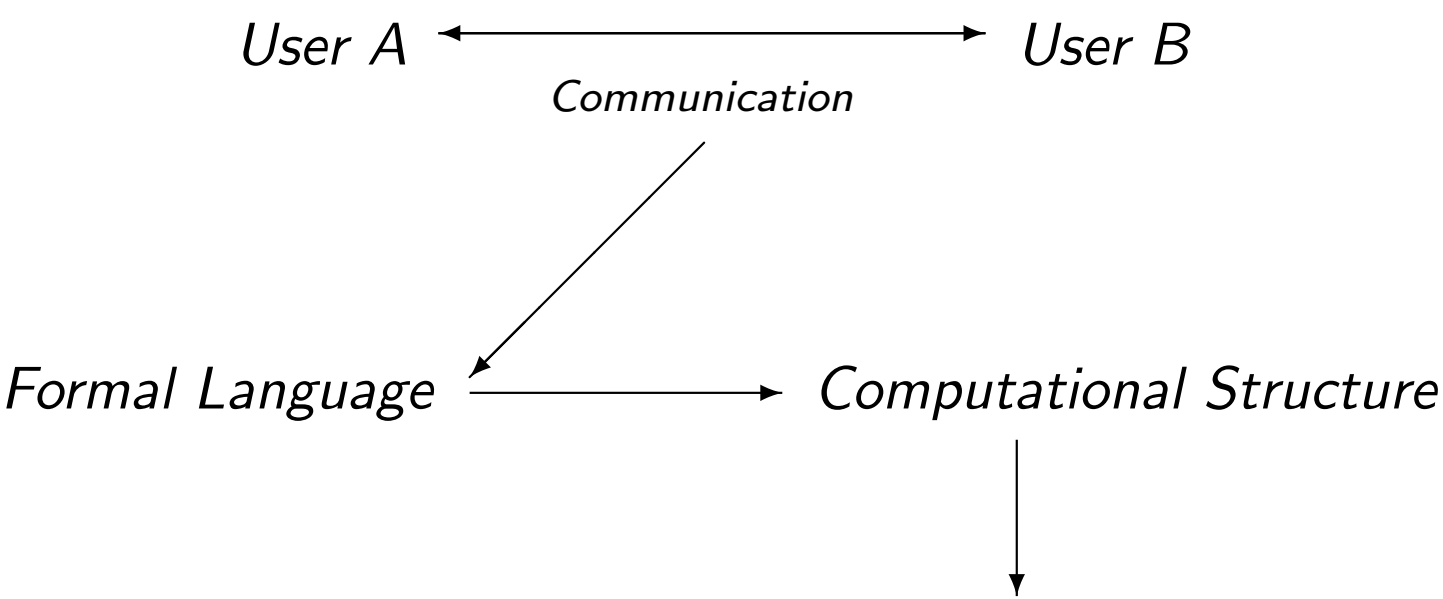
Nasze dotychczasowe podejście



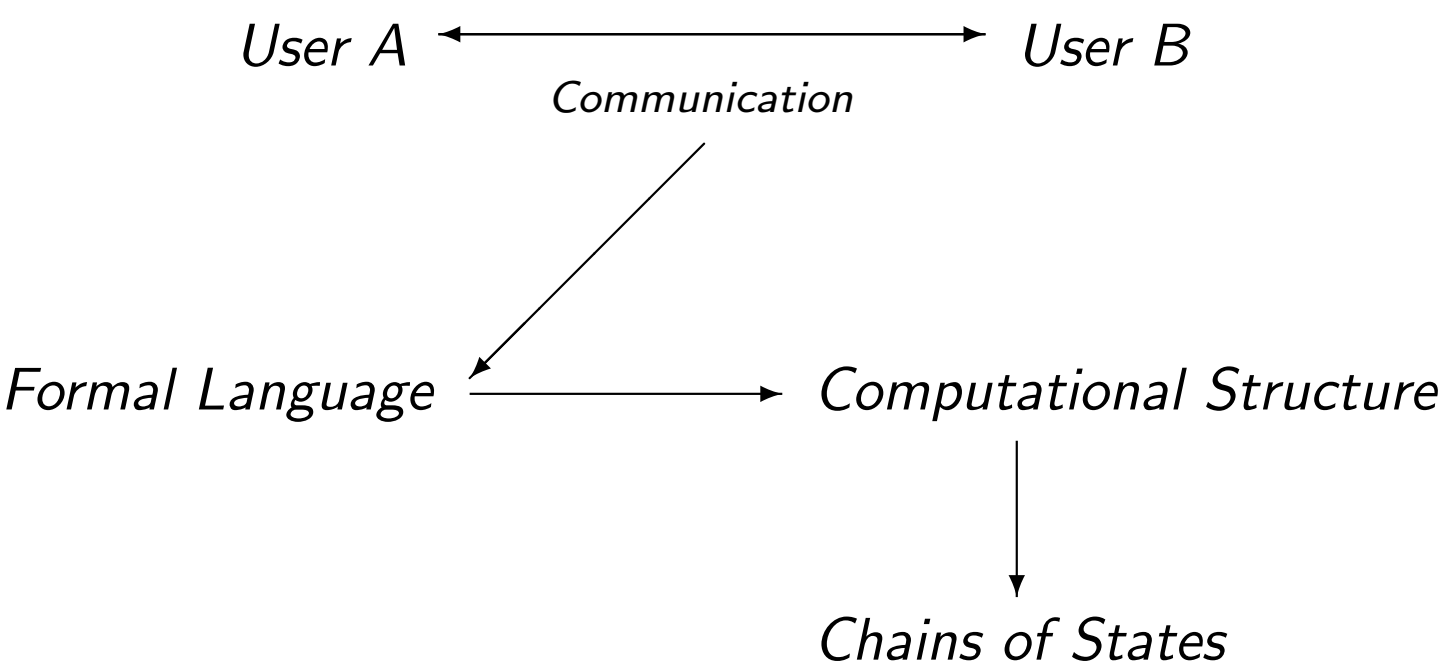
Nasze dotychczasowe podejście



Nasze dotychczasowe podejście



Nasze dotychczasowe podejście



Protokół Needhama-Schroedera i atak Lowe-a

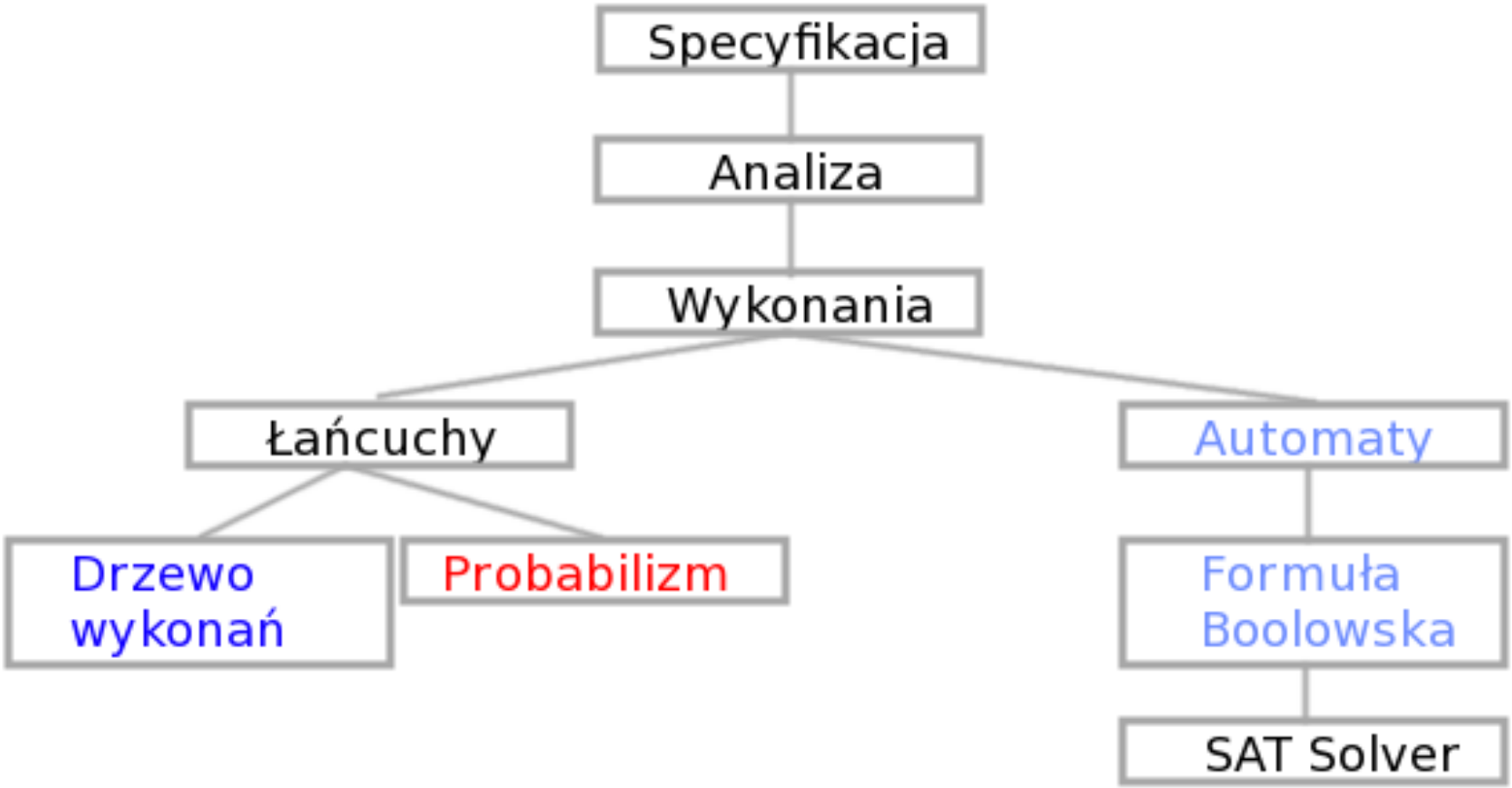
Protokół Needham-Schroeder:

$$\begin{aligned} \alpha_1 \quad & A \rightarrow B : \langle N_A \cdot i(A) \rangle_{K_B}, \\ \alpha_2 \quad & B \rightarrow A : \langle N_A \cdot N_B \rangle_{K_A}, \\ \alpha_3 \quad & A \rightarrow B : \langle N_B \rangle_{K_B}. \end{aligned} \tag{1}$$

Atak Lowe-a:

$$\begin{aligned} \alpha_1^1 \quad & A \rightarrow \iota : \langle N_A \cdot i(A) \rangle_{K_\iota}, \\ \alpha_1^2 \quad & \iota(A) \rightarrow B : \langle N_A \cdot i(A) \rangle_{K_B}, \\ \alpha_2^2 \quad & B \rightarrow \iota(A) : \langle N_A \cdot N_B \rangle_{K_A}, \\ \alpha_2^1 \quad & \iota \rightarrow A : \langle N_A \cdot N_B \rangle_{K_A}, \\ \alpha_3^1 \quad & A \rightarrow \iota : \langle N_B \rangle_{K_\iota}, \\ \alpha_3^2 \quad & \iota(A) \rightarrow B : \langle N_B \rangle_{K_B}. \end{aligned} \tag{2}$$

Aplikacja


Specyfikacja
Analiza
Wykonania
Łańcuchy
Automaty
Drzewo wykonań
Probabilizm
Formuła Boolowska
SAT Solver

Język ProToc

ProToc

Język specyfikacji protokołów komunikacyjnych wykorzystywany dla automatycznej weryfikacji.

Zalety języka:

- w pełni sformalizowany
- udostępnia możliwość pełnej specyfikacji protokołu (opis akcji wewnętrznych i zewnętrznych)
- prosty, intuicyjny, zwarty

Język ProToc


```
users(2);
steps(3);
players(3);
intruder(dy);

protocol:
p_1,p_2;ident_p_1,nonce_p_1,+key_p_2;nonce_p_1;
        <+key_p_2,nonce_p_1|ident_p_1>;
p_2,p_1;nonce_p_1,nonce_p_2,+key_p_1:nonce_p_2;
        <+key_p_1,nonce_p_1|nonce_p_2>;
p_1,p_2;nonce_p_2,+key_p_2;<+key_p_2,nonce_p_2>;

sessions:
(p_1, p_2);
(p_1, p_-2);
(p_-1, p_2);
```

Łańcuchy stanów


```
[1, 1] ; ; n_1(1) ; n_1(1) ;
[1, 2] ; n_1(1) ; n_2(1) ; n_2(1) ;
[1, 3] ; n_2(1) ; ; ;
[2, 1] ; ; n_1(1) ; n_1(1) ;
[2, 2] ; <k+_1(1),n_1(1)|n_0(1)> ; ; n_0(1) ;
[2, 2] ; n_1(1),  n_0(1) ; ; n_0(1) ;
[2, 3] ; n_0(1) ; ; ;
[3, 1] ; ; n_1(1) ; n_1(1) ;
[3, 2] ; <k+_1(1),n_1(1)|n_2(1)> ; ; n_2(1) ;
[3, 2] ; n_1(1),  n_2(1) ; ; n_2(1) ;
[3, 3] ; n_2(1) ; ; ;
[4, 1] ; ; n_1(1) ; <k+_2(1),i(p_1)|n_1(1)> ;
[4, 2] ; <k+_1(1),n_1(1)|n_0(1)> ; ; n_0(1) ;
[4, 2] ; n_1(1),  n_0(1) ; ; n_0(1) ;
[4, 3] ; n_0(1) ; ; <k+_2(1),n_0(1)> ;
[5, 1] ; ; n_1(1) ; <k+_2(1),i(p_1)|n_1(1)> ;
[5, 2] ; <k+_1(1),n_1(1)|n_2(1)> ; ; n_2(1) ;
[5, 2] ; n_1(1),  n_2(1) ; ; n_2(1) ;
[5, 3] ; n_2(1) ; ; <k+_2(1),n_2(1)> ;
[6, 1] ; ; n_2(1) ; n_2(1) ;
...
```

Przeplot wykonań

- Każde wykonanie protokołu składa się z kroków
- W sieci występuje równolegle wiele wykonań tego samego protokołu
- Kroki poszczególnych wykonań mogą się przeplatać:

$$\mathcal{R} = \alpha_1^1, \alpha_2^1, \alpha_1^2, \alpha_3^1, \alpha_2^2, \alpha_3^2.$$

Realne wykonania

- Utrzymana jest chronologia zdarzeń
- Nadający musi posiadać wymaganą do wysłania wiadomości wiedzę (zdobyć ją w trakcie komunikacji, lub wygenerować)
- Otrzymujący poszerza swoją wiedzę o wszystkie składowe przychodzącej wiadomości

Atak


```
[ 3, 1] ; ; n_1(1) ; n_1(1) ;
[18, 1] ; <k+_2(1),i(p_1)|n_1(1)> ; ; n_1(1) ;
[18, 2] ; n_1(1) ; n_2(1) ; <k+_1(1),n_1(1)|n_2(1)> ;
[ 3, 2] ; <k+_1(1),n_1(1)|n_2(1)> ; ; n_2(1) ;
[ 3, 3] ; n_2(1) ; ; ;
[18, 3] ; n_2(1) ; ; ;
```

Typy automatów probabilistycznych

- automaty reaktywne
- automaty generatywne
- automaty I\O
- automaty Vardi'ego
- automaty przemienne Hansson'a
- automaty Segala
- automaty łączone
- automaty Pnueli'ego-Zuck'a

Definicja automatu probabilistycznego

Definicja (PA)

Skończony reaktywny automat probabilistyczny - to piątka $PA = (Q, \Sigma, \delta, q_0, F)$, w której:

- Q to skończony zbiór stanów,
- Σ to skończony zbiór symboli wejściowych,
- δ - funkcja prawdopodbieństwa przejść $\delta : Q \times \{\Sigma \cup \{\epsilon\}\} \mapsto \mathcal{D}(Q)$.
- $q_0 \in Q$ - stan początkowy,
- $F \subseteq Q$ - zbiór stanów końcowych (akceptujących).

Protokół Needhama-Schroedera-Lowe-a

Protokół NSPKL:

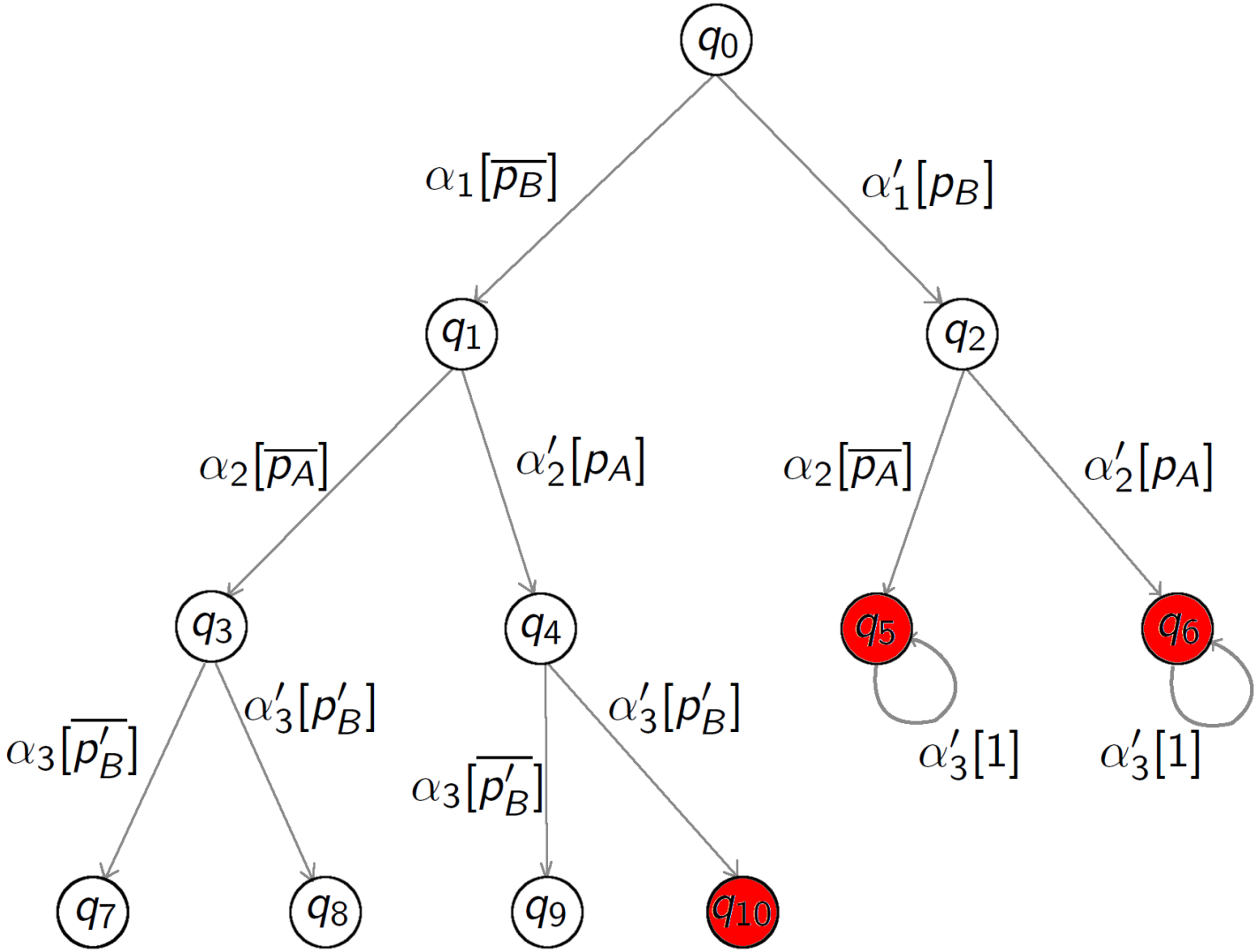
$$\begin{aligned} \alpha_1 \quad & A \rightarrow B : \langle N_A \cdot i(A) \rangle_{K_B}, \\ \alpha_2 \quad & B \rightarrow A : \langle N_A \cdot N_B \cdot i(B) \rangle_{K_A}, \\ \alpha_3 \quad & A \rightarrow B : \langle N_B \rangle_{K_B}. \end{aligned} \tag{3}$$

Załóżmy, że:

- α_1' - α_1 i złamanie klucza K_B^{-1} z prawdopodobieństwem p_B
- α_2' - α_2 i złamanie klucza K_A^{-1} z prawdopodobieństwem p_A
- α_3' - α_3 i złamanie klucza K_B^{-1} z prawdopodobieństwem p_B'

gdzie $p_B + \overline{p_B} = p_A + \overline{p_A} = p_B' + \overline{p_B'} = 1$.

Weryfikacja z wykorzystanie PA



test

Subprotocol	AVISPA				VerICS	PathFinder
	OFMC	CL-AtSe	SATMC	TA4SP		
State 3	SAFE 70 ms.	SAFE <10 ms.	SAFE 30 ms.	SAFE 661 ms.	SAFE 15 ms.	SAFE <10 ms.
State 4	SAFE 70 ms.	SAFE <10 ms.	SAFE 30 ms.	SAFE 661 ms.	SAFE 15 ms.	SAFE <10 ms.

Plany

- Ukończenie poprawianej aplikacji
- Dodanie analizy prawdopobieństw
- PPAM 2015
- CISIM 2015
- Dodanie analizy sieci automatów
- Kwestia czasu...

Źródła I

A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, P.C. Heám, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Vigano, L. Vigneron, The Avispa Tool for the automated validation of internet security protocols and applications, in proceedings of CAV 2005, Computer Aided Verification, LNCS 3576, Springer Verlag, 2005

Behrmann G., David A., Larsen K. G., A Tutorial on Uppaal, In proceedings of the 4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM-RT'04). LNCS 3185

Kurkowski M., Formalne metody weryfikacji własności protokołów zabezpieczających w sieciach komputerowych, s. 208, wyd. Exit, Warszawa, 2013

Kurkowski M., Grosser A., Piątkowski J., Szymoniak S., ProToc - an universal language for security protocols specification, Advances in Intelligent Systems and Computing, vol. 342, pp 237-248, Springer Verlag, 2015

Kurkowski M., Penczek W.: Applying Timed Automata to Model Checking of Security Protocols, in ed. J. Wang, Handbook of Finite State Based Models and Applications, pp. 223–254, CRC Press, Boca Raton, USA, 2012

Kurkowski, M., Penczek, W.: Verifying Security Protocols Modeled by Networks of Automata, Fund. Inform., Vol. 79 (3-4), pp. 453-471, IOS Press 2007

Kurkowski, M., Penczek, W.: Verifying Timed Security Protocols via Translation to Timed Automata, Fund. Inform., vol. 93 (1-3), pp. 245-259, IOS Press 2009

Źródła II

Kurkowski M., Siedlecka-Lamch O., Dudek P., Using Backward Induction Techniques in (Timed) Security Protocols Verification, in Proceedings of 12th International Conference CISIM 2013, Krakow, Poland, Lecture Notes in Computer Science vol. 8104, 265 - 276, 2013

Kurkowski M., Siedlecka-Lamch O., Piech H., A New Effective Approach for Modelling and Verification of Security Protocols, in Proceedings of 21th international Workshop on Concurrency, Specification and Programming (CS&P 2012) Humboldt University Press, Berlin, Germany, 191-202, 2012

Kurkowski M., Siedlecka-Lamch O., Szymoniak S., Piech H., Parallel Bounded Model Checking of Security Protocols, in Proc. of PPAM'13, vol. 8384 of LNCS, Springer Verlag, 224 - 234, 2014

Kurkowski, M., Srebrny, M.: A Quantifier-free First-order Knowledge Logic of Authentication, Fund. Inform., vol. 72, pp. 263-282, IOS Press 2006

Kwiatkowska M., Norman G. and Parker D.. PRISM 4.0: Verification of Probabilistic Real-time Systems. In Proc. 23rd International Conference on Computer Aided Verification (CAV'11), volume 6806 of LNCS, pages 585-591, Springer, 2011

Lowe, G.: Breaking and Fixing the Needham-Schroeder Public-key Protocol Using fdr., In TACAS, LNCS, Springer, 147-166, 1996

Needham, R.M., Schroeder, M.D.: Using encryption for authentication in large networks of computers. Commun. ACM, 21(12), 993-999, 1978.