STUDIA INFORMATICA 2011
Volume 32 Number 3A (98)

Anna FELKNER
Naukowa i Akademicka Sieć Komputerowa
Adam KOZAKIEWICZ
Politechnika Warszawska, Instytut Automatyki i Informatyki Stosowanej
Naukowa i Akademicka Sieć Komputerowa

CZASOWA WAŻNOŚĆ POŚWIADCZEŃ JĘZYKA RT^{T}_{+}

Streszczenie. Język RT^{T} należy do języków zarządzania zaufaniem z rodziny języków role-based trust management, który jest używany do reprezentowania polityk bezpieczeństwa oraz poświadczeń w rozproszonym upoważnianiu. Łączy on zalety kontroli dostępu, opartej na roli i systemów zarządzania zaufaniem. W artykule zaproponowano wprowadzenie ograniczeń czasowych ważności poświadczeń w języku RT^{T}.

Słowa kluczowe: zarządzanie zaufaniem, role-based trust management, system wnioskowania z ograniczeniami

TIME VALIDITY OF CREDENTIALS IN RT^{T}_{+} LANGUAGE

Summary. The topic of this paper is RT^{T}, a language from the family of RT languages, which is used for representing security policies and credentials in distributed access control systems. The goal of this paper is introduction of time validity constraints to show how that can make RT^{T} language more realistic.

Keywords: trust management, role-based trust management, inference system

1. Wstęp

Tradycyjne modele kontroli dostępu (uznaniowa, obowiązkowa i oparta na rolach) podejmują decyzje związane z nadzorem na podstawie tożsamości wnioskującego. W otwartych, rozproszonych środowiskach użytkownicy zmieniają się dynamicznie, a ich tożsamość nie jest a priori znana. Istnieje w nich wiele podmiotów uwierzytelniających i wystawiających poświadczenia. Brak też jednego źródła, do którego można się odwołać, w celu uzyskania

wszystkich dokumentów, zawierających aktualne uprawnienia. Wychodząc naprzeciw tym ograniczeniom, zostały zaproponowane modele *zarządzania zaufaniem* [1, 2], czyli uporządkowane podejście do specyfikowania i interpretowania *polityk bezpieczeństwa*, *poświadczeń* oraz *relacji związanych z zaufaniem*. Decyzje bazują w nich na orzeczeniach wydawanych przez wiele podmiotów. *Poświadczenie* jest dokumentem, wystawionym i podpisanym przez uprawniony podmiot, zawierającym dane, niezbędne do uzyskania uprawnień przez inny podmiot (np. prawo jazdy, legitymacja studencka lub certyfikat). Prawo podmiotu do wykonania danej operacji zależy od przyznanych mu poświadczeń, które może on przekazywać innym podmiotom, co jest nazywane *delegacją*. *Polityka bezpieczeństwa* określa, jak konkretne upoważnienia wynikają z posiadanych poświadczeń, a *relacje związane z zaufaniem* definiują kto ma prawo wystawiać określone poświadczenia.

Do zdefiniowania systemu zarządzania zaufaniem jest potrzebny język opisu podmiotów, poświadczeń i ról, które te podmioty pełnią w systemie. W tym celu w [6, 7] została zdefiniowana rodzina języków Role-based Trust management (RT). Łączy ona zalety kontroli dostępu, opartej na rolach i modeli zarządzania zaufaniem, bazujących na poświadczeniach. RT jest rodziną języków o różnym poziomie złożoności i ekspresyjności. W jej skład wchodzą języki RT_0, RT_1, RT_2 i RT^T. Głównym tematem niniejszego artykułu jest RT^T, oferujący użyteczne właściwości, których nie ma w RT_0, RT_1 i RT_2: role wielorakie, progowanie, polityka podziału obowiązków. *Role wielorakie* rozszerzają pojęcie roli, pozwalając, by członkowie roli byli zbiorem podmiotów, a nie pojedynczym podmiotem. *Struktury progu* wymagają porozumienia kilku podmiotów ze zbioru, żeby potwierdzić jakiś fakt. Polityka *podziału obowiązków* wymaga, aby za wykonanie zadania odpowiedzialnych było dwóch lub więcej użytkowników pochodzących z różnych zbiorów podmiotów.

2. Język RT^T

Podstawowe elementy języków RT to podmioty, nazwy ról, role i poświadczenia. *Podmiot* w RT jest jednoznacznie identyfikowalnym bytem (np. użytkownik lub proces), który może wystawiać role, poświadczenia i przesyłać żądanie dostępu do zasobów. *Nazwa roli* jest jej identyfikatorem. *Rola* jest zbiorem podmiotów, którym uprawnienia przydzielane są wspólnie, co oznacza, że przydzielenie konkretnego uprawnienia danej roli skutkuje otrzymaniem tego uprawnienia przez każdego z członków tej roli. *Poświadczenia* definiują role poprzez wskazanie nowych członków ról lub poprzez delegację uprawnień do członków innych ról. *Nazwa podmiotu* rozpoczyna się (lub po prostu jest) wielką literą, a *nazwa roli* małą literą. *Rola* oznaczana jest przez *nazwę podmiotu* i *nazwę roli* oddzielonych kropką. *Poświadczenia* przyjmują jedną z sześciu poniższych postaci:

$A.r \leftarrow B$ – podmiot B należy do roli $A.r$,
$A.r \leftarrow B.s$ – wszyscy członkowie roli $B.s$ należą również do roli $A.r$,
$A.r \leftarrow B.s.t$ – wszyscy członkowie roli $C.t$ również należą do roli $A.r$, dla każdego C należącego do roli $B.s$,
$A.r \leftarrow B.s \cap C.t$ – tylko członkowie obu ról: $B.s$ i $C.t$ jednocześnie należą do roli $A.r$,
$A.r \leftarrow B.s \bullet C.t$ – do roli $A.r$ należy jeden członek roli $B.s$ i jeden roli $C.t$ jednocześnie,
$A.r \leftarrow B.s \otimes C.t$ – jak w poprzednim przypadku, przy czym są to różniący się członkowie ról.

Przykład 2.1 (zaadaptowany z [6]). Rozpatrzmy przykład, w którym polityka banku BP wymaga, aby pewna transakcja została zatwierdzona przez kierownika, dwóch różnych kasjerów i kontrolera. Kierownik, który jest również kasjerem może świadczyć usługi jednego z dwóch kasjerów. Kontroler natomiast nie może być żadną z pozostałych stron w transakcji.

$$BP.kasjerzy \leftarrow BP.kasjer \otimes BP.kasjer \tag{1}$$

$$BP.kierownikKasjerzy \leftarrow BP.kierownik \bullet BP.kasjerzy \tag{2}$$

$$BP.akceptacja \leftarrow BP.kontroler \otimes BP.kierownikKasjerzy \tag{3}$$

Przyjmując poszczególne role:

$$\text{(a) } BP.kasjer \leftarrow \{Ala\}, \quad \text{(b) } BP.kasjer \leftarrow \{Ola\},$$
$$\text{(c) } BP.kierownik \leftarrow \{Ola\}, \quad \text{(d) } BP.kontroler \leftarrow \{Ela\} \tag{4}$$

możemy wyznaczyć zbiór osób, które muszą wziąć udział w zatwierdzeniu transakcji zgodnie z polityką banku. Są to: $\{Ala, Ola, Ela\}$.

RT^T jest najobszerniejszym językiem z rodziny RT, obejmującym wszystkie reguły składniowe całej rodziny. Opracowana dla niego semantyka teoriomnogościowa została przedstawiona w [3].

W praktyce użyteczniejszy od semantyki teoriomnogościowej jest *system wnioskowania*, czyli reguły uzyskiwania nowych poświadczeń. Dla posiadanego zbioru poświadczeń P, bazę wiedzy tworzy zbiór formuł o postaci: $c \in \mathsf{P}$, dla każdego poświadczenia c, należącego do zbioru P. Wykorzystanie reguł wnioskowania prowadzi do uzyskania nowych poświadczeń. Poświadczenia te są zapisywane za pomocą formuły: $\mathsf{P} \succ c$, którą należy czytać: „używając poświadczeń ze zbioru P, można wywnioskować poświadczenie c". Przesłankami reguł są formuły w obu wymienionych postaciach, a wnioskami formuły o postaci $\mathsf{P} \succ c$. Reguły systemu wnioskowania dla RT^T zostały przedstawione w [4]. Praca zawiera również dowód poprawności i pełności systemu – własności decydujących o jego użyteczności.

Stworzony system wnioskowania języka RT^T jest precyzyjnym odpowiednikiem semantyki teoriomnogościowej tego języka. Pozwala zdecydować, jaką dodatkową wiedzę można wydobyć ze zbioru posiadanych poświadczeń. Jest to niezwykle istotne w sytuacji, gdy w systemie rozproszonym niektóre poświadczenia mogą być w określonym momencie niedostępne lub też dostęp do nich może być, z różnych powodów, ograniczony.

3. Poświadczenia uzależnione od czasu w RT

W artykule [5] rozszerzono język RT_0 o ograniczenia czasu dostępności poświadczeń. Takie ograniczenie ważności ma często miejsce w rzeczywistości. W niniejszym artykule zaproponowano podobne rozszerzenie języka RT^T. Poświadczenie z ograniczeniem można zapisać jako: c **in** v, co oznacza: „poświadczenie c jest dostępne w czasie v". Jako CP oznaczmy skończony zbiór dostępnych poświadczeń, uzależnionych od czasu. Aby uprościć notację, gdy chcemy przedstawić c **in** $(-\infty,+\infty)$, zapisujemy po prostu c. Tak rozszerzony język oznaczamy $RT^T_{+.}$ Ograniczenia czasowe mogą w pewnym stopniu zaspokajać potrzeby, które w systemach niemonotonicznych obsługuje negacja. Ważność czasowa może być wyrażona na różne sposoby, w szczególności: $v=[\tau_1,\tau_2];[\tau_1,\tau_2);(\tau_1,\tau_2];(\tau_1,\tau_2);(-\infty,\tau];(-\infty,\tau);$ $[\tau,+\infty);(\tau,+\infty);(-\infty,+\infty);\ v_1\cup v_2; v_1\cap v_2; v_1\setminus v_2$, gdzie τ oznacza stałą czasową.

Przykład 3.1 (uwarunkowania czasowe wprowadzone do przykładu 2.1). Pierwsze trzy poświadczenia są niezależne od czasu. Jednak prawdopodobne jest, że czas pracy na danym stanowisku jest określony, czyli przynależność do ról jest ograniczona. Poświadczenia (1)-(3) pozostają więc bez zmian, natomiast poświadczenie (4) przyjmuje postać:

(a) $BP.kasjer \leftarrow \{Ala\}$ **in** v_1, (b) $BP.kasjer \leftarrow \{Ola\}$ **in** v_2

(c) $BP.kierownik \leftarrow \{Ola\}$ **in** v_3, (d) $BP.kontroler \leftarrow \{Ela\}$ **in** v_4 (5)

Na tej podstawie możemy wyznaczyć zbiory osób, pełniących dane funkcje w określonym czasie. Będzie to: $\{Ala,Ola\}$ **in** $v_1\cap v_2$ dla roli $BP.kasjerzy$, $\{Ala,Ola\}$ **in** $v_1\cap v_2\cap v_3$, dla roli $BP.kierownikKasjerzy$ i $\{Ala,Ola,Ela\}$ **in** $v_1\cap v_2\cap v_3\cap v_4$ dla roli $BP.akceptacja$. Idąc dalej, można ograniczyć też ważność reguły (3). Przypuśćmy, że jest ona ważna tylko w czasie v_A. Członkowie roli $BP.akceptacja$ mogą wtedy zatwierdzić transakcję w czasie, który jest iloczynem v_A i wyżej wyznaczonego, czyli: $\{Ala,Ola,Ela\}$ **in** $v_A\cap v_1\cap v_2\cap v_3\cap v_4$.

Semantykę teoriomnogościową języka RT^T_+ można zdefiniować podobnie, jak w przypadku RT^T, czyli należy zaadaptować definicję 2 z [3] do obecnej postaci poświadczeń.

Definicja 4.1 (semantyka teoriomnogościowa języka RT^T_+). *Semantyka* S_{CP} *zbioru poświadczeń* CP *jest najmniejszą relacją* S_i*, domkniętą ze względu na funkcję* f *określoną przez warunki:*

1. $\mathsf{S}_0=\phi$
2. $\mathsf{S}_{i+1}=\bigcup_{(c\ \mathbf{in}\ v)\in\mathsf{CP}} f(\mathsf{S}_i,c)$ dla $i=0,1,\ldots$

Wartość funkcji f jest określona dla wszystkich postaci poświadczeń w analogiczny sposób, jak dla języka RT^T w [3].

4. System wnioskowania języka RT^T_+

Adaptacja systemu wnioskowania języka RT^T dla nowego języka jest prosta w przypadku pojedynczej chwili. Niech CP będzie zbiorem poświadczeń RT^T_+ postaci c **in** v. Poświadczenia uzyskane z posiadanej bazy wiedzy przy użyciu reguł wnioskowania będą zapisywane za pomocą formuły: $\mathsf{CP} \succ_\tau c$, którą należy czytać jako: „używając poświadczeń ze zbioru CP, w chwili τ można wywnioskować poświadczenie c". W skład systemu wnioskowania wchodzą następujące reguły ($CW_1 - CW_6$):

$$(CW_1) \quad \frac{c \text{ in } v \in \mathsf{CP} \qquad \tau \in v}{\mathsf{CP} \succ_\tau c}$$

$$(CW_2) \quad \frac{\mathsf{CP} \succ_\tau A.r \leftarrow B.s \qquad \mathsf{CP} \succ_\tau B.s \leftarrow X}{\mathsf{CP} \succ_\tau A.r \leftarrow X}$$

$$(CW_3) \quad \frac{\mathsf{CP} \succ_\tau A.r \leftarrow B.s.t \qquad \mathsf{CP} \succ_\tau B.s \leftarrow C \qquad \mathsf{CP} \succ_\tau C.t \leftarrow X}{\mathsf{CP} \succ_\tau A.r \leftarrow X}$$

$$(CW_4) \quad \frac{\mathsf{CP} \succ_\tau A.r \leftarrow B.s \cap C.t \qquad \mathsf{CP} \succ_\tau B.s \leftarrow X \qquad \mathsf{CP} \succ_\tau C.t \leftarrow X}{\mathsf{CP} \succ_\tau A.r \leftarrow X}$$

$$(CW_5) \quad \frac{\mathsf{CP} \succ_\tau A.r \leftarrow B.s \bullet C.t \qquad \mathsf{CP} \succ_\tau B.s \leftarrow X \qquad \mathsf{CP} \succ_\tau C.t \leftarrow Y}{\mathsf{CP} \succ_\tau A.r \leftarrow X \cup Y}$$

$$(CW_6) \quad \frac{\mathsf{CP} \succ_\tau A.r \leftarrow B.s \otimes C.t \qquad \mathsf{CP} \succ_\tau B.s \leftarrow X \qquad \mathsf{CP} \succ_\tau C.t \leftarrow Y \qquad X \cap Y = \phi}{\mathsf{CP} \succ_\tau A.r \leftarrow X \cup Y}$$

System wnioskowania rozszerza reguły wnioskowania z RT^T na RT^T_+, czyli zamienia (W_i) na (CW_i) i bierze pod uwagę tylko czasowo ważne poświadczenia. Znaczenie zbioru poświadczeń CP, określone przez system wnioskowania, jest dane przez zbiór wszystkich formuł postaci: $\mathsf{CP} \succ_\tau A.r \leftarrow X$. Każda taka formuła zawiera poświadczenie stwierdzające, że zbiór podmiotów X należy do roli $A.r$ w czasie τ. Aby udowodnić prawdziwość takiego poświadczenia, należy wykazać, że trójka (A, r, X) należy do semantyki S_{CP} zbioru poświadczeń CP. Udowodnienie poprawności systemu wnioskowania wymaga wykazania prawdziwości wszystkich poświadczeń $A.r \leftarrow X$, jakie można wywnioskować ze zbioru CP w czasie τ. Zauważmy najpierw, że tak, jak w przypadku poświadczeń P, tak i tutaj trójka (A, r, X) należy do semantyki S_{CP} zbioru poświadczeń CP dla wszystkich poświadczeń $A.r \leftarrow X$, należących do zbioru CP w czasie τ. Dowodzi tego Lemat 5.1.

Lemat 5.1. *Jeżeli* $A.r \leftarrow X \in \mathsf{CP}$, *to* $(A, r, X) \in \mathsf{S}_{\mathsf{CP}}$.

Dowód. *Teza lematu wynika wprost z Definicji 4.1. Relacja* S_{CP} *definiująca semantykę zbioru poświadczeń* CP *w czasie* τ *jest granicą niemalejącego ciągu relacji* $\mathsf{S}_0, \mathsf{S}_1, \ldots,$

w którym $\mathsf{S}_0 = \phi$. *Z warunku* $f(\mathsf{S}_i, A.r \leftarrow B) = \{(A,r,B)\}$ *definicji semantyki* RT_+^T *wynika, że:* $f(\mathsf{S}_0, A.r \leftarrow X) = (A,r,X)$. *Skąd* $(A,r,X) \in \mathsf{S}_1$. *Ponieważ* $\mathsf{S}_1 \subseteq \mathsf{S}_{\mathsf{CP}}$, *więc* $(A,r,X) \in \mathsf{S}_{\mathsf{CP}}$.

Aby udowodnić poprawność systemu wnioskowania, należy wykazać, że stwierdzenia zawarte w poświadczeniach wszystkich formuł $\mathsf{CP} \succ A.r \leftarrow X$, wywnioskowanych za pomocą reguł $(CW_1) - (CW_6)$, są prawdziwe. Dowodzi tego poniższe twierdzenie.

Twierdzenie 5.1 (poprawność). *Jeżeli* $\mathsf{CP} \succ A.r \leftarrow X$, *to* $(A,r,X) \in \mathsf{S}_{\mathsf{CP}}$.

Dowód. *Dowód jest analogiczny do dowodu Twierdzenia 1 z [4], przy czym zamiast Lematu 1 z [4] wykorzystywany jest Lemat 5.1.*

Aby udowodnić pełność systemu wnioskowania, należy wykazać, że formuła $\mathsf{CP} \succ A.r \leftarrow X$ może być wywnioskowana przy użyciu reguł wnioskowania dla każdego elementu $(A,r,X) \in \mathsf{S}_{\mathsf{CP}}$. Dowodzi tego poniższe twierdzenie.

Twierdzenie 5.2 (pełność). *Jeżeli* $(A,r,X) \in \mathsf{S}_{\mathsf{CP}}$, *to* $\mathsf{CP} \succ A.r \leftarrow X$.

Dowód. *Dowód jest analogiczny do dowodu Twierdzenia 2 z [4], przy czym zamiast Lematu 1 z [4] wykorzystywany jest Lemat 5.1.*

Na mocy twierdzeń 5.1 i 5.2 stworzony system wnioskowania jest poprawny i pełny.

Ustalanie maksymalnej ważności czasowej nowych poświadczeń wymaga rozszerzenia systemu wnioskowania. W tym celu formułę $\mathsf{CP} \succ_\tau c$ zastąpi postać $\mathsf{CP} \succ\succ_v c$, co oznacza, że w każdej chwili $\tau \in v$, w której CP posiada semantykę, możliwe jest wywnioskowanie poświadczenia c z CP. Aby uprościć notację, zamiast $\succ\succ_{(-\infty,+\infty)}$, piszemy po prostu $\succ\succ$. W skład systemu wnioskowania wchodzą następujące reguły ($CWP_1 - CWP_7$):

$$(CWP_1) \quad \frac{c \ \mathbf{in}\ v \in \mathsf{CP}}{\mathsf{CP} \succ\succ_v c}$$

$$(CWP_2) \quad \frac{\mathsf{CP} \succ\succ_{v_1} A.r \leftarrow B.s \qquad \mathsf{CP} \succ\succ_{v_2} B.s \leftarrow X}{\mathsf{CP} \succ\succ_{v_1 \cap v_2} A.r \leftarrow X}$$

$$(CWP_3) \quad \frac{\mathsf{CP} \succ\succ_{v_1} A.r \leftarrow B.s.t \qquad \mathsf{CP} \succ\succ_{v_2} B.s \leftarrow C \qquad \mathsf{CP} \succ\succ_{v_3} C.t \leftarrow X}{\mathsf{CP} \succ\succ_{v_1 \cap v_2 \cap v_3} A.r \leftarrow X}$$

$$(CWP_4) \quad \frac{\mathsf{CP} \succ\succ_{v_1} A.r \leftarrow B.s \cap C.t \qquad \mathsf{CP} \succ\succ_{v_2} B.s \leftarrow X \qquad \mathsf{CP} \succ\succ_{v_3} C.t \leftarrow X}{\mathsf{CP} \succ\succ_{v_1 \cap v_2 \cap v_3} A.r \leftarrow X}$$

$$(CWP_5) \quad \frac{\mathsf{CP} \succ\succ_{v_1} A.r \leftarrow B.s \bullet C.t \qquad \mathsf{CP} \succ\succ_{v_2} B.s \leftarrow X \qquad \mathsf{CP} \succ\succ_{v_3} C.t \leftarrow Y}{\mathsf{CP} \succ\succ_{v_1 \cap v_2 \cap v_3} A.r \leftarrow X \cup Y}$$

$$(CWP_6) \quad \frac{\mathsf{CP} \succ\succ_{v_1} A.r \leftarrow B.s \otimes C.t \qquad \mathsf{CP} \succ\succ_{v_2} B.s \leftarrow X \qquad \mathsf{CP} \succ\succ_{v_3} C.t \leftarrow Y \qquad X \cap Y = \phi}{\mathsf{CP} \succ\succ_{v_1 \cap v_2 \cap v_3} A.r \leftarrow X \cup Y}$$

$$(CWP_7) \quad \frac{\mathsf{CP} \succ\succ_{v_1} c \qquad \mathsf{CP} \succ\succ_{v_2} c}{\mathsf{CP} \succ\succ_{v_1 \cup v_2} c}$$

Reguła (CWP_1) mówi, że każde c z CP może być użyte w każdym momencie, w którym jest ważne. Reguły (CWP_2) – (CWP_6) stwierdzają, że reguły wnioskowania mogą być stosowane tylko wtedy, gdy mają miejsce wszystkie przesłanki. Nowa reguła (CWP_7) mówi, że jeśli poświadczenie c może być wywnioskowane zarówno z ważnością v_1, jak i v_2, wtedy c może być wywnioskowane z ważnością $v_1 \cup v_2$. Zauważmy, że $\mathsf{CP} \succ\!\succ_v$ uogólnia $\mathsf{CP} \succ_\tau$. Są one równoważne w przypadku, gdy $v = [\tau, \tau]$. Reguła (CWP_7) może być użyta wiele razy, aby uzyskać maksymalną ważność czasową poświadczenia c.

Definicja 5.1 (wnioskowanie maksymalne). *Wnioskowanie kończące się w* $\mathsf{CP} \succ\!\succ_v c$ *nazywane jest maksymalnym wtedy i tylko wtedy, gdy:*

1. nie istnieje $v' \supset v$ takie, że $\mathsf{CP} \succ\!\succ_{v'} c$, oraz
2. wszystkie pośrednie wnioskowania kończące się w $\mathsf{CP} \succ\!\succ_{v''} c'$, dla $c' \neq c$ są maksymalne.

Pierwsze wymaganie zapewnia, że (CWP_7) zostało użyte maksymalnie wiele razy, aby wywnioskować ważność c. Drugie natomiast propaguje tę właściwość na całe drzewo wnioskowania. Maksymalne wnioskowanie gwarantuje, że v w (CWP_1) jest maksymalną ważnością czasową dla $A.r \leftarrow X$. Dzięki temu możemy udowodnić poprawność i pełność $\mathsf{CP} \succ\!\succ_v$ za pomocą twierdzenia 5.3, którego dowód opiera się na następującym lemacie.

Lemat 5.2. $\mathsf{CP} \succ_\tau c$ *implikuje, że istnieje* v *zawierające* τ *takie, że* $\mathsf{CP} \succ\!\succ_v c$.

Dowód. *Dowód jest analogiczny do przypadku wnioskowania dla* $\mathsf{CP} \succ_\tau c$. *Każde wystąpienie reguły* (CW_i) *należy zastąpić regułą* (CWP_i); v *będzie przecięciem ważności wszystkich poświadczeń* CP *użytych we wnioskowaniu i będzie to co najmniej* $[\tau, \tau]$.

Twierdzenie 5.3 (poprawność i pełność maksymalnego wnioskowania). *Niech* $\mathsf{CP} \succ\!\succ_v c$ *będzie maksymalnym wnioskowaniem i niech będzie zdefiniowany zbiór poświadczeń* c, *które można wywnioskować z* CP *w czasie* τ. *Wtedy* $\mathsf{CP} \succ_\tau c$ *wtedy i tylko wtedy, gdy* $\tau \in v$.

Szkic dowodu. *Dowód przez indukcję po głębokości* $\mathsf{CP} \succ\!\succ_v c$. *W przypadku bazowym,* CP *musi zawierać poświadczenie postaci* c **in** v. *Jeżeli* $\tau \in v$, *to dochodzimy do tego dzięki* (CW_1). *I odwrotnie, przypuśćmy, przez zaprzeczenie, że istnieje* $\tau' \notin v$ *takie, że* $\mathsf{CP} \succ_{\tau'} c$; *ale wtedy wnioskowanie prowadzące do* $\mathsf{CP} \succ\!\succ_v c$ *mogłoby, zgodnie z lematem 5.2, nie być maksymalne, co jest sprzeczne z założeniem, zatem takie* τ' *nie istnieje.*

Dla kroku indukcyjnego, dowodzimy poprzez analizę przypadku użycia ostatniej reguły. Najtrudniejszym przypadkiem jest użycie (CWP_7). *Gdy* $\mathsf{CP} \succ\!\succ_v c$ *kończy się na wystąpieniu tej reguły, to* $v = v_1 \cup v_2$. *Ten przypadek jest szczególny, ponieważ formuły* $\mathsf{CP} \succ\!\succ_{v_1} c$ *i* $\mathsf{CP} \succ\!\succ_{v_2} c$ *na ogół nie są maksymalne. Niech* $\mathsf{CP} \succ_\tau c$. *Z lematu 5.2, istnieje* v' *zawierające* τ *takie, że* $\mathsf{CP} \succ\!\succ_{v'} c$. *Teraz zachodzi* $v' \subseteq v$, *w przeciwnym wypadku* $\mathsf{CP} \succ\!\succ_v c$ *mogłoby nie być maksy-*

malne. I w drugą stronę, niech $\tau \in v$ *i niech* $\mathsf{CP} \succ\!\succ_{v'} c$ *będzie najgłębszym wnioskowaniem pośrednim ze zbioru poświadczeń* $\mathsf{CP} \succ\!\succ_{v} c$*, którego przesłanki nie wymagają* c *(zatem* $\mathsf{CP} \succ\!\succ_{v'} c$ *zostało otrzymane za pomocą wprowadzenia (*CWP_i*), dla* $i \neq 7$*) i takim że* $\tau \in v'$*. Z definicji reguł, każda z tych przesłanek posiada ważność zawierającą* τ_i*, ponieważ zostały one otrzymane dzięki maksymalnemu wnioskowaniu, przez indukcję możemy więc zamienić* $\succ\!\succ \ldots z \succ_{\tau}$*. Teraz trzeba zastosować (*CW_i*) i kontynuować wnioskowanie.*

Przykład 5.1 (system wnioskowania dla przykładu 3.1). Przypuśćmy, że chcemy sprawdzić, w jakim czasie może współpracować dwóch różnych kasjerów. Bierzemy pod uwagę poświadczenia (1), (5a) i (5b) i korzystamy z reguły (CWP_1) zgodnie, z którą:

$$\frac{BP.kasjerzy \leftarrow BP.kasjer \otimes BP.kasjer \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ BP.kasjerzy \leftarrow BP.kasjer \otimes BP.kasjer}$$

$$\frac{BP.kasjer \leftarrow \{Ala\}\ \mathbf{in}\ v_1 \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ_{v_1} BP.kasjer \leftarrow \{Ala\}} \qquad \frac{BP.kasjer \leftarrow \{Ola\}\ \mathbf{in}\ v_2 \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ_{v_2} BP.kasjer \leftarrow \{Ola\}}$$

Następnie biorąc pod uwagę poświadczenia (1), (5a), (5b) i korzystając z reguły (CWP_6):

$$\frac{\begin{array}{cc}\mathsf{CP} \succ\!\succ BP.kasjerzy \leftarrow BP.kasjer \otimes BP.kasjer & \mathsf{CP} \succ\!\succ_{v_1} BP.kasjer \leftarrow \{Ala\} \\ \mathsf{CP} \succ\!\succ_{v_2} BP.kasjer \leftarrow \{Ola\} & \{Ala\} \cap \{Ola\} = \phi\end{array}}{\mathsf{CP} \succ\!\succ_{v_1 \cap v_2} \mathbf{BP.kasjerzy} \leftarrow \mathbf{\{Ala,Ola\}}}$$

uzyskujemy interesujące nas poświadczenie, więc kończymy wnioskowanie, podczas którego były wykorzystane tylko niektóre z posiadanych poświadczeń. Jeśli natomiast chcemy wyznaczyć zbiór osób potrzebnych do zaakceptowania transakcji, to bierzemy pod uwagę poświadczenia (2), (3), (5c i d) i korzystając ponownie z reguły (CWP_1) wnioskujemy:

$$\frac{BP.kierownikKasjerzy \leftarrow BP.kierownik \bullet BP.kasjerzy \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ BP.kierownikKasjerzy \leftarrow BP.kierownik \bullet BP.kasjerzy}$$

$$\frac{BP.akceptacja \leftarrow BP.kontroler \otimes BP.kierownikKasjerzy \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ BP.akceptacja \leftarrow BP.kontroler \otimes BP.kierownikKasjerzy}$$

$$\frac{BP.kierownik \leftarrow \{Ola\}\ \mathbf{in}\ v_3 \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ_{v_3} BP.kierownik \leftarrow \{Ola\}} \qquad \frac{BP.kontroler \leftarrow \{Ela\}\ \mathbf{in}\ v_4 \in \mathsf{CP}}{\mathsf{CP} \succ\!\succ_{v_4} BP.kontroler \leftarrow \{Ela\}}$$

Następnie dodajemy poświadczenia (2) i (5c) i korzystając z reguły (CWP_5):

$$\frac{\begin{array}{cc}\multicolumn{2}{c}{\mathsf{CP} \succ\!\succ BP.kierownikKasjerzy \leftarrow BP.kierownik \bullet BP.kasjerzy} \\ \mathsf{CP} \succ\!\succ \mathbf{in}\ v_3 BP.kierownik \leftarrow \{Ola\} & \mathsf{CP} \succ\!\succ_{v_1 \cap v_2} BP.kasjerzy \leftarrow \{Ala,Ola\}\end{array}}{\mathsf{CP} \succ\!\succ_{v_1 \cap v_2 \cap v_3} \mathbf{BP.kierownikKasjerzy} \leftarrow \mathbf{\{Ala,Ola\}}}$$

uzyskujemy poświadczenie wyznaczające zbiór składający się z kierownika i dwóch różnych kasjerów. Dodając poświadczenia (3) oraz (5d) i korzystając z reguły (CWP_6):

$$\frac{\begin{array}{cc}\mathsf{CP} \succ\!\succ BP.akceptacja \leftarrow BP.kontroler \otimes BP.kierownikKasjerzy & \mathsf{CP} \succ\!\succ_{v_4} BP.kontroler \leftarrow \{Ela\} \\ \mathsf{CP} \succ\!\succ_{v_1 \cap v_2 \cap v_3} BP.kierownikKasjerzy \leftarrow \{Ala,Ola\} & \{Ela\} \cap \{Ala,Ola\} = \phi\end{array}}{\mathsf{CP} \succ\!\succ_{v_1 \cap v_2 \cap v_3 \cap v_4} \mathbf{BP.akceptacja} \leftarrow \mathbf{\{Ala,Ola,Ela\}}}$$

uzyskujemy poświadczenie: $BP.akceptacja \leftarrow \{Ala, Ola, Ela\}$ **in** $v_1 \cap v_2 \cap v_3 \cap v_4$, które wyznacza zbiór ludzi, niezbędnych do zatwierdzenia transakcji, zatem:

$\{(1), (2), (3), (5a), (5b), (5c), (5d)\} \succ\succ_{v_1 \cap v_2 \cap v_3 \cap v_4} BP.akceptacja \leftarrow \{Ala, Ola, Ela\}$.

5. Podsumowanie

Język RT^T jest istotnym wkładem do tematyki zarządzania zaufaniem, gdyż za pomocą ról wielorakich można wyrazić struktury trudne lub wręcz niemożliwe do przedstawienia w pozostałych systemach zarządzania zaufaniem. Wprowadzenie ograniczeń czasowych czyni go językiem bardziej realistycznym. Zasadnicza część pracy opisuje system wnioskowania języka RT^T_+ (czyli RT^T z ograniczeniami czasowymi), pozwalający za pomocą zestawu reguł wnioskowania uzyskać ze zbioru pierwotnie posiadanych poświadczeń nowe. Właściwości poprawności i pełności systemu w odniesieniu do semantyki RT^T_+ zostały sprawdzone, co pokazuje, że jest on prawidłową alternatywą semantyki teoriomnogościowej. W przypadku systemu wnioskowania nie ma potrzeby posiadania wszystkich możliwych poświadczeń, gdyż za pomocą posiadanej bazy wiedzy możemy wywnioskować nowe poświadczenia. System wydaje się być niezbędny, zwłaszcza w przypadku dużych, rozproszonych systemów, gdzie użytkownicy mogą mieć tylko częściową wiedzę na temat swoich aktualnych, możliwych poświadczeń.

BIBLIOGRAFIA

1. Blaze M., Feigenbaum J., Lacy J.: Decentralized Trust Management. In: 17th IEEE Symposium on Security and Privacy, Oakland CA (1996), s. 164÷173.
2. Chapin P., Skalka C., Wang X. S.: Authorization in Trust Management: Features and Foundations. ACM Comput. Surv. 3, (2008), s. 1÷48.
3. Felkner A., Sacha K.: The Semantics of Role-Based Trust Management Languages. CEESET 2009 (preprints), (2009), s. 195÷206.
4. Felkner A., Sacha K.: Deriving RT_T Credentials for Role-Based Trust Management. e-Informatica Software Engineering Journal, Vol. 4, Issue 1, (2010), s. 9÷19.
5. Gorla D., Hennessy M., Sassone V.: Inferring Dynamic Credentials for Role-Based Trust Management. Proc. 8th ACM SIGPLAN *PPDP'06*, (2006), s. 213÷224.
6. Li N., Mitchell J.: RT: A Role-Based Trust-Management Framework. Proc. 3rd DARPA Information Survivability Conference and Exposition, (2003), s. 201÷212.

7. Li N., Winsborough W., Mitchell J.: Distributed Credential Chain Discovery in Trust Management. J. Comput. Secur. 1, (2003), s. 35÷86.

Recenzenci: Prof. dr hab. inż. Bolesław Pochopień
Dr inż. Mirosław Skrzewski

Wpłynęło do Redakcji 11 marca 2011 r.

Abstract

The topic of this paper is RT^{T}, a language from the family of role-based trust management languages, which is used for representing security policies and credentials in distributed large scale access control systems. A credential provides information about the privileges of users and the security policies issued by one or more trusted authorities. RT languages combine trust management and Role Based Access Control features. RT^{T} provides manifold roles to express threshold and separation of duties policies. A manifold role defines sets of entities whose cooperation satisfies the manifold role. The goal of this paper is introduction of time validity constraints to show how that can make RT^{T} language more realistic. The new language is denoted as RT^{T}_{+}. The core part of the paper describes an inference system, in which credentials can be derived from an initial set of credentials using a set of inference rules. The properties of soundness and completeness of the inference system with respect to the semantics of RT^{T}_{+} are proven. Inference systems presented in this paper are simple, but well-founded theoretically. It turns out to be fundamental mainly in large-scale distributed systems, where users have only partial view of their execution context.

Adresy

Anna FELKNER: Naukowa i Akademicka Sieć Komputerowa, ul. Wąwozowa 18, 02-796 Warszawa, Polska, anna.felkner@nask.pl
Adam KOZAKIEWICZ: Instytut Automatyki i Informatyki Stosowanej, ul. Nowowiejska 15/19, 00-665 Warszawa, Polska, akozakie@elka.pw.edu.pl