

# Systemy Operacyjne — Ochrona

Arkadiusz Chrobot

Katedra Systemów Informatycznych, Politechnika Świętokrzyska w Kielcach

Kielce, 22 stycznia 2023

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 3 Ochrona
  - 4 Bezpieczeństwo
  - 5 Polityka bezpieczeństwa i mechanizmy
  - 6 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 4 Tablica globalna
  - 5 Wykaz dostępu
  - 6 Wykaz możliwości
  - 7 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 3 Ochrona
  - 4 Bezpieczeństwo
  - 5 Polityka bezpieczeństwa i mechanizmy
  - 6 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 4 Tablica globalna
  - 5 Wykaz dostępu
  - 6 Wykaz możliwości
  - 7 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
  - 2 Bezpieczeństwo
  - 3 Polityka bezpieczeństwa i mechanizmy
  - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
    - 2 Bezpieczeństwo
    - 3 Polityka bezpieczeństwa i mechanizmy
    - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
  - 2 Bezpieczeństwo
  - 3 Polityka bezpieczeństwa i mechanizmy
  - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- ❶ Wstęp
- ❷ Podstawowe pojęcia
  - ❶ Ochrona
  - ❷ Bezpieczeństwo
  - ❸ Polityka bezpieczeństwa i mechanizmy
  - ❹ Domeny i wiedza konieczna
- ❸ Macierze dostępu
  - ❶ Tablica globalna
  - ❷ Wykaz dostępu
  - ❸ Wykaz możliwości
  - ❹ Mechanizm zamka z kluczem
- ❹ Dynamiczne struktury ochrony
- ❺ Unieważnianie praw dostępu

# Plan wykładu

- ❶ Wstęp
- ❷ Podstawowe pojęcia
  - ❶ Ochrona
  - ❷ Bezpieczeństwo
  - ❸ Polityka bezpieczeństwa i mechanizmy
  - ❹ Domeny i wiedza konieczna
- ❸ Macierze dostępu
  - ❶ Tablica globalna
  - ❷ Wykaz dostępu
  - ❸ Wykaz możliwości
  - ❹ Mechanizm zamka z kluczem
- ❹ Dynamiczne struktury ochrony
- ❺ Unieważnianie praw dostępu



# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
  - 2 Bezpieczeństwo
  - 3 Polityka bezpieczeństwa i mechanizmy
  - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
  - 2 Bezpieczeństwo
  - 3 Polityka bezpieczeństwa i mechanizmy
  - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
  - 2 Bezpieczeństwo
  - 3 Polityka bezpieczeństwa i mechanizmy
  - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- 1 Wstęp
- 2 Podstawowe pojęcia
  - 1 Ochrona
  - 2 Bezpieczeństwo
  - 3 Polityka bezpieczeństwa i mechanizmy
  - 4 Domeny i wiedza konieczna
- 3 Macierze dostępu
  - 1 Tablica globalna
  - 2 Wykaz dostępu
  - 3 Wykaz możliwości
  - 4 Mechanizm zamka z kluczem
- 4 Dynamiczne struktury ochrony
- 5 Unieważnianie praw dostępu

# Plan wykładu

- ❶ Wstęp
- ❷ Podstawowe pojęcia
  - ❶ Ochrona
  - ❷ Bezpieczeństwo
  - ❸ Polityka bezpieczeństwa i mechanizmy
  - ❹ Domeny i wiedza konieczna
- ❸ Macierze dostępu
  - ❶ Tablica globalna
  - ❷ Wykaz dostępu
  - ❸ Wykaz możliwości
  - ❹ Mechanizm zamka z kluczem
- ❹ Dynamiczne struktury ochrony
- ❺ Unieważnianie praw dostępu

# Plan wykładu

- ❶ Wstęp
- ❷ Podstawowe pojęcia
  - ❶ Ochrona
  - ❷ Bezpieczeństwo
  - ❸ Polityka bezpieczeństwa i mechanizmy
  - ❹ Domeny i wiedza konieczna
- ❸ Macierze dostępu
  - ❶ Tablica globalna
  - ❷ Wykaz dostępu
  - ❸ Wykaz możliwości
  - ❹ Mechanizm zamka z kluczem
- ❹ Dynamiczne struktury ochrony
- ❺ Unieważnianie praw dostępu

# Plan wykładu

- ❶ Wstęp
- ❷ Podstawowe pojęcia
  - ❶ Ochrona
  - ❷ Bezpieczeństwo
  - ❸ Polityka bezpieczeństwa i mechanizmy
  - ❹ Domeny i wiedza konieczna
- ❸ Macierze dostępu
  - ❶ Tablica globalna
  - ❷ Wykaz dostępu
  - ❸ Wykaz możliwości
  - ❹ Mechanizm zamka z kluczem
- ❹ Dynamiczne struktury ochrony
- ❺ Unieważnianie praw dostępu

# Wstęp

Wraz z rozwojem systemów komputerowych narastała konieczność zapewnienia ochrony zasobów przed nieuprawnionym użyciem. Po raz pierwszy taki wymóg pojawił się na większą skalę w systemach wieloprogramowych i dotyczył ochrony pamięci. Na szeroką skalę sprawa bezpieczeństwa obecna jest w systemach wielodostępnych, których użytkownicy zainteresowani są kwestią poufności swoich danych.



# Ochrona

*Ochrona* jest mechanizmem służącym do kontrolowania dostępu procesów, wątków i użytkowników (ludzi) do zasobów systemu komputerowego. Dostarcza ona środków pozwalających określić rodzaj i zakres stosowanej kontroli, jak również pozwalających ją egzekwować. Dobrze zaimplementowana ochrona przyczynia się również do wykrywania i usuwania usterek obecnych w systemie.

# Bezpieczeństwo

Pojęciem szerszym znaczeniowo w stosunku do ochrony jest *bezpieczeństwo*, które możemy rozważyć w dwóch aspektach: niezawodnościowym (ang. safety) i związanym z kontrolą dostępu do danych (ang. security). Bezpieczeństwo niezawodnościowe ma na celu zapewnienie ciągłości usług oferowanych przez system komputerowy, nawet wtedy, kiedy doszło w nim do wystąpienia określonych usterek. Bezpieczeństwo związane z dostępem do danych ma na celu zapewnienie ich poufności. Oba te aspekty bezpieczeństwa są z sobą powiązane w sposób nierozzerwalny. Pojęcie bezpieczeństwa jest również związane z prawie każdą dziedziną informatyki, a nawet wykracza poza nią (np.: kwestie prawne).

# Polityka bezpieczeństwa

*Polityka bezpieczeństwa* jest zbiorem postanowień odnośnie zakresu stosowanych środków bezpieczeństwa i postępowania w przypadku ich naruszenia. Dobrze zdefiniowana polityka bezpieczeństwa swoim zakresem obejmuje nie tylko system komputerowy, ale całość środków instytucji w której ten system jest zainstalowany. Sformułowanie poprawnej polityki bezpieczeństwa nie może być obowiązkiem wyłącznie jednej osoby. Do jej określenia potrzebny jest zespół składający się z osób zarówno z wiedzą techniczną, jak i prawniczą.

# Mechanizmy

Zadaniem programisty systemowego jest dostarczenie środków, zwanych *mechanizmami* pozwalających na realizację polityki bezpieczeństwa w zakresie systemów komputerowych. Zbiór mechanizmów zaimplementowanych w systemie operacyjnym powinien być na tyle elastyczny, aby umożliwić realizację każdego z możliwych wariantów polityki bezpieczeństwa, dlatego ważne jest rozdzielenie tych dwóch pojęć. Dobrym przykładem rozdzielenia mechanizmu od polityki jest system uwierzytelniania użytkowników wykorzystujący moduły PAM, który został zaimplementowany między innymi w systemach Solaris, Linux, FreeBSD. Dalsza część wykładu będzie dotyczyła mechanizmów ochrony.

# Wiedza konieczna

System komputerowy składa się z zasobów i procesów. Jeśli zasoby powiążemy z operacjami, które mogą być na nich wykonywane, to możemy je traktować jako obiekty. Proces, który odwołuje się do takiego obiektu może wykonywać wyłącznie operacje, które są dla niego zdefiniowane. Co więcej, nie każdy proces powinien móc w każdej chwili wykonać wszystkie operacje przewidziane dla danego obiektu. W tej kwestii stosowana jest zasada *wiedzy koniecznej*, czyli *proces powinien móc wykonać na zasobie tylko te operacje, które są konieczne w danej chwili do kontynuacji jego pracy*.

## Domeny ochrony

Aby móc zastosować w praktyce schemat wiedzy koniecznej należy wprowadzić pojęcie *domeny ochrony*. Domena definiuje jakie zasoby są dostępne i jakie operacje może na nich wykonać proces w tej domenie pracujący. Innymi słowy domena określa prawa dostępu do obiektów. Domenę opisujemy jako zbiór par uporządkowanych  $\langle \textit{nazwa-obiektu}, \textit{zbiór-praw} \rangle$ . Domeny nie muszą być rozłączne, mogą mieć części wspólne.

## Macierze dostępu

Do modelowania systemu ochrony służą *macierze dostępu*. Wiersze tych macierzy zawierają nazwy domen, kolumny nazwy obiektów, natomiast każdy element zbiór praw dostępu od obiektu. Element macierzy o współrzędnych  $[i,j]$  określa jakie czynności mogą być wykonane w domenie o numerze  $i$  na obiekcie o numerze  $j$ . Na następnej planszy znajduje się przykładowa macierz dostępu. Wynika z niej np. że jeśli proces działa w domenie  $D_3$ , to może na obiekcie  $F_2$  wykonać jedynie operację odczytu.

# Macierze dostępu

<i>domena</i> \ <i>obiekt</i>	<i>F<sub>1</sub></i>	<i>F<sub>2</sub></i>	<i>F<sub>3</sub></i>	<i>cdrom</i>	<i>drukarka</i>
<i>D<sub>1</sub></i>	czytaj		czytaj		
<i>D<sub>2</sub></i>				czytaj	drukuj
<i>D<sub>3</sub></i>		czytaj	wykonaj		
<i>D<sub>4</sub></i>	czytaj pisz		czytaj pisz		



## Macierze dostępu

Z ilustracji umieszczonej na poprzedniej planszy wynika, że macierz dostępu jest macierzą rzadką. Choć istnieje efektywna reprezentacja takich struktur, bazująca na dynamicznych listach cyklicznych, to w opisywanych zastosowaniach jest mało praktyczna. Są za to stosowane inne formy implementacji macierzy dostępu.

## Tablica globalna

Macierz dostępu można zaimplementować w postaci tablicy składającej się z trójek uporządkowanych  $\langle \text{domena}, \text{obiekt}, \text{prawa} \rangle$ . Taka tablica ma dwie główne wady. Pierwszą wadą jest jej rozmiar, który może powodować, że nie będzie się mieściła w całości w pamięci operacyjnej, a drugi to powielanie informacji dotyczących domen i obiektów.

## Wykaz dostępu

Z każdym obiektem można związać wykaz domen i praw jakie w nich przysługują użytkownikowi obiektu. Jeśli w pewnej domenie nie przysługują żadne prawa do obiektu, to można ją pominąć. Dodatkowo schemat ten można rozszerzyć o pewne standardowe prawa, które obowiązują zawsze, niezależnie w jakiej domenie znajduje się użytkownik obiektu. Aby przyspieszyć wykonywanie operacji na obiekcie, najpierw powinien być przeszukiwany zbiór praw standardowych, a dopiero potem wykaz związany z danym obiektem.

## Wykaz możliwości

W systemie można stworzyć chronione obiekty, które reprezentują domeny. Jeśli użytkownik (proces) znajduje się w domenie, to jest związany z nią taki obiekt i dzięki niemu może on wykonać operacje na innych obiektach, do których dana domena zapewnia dostęp. Proces nigdy nie ma bezpośredniego dostępu do wykazu możliwości, ale zlecając systemowi operacyjnemu wykonanie operacji na obiekcie podaje jako parametr tej operacji obiekt wykazu możliwości. Jeśli zawiera on wskaźnik na obiekt, na którym ma być wykonana operacja, to jest ona realizowana, w przeciwnym wypadku następuje odmowa dostępu. Aby zapewnić ochronę wykazów możliwości należy zastosować środki pozwalające odróżnić je od innych obiektów. Rozwiązania tego problemu są dwa:

- 1 Każdy obiekt jest etykietowany. Etykieta jest znacznikiem bitowym, który pozwala odróżnić możliwości od innych obiektów. Do wprowadzenia takiego rozwiązania wystarczy jeden bit, ale w systemach je stosujących używa się kilku bitów na etykietę, dzięki temu możliwe jest odróżnianie nie tylko możliwości od innych zmiennych, ale również rozróżnianie zmiennych całkowitych, zmiennopozycyjnych, wskaźników. Etykieta stanowi więc reprezentację typu zmiennej na poziomie kodu maszynowego, a nie źródłowego. Etykiety mogą być interpretowane za pomocą środków programowych i sprzętowych.
- 2 Rozdzielenie przestrzeni adresowej procesu na część zawierającą jego kod i dane, oraz część zawierającą wykazy możliwości. To rozwiązanie szczególnie dobrze nadaje się do zastosowania w systemach stosujących segmentację pamięci.

## Wykaz możliwości

W systemie można stworzyć chronione obiekty, które reprezentują domeny. Jeśli użytkownik (proces) znajduje się w domenie, to jest związany z nią taki obiekt i dzięki niemu może on wykonać operacje na innych obiektach, do których dana domena zapewnia dostęp. Proces nigdy nie ma bezpośredniego dostępu do wykazu możliwości, ale zlecając systemowi operacyjnemu wykonanie operacji na obiekcie podaje jako parametr tej operacji obiekt wykazu możliwości. Jeśli zawiera on wskaźnik na obiekt, na którym ma być wykonana operacja, to jest ona realizowana, w przeciwnym wypadku następuje odmowa dostępu. Aby zapewnić ochronę wykazów możliwości należy zastosować środki pozwalające odróżnić je od innych obiektów. Rozwiązania tego problemu są dwa:

- 1 Każdy obiekt jest etykietowany. Etykieta jest znacznikiem bitowym, który pozwala odróżnić możliwości od innych obiektów. Do wprowadzenia takiego rozwiązania wystarczy jeden bit, ale w systemach je stosujących używa się kilku bitów na etykietę, dzięki temu możliwe jest odróżnianie nie tylko możliwości od innych zmiennych, ale również rozróżnianie zmiennych całkowitych, zmiennopozycyjnych, wskaźników. Etykieta stanowi więc reprezentację typu zmiennej na poziomie kodu maszynowego, a nie źródłowego. Etykiety mogą być interpretowane za pomocą środków programowych i sprzętowych.
- 2 Rozdzielenie przestrzeni adresowej procesu na część zawierającą jego kod i dane, oraz część zawierającą wykazy możliwości. To rozwiązanie szczególnie dobrze nadaje się do zastosowania w systemach stosujących segmentację pamięci.

## Wykaz możliwości

W systemie można stworzyć chronione obiekty, które reprezentują domeny. Jeśli użytkownik (proces) znajduje się w domenie, to jest związany z nią taki obiekt i dzięki niemu może on wykonać operacje na innych obiektach, do których dana domena zapewnia dostęp. Proces nigdy nie ma bezpośredniego dostępu do wykazu możliwości, ale zlecając systemowi operacyjnemu wykonanie operacji na obiekcie podaje jako parametr tej operacji obiekt wykazu możliwości. Jeśli zawiera on wskaźnik na obiekt, na którym ma być wykonana operacja, to jest ona realizowana, w przeciwnym wypadku następuje odmowa dostępu. Aby zapewnić ochronę wykazów możliwości należy zastosować środki pozwalające odróżnić je od innych obiektów. Rozwiązania tego problemu są dwa:

- 1 Każdy obiekt jest etykietowany. Etykieta jest znacznikiem bitowym, który pozwala odróżnić możliwości od innych obiektów. Do wprowadzenia takiego rozwiązania wystarczy jeden bit, ale w systemach je stosujących używa się kilku bitów na etykietę, dzięki temu możliwe jest odróżnianie nie tylko możliwości od innych zmiennych, ale również rozróżnianie zmiennych całkowitych, zmiennopozycyjnych, wskaźników. Etykieta stanowi więc reprezentację typu zmiennej na poziomie kodu maszynowego, a nie źródłowego. Etykiety mogą być interpretowane za pomocą środków programowych i sprzętowych.
- 2 Rozdzielenie przestrzeni adresowej procesu na część zawierającą jego kod i dane, oraz część zawierającą wykazy możliwości. To rozwiązanie szczególnie dobrze nadaje się do zastosowania w systemach stosujących segmentację pamięci.

## Mechanizm zamka i klucza

Ten mechanizm jest rozwiązaniem pośrednimi między wykazem możliwości, a wykazem dostępu. Każda domena dysponuje zbiorem unikatowych wzorców binarnych nazywanych *kluczami*. Również z każdym obiektem związany jest zbiór unikatowych wzorców binarnych nazywanych *zamkami*. Jeśli proces pracujący w określonej domenie chce wykonać na obiekcie jakąś operację, to może dokonać tego tylko wtedy, gdy w tej domenie znajduje się klucz pasujący do określonego zamka w wykazie należącym do obiektu.

## Dynamiczne struktury ochrony

Większość złożonych procesów podczas pracy korzysta z dużej liczby obiektów i dodatkowo zmienia w trakcie działania sposób korzystania z nich. W systemach ochrony, w których zawartość domen jest statyczna oznaczałoby to konieczność umieszczenia w domenie wszystkich praw dostępu, jakie mogą być przydatne podczas całego cyklu wykonania danemu procesowi. Takie rozwiązanie stanowi jawne naruszenie reguły wiedzy koniecznej. Aby do tego nie dopuścić należy umożliwić dynamiczne tworzenie nowych domen lub zmianę zawartości istniejących domen i umożliwić procesom przełączanie między domenami podczas pracy. To ostatnie rozwiązanie wymaga potraktowania domen jako obiektów i włączenia ich do macierzy dostępu. Dla domen musi być też sformułowane nowe prawo *przełącz*, które pozwala procesowi na przełączanie się między domenami. Na następnej planszy znajduje się ilustracja macierzy dostępu z domenami.



# Dynamiczne struktury ochrony

<i>obiekt</i>	<i>F<sub>1</sub></i>	<i>F<sub>2</sub></i>	<i>F<sub>3</sub></i>	<i>cdrom</i>	<i>drukarka</i>	<i>D<sub>1</sub></i>	<i>D<sub>2</sub></i>	<i>D<sub>3</sub></i>	<i>D<sub>4</sub></i>
<i>domena</i>									
<i>D<sub>1</sub></i>	czytaj		czytaj				przełącz		
<i>D<sub>2</sub></i>				czytaj	drukuj			przełącz	przełącz
<i>D<sub>3</sub></i>		czytaj	wykonaj						
<i>D<sub>4</sub></i>	czytaj pisz		czytaj pisz			przełącz			

## Dynamiczne struktury ochrony

Dopełnieniem włączania domen jako obiektów do macierzy dostępu jest zezwolenie na modyfikację zawartości domen. Można to zrealizować wprowadzając trzy dodatkowe prawa: *kopiowania*, *właściciela* i *kontroli*. Prawo kopiowania występuje w połączeniu z innymi prawami. Jeśli użytkownik (proces) pracuje w domenie, w której występuje prawo kopiowania, to może on przenieść prawo, które występuje wraz z prawem kopiowania do innej domeny. To przeniesienie najczęściej odbywa się na jeden z dwóch sposobów:

- prawo jest kopiowane do domeny docelowej, a z domeny źródłowej usuwane,
- prawo jest kopiowane, ale bez prawa kopiowania.

Ostatni wariant nazywamy *ograniczonym kopiowaniem*. W macierzach dostępu prawo kopiowania oznaczane jest gwiazdką i obowiązuje w zakresie kolumny, w której występuje. Dwie następne plansze pokazują sposób funkcjonowania prawa ograniczonego kopiowania.

## Dynamiczne struktury ochrony

Dopełnieniem włączania domen jako obiektów do macierzy dostępu jest zezwolenie na modyfikację zawartości domen. Można to zrealizować wprowadzając trzy dodatkowe prawa: *kopiowania*, *właściciela* i *kontroli*. Prawo kopiowania występuje w połączeniu z innymi prawami. Jeśli użytkownik (proces) pracuje w domenie, w której występuje prawo kopiowania, to może on przenieść prawo, które występuje wraz z prawem kopiowania do innej domeny. To przeniesienie najczęściej odbywa się na jeden z dwóch sposobów:

- prawo jest kopiowane do domeny docelowej, a z domeny źródłowej usuwane,
- prawo jest kopiowane, ale bez prawa kopiowania.

Ostatni wariant nazywamy *ograniczonym kopiowaniem*. W macierzach dostępu prawo kopiowania oznaczane jest gwiazdką i obowiązuje w zakresie kolumny, w której występuje. Dwie następne plansze pokazują sposób funkcjonowania prawa ograniczonego kopiowania.

## Dynamiczne struktury ochrony

Dopełnieniem włączania domen jako obiektów do macierzy dostępu jest zezwolenie na modyfikację zawartości domen. Można to zrealizować wprowadzając trzy dodatkowe prawa: *kopiowania*, *właściciela* i *kontroli*. Prawo kopiowania występuje w połączeniu z innymi prawami. Jeśli użytkownik (proces) pracuje w domenie, w której występuje prawo kopiowania, to może on przenieść prawo, które występuje wraz z prawem kopiowania do innej domeny. To przeniesienie najczęściej odbywa się na jeden z dwóch sposobów:

- prawo jest kopiowane do domeny docelowej, a z domeny źródłowej usuwane,
- prawo jest kopiowane, ale bez prawa kopiowania.

Ostatni wariant nazywamy *ograniczonym kopiowaniem*. W macierzach dostępu prawo kopiowania oznaczane jest gwiazdką i obowiązuje w zakresie kolumny, w której występuje. Dwie następne plansze pokazują sposób funkcjonowania prawa ograniczonego kopiowania.

# Dynamiczne struktury ochrony

<i>domena</i> \ <i>obiekt</i>	$F_1$	$F_2$	$F_3$
$D_1$	wykonaj		pisz*
$D_2$	wykonaj	czytaj*	wykonaj
$D_3$	wykonaj		

# Dynamiczne struktury ochrony

<i>domena</i> \ <i>obiekt</i>	$F_1$	$F_2$	$F_3$
$D_1$	wykonaj		pisz*
$D_2$	wykonaj	czytaj*	wykonaj
$D_3$	wykonaj	czytaj	

# Dynamiczne struktury ochrony

Jeśli proces działa w domenie w której obowiązuje prawo właściciela dla określonego obiektu, to może on usuwać i dodawać nowe prawa do innych domen. To prawo, podobnie jak prawo kopiowania działa w obrębie kolumn. Następane dwie plansze ilustrują sposób działania takiego prawa.

# Dynamiczne struktury ochrony

<i>domena</i> \ <i>obiekt</i>	$F_1$	$F_2$	$F_3$
$D_1$	właściciel wykonaj		pisz
$D_2$		właściciel czytaj*	właściciel czytaj* pisz*
$D_3$	wykonaj		



# Dynamiczne struktury ochrony

<i>domena</i> \ <i>obiekt</i>	$F_1$	$F_2$	$F_3$
$D_1$	właściciel wykonaj		
$D_2$		właściciel czytaj* pisz*	właściciel czytaj* pisz*
$D_3$		pisz	pisz

# Dynamiczne struktury ochrony

Uzupełnieniem dwóch przedstawionych praw jest prawo *kontroli* pozwalające modyfikować domeny. Jest ono stosowane tylko w odniesieniu do obiektów domen. W odniesieniu do macierzy dostępu oznacza to, że proces działający w domenie w której zostało umieszczone prawo kontroli innej domeny może w niej dodawać i usuwać prawa dla wszystkich należących do niej obiektów. Innymi słowy, prawo kontroli działa w obrębie wierszy. Następną plansza ilustruje działanie tego prawa.

# Dynamiczne struktury ochrony

<i>obiekt</i>	$F_1$	$F_2$	$F_3$	<i>cdrom</i>	<i>drukarka</i>	$D_1$	$D_2$	$D_3$	$D_4$
<i>domena</i>									
$D_1$	czytaj		czytaj				przełącz		
$D_2$				czytaj	drukuj			przełącz	kontroluj przełącz
$D_3$		czytaj	wykonaj						
$D_4$	pisz		pisz			przełącz			

## Unieważnianie praw dostępu

Zezwolenie na modyfikację zawartości domen wymusza zastosowanie unieważniania praw dostępu. Implementując taką możliwość należy rozstrzygnąć następujące kwestie:

- Czy prawa mają być unieważniane natychmiast, czy z opóźnieniem?
- Czy unieważnienie dotyczy wszystkich użytkowników (unieważnienie ogólne), czy tylko określonej grupy (unieważnienie wybiórcze)?
- Czy unieważniane są wszystkie prawa (unieważnienie całkowite), czy tylko część z nich (unieważnienie częściowe)?
- Czy odbieramy prawa użytkownikom permanentnie (unieważnienie na stałe), czy też istnieje możliwość ich przywrócenia (unieważnienie czasowe)?

## Unieważnianie praw dostępu

Zezwolenie na modyfikację zawartości domen wymusza zastosowanie unieważniania praw dostępu. Implementując taką możliwość należy rozstrzygnąć następujące kwestie:

- Czy prawa mają być unieważniane natychmiast, czy z opóźnieniem?
- Czy unieważnienie dotyczy wszystkich użytkowników (unieważnienie ogólne), czy tylko określonej grupy (unieważnienie wybiórcze)?
- Czy unieważniane są wszystkie prawa (unieważnienie całkowite), czy tylko część z nich (unieważnienie częściowe)?
- Czy odbieramy prawa użytkownikom permanentnie (unieważnienie na stałe), czy też istnieje możliwość ich przywrócenia (unieważnienie czasowe)?

## Unieważnianie praw dostępu

Zezwolenie na modyfikację zawartości domen wymusza zastosowanie unieważniania praw dostępu. Implementując taką możliwość należy rozstrzygnąć następujące kwestie:

- Czy prawa mają być unieważniane natychmiast, czy z opóźnieniem?
- Czy unieważnienie dotyczy wszystkich użytkowników (unieważnienie ogólne), czy tylko określonej grupy (unieważnienie wybiórcze)?
- Czy unieważniane są wszystkie prawa (unieważnienie całkowite), czy tylko część z nich (unieważnienie częściowe)?
- Czy odbieramy prawa użytkownikom permanentnie (unieważnienie na stałe), czy też istnieje możliwość ich przywrócenia (unieważnienie czasowe)?

## Unieważnianie praw dostępu

Zezwolenie na modyfikację zawartości domen wymusza zastosowanie unieważniania praw dostępu. Implementując taką możliwość należy rozstrzygnąć następujące kwestie:

- Czy prawa mają być unieważniane natychmiast, czy z opóźnieniem?
- Czy unieważnienie dotyczy wszystkich użytkowników (unieważnienie ogólne), czy tylko określonej grupy (unieważnienie wybiórcze)?
- Czy unieważniane są wszystkie prawa (unieważnienie całkowite), czy tylko część z nich (unieważnienie częściowe)?
- Czy odbieramy prawa użytkownikom permanentnie (unieważnienie na stałe), czy też istnieje możliwość ich przywrócenia (unieważnienie czasowe)?

## Unieważnianie praw dostępu

Zezwolenie na modyfikację zawartości domen wymusza zastosowanie unieważniania praw dostępu. Implementując taką możliwość należy rozstrzygnąć następujące kwestie:

- Czy prawa mają być unieważniane natychmiast, czy z opóźnieniem?
- Czy unieważnienie dotyczy wszystkich użytkowników (unieważnienie ogólne), czy tylko określonej grupy (unieważnienie wybiórcze)?
- Czy unieważniane są wszystkie prawa (unieważnienie całkowite), czy tylko część z nich (unieważnienie częściowe)?
- Czy odbieramy prawa użytkownikom permanentnie (unieważnienie na stałe), czy też istnieje możliwość ich przywrócenia (unieważnienie czasowe)?



## Unieważnienie praw dostępu

Jeśli macierz dostępu jest zaimplementowana jako wykaz dostępu, to unieważnianie stosunkowo łatwo jest zaimplementować i jest ono natychmiastowe, może być ogólne lub wybiórcze, całkowite lub częściowe, stałe lub czasowe. Wykazy możliwości natomiast są rozproszone po całym systemie, co utrudnia przeprowadzanie unieważniania. Dlatego stosuje się jeden z następujących schematów:

## Wtórne pozyskiwanie

Możliwości są okresowo usuwane ze wszystkich domen. Proces, który potrzebuje danej możliwości powinien wykryć jej usunięcie i spróbować ją ponownie pozyskać. Jeśli została ona usunięta to nie będzie mógł jej uzyskać.

## Wskaźniki zwrotne

Każdy obiekt posiada listę wskaźników na związane z nim możliwości. Jeśli należy przeprowadzić unieważnienie, to dokonuje się modyfikacji odpowiedniego wskaźnika, tak aby wskazywał on na inną możliwość. Schemat ten jest elastyczny i ogólny, ale kosztowny w realizacji.

## Sposób pośredni

Możliwości powiązane są z obiektami za pomocą elementów tablicy globalnej. Jeśli należy przeprowadzić unieważnienie, to wyszukuje się i usuwa zawartość odpowiedniego elementu tej tablicy. Tak opróżniony element można wykorzystać później dla innych możliwości. Ten schemat nie pozwala na częściowe unieważnianie.

## Klucze

Z każdym obiektem wiązany jest *klucz główny*, który jest unikatowym wzorcem binarnym (ang. binary pattern). Ten klucz może być ustalany lub zmieniany za pomocą *operacji ustawienia klucza* (ang. set-key). Tę operację może wykonywać tylko ustalona grupa procesów (np. właściciel obiektu do którego zmieniane są prawa). Każda możliwość podczas tworzenia „stemplowana” jest bieżącą wartością klucza. Zanim zostanie wykonana na obiekcie operacja, na którą zezwala możliwość porównywana jest wartość klucza tej możliwości z kluczem głównym obiektu. Jeśli te wartości się nie zgadzają to następuje odmowa dostępu do obiektu. Aby więc unieważnić prawa dostępu wystarczy zmienić wartość klucza głównego. Ten schemat nie pozwala na częściowe odbieranie praw. Można usunąć tę niedogodność stosując nie jeden klucz główny, a listę takich kluczy. Jeszcze elastyczniejsze rozwiązanie zakłada zgromadzenie wszystkich kluczy głównych w tablicy globalnej. Dzięki temu jeden klucz mógłby być powiązany z kilkoma obiektami jednocześnie. Ustalenie, czy możliwość jest prawomocna wymagałoby odnalezienia w tablicy globalnej klucza o takiej samej wartości, jak jej klucz.

# Pytania

?

Dziękuję Państwu za uwagę!