Безпека операційних систем і комп'ютерних мереж

Лекція 10

Довірені UNIX-подібні системи (зокрема, Trusted Solaris і SELinux)

Trusted Solaris: додаткові можливості

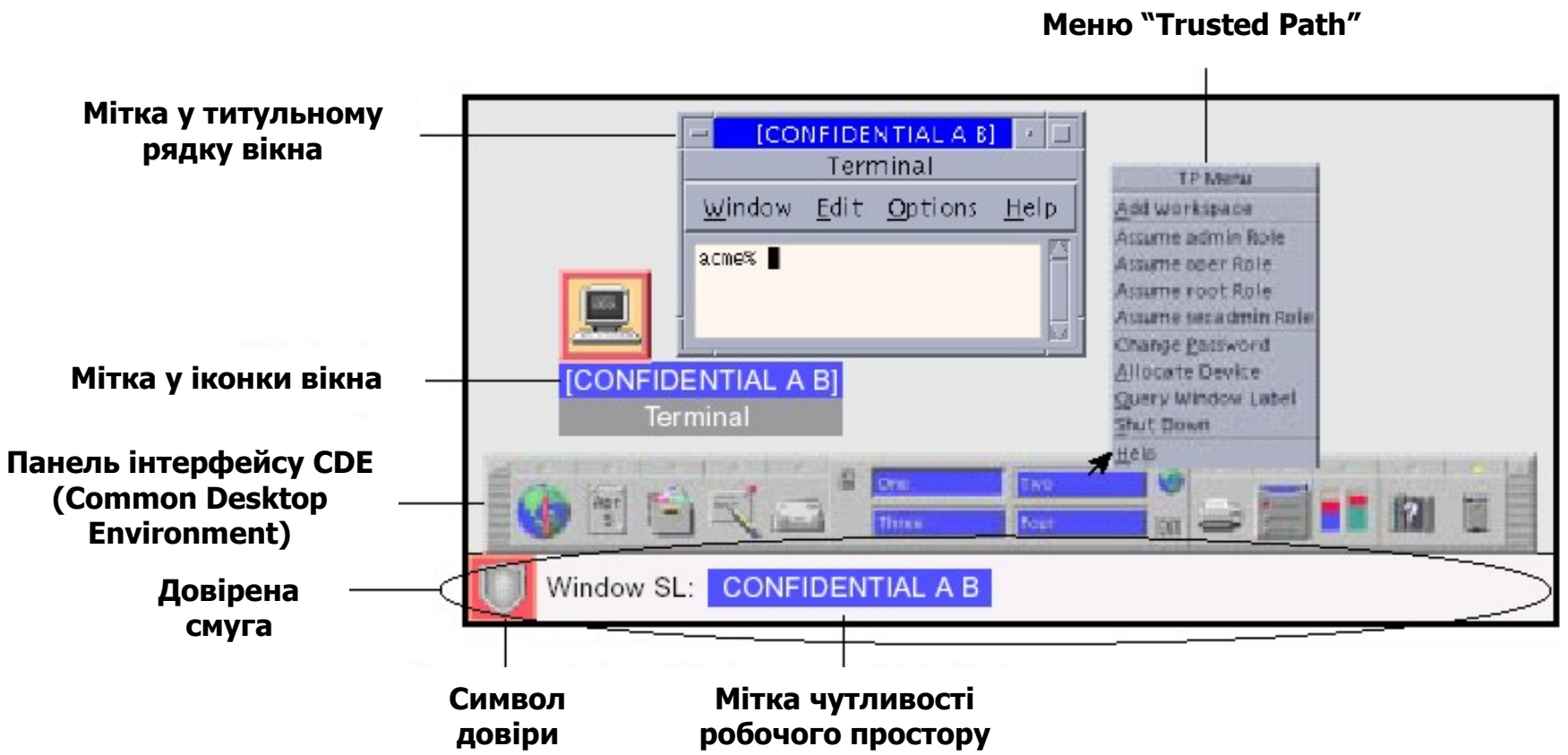
- обмеження доступу до КЗЗ
- додатковий захист паролів, що утруднює їх викрадення
- захист інформації в системі шляхом удосконаленого контролю доступу
- забезпечення аудиту
- запобігання підміні програм
- захист локальних периферійних пристроїв від НСД:
  - віддалені користувачі не можуть керувати пристроями, такими як мікрофони або пристрої запису на магнітну стрічку; для того, щоби використовувати ці пристрої користувачі повинні увійти в систему локально;
  - лише користувачі зі спеціальною авторизацією можуть отримувати доступ до пристроїв із знімними носіями.

Мандатне керування доступом у Trusted Solaris

Допуски та мітки:

- класифікація (наприклад, для урядових організацій США):
  - UNCLASSIFIED (нетаємна)
  - CONFIDENTIAL (конфіденційна)
  - SECRET (таємна)
  - TOP SECRET (цілком таємна)
- категорії
  - робоча група, відділ, проект або тематика

Типове середовище, в якому відображаються мітки



Приклади відношень міток

Мітка 1	**Відношення**	**Мітка 2**
Top Secret A B	(строго) домінує	Secret A
Top Secret A B	(строго) домінує	Secret A B
Top Secret A B	(строго) домінує	Top Secret A
Top Secret A B	домінує (дорівнює)	Top Secret A B
Top Secret A B	не пов’язана	Top Secret C
Top Secret A B	не пов’язана	Secret C
Top Secret A B	не пов’язана	Secret A B C

Правила моделі конфіденційності

- У транзакції **зчитування** мітка суб’єкта повинна домінувати над міткою об’єкта
  - Це правило гарантує, що рівень довіри до суб’єкта виконує вимоги щодо доступу до об’єкта, і що мітка суб’єкта включає всі категорії, яким дозволений доступ до об’єкта
- У транзакції **записування**, тобто коли суб’єкт створює або модифікує об’єкт, мітка об’єкта повинна домінувати над міткою суб’єкта
  - Це правило не дає можливості суб’єкту знизити рівень мітки об’єкта

Відокремлення поміченої інформації у Trusted Solaris

- Середовище Trusted Solaris відокремлює інформацію, що помічена різними мітками, такими засобами:
  - дозволяючи користувачам обирати одно- або багаторівневі сеанси
  - надаючи помічені робочі простори (*labelled workspaces*)
  - зберігаючи файли в окремих каталогах відповідно до міток
  - застосовуючи мандатне керування доступом до трансакцій електронною поштою
  - очищуючи об’єкти перед їх повторним використанням

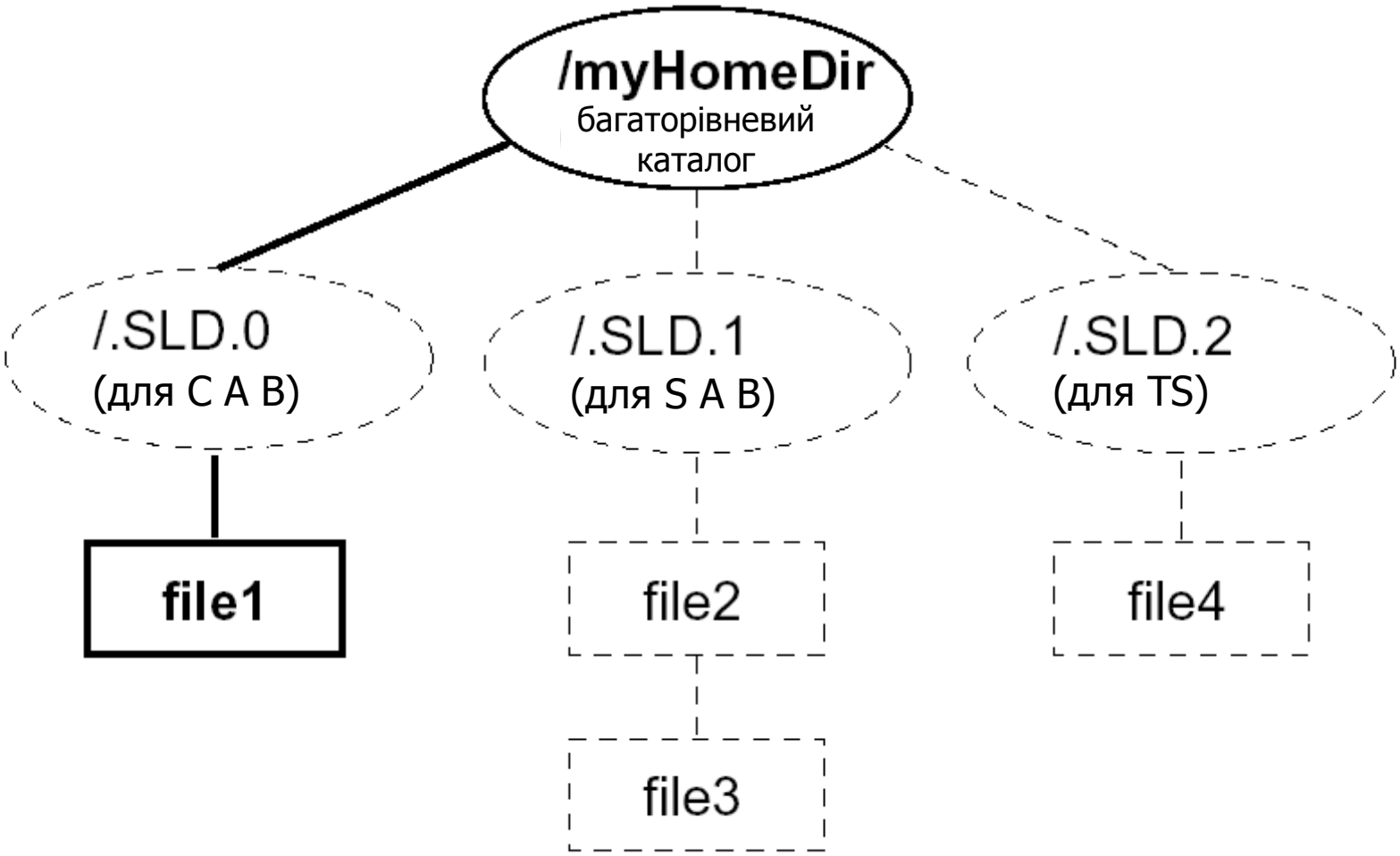
Одно- та багаторівневі сеанси

- Коли користувач відкриває сеанс роботи в Trusted Solaris, він указує, чи буде він працювати з одною міткою, чи з різними мітками (якщо останнє йому дозволено)
- Після цього користувач встановлює рівень безпеки, на якому він планує працювати:
  - для однорівневого сеансу – мітку
  - для багаторівневого сеансу – допуск
- В однорівневому сеансі користувач може отримати доступ лише до тих об’єктів, над мітками яких домінує мітка сеансу
- В багаторівневому сеансі користувач має доступ до інформації на різних рівнях чутливості, доки вони дорівнюють або нижчі за допуск сеансу
- В середовищі Trusted Solaris користувач може вказати різні мітки для різних робочих просторів

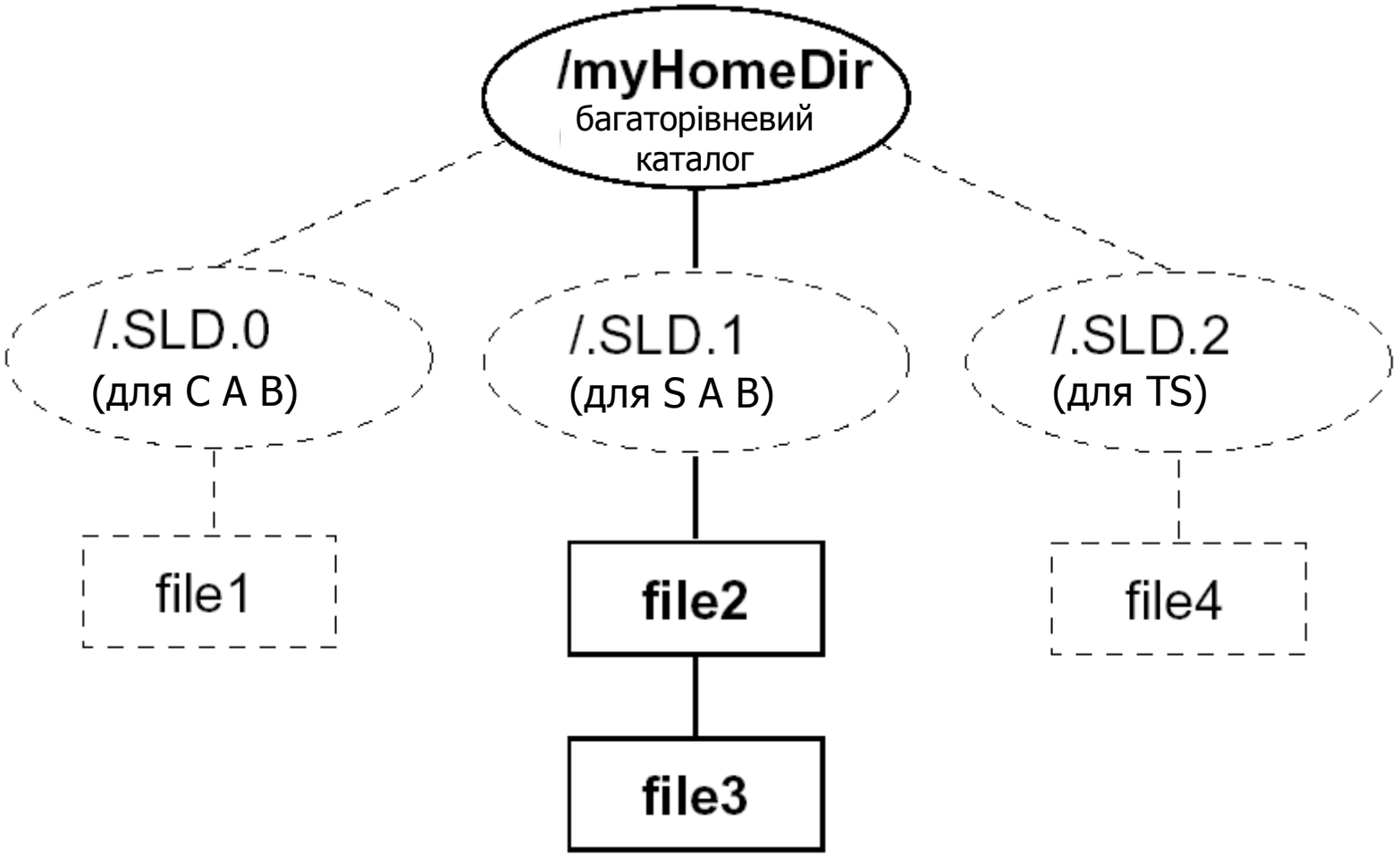
Зберігання файлів в окремих каталогах відповідно до міток

- Trusted Solaris надає два спеціальних типи каталогів для зберігання файлів і підкаталогів з різними мітками та їх відокремлення:
  - **Багаторівневий каталог** (multilevel directory, MLD) – спеціальний тип каталогу, що прозоро зберігає інформацію за її мітками в окремих підкаталогах, що називаються одноріневими
  - **Одноріневий каталог** (single-level directory, SLD) – це прихований підкаталог всередині багаторівневого каталогу, що містить файли та, можливо, підкаталоги, які мають лише однакові мітки
- Коли користувачі намагаються переглянути список файлів або отримати доступ до файлів у багаторівневому каталозі (як за допомогою прикладних програм, таких як File Manager, так і застосовуючи стандартну командну оболонку і стандартні команди), видимими у списку і доступними будуть лише ті файли, які відповідають поточній мітці сеансу користувача

Багаторівневий каталог: сеанс з міткою CONFIDENTIAL A B



Багаторівневий каталог: сеанс з міткою SECRET A B



Адміністрування безпеки у середовищі Trusted Solaris

- **Роль** – це спеціальний обліковий запис, який надає користувачеві доступ до певних утиліт з авторизацією, привілеями та ефективними ідентифікаторами UID/GID, необхідними для виконання спеціальних задач
- Програмний механізм, що називається **авторизація**, надає окремому користувачеві право долати певне окреме обмеження політики безпеки
- Механізм подолання обмежень, що називається **привілеєм**, асоціюється з певними програмами і дозволяється лише окремим користувачам
- Адміністратор забезпечує доступ, призначаючи обліковому запису користувача один чи кілька **профілів виконання** – спеціальних пакетів дій, команд та авторизацій графічного інтерфейсу CDE (Common Desktop Environment)
- Адміністратор може призначити користувачеві в якості командної оболонки за умовчанням **профільну оболонку** – спеціальну версію оболонки Bourne shell, що забезпечує доступ лише до обмеженого набору програм і можливостей

Наперед визначені адміністративні ролі у Trusted Solaris

- **суперкористувач (root)**
  - використовується, головним чином, лише для первинної інсталяції середовища
- **адміністратор безпеки (security administrator)**
  - використовується для виконання задач, безпосередньо пов’язаних з безпекою, таких як призначення міток або аудит дій користувачів
- **системний адміністратор (system administrator)**
  - використовується для виконання стандартних задач адміністрування, таких як встановлення тих частин облікових записів користувачів, які не пов’язані з безпекою
- **системний оператор (system operator)**
  - використовується для створення резервних копій, адміністрування принтерів, монтування знімних носіїв
- **первинний адміністратор (primary administrator)**
  - використовується для виконання будь-яких задач, що вимагають привілеїв поза межами можливостей інших ролей

Security Enhanced Linux (SE Linux)

- SELinux — набір технологій розширення системи безпеки Linux
- Основу набора складають три технології:
  - мандатне керування доступом
  - ролевий доступ RBAC
  - система типів (доменів)
- SELinux містить:
  - модулі ядра
  - поділювані бібліотеки для створення застосувань, що використовують SELinux
  - утиліти
  - інші файли
- SELinux можна встановити з будь-яким дистрибутивом Linux, починаючи з ядра версії 2.2.x
- Архітектурно SELinux слідує трьом принципам, що сприяють максимально безболісній інтеграції SELinux у Linux-системи:
  - паралельне співіснування з класичною системою безпеки Linux
  - незалежність від класичної системи безпеки Linux
  - пріоритет заборон класичної системи безпеки Linux над SELinux (те, що заборонено класичною системою безпеки, не може бути дозволено SELinux)

Історія і застосування SE Linux

- SELinux був започаткований в Агентстві національної безпеки США
- Безпосередньо розробку вели компанії Network Associates і MITRE
- SELinux був випущений у вигляді загальнодоступного відкритого програмного продукту у грудні 2000 року
- Для систем з ядрами 2.2 и 2.4 SELinux випускався у вигляді заплати
- Після введення модулю Linux Security Module (LSM) в ядрі 2.4 була випущена версія SELinux для LSM
- В ядрі 2.6 SELinux також підтримується LSM. Крім того, деякі елементи SELinux включені в саме ядро
  - Однак, якщо ОС має ядро 2.6, це не означає, що там обов'язково є SELinux або що його легко активувати. Це означає лише, що встановити SELinux буде легше.
- SELinux повністю підтримується у дистрибутивах RedHat Enterprise Linux 4, Fedora Core 2 і 3, Gentoo Hardened Linux, і усіх наступних
  - В кожному з цих дистрибутивів є параметр включення SELinux під час інсталяції ОС

Суб’єкти, об’єкти і дії SE Linux

- Три головні концепції моделі безпеки SELinux:
  - Суб’єкти (subject)
    - Процеси, що діють як від імені окремих користувачів, так і самостійно (серверні процеси)
  - Об’єкти (object)
    - Об’єкти файлової системи (файли, каталоги, посилання)
    - Процеси (коли один процес-суб’єкт виконує дії з іншим процесом, який виступає в ролі об’єкта
    - Дескриптори файлів (у тому числі сокети)
    - Об’єкти міжпроцесової взаємодії, не пов’язані з дескрипторами файлів
  - Дії (action)
    - Будь-які операції, які суб’єкт може виконати над об’єктом
- З точки зору SELinux всю роботу системи можна описати як виконання суб’єктами дій над об’єктами
  - Основна частина роботи системи безпеки полягає в ухваленні рішення про те, чи має право цей суб’єкт виконати цю дію над цим об’єктом
- Рішення про припустимість чи неприпустимість дії система приймає на основі політик

Політики SE Linux

- Політики – це спосіб описання поведінки системи безпеки, що абстрагується від таких понять низького рівня як вектори доступу
- Формування політик безпеки в SELinux подібно до програмування:
  - Політика описується спеціальною мовою
  - Файл політики компілюється у бінарний модуль (для цього використовується утиліта make)
  - Скомпільований файл динамічно завантажується в ядро операційної системи
- Політики SELinux дозволяють визначити, яке рішення слід прийняти для певних операцій і класів операцій:
  - allow (дозволити операцію)
  - auditallow (занести операцію в журнал, незалежно від того, дозволена вона чи ні)
  - dontaudit (заборонити операцію, але не вносити дані про спробу в журнал)
- В SELinux політика дозволу операцій тісно пов’язана з їх журналюванням
- Логіка сумісної роботи така:
  - якщо операція дозволена, вона заноситься в журнал лише тоді, коли прийнято рішення auditallow
  - якщо операція не дозволена, вона не заноситься в журнал лише тоді, коли прийнято рішення dontaudit

Контексти безпеки SE Linux (1/3)

- Права суб'єктів і об'єктів визначаються в SELinux контекстами безпеки, що складаються з:
  - ідентифікатора
    - Ідентифікатор суб'єкта – це ідентифікатор користувача SELinux, що створив процес-суб'єкт
    - Ідентифікатор об'єкта – це ідентифікатор користувача-власника об'єкта (як правило, це користувач, що створив об'єкт)
  - ролі
    - Ролі – це набори привілеїв
    - У будь-який момент кожний користувач може виступати в одній з доступних йому ролей
    - Ролі дозволяють надавати користувачеві додаткові привілеї, не втрачаючи його ідентичності (на відміну від команди su)
    - Політика безпеки SELinux може накладати обмеження на кількість ролей, доступних процесу, і визначати правила переходу з однієї ролі в іншу. Не всякий перехід є припустимим.
    - Ролі частіше використовуються суб'єктами, ніж об'єктами, а деякі об'єкти (наприклад, дискові файли) взагалі не потребують ролей. Таким об'єктам надають «пусті ролі»
  - типи об'єкта
    - Типи об'єднують групи суб'єктів і об'єктів, надаючи їм певні права
    - Важлива функція типів – обмеження можливих дій суб'єкта над об'єктом. Типи іноді називають «пісочницями SELinux» (SELinux sandbox)

Контексти безпеки SE Linux (2/3)

- В SELinux також застосовується термін «домен»
  - На відміну від класичних моделей безпеки систем, в SELinux поняття «тип» і «домен» є майже синонімами
  - Про типи кажуть, коли мова йде про об’єкти, а про домени – коли мова йде про суб’єкти
- Домени можна описати як множини процесів (суб’єктів), що мають однакові права
  - Наприклад, Web-сервер Apache належить домену (типу) httpd_t і має усі права, що пов’язані з цим доменом. До цього ж типу належать файли, до яких демон httpd повинен мати повний доступ.
- В SELinux діє механізм примусового присвоєння типів (type enforcement)
  - Кожний процес належить певному типу (домену), що визначає права цього процесу
  - Без примусового присвоєння типів система мандатного керування доступом не могла б функціонувати

Контексти безпеки SE Linux (3/3)

- Кожний суб'єкт і об'єкт має власний контекст безпеки, якому відповідає ідентифікатор безпеки SID
- Система контекстів сильно відрізняється від традиційної системи облікових записів в ОС Linux
  - Відповідно до принципу незалежності SELinux підтримує таблицю контекстів безпеки, незалежну від таблиці облікових записів Linux
  - Можливо відображення кількох облікових записів Linux на один обліковий запис SELinux
  - Зміни в облікових записах Linux не впливають на параметри безпеки SELinux
- Модель безпеки SELinux вимагає, щоби кожний файл в системі був пов'язаний з певним контекстом безпеки
  - Для цього під час інсталяції завжди виконується маркування (labeling) об'єктів файлової системи
  - Відповідно до принципу незалежності маркування файлів не впливає на маски доступу до файлів
  - Відповідно до принципу пріоритету традиційної системи безпеки заборони, накладені маскою доступу, відміняють дозволи SELinux

Операції доступу і операції перетворення

- Операції SELinux поділяються на
  - операції доступу (access)
    - наприклад, відкриття та зчитування даних з файлу
  - операції перетворення (transition)
    - операції, що пов’язані зі зміною контексту безпеки об’єктів
- Внаслідок операції перетворення об’єкт отримує контекст безпеки, відмінний від того, який він отримав би за умовчанням
  - Приклад 1
    - Під час створення файл за умовчанням отримує той самий контекст безпеки, що й каталог, в якому цей файл створено (файл успадковує контекст безпеки)
    - Застосовуючи операцію перетворення, файлу можна надати інший контекст безпеки
  - Приклад 2
    - Процес, що здійснює авторизацію користувача в системі, повинен мати змогу надати користувачеві інший рівень привілеїв, відмінний від його власного
    - Для цього він має виконати операцію перетворення
  - Приклад 3
    - Важливою операцією перетворення в SELinux є операція перетворення типів, за якої процес переходить з одного домену в інший
    - З типами пов’язані групи ролей, тому процес зміни типу зазвичай виконують шляхом зміни ролі

Обмеження SE Linux

- Модель безпеки SELinux вирішує головним чином лише проблеми керування доступом користувачів до об’єктів операційної системи
- Іншим класом проблем безпеки є порушення цілісності виконуваного коду
  - В ході подібних атак зловмисник впроваджує свій код у програму, змушуючи її виконувати його команди
  - Класичний приклад – переповнення стеку
  - Вразливості для впровадження стороннього коду виявляють у програмах регулярно
- Для вирішення проблем порушення цілісності виконуваного коду розроблені розширення безпеки Linux, що дозволяють мінімізувати наслідки атак
  - Система PAX
  - Bastille Linux

Система PaX *(pax.grsecurity.net)*

- PaX застосовує технологію випадкового розташування програми в адресному просторі (address space layout randomization, ASLR)
  - Система PaX, що функціонує на рівні ядра Linux, відстежує запуск певних процесів і виконує в них «перестановки», керуючись принципом випадковості
  - Linux застосовує технологію ASLR, починаючи з ядра версії 2.6.12 (2005 рік). Але це “простий” варіант ASLR. PaX реалізує більш складний і повний варіант (хоча деякі сучасні дистрибутиви, зокрема ті, що мають слово Hardened у назві, застосовують повний варіант без додаткових патчів.
- Інший напрям розвитку PaX – це запобігання виконанню невиконуваних сторінок пам’яті
- Основна процесорна архітектура для PaX – IA-32, існують реалізації і для інших платформ

Проект grsecurity *(www.grsecurity.org)*

- Система PaX є частиною проекту grsecurity
- Проект має ті ж цілі, що й SELinux
  - «Якщо система безпеки не дружня по відношенню до користувача, вона марна»
  - Реалізацією цього принципу можна вважати читабельну з точки зору людини систему конфігурації
- Система grsecurity реалізує рольовий контроль доступу
  - Це, мабуть, у найближчому майбутньому стане стандартом для Unix-систем
- Головні особливості grsecurity:
  - Система аудиту ядра
  - Особливі засоби захисту поділюваної пам’яті (один з механізмів міжпроцесової взаємодії
  - Жорсткий контроль операції chroot

Bastille Linux *(www.bastille-linux.org)*

- Bastille Linux – це набор інструментів безпеки, що увібрав у себе багаторічний досвід експлуатації Linux-систем
- Bastille Linux має широку підтримку
  - Bastille Linux підтримується дистрибутивами Fedora Core, Red Hat Enterprise, SuSE, SuSE Enterprise, Mandrake, Debian і Gentoo
  - Bastille Linux перенесена в HP-UX і MacOS X
- Bastille Linux включає:
  - систему регулярних автоматичних оновлень безпеки
  - захист каталогів спільного доступу
- В процесі інсталяції Bastille Linux виконується сценарій, що перевіряє і налаштовує параметри безпеки системи (доступ до каталогів, біти suid і т.і.)