ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

1

Кафедра «Електронних обчислювальних машин»

Кваліфікаційна робота на тему:
«Методи забезпечення безпеки в операційних системах з відкритим кодом»

Виконав: ст. Групи КСМм-23-1 Цуканов Є.Є.
Керівник: доц. Сорокін А. Р.

2025

Актуальність проблеми

2

- З розвитком інформаційних технологій і постійним зростанням кількості даних, які обробляються комп'ютерними системами, питання безпеки інформації набуває все більшої актуальності.
- Операційні системи з відкритим кодом, такі як Linux, мають широке поширення завдяки їх гнучкості, можливостям налаштування та підтримці з боку глобальної спільноти розробників.
- Водночас відкритість коду робить такі системи вразливими до різних видів атак, що створює потребу в розробці ефективних методів забезпечення їхньої безпеки.

Мета та задачі

3

- Дослідження методів забезпечення безпеки в операційних системах з відкритим кодом, оцінка їх ефективності та пропозиція оптимізації одного з методів для підвищення рівня захисту .
- Для досягнення цієї мети в роботі буде здійснено огляд існуючих методів захисту, вивчено основні підходи до управління безпекою, проведено оптимізацію методу контролю доступу та представлено результати тестування реалізованих покращень.

Принципи безпеки в ОС з відкритим кодом

4

Основні принципи безпеки включають:

- Конфіденційність
- Цілісність
- Доступність даних

Всі ці принципи повинні зберігатися в ОС з відкритим кодом

Класифікація методів безпеки

5

Рівень безпеки	Превентивні механізми	Детекційні механізми	Реакційні механізми
Фізична безпека	- Контроль доступу (замки, картки доступу, біометрія) - Обмеження фізичного доступу - Протипожежне обладнання	- Камери відеоспостереження - Датчики руху - Сигналізація	- Активізація сигналізації - Виклик охорони або служб безпеки - Локалізація небезпечних зон
Логічна безпека	- Сильні паролі - Двофакторна аутентифікація - SELinux, AppArmor	- Журнали доступу (auditd, syslog) - Моніторинг підозрілої активності	- Блокування облікового запису при виявленні підозрілої активності - Зміна паролів
Безпека даних	- Шифрування даних (AES, RSA) - Резервне копіювання - Обмеження доступу до файлів	- Контроль хешів файлів - Моніторинг доступу до файлів (inotify)	- Відновлення даних із резервної копії - Оновлення ключів шифрування
Мережева безпека	- Брандмауери (iptables, ufw) - VPN - Використання TLS/SSL	- Виявлення атак (Snort, Suricata) - Логи мережевої активності	- Блокування підозрілих IP-адрес - Ізоляція скомпрометованих сегментів мережі
Програмна безпека	- Аналіз коду перед впровадженням - Регулярне оновлення ПЗ - Використання надійних бібліотек	- Виявлення вразливостей (OpenVAS, Nessus) - Моніторинг аномальної поведінки програм	- Патчинг виявлених вразливостей - Оновлення або видалення скомпрометованого ПЗ

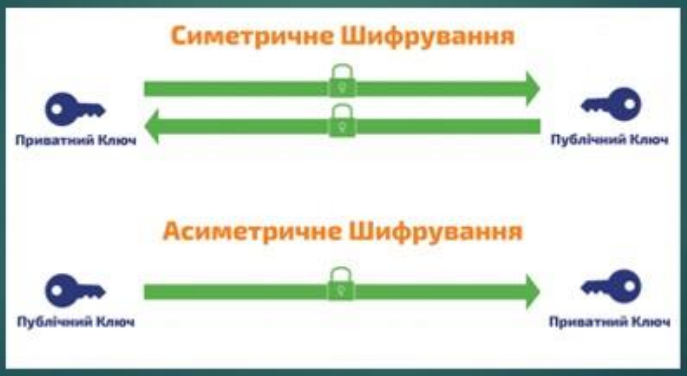
Методи контролю доступу

6

Критерій	DAC (Discretionary Access Control)	MAC (Mandatory Access Control)	RBAC (Role-Based Access Control)
Безпека	- Менш надійний: доступ визначається власником ресурсу - Вразливий до зловживань користувачів	- Висока безпека: доступ базується на політиках, визначених адміністратором - Захищає від несанкціонованого доступу	- Середній рівень безпеки: залежить від правильного налаштування ролей - Мінімізує людський фактор
Ефективність	- Простий у налаштуванні для окремих ресурсів - Складно масштабувати в великих системах	- Менш ефективний у великих системах через суворі політики - Вимагає значних ресурсів для управління	- Висока ефективність у великих організаціях - Спрощує управління доступом через ролі
Простота реалізації	- Легкий для реалізації в невеликих системах - Не потребує складних політик	- Складний у реалізації через сувору структуру - Вимагає глибоких знань безпеки	- Відносно простий: достатньо визначити ролі та права доступу - Легко інтегрується з існуючими системами
Гнучкість	- Висока: власник ресурсу може вільно керувати доступом	- Низька: правила доступу суворо визначені адміністраторами - Неможливо змінювати політики без дозволу	- Середня: зміни в доступі здійснюються через модифікацію ролей
Масштабованість	- Обмежена: складно керувати великою кількістю користувачів і ресурсів	- Низька: управління політиками може стати надмірно складним у великих мережах	- Висока: дозволяє легко масштабувати доступ через управління ролями

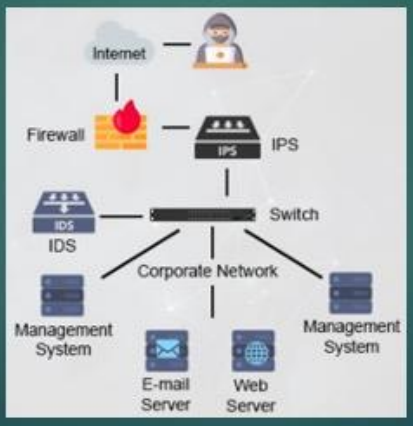
Криптографічні методи захисту

7



Системи виявлення та запобігання вторгненням (IDS та IPS)

8



Забезпечення цілісності даних

9

```
dave@Ubuntu-22-04:~$ sha256sum binary_file1.so
b77a0ca7ff229b8f752548c9e5a66762b61fdf2f089042e5fd6183c9c48c12a0  bina
ry_file1.so
dave@Ubuntu-22-04:~$ sha256sum binary_file2.so
eb210b05f74f92e7d6ed2ce2722209fa7abf056eaabbdcd3431435a11895b64e  bina
ry_file2.so
dave@Ubuntu-22-04:~$
```

Оптимізація методу контролю доступу

10

Показник	Опис	Оцінка (за шкалою 1–5)	Коментар
Рівень безпеки	Наскільки метод забезпечує захист від несанкціонованого доступу	4	Забезпечує базову безпеку, але є ризик обхідних шляхів
Гнучкість	Легкість адаптації під різні сценарії використання	3	Складність у налаштуванні для складних систем
Простота використання	Зручність у впровадженні та адмініструванні	5	Простий для користувачів, не вимагає значних технічних знань
Масштабованість	Здатність працювати ефективно у великих системах	3	Потребує додаткових ресурсів для великих організацій
Продуктивність	Вплив на продуктивність системи	4	Негативний вплив мінімальний
Захист від внутрішніх загроз	Рівень захисту від дій привілейованих користувачів	3	Немає достатнього контролю за діями адміністратора
Сумісність	Наскільки метод інтегрується з іншими системами або технологіями	4	Добре працює з більшістю сучасних ОС

Оптимізація методу контролю доступу

11

Конфігурація для контролю доступу

```
access_control:
  roles:
    - role: admin
      permissions:
        - manage_users
        - manage_system
    - role: user
      permissions:
        - view_content
        - edit_own_profile
    - role: operator
      permissions:
        - monitor_system
        - manage_tasks
  users:
    - username: alice
      role: admin
    - username: bob
      role: user
    - username: charlie
      role: operator
```

Шифрування конфігурації

```
encryption:
  algorithm: AES-256
  key: "your-encryption-key-here"
```

Моніторинг і аудитор доступу

```
monitoring:
  enable: true
  log_level: DEBUG
  audit_logs:
    - event: access_granted
    - event: access_denied
    - event: role_change
```

Біометрична ідентифікація

```
biometric:
  enabled: true
  method: fingerprint
  device: biometric-fingerprint-scanner
```

Конфігурація автентифікації з MFA

```
authentication:
  method: multi-factor
  factors:
    - password
    - sms
    - google_authenticator
```

Перевірка цілісності даних

```
integrity_check:
  enabled: true
  algorithm: SHA-256
  monitored_files:
    - /path/to/critical_data_1
    - /path/to/critical_data_2
```

Порівняння оптимізованої та вихідної систем контролю доступу

12

Критерій	Стара система контролю доступу	Нова система контролю доступу	Покращення
Типи доступу	Базовий доступ лише на основі ролей	Рольовий доступ з умовами часу, місця, типу запиту	Багаторівнева система з гнучкими політиками доступу
Автентифікація	Парольна автентифікація	Многофакторна автентифікація (MFA)	Посилення безпеки за рахунок додаткових факторів
Шифрування	Шифрування на основі базових алгоритмів	Сучасне шифрування AES-256	Збільшення рівня безпеки даних при передачі і зберіганні
Моніторинг	Моніторинг без детального аудиту	Постійний моніторинг з веденням аудиту	Виявлення аномалій у доступі та контроль активності
Біометрія	Відсутня	Використання біометричних даних (відбитки пальців)	Додатковий рівень ідентифікації користувачів
Цілісність даних	Відсутня	Перевірка цілісності даних за допомогою хешування	Захист даних від модифікацій
Гнучкість	Обмежена зміна політик доступу	Гнучкі налаштування доступу на основі контексту	Адаптація системи до змін в політиках доступу

План розвитку та покращення системи оманливих рацій

13

- До можливих покращень можна віднести:
  1. Інтеграція з системами моніторингу
- 2. Перевірка на додаткові загрози
- 3. Підтримка постійного вдосконалення

Висновки

14

- Оптимізація методів контролю доступу в операційних системах з відкритим кодом сприяє значному підвищенню безпеки, покращенню продуктивності та зручності адміністрування. Однак для того, щоб досягти максимального рівня захисту, важливо не лише впроваджувати нові технології, але й постійно підтримувати і вдосконалювати ці методи, забезпечуючи їх сумісність з іншими системами та адаптацію до нових загроз.
- Наукова новизна розробленого методу оптимізації контролю доступу для операційних систем з відкритим кодом полягає у вдосконаленні існуючих механізмів захисту, інтеграції кількох рівнів контролю доступу з врахуванням специфіки роботи відкритих систем, а також оптимізації продуктивності без значного зниження рівня безпеки.
- Запропонований метод інтегрує кілька рівнів контролю доступу, зокрема рольовий (RBAC), атрибутивний (ABAC) та контроль доступу на основі політик (MAC), що дозволяє поєднати їхні переваги.