Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

БАКАЛАВРСЬКА ДИПЛОМНА РОБОТА
на тему:

Розробка системи захисту інформації від несанкціонованого доступу з використанням мандатного керування на основі ОС Linux Mint

Виконав: студент 4 курсу, гр.1КІТС-20б
спеціальності 125– Кібербезпека
Освітня програма – Кібербезпека
інформаційних технологій та систем
(шифр і назва напряму підготовки, спеціальності)

Дячук О. О.
(прізвище, ім`я, по-батькові)

Керівник: д.ф., доц., доцент каф. МБІС
Салієва О. В.
(прізвище та ініціали)
«____»_____ 2024 р.

Рецензент: к.т.н., професор каф. ОТ
Захарченко С. М.
(прізвище та ініціали)
«____»_____ 2024 р.

Допущено до захисту
Голова секції УБ кафедри МБІС
д.т.н., проф. Яремчук Ю.Є.
«_____»_____ 2024 р.

Вінниця ВНТУ – 2024

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

Рівень вищої освіти І-й (бакалаврський)
Галузь знань 12 – Інформаційні технології
Спеціальність 125 – Кібербезпека
Освітньо-професійна програма - Кібербезпека інформаційних технологій та систем

ЗАТВЕРДЖУЮ
Голова секції УБ, кафедра МБІС

_____ **д.т.н., проф. Яремчук Ю.Є.**
«22» березня 2024 р.

З А В Д А Н Н Я
на бакалаврську дипломну роботу студенту
Дячуку Олександру Олександровичу
(прізвище, ім`я, по-батькові)

1. Тема роботи: «Розробка системи захисту інформації від несанкціонованого доступу з використанням мандатного керування на основі ОС Linux Mint»
Керівник роботи д.ф., доц. каф. МБІС Салієва Ольга Володимирівна
(прізвище, ім`я, по-батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від «11» березня 2024 року №80
2. Термін подання студентом роботи за тиждень до захисту.
3. Вихідні дані до роботи:
Метою роботи є здійснення розробки системи захисту інформації на базі операційної системи Linux Mint в якій реалізоване мандатне керування доступом.
4. Зміст текстової частини:
В першому розділі дипломної роботи здійснюється аналіз понять автоматизованої системи, системи захисту інформації, моделей керування доступом та існуючі засоби захисту від несанкціонованого доступу. В другому розділі виконується розробка моделі загроз розроблюваної системи, проводиться робота по розробці профілів мандатного керування доступом, політик та правил основних складових системи захисту інформації. У третьому розділі дипломної роботи на основі обраного дистрибутиву операційної системи та обраних середовищ здійснюється реалізація розроблених профілів мандатного керування доступом, політики аутентифікації та правил аудиту, а також реалізовано і наведено механізм швидкого розгортання реалізованої конфігурації на інших

системах. Здійснено перевірку коректності роботи реалізованих складових, висновках підбиваються усі результати проведеної роботи.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень)
У першому розділі бакалаврської дипломної роботи наявно 3 рисунка, 2 таблиці у другому розділі – 7 рисунків, 1 таблиця; у третьому розділі – 39 рисунків.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Спеціальна частина			
I	Салієва О. В., д.ф., доц. каф. МБІС		
II	Салієва О. В., д.ф., доц. каф. МБІС		
III	Салієва О. В., д.ф., доц. каф. МБІС		

7.Дата видачі завдання 22 березня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Визначення напрямку бакалаврської роботи, формулювання теми	23.03.2024	30.03.2024	Виконано
2	Аналіз предметної області обраної теми	1.04.2024	11.04.2024	Виконано
3	Розробка, реалізація	14.04.2024	04.05.2024	Виконано
4	Написання бакалаврської роботи на основі розробленої теми	08.05.2024	27.05.2024	Виконано
5	Передзахист бакалаврської дипломної роботи	30.05.2024	10.06.2024	Виконано
6	Виправлення, уточнення, корегування бакалаврської дипломної роботи	12.06.2024	16.06.2024	Виконано
7	Захист бакалаврської дипломної роботи	17.06.2024	18.06.2024	Виконано

Студент _____ Дячук О.О.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Салієва О.В.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

В даній бакалаврській дипломній роботі представлено розробку та реалізацію системи захисту інформації на основі операційної системи з відкритим вихідним кодом Linux Mint з використанням мандатного керуванням доступу.

У роботі проаналізовано основні аспекти захисту інформації, включаючи моделі керування доступом та існуючі засоби захисту інформації в автоматизованих системах. Було розроблено модель загроз і на її основі реалізовано профілі мандатного керування доступом, а також конфігурації компонентів системи захисту інформації.

Здійснено обґрунтування вибору складових системи захисту, включаючи вибір дистрибутива ОС, середовища реалізації мандатного керування доступом, програм для реалізації політики розширеної аутентифікації та ведення аудиту системи. Реалізовано компоненти системи захисту на базі модулів та ядра операційної системи Linux Mint, включаючи профілі доступу, правила аудиту та розширену аутентифікацію. Робота виконувалась в середовищі операційної системи Linux Mint та програм які входять до її складу.

Ключові слова: технічний захист інформації, автоматизована система, система захисту інформації, модель керування доступом, системи керування доступом, модель загроз, профілі мандатного керування доступом, аутентифікація, аудит, linux.

Abstract

In this bachelor's thesis, the development and implementation of an information security system based on the open-source operating system Linux Mint using mandatory access control is presented.

The thesis analyzes the main aspects of information security, including access control models and existing means of information protection in automated systems. A threat model was developed, and based on it, mandatory access control profiles and configurations of the information security system components were implemented.

The selection of the components for the security system was justified, including the choice of the OS distribution, the environment for implementing mandatory access control, programs for implementing an enhanced authentication policy, and system audit management. The components of the security system were implemented based on modules and the kernel of the Linux Mint operating system, including access profiles, audit rules, and enhanced authentication. The work was carried out in the Linux Mint operating system environment and the programs included in its composition.

Keywords: technical information protection, automated system, information security system, access control model, access control systems, threat model, mandatory access control profiles, authentication, audit, Linux.

ЗМІСТ

ВСТУП

Актуальність. У сучасному світі процес інформатизації охоплює практично всі сфери людської діяльності, включаючи соціальну, економічну, освітню тощо. Зростання значущості інформації є невід'ємною частиною розвитку технологій.

У кожній інформаційній системі сьогодні обробляється велика кількість даних, розголошення яких може призвести до серйозних збитків. Тому створення ефективних заходів захисту інформації є однією з найбільш актуальних проблем.

Для запобігання витоку інформації під час її обробки в автоматизованих системах використовуються комплекси засобів захисту та операційні системи з вбудованими засобами захисту. Ці комплекси забезпечують реалізацію політики безпеки, яка регулює порядок захисту інформації.

Аналіз поточних тенденцій у сфері операційних систем вказує на поступовий перехід користувачів автоматизованих систем від пропрієтарних програмних рішень до систем з відкритим вихідним кодом. Це спонукало до розробки відповідних комплексів захисту з різноманітними сервісами забезпечення безпеки, які не лише вбудовані в операційну систему, але й постачаються сторонніми розробниками.

Операційна системи є важливим складником для обробки конфіденційної інформації, але за замовчуванням в ОС використовуються стандартні параметри системи та виділених компонентів, що не відповідає критеріям безпеки, встановленим багатьма організаціями та комерційними підприємствами. Використання системи з неправильними налаштуваннями може призвести до критичних порушень безпеки, дозволяючи зловмисникам перехоплювати конфіденційну інформацію або викликати збій системного обладнання.

Створення систем захисту інформації на базі операційних систем з відкритим вихідним кодом залишається актуальним завданням у сфері безпеки інформації. Дані системи стали широко використовуваною платформою як для

серверних, так і для настільних комп'ютерів. З ростом їх популярності збільшується і інтерес до захисту інформації на цій платформі.

Відкритий код також дозволяє розробникам створювати спеціалізовані інструменти та рішення для захисту інформації в переважній більшості випадків без значних витрат. Системи захисту інформації можуть бути розроблені та налаштовані для відповідності конкретним потребам та вимогам у різних галузях.

Метою даної бакалаврської роботи є розробка системи захисту інформації яка буде використовувати сучасні рішення, що надаються системами з відкритим вихідним кодом, яка буде доступною, гнучкою, налаштовуваною та багатофункціональною.

Для досягнення поставленої мети необхідно вирішити наступні ***задачі***:

- визначити поняття автоматизованої системи;
- проаналізувати аспекти інформаційної безпеки в автоматизованих системах;
- визначити поняття систем захисту інформації та її складових;
- проаналізувати існуючі методи та засоби захисту від несанкціонованого доступу;
- вибрати середовище реалізації системи безпеки;
- визначити загрози безпеки та на їх основі розробити модель загроз, яка ляже в основу розроблюваної системи;
- вибрати середовища реалізації складових системи захисту інформації;
- розробити та реалізувати конфігурацію обраної моделі керування доступом;
- розробити та реалізувати конфігурації елементів системи захисту;
- реалізувати механізм розгортання реалізованої конфігурації в нових системах.

Об’єктом дослідження є операційна система з відкритим вихідним кодом та моделі керування доступом які в них використовуються.

Предметом дослідження є сукупність методів, засобів та компонентів операційних систем з відкритим вихідним кодом.

Новизна роботи полягає у розробці системи захисту інформації з використанням власних профілів мандатного керування доступом на базі операційної системи з відкритим вихідним кодом для забезпечення захисту від несанкціонованого доступу в автоматизованих системах на об’єктах інформаційної діяльності.

Практична цінність роботи полягає у використанні на об’єктах інформаційної діяльності для запобіганню несанкціонованого доступу до інформації. Також дану систему можна використовувати для проведення досліджень та тестувань при підготовці кваліфікованих кадрів в сфері захисту інформації.

1 АНАЛІЗ ОСНОВНИХ АСПЕКТІВ ТА ІСНУЮЧИХ ЗАСОБІВ ЗАХИСТУ ІНФОРМАЦІЇ В АВТОМАТИЗОВАНИХ СИСТЕМАХ

В даному розділі буде проведено аналіз питань захисту інформації в автоматизованих системах, систем захисту інформації, а також правової бази яка регулює дані поняття. Буде розглянуто та проаналізовано особливості моделей керування доступом. Також буде проаналізовано існуючі рішення які використовуються для захисту інформації від несанкціонованого доступу.

1.1 Захист інформації в автоматизованих системах

В сучасному світі, де автоматизовані системи здійснюють обробку та зберігання великих обсягів інформації, захист цієї інформації від несанкціонованого доступу стає надзвичайно важливим завданням.

Автоматизована система - це комплексно організована система, яка включає в себе комп'ютерні технології та програмне забезпечення, призначене для автоматизації виконання певних функцій або процесів. Основна мета автоматизованих систем полягає у полегшенні рутинних завдань, підвищенні продуктивності та забезпеченні ефективного управління різними видами діяльності [1].

Вони можуть бути реалізовані у різних сферах, включаючи виробництво, торгівлю, фінанси, логістику, охорону здоров'я, освіту та інші. Автоматизовані системи можуть включати в себе комп'ютери, сенсори, пристрої зчитування даних та інші електронні пристрої, які взаємодіють для забезпечення автоматизованої роботи [2].

Одним із ключових аспектів автоматизованих систем є їхній здатність до автоматичного аналізу та обробки великих обсягів даних, що дозволяє приймати швидкі та обґрунтовані рішення. Крім того, автоматизовані системи часто мають можливість інтеграції з іншими системами та мережами, що дозволяє покращувати їхню функціональність та розширювати їхні можливості [3].

Загалом, автоматизовані системи відіграють важливу роль у сучасному суспільстві, забезпечуючи ефективне управління та підвищуючи продуктивність

у різних галузях діяльності. Їхнє значення зростає з кожним роком, і вони стають необхідним елементом для досягнення конкурентних переваг у сучасному бізнесі та управлінні організаціями, а це в свою чергу викликає потребу в захисті інформації яка циркулює в них.

Згідно з НД ТЗІ 2.5-005-99 «Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу» [4], АС поділяються на три класи:

- АС-1 (клас «1) – одномашинний однокористувачевий комплекс, який обробляє інформацію однієї або кількох категорій конфіденційності. Особливості: в кожен момент часу з комплексом може працювати тільки один користувач;

- АС-2 (клас «2») – локалізований багатомашинний багатокористувачевий комплекс, який обробляє інформацію різних категорій конфіденційності. Особливості: в кожен момент часу з комплексом можуть працювати декілька користувачів з різними повноваженнями, які можуть одночасно виконувати обробку інформації різних категорій конфіденційності;

- АС-3 (клас «3») – розподілений багатомашинний багатокористувачевий комплекс, який обробляє інформацію різних категорій конфіденційності. Особливості: передача інформації через незахищене середовище або наявність вузлів, що реалізують різну політику безпеки.

Для обробки інформації з обмеженим доступом (ІзОД) частіше за все використовуються АС класу «1», оскільки не завжди необхідно виконувати передачу ІзОД через незахищені канали зв’язку. Також вартість створення та обслуговування КСЗІ АС класу «1» набагато нижче.

АС класу «1» найчастіше використовується для обробки інформації, імпорту/експорту на зовнішні носії та друку. Таким чином, до складу прикладного програмного забезпечення (ПЗ) типової АС класу «1» можуть входити:

- офісні пакети (Microsoft Office, LibreOffice);

- ПЗ для обробки графіки (GIMP, AutoCAD);

- засоби розборки (gcc, java);

- ПЗ для перегляду документів (Adobe Acrobat, PDF-viewer);

- бази даних (PostgreSQL, MySQL).

Потенційними загрозами для АС класу «1» є:

- помилки, що виникають при роботі ПЗ, можуть призвести до втрати або пошкодження інформації;

- помилки при введені даних користувачем;

- збої та відмови в роботі апаратного забезпечення;

- встановлення і використання стороннього ПЗ, що не дозволено політикою безпеки;

- встановлення шкідливого ПЗ;

- викрадення носіїв інформації;

- несанкціоноване копіювання інформації на зовнішні носії;

- несанкціоноване копіювання інформації у каталоги, що мають загальний доступ;

- доступ до інформації, яка залишилась в оперативній пам’яті чи запам’ятовуючих пристроях після її видалення;

- розголошення даних автентифікації користувачів;

- несанкціонований доступ до інформації та її модифікація.

Захист інформації в автоматизованих системах включає в себе широкий спектр аспектів, від технічних до організаційних, і потребує врахування відповідних правових норм та стандартів.

Одним з ключових аспектів захисту інформації в автоматизованих системах є контроль доступу. Це включає в себе встановлення механізмів аутентифікації та авторизації, обмеження прав доступу користувачів та впровадження систем моніторингу доступу. Забезпечення конфіденційності та цілісності даних здійснюється за допомогою шифрування даних, яке запобігає несанкціонованому доступу до чутливої інформації під час її передачі та зберігання [5].

Інший важливий аспект - захист від мережевих загроз. Він включає в себе використання файерволів, виявлення вторгнень та інші механізми, які допомагають уникнути атак на мережу і зберегти цілісність системи. Правова база визначає стандарти та вимоги до захисту інформації в мережах, зокрема через закони, які регулюють цю сферу [6].

Нормативно-правова база України в цьому плані є відносно добре розвиненою. Закони "Про захист персональних даних" [7] та "Про захист інформації в інформаційно-телекомунікаційних системах"[8] встановлюють основні правила та вимоги щодо захисту інформації. Крім того, регулювання кіберзлочинності визначається Кримінальним кодексом України [9], де закріплені статті, що стосуються кіберзлочинів. Питання захисту інформації в автоматизованих системах регламентується зокрема такими документами:

- «Загальні вимоги до захисту інформації в інформаційно-телекомунікаційних системах»: цей нормативний документ встановлює загальні вимоги до захисту інформації в інформаційно-телекомунікаційних системах (ІТС). Він охоплює такі аспекти, як організація захисту інформації, технічні та криптографічні засоби захисту, а також вимоги до забезпечення безпеки персональних даних [10].
- «Правила організації захисту інформації в інформаційних системах»: цей документ визначає правила та процедури організації захисту інформації в інформаційних системах. Він встановлює вимоги до захисту даних на різних етапах їх обробки та передачі в інформаційних системах [11].
- «Вимоги до захисту інформації, що обробляється в інформаційних системах з обмеженим доступом»: цей нормативний документ встановлює специфічні вимоги до захисту інформації в інформаційних системах з обмеженим доступом. Він охоплює такі аспекти, як захист інформації під час її обробки та зберігання в обмежених доступом системах [12].

Ці нормативні документи мають значення для забезпечення безпеки інформації в Україні, оскільки вони встановлюють стандарти та вимоги до

захисту інформації в інформаційно-телекомунікаційних системах та інформаційних системах з обмеженим доступом.

Державна служба спеціального зв'язку та захисту інформації України виступає як головний орган, що контролює і забезпечує дотримання вимог до захисту інформації в країні. Нормативні акти та положення, які вона приймає, визначають стандарти та вимоги до захисту інформації в автоматизованих системах.

Загальна реалізація заходів захисту інформації в автоматизованих системах вимагає комплексного підходу, що об'єднує технічні, організаційні та правові аспекти. Врахування відповідної правової бази є ключовим елементом успішної реалізації цих заходів.

1.2 Поняття систем захисту інформації

Система захисту інформації (СЗІ) - це комплексний підхід до захисту інформації в автоматизованих системах (АС), що включає в себе не лише технічні засоби, але й організаційні заходи, процедури та політики безпеки. Основна мета СЗІ - забезпечити конфіденційність, цілісність та доступність інформації в системі, а також захистити її від несанкціонованого доступу, зміни та втрати [13].

Роль системи захисту інформації полягає в запобіганні можливим загрозам безпеці, що можуть виникнути як внаслідок дій людини (наприклад, зловмисного вторгнення чи несанкціонованого доступу), так і внаслідок технічних проблем (наприклад, вірусів, хакерських атак тощо). СЗІ також відіграє важливу роль у впровадженні вимог законодавства щодо захисту персональних даних та інших конфіденційних інформаційних ресурсів [14].

Основні елементи системи захисту інформації включають:

1. Політики безпеки. Установлення правил, процедур та вимог щодо захисту інформації.

2. Технічні засоби захисту. Використання програмного та апаратного забезпечення для захисту інформації, таких як фаєрволи, антивірусне програмне забезпечення, системи виявлення вторгнень, системи шифрування тощо.

3. Організаційні заходи. Навчання персоналу, розподіл доступу до інформації, контроль і аудит доступу, регулярні аналізи загроз безпеці.

4. Фізичні заходи безпеки. Захист приміщень, серверних кімнат та інших фізичних об'єктів, які містять інформацію.

В сучасних автоматизованих системах, де обробка та передача інформації здійснюються великою мірою через мережі, значення системи захисту інформації набуває особливого значення, оскільки зростає ймовірність атак на систему.

Важливою складовою систем захисту інформації є комплекси засобів захисту (КЗЗ). Вони становлять основний або навіть центральний елемент у забезпеченні безпеки інформації в сучасних автоматизованих системах. Вони представляють собою комплексні технічні, організаційні та процедурні заходи, спрямовані на захист конфіденційності, цілісності та доступності даних в системі.

Одним з ключових принципів роботи КЗЗ є інтеграція різноманітних заходів та технологій в єдину систему, яка працює в комплексі для забезпечення максимального рівня захисту. Наприклад, антивірусне програмне забезпечення може виявляти та блокувати загрози з боку шкідливих програм, водночас фаєрвол може контролювати мережевий трафік та блокувати небажаний доступ до системи [15].

Комплекси засобів захисту (КЗЗ) можуть включати в себе:

1) Антивірусне програмне забезпечення: вид захисту призначений для виявлення, блокування та видалення вірусів, троянів, шпигунського програмного забезпечення та інших шкідливих програм, що можуть завдати шкоди системі.

2) Фаєрволи: контролюють мережевий трафік між комп'ютерами та мережами, встановлюючи правила фільтрації, які визначають, який трафік дозволяється, а який блокується [16].

3) Системи виявлення та запобігання вторгненням (IDS/IPS): ці системи аналізують мережевий трафік та виявляють аномальні або підозрілі дії, які можуть свідчити про потенційний вторгнення. IDS виявляють вторгнення, тоді як IPS може автоматично блокувати або відхиляти такі атаки [17].

4) Шифрування даних: шифрування даних використовується для захисту конфіденційної інформації шляхом перетворення її у незрозумілий формат для осіб без відповідного доступу [18].

5) Контроль доступу: вид захисту, що визначає, хто має доступ до системи або її ресурсів, та обмежує цей доступ відповідно до встановлених правил та політик безпеки [19].

6) Аудит безпеки: системи аудиту безпеки збирають та аналізують дані про дії користувачів та події в системі з метою виявлення можливих загроз та атак.

7) Безпека додатків: цей вид захисту включає в себе перевірку безпеки програмного забезпечення та застосунків на наявність вразливостей, а також виконання процедур тестування на проникнення для виявлення слабких місць.

Наведені вище елементи лише деякі з основних, які використовуються для захисту інформації в автоматизованих системах. Зазвичай комплекси засобів захисту поєднують у собі декілька з цих технологій та методів для максимального рівня безпеки.

Нормативно-правове забезпечення України в сфері технічного захисту інформації є важливими для регулювання поняття комплексів засобів захисту (КЗЗ). Нормативні документи, які регулюють поняття КЗЗ:

- НД ТЗІ 2.5-004-99 Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу, наказ ДСТСЗІ СБУ від 28.04.99 № 22 (Зміна № 1 наказ від 28.12.2012 № 806);

- НД ТЗІ 2.5-005-99 Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблювальної інформації від несанкціонованого доступу, наказ ДСТСЗІ СБУ від 28.04.99 № 22 (Зміна № 1 наказ від 15.10.2008 № 172).

Організаційні заходи також грають важливу роль у реалізації систем захисту інформації, включаючи розробку політик безпеки, навчання персоналу, встановлення процедур безпеки та створення системи моніторингу та реагування на інциденти [20].

1.3 Огляд існуючих моделей керування доступом

Системи керування доступом (СКД) відіграють ключову роль у побудові систем захисту інформації, забезпечуючи контроль і обмеження доступу користувачів до ресурсів системи. Системи керування доступом є невід'ємною складовою систем захисту інформації, допомагаючи забезпечити безпеку, конфіденційність та цілісність даних та ресурсів системи.

СКД встановлюють правила та політики, що регулюють доступ користувачів до файлів, каталогів, програм, мережевих ресурсів та інших об'єктів системи. Це дозволяє обмежити доступ лише для авторизованих користувачів або груп користувачів та запобігти несанкціонованому доступу [21].

В операційних системах на базі Linux можуть використовуватись наступні моделі керування доступом:

1. DAC (Discretionary Access Control). Це модель контролю доступу, в якій власник ресурсу (наприклад, файлу або каталогу) має повний контроль над правами доступу до цього ресурсу і може надавати доступ іншим користувачам. В основі DAC лежить ідея дозволу або відмови в доступі, який встановлює власник ресурсу. В основному для налаштування DAC використовуються такі системи як sudo і PAM. Вони дозволяють адміністраторам системи контролювати доступ користувачів до підвищених привілеїв або ресурсів на основі прав доступу, встановлених самими адміністраторами або користувачами [22].

2. MAC (Mandatory Access Control). Це більш строга модель контролю доступу, в якій права доступу встановлюються системним адміністратором або політиком безпеки, а не власником ресурсу. У системах MAC кожен об'єкт та суб'єкт мають мітки безпеки, які визначають рівень дозволу або відмови в

доступі. Приклади систем, які використовують MAC, включають SELinux та AppArmor [23].

3. RBAC (Role-Based Access Control). У RBAC доступ до ресурсів системи призначається на основі ролей, які мають користувачі. Користувачі призначаються ролям, а ролі мають відповідні дозволи. Цей підхід полегшує управління доступом до ресурсів у великих системах і зменшує ризик неправильної адміністрації [24].

4. ABAC (Attribute-Based Access Control). У ABAC доступ до ресурсів системи призначається на основі атрибутів користувачів, об'єктів та контексту. Наприклад, доступ може бути призначений на основі характеристик користувача (роль, відділ), атрибутів об'єкту (класифікація даних) та інших факторів [25].

5. Rule-Based Access Control. У цьому підході доступ до ресурсів системи призначається на основі заздалегідь визначених правил або умов. Користувачі отримують доступ до ресурсу, якщо вони відповідають заданим умовам, які можуть бути залежні від різних факторів, таких як час, місцезнаходження, аутентифікація тощо [26].

6. ABE (Attribute-Based Encryption). Це модель контролю доступу, яка використовується в криптографії. Вона базується на шифруванні даних за допомогою ключів, які залежать від атрибутів користувача та/або контексту, і відповідному розшифруванні цих даних лише тим користувачам, які мають відповідні атрибути [27].

Кожна з цих моделей має свої особливості, переваги та недоліки. Порівняємо зазначені моделі між собою, результат порівняння відображено в табл.1.1.

Таблиця 1.1 – Порівняльна характеристика моделей керування доступом

Характеристика	DAC (Discretionary Access Control)	MAC (Mandatory Access Control)	RBAC (Role-Based Access Control)	ABAC (Attribute-Based Access Control)	Rule-Based Access Control	ABE (Attribute-Based Encryption)
Основний принцип	Контроль доступу на розсуд власника об'єкта	Контроль доступу на основі політик, встановлених системою	Контроль доступу на основі ролей користувачів	Контроль доступу на основі атрибутів користувача та об'єкта	Контроль доступу на основі правил	Шифрування даних на основі атрибутів
Гнучкість	Висока	Низька	Середня	Висока	Висока	Висока
Безпека	Низька	Висока	Висока	Висока	Висока	Висока
Простота налаштування	Висока	Низька	Середня	Низька	Середня	Низька
Контроль адміністратором	Низький	Високий	Високий	Високий	Високий	Середній
Підходить для великих організацій	Середньо	Так	Так	Так	Так	Так
Підходить для малих організацій	Так	Ні	Можливо	Можливо	Можливо	Ні
Приклади використання	Типові файлові системи Linux	SELinux, AppArmor	Багато корпоративних систем	Складні інформаційні системи	Мережеві брандмауери	Захист даних у хмарних системах
Вразливість до помилок користувача	Висока	Низька	Середня	Низька	Середня	Низька
Складність підтримки	Низька	Висока	Середня	Висока	Середня	Висока

Традиційно, операційні системи сімейства Linux використовують модель безпеки засновану на вибірковому керуванні доступом (Discretionary Access Control, DAC). Для забезпечення більш високого рівню захисту, в деяких Linux системах опціонально або за замовченням використовується модель мандатного керування доступом (Mandatory Access Control, MAC) [28].

Модель MAC має ряд принципових переваг над іншими моделями контролю доступу. Дана модель може надати захист від просунутих загроз, таких як внутрішні атаки або експлуатація вразливостей, оскільки вона забезпечує додаткові шари контролю та обмеження доступу. Через те, що доступ до ресурсів у MAC визначається на рівні системи або політики безпеки, зменшується ризик недопущення або уникнення недбалого призначення прав доступу, що може виникнути в DAC, де власник ресурсу встановлює права доступу.

З точки зору поширеності використання, MAC є менш поширеною моделлю в порівнянні з DAC. Однак вона широко використовується в деяких високоризикових секторах, таких як військові системи, урядові установи,

фінансові установи та інші області, де важливий високий рівень безпеки та строгий контроль доступу до конфіденційної інформації. Це через те, що МАС дозволяє забезпечити більш високий рівень безпеки та контролю над доступом до критичних ресурсів, що є критичним для захисту важливих даних та інформації в цих галузях.

Попри недоліки, пов'язані з централізованим і жорстко регульованим підходом до контролю доступу, модель мандатного керування доступом є більш доцільною при побудові систем захисту інформації автоматизованих систем, оскільки вона пріоритетно орієнтована на забезпечення безпеки і контролю.

1.4 Огляд існуючих систем захисту інформації

Відповідно до «Переліку засобів технічного захисту інформації, дозволених для забезпечення технічного захисту державних інформаційних ресурсів та інформації, вимога щодо захисту якої встановлена законом» («Перелік...») [29] наразі існують наступні рішення представлені як ті, які базуються на операційних системах, так і системи, що є свого роду надбудовами до них:

– комплекс засобів захисту операційної системи Microsoft Windows 10 Professional – ТОВ «Майкрософт Україна» [30];

– комплекс засобів захисту програмного забезпечення «Операційна система Ubuntu*Pack 20.04» – ТОВ «УАЛІНУКС»[31];

– комплекс засобів захисту програмного забезпечення «Операційна система Січ» – ТОВ «Трайбекс» [32];

– засіб технічного захисту інформації від несанкціонованого доступу «Комплекс «Гриф» версії 4» – ТОВ «Інститут комп'ютерних технологій» [33];

– система захисту інформації ЛОЗА™-1, версія 4 – ТОВ НДІ «АВТОПРОМ» [34].

Так як, найближчими до тематики даної бакалаврської дипломної роботи є саме операційні системи та їх засоби, розглянемо детально їх особливості та порівняємо їх між собою.

Операційна система Microsoft Windows 10 Professional (рис.1.1) – є клієнтською багатозадачною, багатокористувацькою, багатопроцесорною, мережевою ОС. ОС Microsoft Windows 10 Professional призначена для використання у великих підприємствах та організаціях та має відповідні засоби розгортання та підтримки. Комплекс засобів захисту операційної системи Microsoft Windows 10 Professional – це сервіси безпеки операційної системи Microsoft Windows 10 Professional призначені для забезпечення конфіденційності, цілісності, доступності та спостережності об’єктів захисту [35].

Дана операційна система включає широкий набір засобів захисту та функцій безпеки, які спрямовані на захист системи від різних загроз, таких як:

– Windows Defender. Вбудований антивірус та антишпигун для операційних систем Windows 10. Даний продукт забезпечує захист від різних видів загроз, включаючи віруси, шпигунське програмне забезпечення, рекламне ПЗ тощо.

– Firewall Windows. Вбудований файрвол, який допомагає контролювати трафік мережі та захищає ваш комп'ютер від небажаного доступу з мережі.

– Bitlocker. Рішення для шифрування даних на диску, що забезпечує захист від несанкціонованого доступу до інформації в разі втрати або крадіжки пристрою.

– Аудит безпеки Windows. Інструмент, який дозволяє вести журнал подій безпеки та відстежувати дії користувачів та системних процесів.

– Оновлення безпеки Windows. Важлива складова безпеки, яка забезпечує вчасне встановлення патчів та оновлень для вирішення виявлених вразливостей.

– Система контролю доступу до об'єктів (DAC).

– Система контролю доступу на основі ролей (RBAC).

Рисунок 1.1 – Користувацький інтерфейс ОС Windows 10 Professional

Даний набір засобів покриває різні аспекти захисту і допомагає забезпечити високий рівень захисту для користувачів та організацій, роблячи систему стійкою до сучасних кіберзагроз.

Недоліки використання операційної системи Windows 10 Pro здебільшого полягають у тому, що це серйозний комерційний продукт який здебільшого орієнтований на бізнес-користувачів. Висока вартість використання обумовлена ліцензійними зборами за використання ОС і додаткових інструментів безпеки. Серед недоліків слід виділити те, що використання ОС Windows 10 Pro накладає високі вимоги до апаратних ресурсів для повноцінного функціонування засобів безпеки. Також, одним з основних недоліків є можливі проблеми з оновленнями, які можуть впливати як на продуктивність, так і відкривати небажані вразливості.

Комплекс засобів захисту програмного забезпечення «Операційна система Ubuntu*Pack є спеціалізованою збіркою операційної системи на базі Ubuntu, яка створена для підвищення безпеки та конфіденційності користувачів. Вона розроблена українською компанією UALinux і пропонує ряд додаткових функцій і налаштувань для посилення захисту. На рис.1.2 зображено вигляд інтерфейсу захищеної операційної системи Ubuntu*Pack.

Рисунок 1.2 – Інтерфейс захищеної операційної системи Ubuntu*Pack

Посилена безпека реалізується використанням інструментів для мандатного керування доступом (MAC), що забезпечують додатковий рівень захисту, а також вбудованим антивірусним захистом ClamAV.

Конфіденційність операційної системи забезпечується підтримкою шифрування диска з використанням таких технологій, як LUKS (Linux Unified Key Setup), яка забезпечує захист даних у разі втрати або крадіжки пристрою, а також інструментами Tor і VPN у мережевих активностях.

Також операційна система Ubuntu*Pack надає такі додаткові інструменти безпеки як Firewall та система контролю доступу для запобігання виконанню несанкціонованих програм або сценаріїв.

Ubuntu*Pack версії 20.04 є LTS (Long Term Support) версією, що означає довготривалу підтримку та регулярні оновлення безпеки в автоматичному режимі.

Дана ОС хоч і базується на операційній системі з відкритим вихідним кодом, проте є комерційним продуктом, що крім вартості накладає обмеження на її перебудову під власні специфічні задачі. Налаштування безпекових параметрів присутнє, але обмежене. Також, серед недоліків слід виділити обмежену підтримку комерційного ПЗ та можливі проблеми з драйверами для деяких специфічних апаратних засобів.

Комплекс засобів захисту програмного забезпечення «Операційна система Січ» є захищеною операційною системою, що створена на базі ОС SUSE Linux

Enterprise 15 (рис.1.3). Вона була створена для використання в державних установах та організаціях, де вимоги до захисту інформації є особливо високими. Дана ОС забезпечує захист за допомогою двох основних властивостей - модульної архітектури та набору інструментів забезпечення безпеки. Різні елементи операційної системи слабо пов'язані між собою, що значно ускладнює недолік безпеки однієї частини системи, який впливає на іншу частину системи. В ОС Січ встановлено за замовчуванням набір інструментів AppArmor, що забезпечує безпеку та контроль доступу до системи зі сторони додатків та програм. Також, в ОС Січ за замовчуванням не надаються користувачу адміністративні права.

Рисунок 1.3 – Інтерфейс користувача ОС «Січ»

Набір інструментів, що забезпечують посилену безпеку, конфіденційність подібний до Ubuntu*Pack. Однією з особливостей і відмінностей даної ОС є використання YaST для шифрування розділів, додаткових носіїв або частин файлової системи під час встановлення або у вже встановленій системі, що підвищує рівень безпеки даних.

В ОС «Січ» реалізовано просунуту системи ведення аудиту дій користувачів і моніторингу активності в системі для виявлення і запобігання загрозам.

Так як і в Ubuntu*Pack присутня можливість постійних оновлень безпеки і виправлень вразливостей для підтримки актуального стану системи.

Операційна система «Січ» є комерційним продуктом, що варто враховувати при побудові системи захисту інформації.

Серед недоліків ОС «Січ» слід виділити:

- необхідність у високій кваліфікації для налаштування політик безпеки;
- менш розвинута екосистема, зумовлена низькою популярністю дистрибутива;
- обмежена документація.

Визначивши основні особливості та характеристики розглянутих вище засобів, порівняємо їх між собою. Результат порівняння відображено в табл.1.2.

Таблиця 1.2 – Порівняльна характеристика існуючих засобів

Критерій	**Ubuntu*Pack 20.04**	**ОС "Січ"**	**Microsoft Windows 10 Professional**
Основний фокус	Безпека і конфіденційність	Безпека для державних установ	Комплексний захист для бізнесу
Модель керування доступом	MAC (SELinux або AppArmor)	MAC (SELinux)	DAC, RBAC, частково MAC (Device Guard, Credential Guard)
Шифрування	LUKS	LUKS	BitLocker
Антивірусний захист	Вбудовані рішення, ClamAV	Вбудовані рішення	Windows Defender Antivirus
Мережевий захист	Підтримка VPN і Tor	Підтримка VPN і Tor	Windows Firewall, VPN підтримка
Контроль доступу	Багаторівневий контроль	Багаторівневий контроль	Багаторівневий контроль, Credential Guard
Підтримка комерційного ПЗ	Обмежена	Обмежена	Широка підтримка
Сумісність обладнання	Можливі проблеми з деяким обладнанням	Можливі проблеми з деяким обладнанням	Широка підтримка обладнання
Ліцензійна вартість	Відкритий код, комерційний продукт	Відкритий код, комерційний продукт	Висока вартість ліцензій
Вимоги до ресурсів	Середні	Середні	Високі
Зручність використання	Складне налаштування	Складне налаштування	Інтуїтивний інтерфейс
Документація та підтримка	Обмежена підтримка	Обмежена підтримка	Розширена документація, технічна підтримка
Регулярність оновлень	Регулярні оновлення	Регулярні оновлення від розробників	Централізовані оновлення від Microsoft

1.5 Висновки та постановка задачі

В даному розділі бакалаврської дипломної роботи було розглянуто питання захисту інформації від несанкціонованого доступу в автоматизованих системах. Було розглянуто поняття автоматизованих систем, систем захисту інформації, чинну нормативну-правову базу яка регулює аспекти їх впровадження та використання.

При розгляді питання поняття систем захисту інформації, в п.1.2., було визначено поняття системи захисту інформації. Система захисту інформації це широке поняття, яке охоплює всі структури, процеси та заходи, спрямовані на захист інформації в організації чи системі. Це може включати політики безпеки, процедури, технічні засоби захисту, контроль доступу, обробку даних та навчання персоналу. Основу систем захисту інформації складає технічна складова, яка представлена конкретними технічні, програмні та організаційні засоби, що використовуються для реалізації системи захисту інформації. Це може включати файєрволи, антивірусне програмне забезпечення, системи контролю доступом, системи виявлення вторгнень, шифрування даних, аутентифікаційні механізми та інші технічні засоби.

Серед питань які розглядалися в даному розділі, було питання систем контролю доступу, що є основою технічної складової будь-якої системи захисту інформації. Виділено основні моделі керування доступом та системи які їх використовують, зокрема і ті які реалізовані в операційних системах сімейства Linux. В ході аналізу цього питання було встановлено, що системи керування доступом на базі моделі MAC (Mandatory Access Control) є більш доцільними при розробці систем захисту інформації.

Аналіз існуючих систем та засобів захисту інформації від несанкціонованого доступу які входять до переліку дозволених Адміністрацією Держспецзв’язку показує, що на українському ринку представлено обмежена кількість таких рішень. Це в свою чергу обмежує вибір для кінцевих користувачів, оскільки задачі та вимоги до захисту можуть бути різними.

Серед розглянутих рішень в п.1.5 є комплекси засобів захисту які являються по своїй суті надбудовами до операційної системи. Як правило це операційна система Windows. Важливою вимогою до систем захисту інформації, регламентованою нормативно-правовою базою в цій галузі є використання виключно ліцензійного програмного забезпечення або вільно поширюваного ПЗ (freeware). Операційна система Windows є комерційним продуктом, вартість якого залежить від використовуваної версії. Таким чином, витрати на захист будуть включати витрати на операційну систему та на саму систему захисту.

Альтернативою таких надбудов є використання засобів і програмного забезпечення які включені в саму операційну систему, але при цьому необхідно виконувати конфігурування ОС відповідно до вимог захисту і визначеного політики безпеки, а це потребує навичок системного адміністрування та поглиблених знань в сфері технічного захисту інформації. Крім того, сучасні версії операційної системи Windows мають високі вимоги до апаратної частини. Популярність операційної системи Windows зумовлює високий інтерес збоку зловмисників, що на ряду з можливими проблемами централізованого оновлення безпеки створює високий рівень загрози конфіденційності та цілісності даних.

Іншим варіантом є використання захищених операційних систем. Серед представлених на ринку це операційні системи на базі Linux які включають у себе необхідні засоби і конфігурації для захисту від несанкціонованого доступу. Не дивлячись на те, що переважна більшість дистрибутивів сімейства Linux є безкоштовними, розглянуті рішення є комерційними продуктами.

На ряду з вартістю, недоліком розглянутих рішень є їх обмеженість та закритість, що впливає на можливість вдосконалення чи переконфігурування у разі зміни політик безпеки. Зміна поточних конфігурацій захищених операційних систем потребує необхідність у високій кваліфікації для налаштування політик безпеки.

Хоч і розглянуті існуючі рішення дозволяють забезпечувати належний рівень захищеності інформації попри витрати на їх використання, виникає потреба у створенні концептуально нових рішень, які давали можливість не

тільки зменшити витрати на забезпечення захисту інформації від несанкціонованого доступу, а й також які відповідали вимогам забезпечення захищенності інформації від несанкціонованого доступу і чинної нормативно-правової бази.

Тому ***метою*** бакалаврської дипломної роботи є розробка системи захисту інформації яка була б доступнішою в реалізації та використанні в цілому, була гнучкою, використовувала доступні сучасні підходи та засоби, забезпечувала високий рівень захищеності.

Для досягнення поставленої мети необхідно вирішити наступні ***задачі***:

- розробити модель загроз, відносно якої розроблюватимуться і реалізовуватимуться складові системи захисту;
- вибрати середовища реалізації складових системи захисту інформації;
- розробити профілі безпеки мандатного керування доступом;
- розробити основу конфігурації елементів системи захисту у вигляді розширеної політики аутентифікації та правил аудиту системи;
- реалізувати розроблені профілі безпеки;
- реалізувати розроблену політику аутентифікації;
- реалізувати правила аудиту системи
- реалізувати механізм розгортання реалізованої конфігурації в нових системах.

2 РОЗРОБКА ПРОФІЛІВ МАНДАТНОГО КЕРУВАННЯ ДОСТУПОМ ТА ОСНОВИ КОНФІГУРАЦІЇ КОМПОНЕНТІВ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ

В даному розділі буде здійснено розробку моделі загроз для визначення потенційних загроз розроблюваній системі. Здійснено та аргументовано вибір середовищ реалізації мандатного керування доступом, політик аутентифікації та правил аудиту системи.

Також буде розроблено профілі мандатного керування доступом, політики аутентифікації та правила аудиту, які будуть основою розроблюваної системи захисту інформації.

2.1 Розробка моделі загроз

Модель загроз є критично важливою при розробці системи захисту інформації. За допомогою неї можна проаналізувати та оцінити потенційні загрози, які можуть впливати на розроблювану систему, а це, в свою чергу, дає можливість створення ефективних контрзаходів [36].

Тематикою даної бакалаврської дипломної роботи передбачено розробку системи захисту інформації від несанкціонованого доступу з використанням мандатного керування на основі операційної системи з відкритим вихідним кодом. Модель загроз для такої системи повинна враховувати специфіку операційної системи, особливості мандатного керування доступом, а також можливі загрози і вразливості.

Розробка моделі загроз передбачає наступні етапи:

1) визначення активів;
2) визначення меж системи;
3) ідентифікація загроз;
4) аналіз загроз;
5) розробка контрзаходів;
6) реалізація контрзаходів та моніторинг.

Впроваджені контрзаходи повинні відповідати вимогам чинній нормативно правовій базі, що регулює питання в даний сфері.

Враховуючи вище зазначене, модель загроз передбачатиме наступне (рис.2.1).

1) Активи:

- конфіденційні дані - файли, бази даних, документи, що містять критично важливу інформацію;
- системні ресурси - ядро ОС, системні бібліотеки, конфігураційні файли;
- користувацькі дані - особисті файли користувачів, налаштування;
- мережеві ресурси - з’єднання та комунікації з іншими системами та мережами.

2) Межі системи:

- апаратні межі - робоча станція, периферійні пристрої;
- програмні межі - встановлені програми та служби, які працюють в обраній ОС;
- мережеві межі - маршрутизатори, брандмауери, точки доступу.

3) Загрози:

- зовнішні загрози - хакери, шкідливе програмне забезпечення, мережеві атаки;
- внутрішні загрози - недобросовісні або недбалі співробітники, які мають доступ до системи;
- загрози конфіденційності - несанкціонований доступ до конфіденційних даних;
- загрози цілісності - зміни або видалення важливих даних.
- загрози доступності - порушення доступності критично важливих сервісів та даних;
- фізичні загрози - крадіжка обладнання, фізичне пошкодження апаратних компонентів.

4) Контрзаходи:

- технічні заходи:
  - мандатне керування доступом (MAC);
  - використання шифрування для захисту конфіденційних даних;
  - розширена політика аутентифікації;
  - налаштування системи журналювання та моніторингу для відстеження дій користувачів і аномальної активності.
- організаційні заходи;
- фізичні заходи.

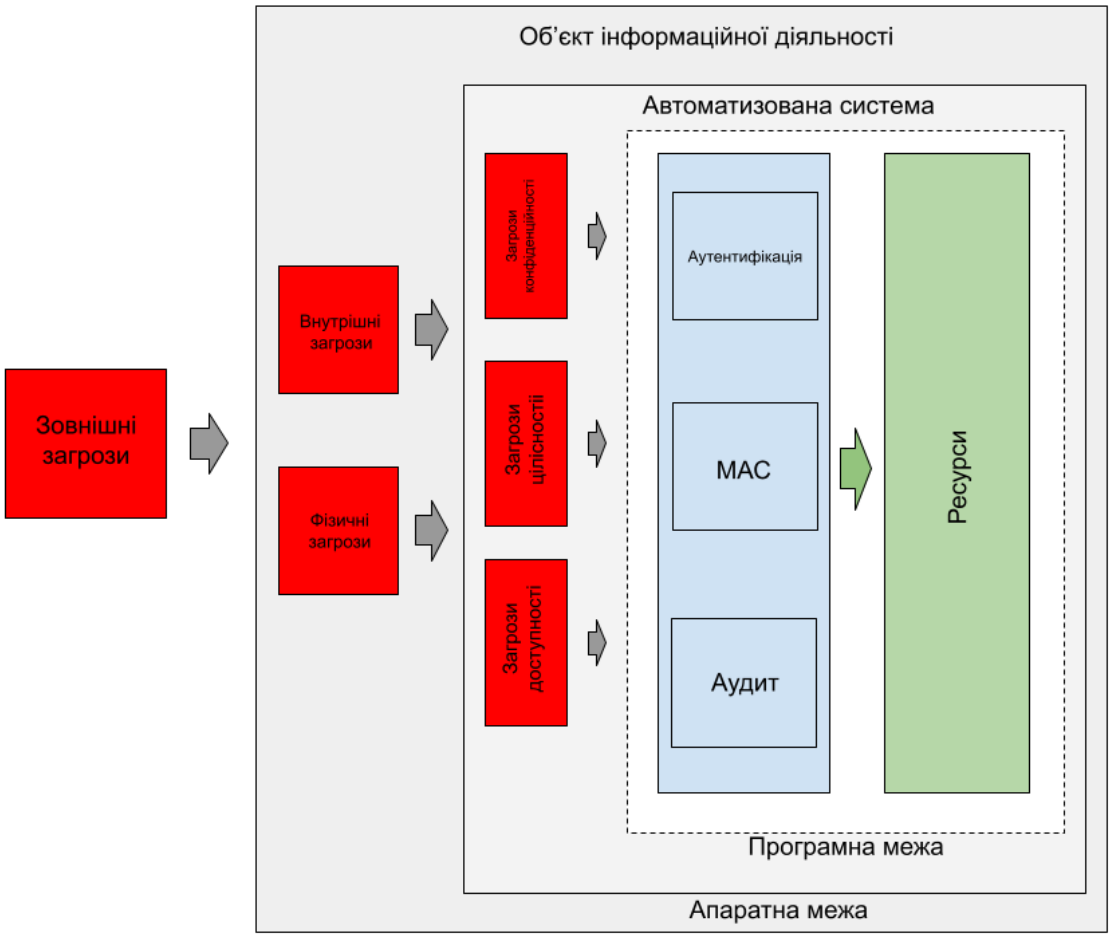


Рисунок 2.1 – Модель загроз розроблюваної системи безпеки від несанкціонованого доступу

Як бачимо, розроблена модель загроз враховує різноманітні загрози, дає можливість оцінки їхні ризиків та передбачає відповідні контрзаходи для забезпечення надійної безпеки системи на базі ОС відкритим вихідним кодом з

використанням мандатного керування доступом від несанкціонованого доступу до інформації.

2.2 Обґрунтування вибору складових системи захисту інформації

Правильний вибір складових при розробці системи захисту інформації має вирішальне значення для забезпечення ефективного захисту від різних загроз. Від цього залежить стійкість системи до атак, її надійність, конфіденційність, цілісність і доступність даних.

Відповідно до тематики роботи та визначеної моделі загроз для розробки і реалізації технічної складової системи захисту інформації потрібно обрати:

- дистрибутив ОС з відкритим вихідним кодом;
- середовище реалізації моделі мандатного керування доступом;
- модуль/програму реалізації політики розширеної аутентифікації;
- налаштовуваний модуль/програму ведення аудиту системи.

2.2.1 Вибір дистрибутиву ОС

Вибір дистрибутива операційної системи на базі Linux при побудові комплексу засобів захисту та системи захисту інформації має базуватися на наступних ключових критеріях:

1) безпека;
2) стабільність та надійність;
3) відповідність нормативним вимогам;
4) підтримка;
5) інструменти та сумісність;
6) простота адміністрування.

Для побудови системи захисту інформації пропонується дистрибутив Linux Mint, який відповідає зазначеним критеріям.

Linux Mint є одним з найпопулярніших дистрибутивів Linux, особливо серед новачків та користувачів, які переходять з інших операційних систем, таких як

Windows або macOS. При цьому, він є повністю безкоштовним. Як видно з рис.2.2, інтерфейс має багато спільних рис з добре відомою ОС Windows.

Рисунок 2.2 - Інтерфейс операційної системи Linux Mint

Linux Mint базується на Ubuntu та Debian, що забезпечує йому стабільність і доступ до великої кількості програмного забезпечення. Основна мета цього дистрибутива - надати зручне та функціональне середовище для щоденного використання з мінімальними налаштуваннями для звичайного користувача та надавати широкий спектр можливостей для адміністрування [37].

Переваги:

– інтуїтивний інтерфейс;

– більшість налаштувань можна зробити через графічні інтерфейси;

– базування на Ubuntu LTS забезпечує стабільність та регулярні оновлення безпеки;

– мінімальні вимоги до ресурсів;

– доступність документації та ресурсів;

– з точки зору звичайного користувача, система готова до використання одразу після встановлення, без потреби додаткового налаштування або встановлення програмного забезпечення.

Окрім вище зазначеного, даний дистрибутив має інструменти для резервного копіювання та відновлення даних, такі як Timeshift, дозволяють легко зберігати та відновлювати стан системи.

Linux Mint володіє підтримкою популярних онлайн-сервісів для роботи з електронною поштою, календарями та іншими сервісами.

Також даний дистрибутив має широкі можливості налаштування зовнішнього вигляду та функціоналу системи відповідно до потреб користувача.

Таким чином, Linux Mint є доцільним вибором, завдяки своїм численним перевагам та зручностям. Зокрема вибір даного дистрибутиву є доцільним і для специфічних задач, таких як побудова систем захисту інформації де необхідно забезпечити належний рівень захищеності від несанкціонованого доступу.

2.2.2 Вибір середовища реалізації мандатного керування доступом

Для реалізації мандатного керування доступом (MAC) в Linux Mint доступні два механізми AppArmor і SELinux, які надають можливості обмеження дій процесів для підвищення безпеки системи. Розглянемо їх детальніше, щоб визначити, який з них краще підходить для побудови системи захисту інформації на базі обраної ОС.

AppArmor використовує профілі на основі шляхів файлів, що спрощує створення та управління політиками безпеки. Профілі AppArmor можуть бути налаштовані для конкретних програм, обмежуючи доступ до файлів та ресурсів системи [38].

AppArmor менш ресурсномісткий, що може бути перевагою для систем з обмеженими ресурсами і краще інтегрований в деякі дистрибутиви Linux, такі як Ubuntu і, відповідно, Linux Mint.

SELinux забезпечує більш детальний і тонкий контроль над політиками безпеки завдяки контекстам безпеки, що застосовуються до файлів, процесів та інших об'єктів. Налаштування SELinux є складнішим процесом, що вимагає більш глибокого розуміння системи та її безпеки. SELinux пропонує потужніші механізми контролю, що дозволяє створювати більш детальні політики безпеки [39].

Проте, SELinux має вплив на продуктивність системи через більш детальний моніторинг, є складним в налаштуванні і управлінні, що, в свою чергу, вимагає додаткового навчання та досвіду для ефективного використання.

Linux Mint є дистрибутивом, що базується на Ubuntu, яка за замовчуванням використовує AppArmor. Це робить AppArmor більш природним вибором для Linux Mint через кілька причин:

- інтеграція та підтримка;
- простота використання;
- продуктивність.

Отже, для побудови системи захисту інформації на базі Linux Mint використовуватимемо AppArmor через його простоту інтеграції, легкість налаштування, менший вплив на продуктивність та відсутність завищених вимог політики безпеки.

2.2.3 Вибір модуля реалізації політик розширеної аутентифікації

Стандартно в Linux Mint для аутентифікації користувачів використовується традиційний метод аутентифікації за допомогою пароля. Парольні файли зберігаються у файлах директорій /etc/passwd і /etc/shadow, де зберігаються хеші паролів користувачів.

У Linux Mint (як і в інших дистрибутивах на базі Ubuntu) для аутентифікації користувачів може бути перед встановлено модуль PAM. PAM (Pluggable Authentication Modules) — це потужна система аутентифікації, яка забезпечує модульний підхід до аутентифікації користувачів [40].

Використання PAM в системах на базі Linux Mint дозволяє легко додавати, видаляти або змінювати методи аутентифікації, що забезпечує гнучкість у налаштуванні системи безпеки. Він підтримує різноманітні методи аутентифікації, включаючи паролі, одноразові паролі (OTP), смарт-карти, біометричні дані тощо, а також дозволяє централізовано керувати політиками аутентифікації для різних сервісів, що спрощує адміністрування системи. Даним

модулем забезпечується можливість налаштування багатофакторної аутентифікації (MFA), що значно може підвищити рівень безпеки системи.

Використання PAM в якості модуля аутентифікації для системи захисту інформації на базі Linux Mint є доцільним вибором через його гнучкість, модульність і високий рівень безпеки. Оскільки PAM використовується за замовчуванням у Linux Mint, його налаштування та використання є природним і зручним рішенням для адміністраторів безпеки.

2.2.4 Вибір модуля ведення аудиту системи

В операційних системах на базі Linux стандартно для моніторингу та запису подій використовується інструмент gnome-logs. Gnome-logs має графічний інтерфейс та призначений для загального моніторингу системних подій, які корисні з точки зору відстеження загального стану системи та діагностики проблем.

З точки зору безпеки системи, gnome-logs є неприйнятним. Тому потрібно обрати інструмент який матиме достатній функціонал забезпечення високого рівня деталізації подій регламентованими політикою безпеки.

Найбільш відповідним інструментом, в зазначеному вище випадку, є модуль auditd.

Auditd (Audit Daemon) – це компонент Linux Auditing System, який забезпечує докладний запис різноманітних системних подій. Він призначений для аудиту дій користувачів, моніторингу доступу до файлів та системних викликів, а також для збору інформації, для проведення розслідувань інцидентів порушення правил безпеки [41].

Основні відмінності між gnome-logs та auditd наведено в табл.2.1.

Таблиця 2.1 – Порівняльна характеристика систем аудиту

Характеристика	Auditd	gnome-logs
Призначення	Система аудиту для відстеження подій ядра та системних викликів на низькому рівні	Графічний інтерфейс для перегляду системних журналів
Рівень роботи	Низькорівнева робота на рівні ядра	Високий рівень роботи з користувацьким інтерфейсом
Журналювання	Веде журнали аудиту в **/var/log/audit/audit.log**	Відображає журнали з системного журналу (journald)
Налаштування	Вимагає ручного налаштування конфігураційного файлу та правил аудиту	Налаштовується через графічний інтерфейс
Гнучкість	Дуже гнучкий, можна налаштувати детальні правила для відстеження конкретних дій	Менш гнучкий, націлений на перегляд загальних системних журналів
Повідомлення	Можливість налаштування дій при досягненні певних умов (наприклад, відправка електронних листів)	Відсутні можливості для налаштування автоматичних дій
Продуктивність	Висока продуктивність завдяки низькорівневій інтеграції з ядром	Використовує ресурси для відображення графічного інтерфейсу
Цільова аудиторія	Системні адміністратори, адміністратори безпеки	Звичайні користувачі, які потребують перегляду системних логів
Типи подій	Відстежує системні виклики, доступ до файлів, зміни файлів та інші критичні події	Відображає загальні системні журнали, включаючи повідомлення про помилки, попередження та інші події
Зберігання логів	Журнали зберігаються в текстових файлах у форматі RAW або ENRICHED	Журнали зберігаються в системному журналі (journald)
Інтеграція з іншими системами	Може бути інтегрований з іншими системами безпеки та моніторингу через плагіни	Обмежена інтеграція, основний фокус - перегляд журналів

Як бачимо, auditd є потужною і гнучкою системою для низькорівневого моніторингу та аудиту дій у системі, що робить його більш корисним для адміністраторів безпеки та системних адміністраторів.

2.3 Розробка профілів мандатного керування доступом обраного середовища

Для розробки захисту автоматизованої системи на базі ОС Linux Mint з використанням мандатного керування доступом за допомогою AppArmor, перш за все необхідно визначити ролі та права доступу користувачів.

Відповідно до вимог нормативної документації однокористувацькі автоматизовані системи повинні передбачати розподіл їх користувачів на адміністраторів та звичайних користувачі. В свою чергу, ролі адміністраторів також підлягають розподіленню на адміністратора безпеки та системного адміністратора. Є певні випадки, коли ще вводиться роль адміністратора документів, але зазвичай в системах де не високі вимоги до захисту, дана роль суміщається з адміністратором безпеки. Виходячи з цього, розроблювана система має передбачати такі ролі і права:

- Адміністратор безпеки. Має повний доступ до налаштувань безпеки системи, зокрема профілів AppArmor.
- Системний адміністратор. Має доступ до системних ресурсів, але обмежений у доступі до налаштувань безпеки.
- Звичайний користувач. Має обмежений доступ до системи та налаштувань безпеки відповідно, а має доступ тільки до необхідних для роботи програм і ресурсів.

Наступним етапом є розробка профілів, що передбачає визначення політик для додатків та процесів, які використовуються в системі. Як зазначалося раніше, AppArmor профілі створюються для конкретних програм. Такий підхід дає змогу точніше контролювати, які дії дозволені кожній програмі.

Алгоритм створення профілів (рис.2.3) передбачає виконання наступних кроків.

Крок 1. Визначення програм, які використовуються кожним типом користувачів.

Крок 2. Створення базових профілів для кожної програми, що використовується різними типами користувачів.

Крок 3. Конфігурування профіля таким чином, щоб він дозволяв лише ті дії, які необхідні для виконання завдань користувача.

Крок 4. Запуск програми з профілями в режимі complain mode для збору інформації про їх роботу. Коригування профілів, виходячи з отриманих даних.

Крок 5. Переведення профілів в режим примусового виконання (enforce mode) після їх тестування та налаштування.

Крок 6. Перегляд журналів безпеки AppArmor, для виявляти та усунення можливих проблеми. Оновлення профілів при зміні вимог або додатків.

Рисунок 2.3 – Алгоритм створення профілів мандатного керування доступом AppArmor

Операційна система Linux Mint містить у собі достатню кількість інструментів та утиліт, для забезпечення базового, зручного керування та користування. Виключенням є безпекові модулі та програми, які потрібно додатково встановлювати.

Складемо список базових програм, що використовуватимуться в системі.

1) Програми безпеки:

- Firewall Gufw;
- AppArmor;

- Auditd;
- Pluggable Authentication Modules (PAM).

2) Програми системного адміністрування:

- System Settings;
- Update Manager;
- Driver Manager;
- Software Manager;
- Synaptic Package Manager;
- Timeshift;
- System Monitor;
- Disk Usage Analyzer;
- Disks;
- Printers.

3) Програми загального користування:

- офісний пакет LibreOffice;
- браузер Firefox;
- текстовий редактор gedit.

Загальна структура профілю AppArmor зображена на рис.2.4.

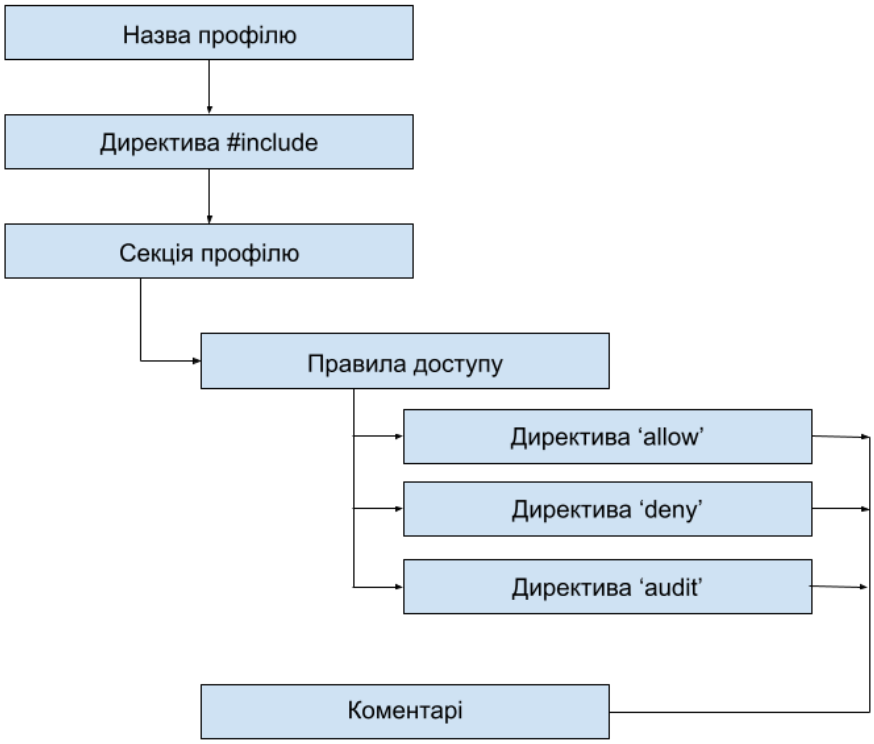


Рисунок 2.4. – Загальна структура профілю мандатного керування доступом

Структура профілю включає:

1) Назва профілю. Ім'я файлу профілю, яке зазвичай відповідає шляху до виконуваного файлу програми.

2) Директива #include. Використовується для включення інших файлів профілю чи абстракцій AppArmor, які визначають додаткові правила або налаштування.

3) Секція профілю. Секція, яка містить правила доступу для конкретного процесу чи програми.

4) Правила доступу. Основні правила, які визначають доступ до файлів, директорій, мережі тощо. Вони включають такі директиви, як allow (дозволяє доступ), deny (забороняє доступ), audit (включає аудит доступу).

5) Коментарі. Коментарі можуть опціонально додаватись для пояснення кожного правила або секції, щоб роз'яснити їхнє призначення або специфіку використання.

Визначивши ролі користувачів і їх права, а також програм які використовуватимуться ними, приступимо до розробки відповідних базових профілів.

Забезпечення необхідного захисту профілями мандатного керування доступом будуватиметься на обмежені доступу визначених програм до директорій з якими можуть працювати визначені користувачі. З урахуванням цього список директорій складається наступним чином:

1) Адміністратор безпеки:

- /etc/ - конфігураційні файли системи. Ця директорія містить конфігураційні файли для багатьох системних служб та програм, які адміністратор безпеки може використовувати для налаштування безпеки системи.
- /var/log/ - лог-файли системи. Доступ до цієї директорії дозволяє адміністратору безпеки переглядати лог-файли для виявлення можливих атак або проблем безпеки.
- /etc/apparmor.d/ - конфігураційні файли AppArmor. Тільки адміністратор безпеки може вносити зміни до профілів безпеки AppArmor, які використовуються для обмеження доступу до ресурсів системи.
- /var/lib/apparmor/ - дані AppArmor. Ця директорія містить дані профілів та усі важливі файли, пов'язані з роботою AppArmor. Адміністратор безпеки може використовувати її для аналізу та моніторингу безпеки.
- /etc/security/ - конфігураційні файли PAM. Директорія містить конфігураційні файли для PAM, які керують політиками автентифікації. Адміністратор безпеки налаштовує ці файли для забезпечення правильних процедур автентифікації.
- /etc/pam.d/ - конфігураційні файли PAM. Містить конфігураційні файли для окремих програм, які використовують PAM для автентифікації.
- /var/lib/pam/ - дані PAM. Містить дані, що використовуються PAM для автентифікації.

- /var/log/audit/ - логи auditd. Ця директорія містить лог-файли, створені auditd. Логи використовуються для детального аудиту та моніторингу активності в системі.
- /etc/audit/ - конфігураційні файли auditd. Містить конфігураційні файли для налаштування auditd. Адміністратор безпеки налаштовує ці файли для визначення політик аудиту.
- /usr/sbin/ - виконувані файли системних програм. Містить виконувані файли для системних утиліт, таких як auditd, які адміністратор безпеки використовує для керування безпекою системи.
- /var/spool/ - директорія для тимчасового зберігання файлів. Може використовуватися для тимчасового зберігання файлів, пов'язаних із безпекою, наприклад, черги обробки логів або даних для перевірки.
- /etc/rsyslog.d/ - конфігураційні файли rsyslog. Містить конфігураційні файли для rsyslog, який використовується для системного логування. Адміністратор безпеки може налаштовувати ці файли для забезпечення відповідного логування безпекових подій.

2) Системний адміністратор:

- /etc/ - конфігураційні файли системи. Системний адміністратор може мати доступ до цієї директорії для налаштування системних служб та інших параметрів.
- /var/log/ - лог-файли системи. Доступ до цієї директорії дозволяє системному адміністратору переглядати лог-файли для відстеження роботи системних служб та виявлення проблем.
- /usr/sbin/ - виконувані файли системних програм. Ця директорія містить виконувані файли для системних утиліт та програм, які системний адміністратор може використовувати для керування системою.
- /var/lib/dpkg/ - дані про встановлені пакети (dpkg).
- /var/lib/systemd/ - дані про сервіси системного менеджера Linux.
- /var/spool/
- /etc/systemd/ - конфігураційні файли для systemd.

- /etc/network/ - конфігураційні файли для мережевих налаштувань.
- /usr/local/sbin/ - локальні виконувані файли системних утиліт.
- /etc/cron.d/ - конфігураційні файли для cron-завдань (завдань планувальника cron для виконання завдань у фоновому режимі.
- /etc/init.d/ - скрипти ініціалізації системних служб.
- /var/backups/ - резервні копії файлів та налаштувань системи.

3) Звичайний користувач:

- /home/username/ - домашня директорія користувача. Кожен користувач має свою домашню директорію, в якій він зберігає свої особисті файли та налаштування.
- /tmp/ - тимчасові файли. Користувач може мати доступ до цієї директорії для збереження тимчасових файлів, наприклад, завантажених документів або виконуваних файлів.
- ~/Documents/ - директорія для документів. Ця директорія може використовуватися користувачем для зберігання основних документів та файлів.
- ~/Downloads/ - директорія для завантажень. Користувач може мати доступ до цієї директорії для збереження файлів, які він завантажує з мережі Інтернет.
- ~/Desktop/ - робочий стіл. Ця директорія може використовуватися для зберігання ярликів до часто використовуваних програм або файлів.
- ~/Pictures/
- ~/Videos/
- ~/Music/

Відповідно до загальної структури профілів, визначених ролей, програм та директорій, що використовуватимуться зазначеними користувачами, загальні профілі мандатного керування доступом системи матимуть вигляд зображеному на рис.2.5 – 2.7. На представлених рисунках наглядно видно як відбуватиметься керування доступом на рівні використовуваних програм адміністратором безпеки, системного адміністратора та звичайного користувача.

Програми які відповідають за безпеку підпорядковуються правилам доступу до відповідних їм директоріям і вони мають права як на запис так і читання файлів з цих директорій (рис.2.5).

Програми системного адміністратора частково обмежені в доступі до директорій з якими працюють безпекові програми. Операційна система Linux Mint як і всі ОС основані на Ubuntu і Debian, влаштована таким чином, що системні і безпекові програми використовують спільні директорії (рис.2.6).

Програми які визначені для використання звичайними користувачами за правилами профілю (рис.2.7) не матимуть доступу, а ні до директорій з якими працюють безпекові програми, а ні до системних директорій.

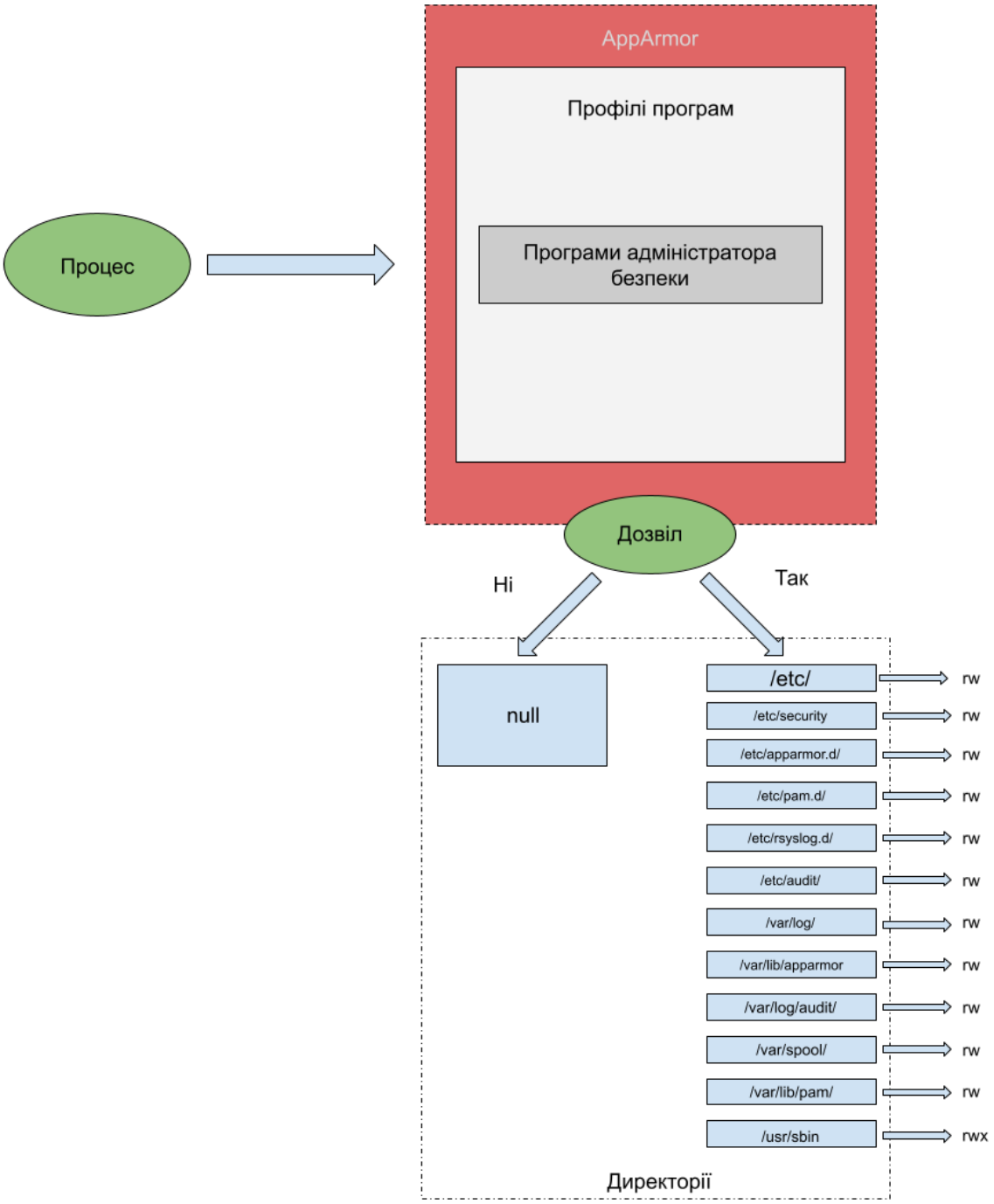


Рисунок 2.5. – Базовий профіль для програм адміністратора безпеки

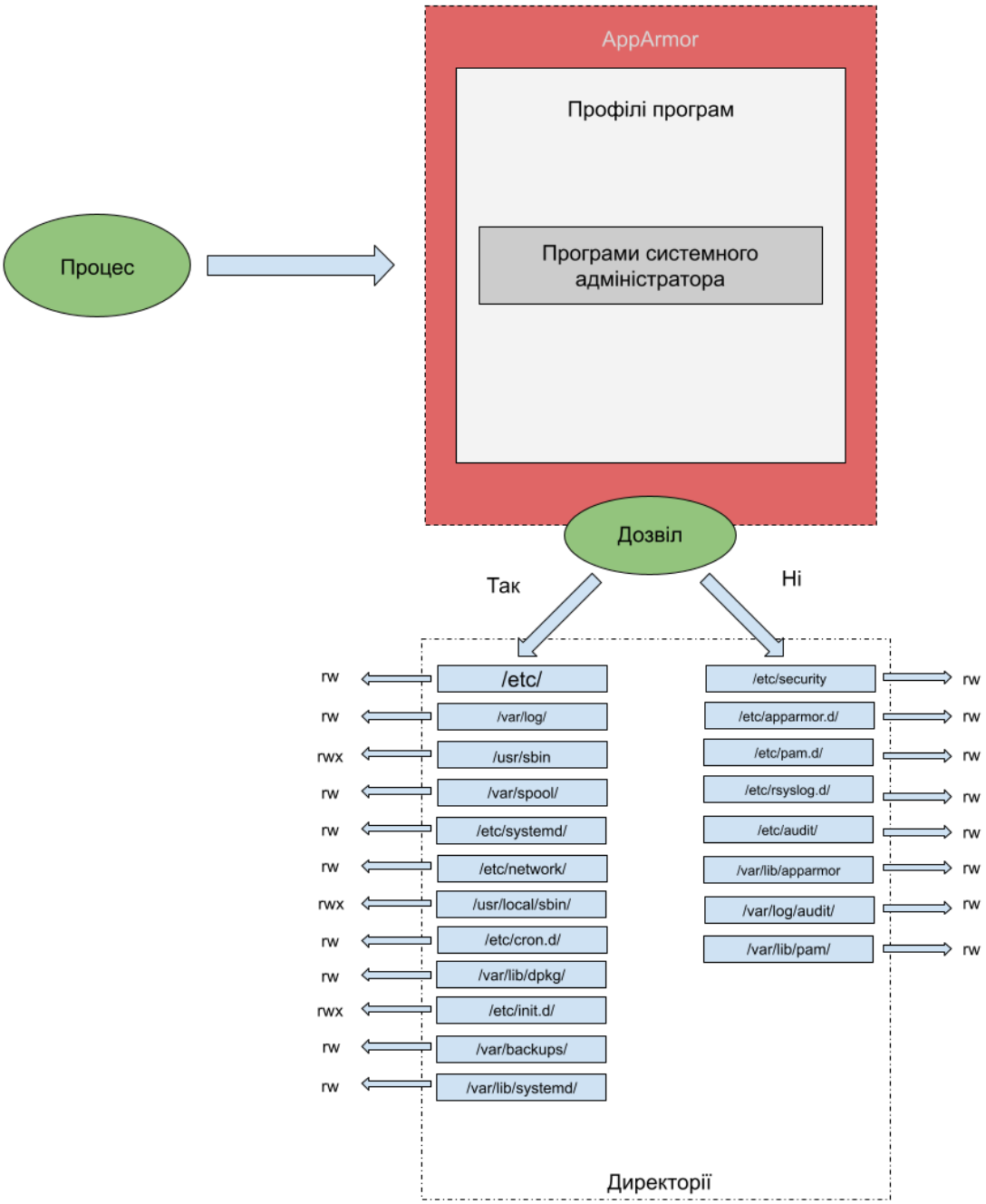


Рисунок 2.6 - Базовий профіль для програм системного адміністратора

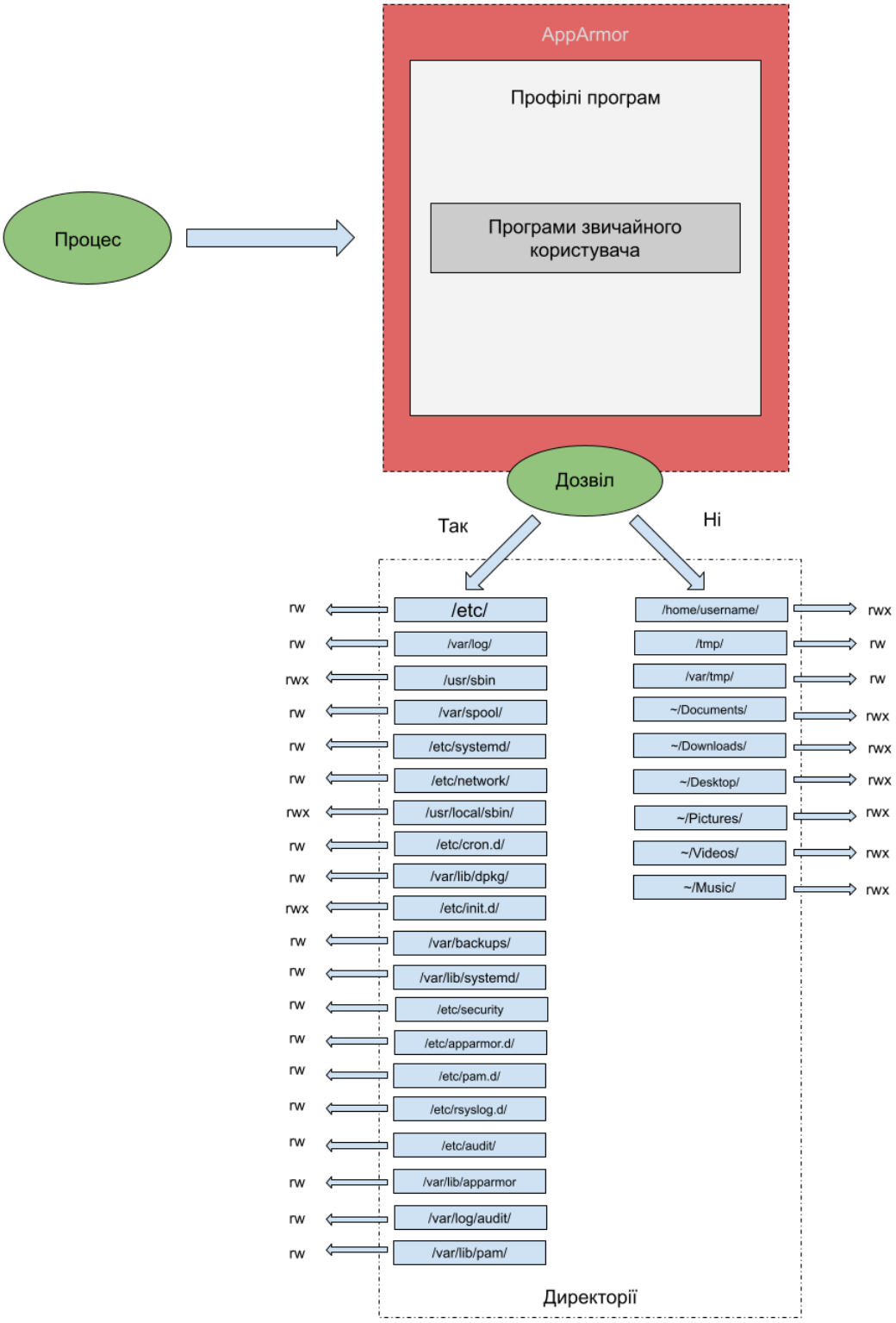


Рисунок 2.7 - Базовий профіль для програм звичайного користувача

Для кожного правила профілів (рис.2.5- 2.7) вказано, які дії дозволені, а які ні: читання – «r», запис – «w», виконання – «x».

Слід зазначити, що розроблені профілі можна доповнювати відповідно до політик безпеки, які можуть встановлюватися середовищем функціонування автоматизованої системи. В такому разі, адміністратору безпеки, після здійснення аналізу, надана можливість зміни профілів, щоб звести до мінімуму ризик несанкціонованого доступу.

2.4 Розробка основи конфігурації компонентів захисту операційної системи

Одними з ключових елементів побудови ефективної системи захисту інформації на базі операційних систем є розробка політики паролів та правил аудиту.

Політика паролів забезпечує захист від несанкціонованого доступу, а правила аудиту дозволяють моніторити дії користувачів і забезпечують можливість вчасно реагувати на інциденти безпеки. В комплексі з іншими заходами, ці заходи сприяють створенню більш безпечного середовища та дозволяють знизити ризики визначеними моделлю загроз.

2.4.1 Розробка правил аудиту системи

На початковому етапі, при розробці правил аудиту потрібно визначити вихідні умови відносно яких вони мають функціонувати.

В загальному, розробка системи захисту інформації на базі операційної системи Linux Mint, яка планується використовуватись для обробки і зберігання інформації. В системі передбачатиметься 3 ролі користувачів: адміністратор безпеки, системний адміністратор і звичайний користувач. В ОС включатимуться такі обрані безпекові програми як: PAM та Apparmor зі створеними власними профілями безпеки. Відносно цих умов правила аудиту системи мають включати:

- моніторинг змін у важливих файлах;
- аудит дій специфічних ролей;
- моніторинг використання правил з підвищеними привілеями;

- моніторинг важливих системних викликів.

Моніторинг системних файлів важливий для забезпечення безпеки системи, оскільки вони містять критичну інформацію про користувачів і їхні паролі. Будь-які несанкціоновані зміни цих файлів можуть свідчити про злом або іншу небажану активність. Ці правила мають встановлювати моніторинг для файлів /etc/passwd, /etc/shadow, та /etc/group. Кожне правило повинно вказувати на файл (-w), права доступу, які потрібно моніторити (-p wa), та ключ (-k), за яким можна буде знайти ці події в журналі аудиту.

Адміністратор безпеки має високий рівень привілеїв і доступ до критичних частин системи. Відстеження всіх дій цього користувача дозволяє забезпечити, що всі його дії є санкціонованими і немає спроб зловживань.

Системний адміністратор має доступ до широкого спектру системних функцій та ресурсів, окрім безпекових аспектів. Як і у випадку адміністратора безпеки, відстеження всіх його дій є критичним для забезпечення, що всі виконувані операції є санкціонованими і немає спроб зловживань або помилкових дій, які можуть вплинути на стабільність або безпеку системи.

Правилами має бути встановлено моніторинг для всіх системних викликів, виконаних користувачем з UID (User ID) який відповідає адміністратору безпеки і системному адміністратору, незалежно від архітектури (64-бітова чи 32-бітова).

Використання високопривілейованої команди sudo дозволяє користувачам виконувати команди з правами адміністратора, що може бути небезпечним у випадку компрометації облікового запису. Відстеження привілейованих ескалацій допомагатиме виявити спроби отримання прав адміністратора несанкціонованими користувачами.

В операційних системах Linux, директорія /etc містить критично важливі конфігураційні файли системи. Відстеження системних викликів open, write та execve у цій директорії дозволяє виявляти спроби несанкціонованого доступу, змін або виконання програм, які можуть вплинути на конфігурацію та безпеку системи.

Наведені вище правила аудиту спрямовані на забезпечення цілісності та безпеки системи Linux Mint. Реалізація та дотримання їх дозволить відстежувати критичні дії та зміни в системі, забезпечуючи можливість своєчасного реагування на потенційні загрози та порушення, а це є невід'ємною частиною системи захисту інформації, що допомагає підтримувати високий рівень безпеки та контролю у системі.

2.4.2 Розробка політики паролів

Розробка надійної політики паролів є важливим аспектом захисту інформації в будь-якій операційній системі, включаючи Linux Mint. Для забезпечення базової надійної політики паролів пропонується встановити наступні вимоги, які можуть бути в подальшому реалізовані та ще більш розширені:

1) Мінімальна довжина пароля: 14 символів.
2) Складність пароля:

- принаймні одна велика літера (A-Z);
- принаймні одна мала літера (a-z);
- принаймні одна цифра (0-9);
- принаймні один спеціальний символ (наприклад, !@#$%^&*()_+).

3) Унікальність пароля:

- заборона використання старих паролів (як мінімум останні 5 паролів не повинні повторюватися).

4) Обмеження повторюваних символів:

- не більше 3 повторюваних символів підряд;
- не більше 4 однакових символів з однієї категорії (велика літера, мала літера, цифра, спеціальний символ) підряд.

5) Зміна паролів:

- примусова зміна пароля кожні 90 днів;
- обов'язкова зміна пароля після першого входу.

Запропонована політика паролів повинна забезпечити високий базовий рівень безпеки розроблюваної системи.

2.5 Висновки

В даному розділі дипломної роботи проведено аналіз загроз системи захисту інформації на базі операційної системи з відкритим вихідним кодом та представлено модель загроз, на основі якої будуватиметься розроблювана система.

Було здійснено обґрунтування вибору складових системи захисту інформації, що включає вибір дистрибутива ОС, середовища реалізації моделі мандатного керування доступом, а також програм для реалізації політики розширеної аутентифікації та ведення аудиту системи.

Особлива увага у цьому розділі приділяється розробці профілів мандатного керування доступом, що реалізуватиметься в обраному середовищі. Розроблені профілі дозволяють детально контролювати дії визначених програм відповідно ролям, що використовуються у системі. Подальше впровадження профілів мандатного керування доступом дасть змогу підвисити надійність системи, забезпечуючи високий рівень безпеки для користувачів та їх даних.

Також було розроблено розширену політику аутентифікації та правил системи журналювання та моніторингу для відстеження дій користувачів і аномальної активності.

3 РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ НА БАЗІ МОДУЛІВ ТА ЯДРА ОПЕРАЦІЙНОЇ СИСТЕМИ LINUX MINT

В третьому розділі бакалаврської дипломної роботи на основі розроблених профілів, політики аутентифікації та правил аудиту здійснюється реалізація мандатного керування доступом, розширеної аутентифікації та аудиту системи захисту інформації на базі операційної системи Linux Mint.

В даному розділі буде проведено і відображено процес встановлення дистрибутиву Linux Mint та його попереднє конфігурування.

Також буде проведено збереження реалізованої конфігурації та реалізовано механізм її розгортання.

3.1 Встановлення обраного дистрибутиву та попереднє конфігурування

Для встановлення і подальшої реалізації було обрано дистрибутив Linux Mint версії 21.3 64-bit з робочим середовищем Cinnamon. Тип інсталяції – стандартний, в який включено всі необхідні базові компоненти для налаштування безпеки, адміністрування та базові користувацькі утиліти.

Перед початком інсталяції необхідно задати початкові параметри облікового запису (рис.3.1). В подальшому даний обліковий запис використовуватиметься як обліковий запис адміністратора безпеки, а перед введенням всіх безпекових компонентів буде слугувати для їх реалізації.

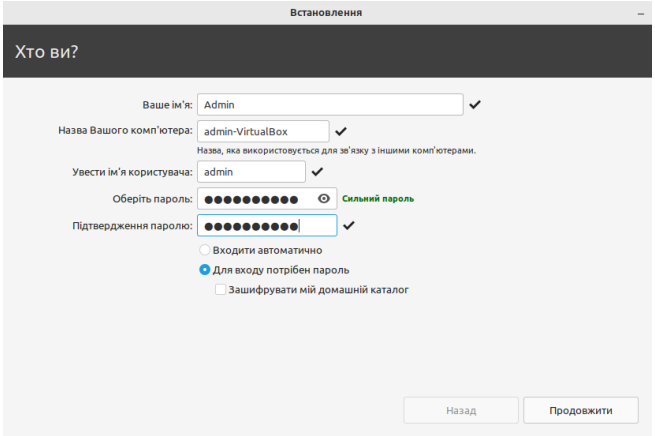


Рисунок 3.1 – Початок інсталяції ОС

На рис.3.2 зображено інтерфейс ОС Linux Mint після завершення процесу інсталяції. Інсталювавши обрану ОС необхідно виконати наступні попередні встановлення та налаштування:

1) інсталювання пакетів безпекових утиліт:

- AppArmor;
- PAM модуль;
- Auditd;

2) створення адміністраторів, користувачів та визначення груп доступу яким вони належатимуть;

3) встановлення відповідних дозволів користувачам та групам до програм безпекового адміністрування.

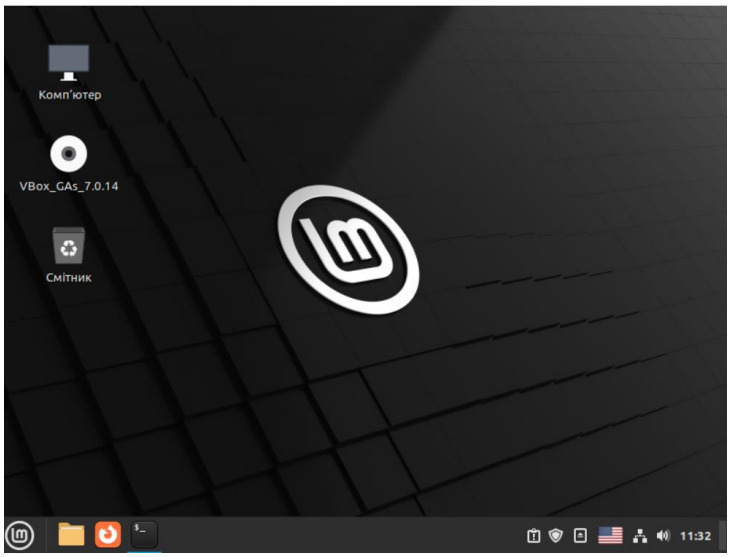


Рисунок 3.2 – Користувацький інтерфейс інстальованої ОС

Для використання і подальшої реалізації мандатного керування доступом в ОС Linux Mint потрібно встановити програмний пакет apparmor та apparmor-utils через термінал (консоль) в репозиторії apt. Перевірка стану AppArmor виконується командою sudo systemctl status apparmor (рис.3.3).

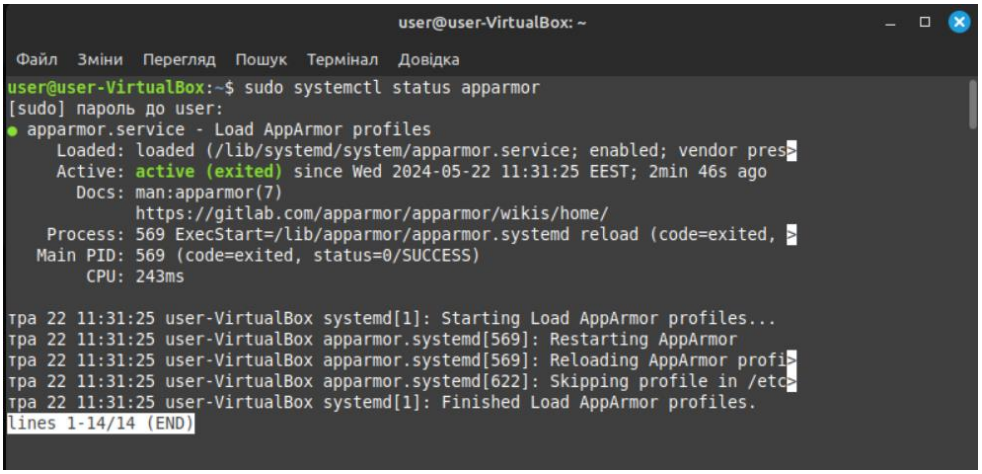


Рисунок 3.3 – Статус роботи AppArmor

Для налаштування політики паролів в Linux Mint необхідна інсталяція пакунків libpam-modules, libpam-pwquality, libpam-cracklib, інструмента Pluggable Authentication Modules (PAM) (рис.3.4), дозволяє налаштовувати правила аутентифікації та політику паролів. Як у випадку з AppArmor, інсталяція виконується через термінал і репозиторій apt.

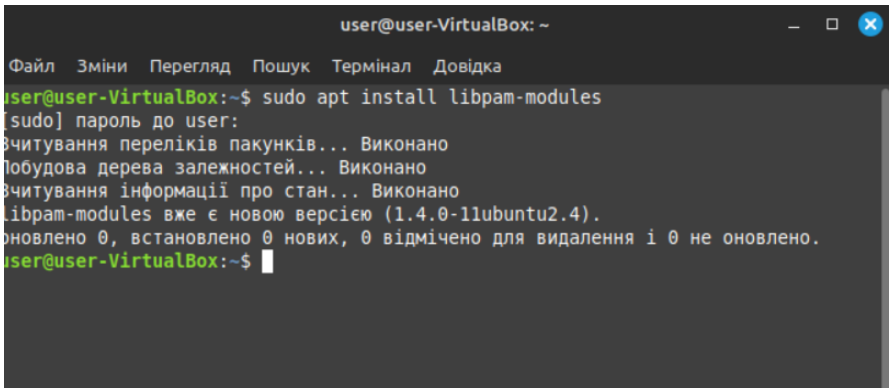


Рисунок 3.4 – Інсталяція PAM-модуля

Операційна система Linux Mint містить у своєму складі достатню кількість програм адміністрування, як безпекою так і роботою ОС, які мають графічний інтерфейс, що значно спрощує роботу з ними. Зокрема, в базовому наборі програм передбачена утиліта *Журнали*, яка дозволяє переглядати детальний журнал системних подій. Проте функціонал даної утиліти обмежений і передбачає лише експорт журналу подій. Операційні системи на базі Linux є досить гнучкими, де будь-яка зміна має вплив на забезпечення захищеності, адміністратору безпеки потрібно детально проводити аудит системи, щоб вчасно примати рішення щодо проведення відповідних заходів. Таким чином, детальний

аудит системи є необхідним. Потужною утилітою яка може використовуватися в такому випадку є auditd. Дана програма дозволяє налаштовувати свої сценарії журналювання подій і є має високу сумісність з іншими програми. На рис. 3.5 зображено процес інсталяції auditd.

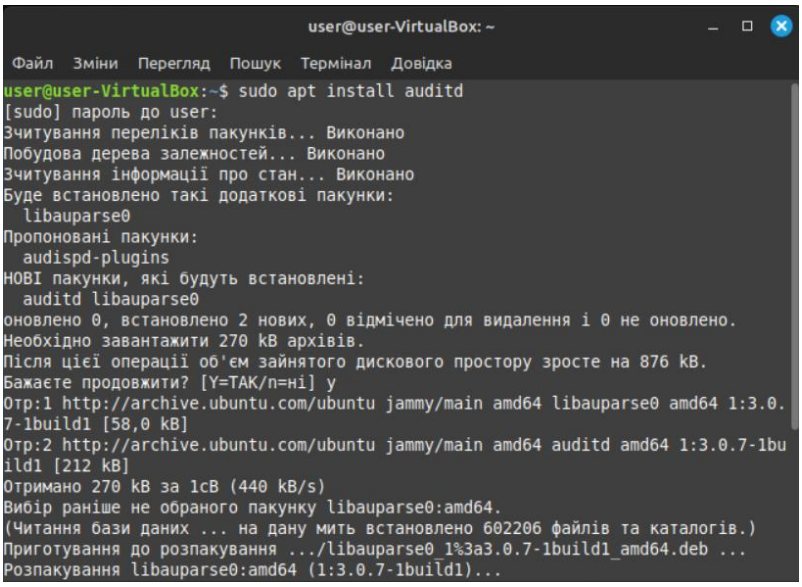


Рисунок 3.5 – Інсталяція програми аудиту auditd

Наступним кроком попереднього конфігурування системи є створення облікових записів, передбачені політикою безпеки та надання відповідних дозволів, щоб забезпечити перший шар безпеки системи (рис.3.6). Таким чином, було створено наступні облікові записи та групи яким вони належатимуть:

1) адміністратор безпеки – Sec_admin (група security_admin);
2) адміністратор системи – Syst_admin (група sys_admin);
3) користувач – Stand_user (група regular_user).

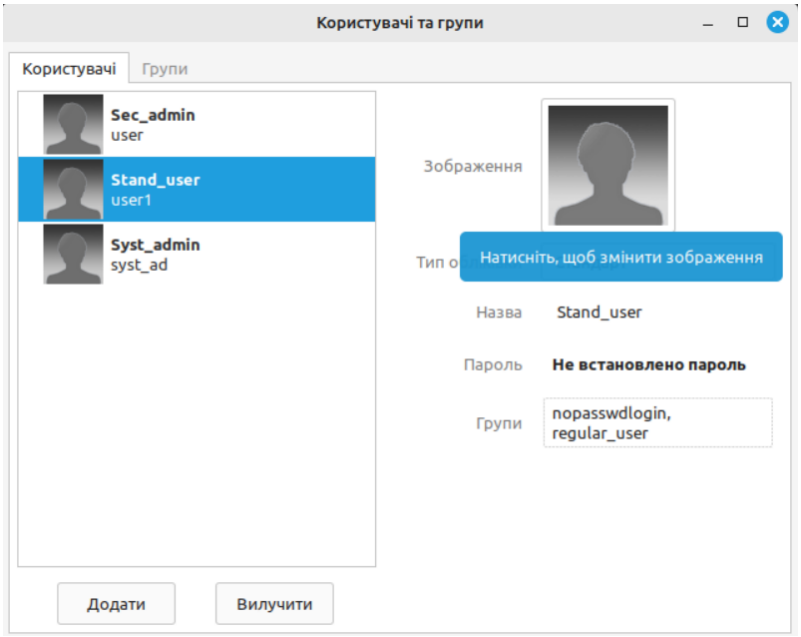


Рисунок 3.6 – Створення облікових записів передбачених політикою безпеки

Створивши відповідні облікові записи вікно входу набуде вигляду зображеного на рис.3.7.

Рисунок 3.7 – Вікно входу в систему

Подальші дії з реалізації виконуватимуться обліковим записом адміністратора безпеки, так як після створення інших, даний акаунт має необхідні права надані операційною системою.

Основою системи захисту інформації є мандатне керування доступом. Перед введенням даної моделі розмежування доступом потрібно встановити перший шар захисту, що полягає у встановлені виключних прав якими може

володіти тільки супер користувач (root) на роботу з безпековими програмами, описаними вище та безпосередньо терміналом (консоллю).

Як зазначалось раніше, в даній ОС визначено групу, що відповідає за безпеку - security_admin. Для обмеження доступу до програм, що відповідають за безпеку в Linux Mint, групі користувачів security_admin, потрібно змінити власника та групу командами sudo chown та sudo chmod для кожної програми, що відповідає за безпеку, щоб вони належали користувачу root та групі security_admin, а також налаштувати дозволи так, щоб тільки користувач root та члени групи security_admin могли виконувати ці програми (рис 3.8).

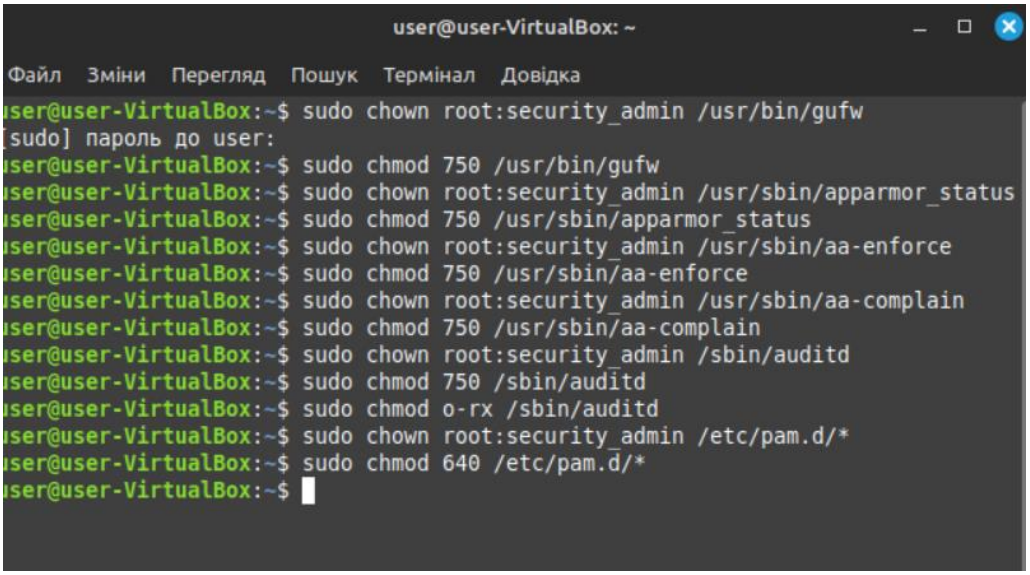


Рисунок 3.8 – Визначення програм, прав та відповідних доступів

Команда chmod 750 встановлює такі права:

- Власник має всі права: читання, запис і виконання.
- Група має права на читання і виконання, але не на запис.
- Інші користувачі не мають жодних прав доступу до файлу чи каталогу.

Команда chmod 640 встановлює такі права:

- Власник має права на читання і запис, але не виконання.
- Група має лише права на читання.
- Інші користувачі не мають жодних прав доступу до файлу чи каталогу.

Процес перевірки доступів зображено на рис.3.9.

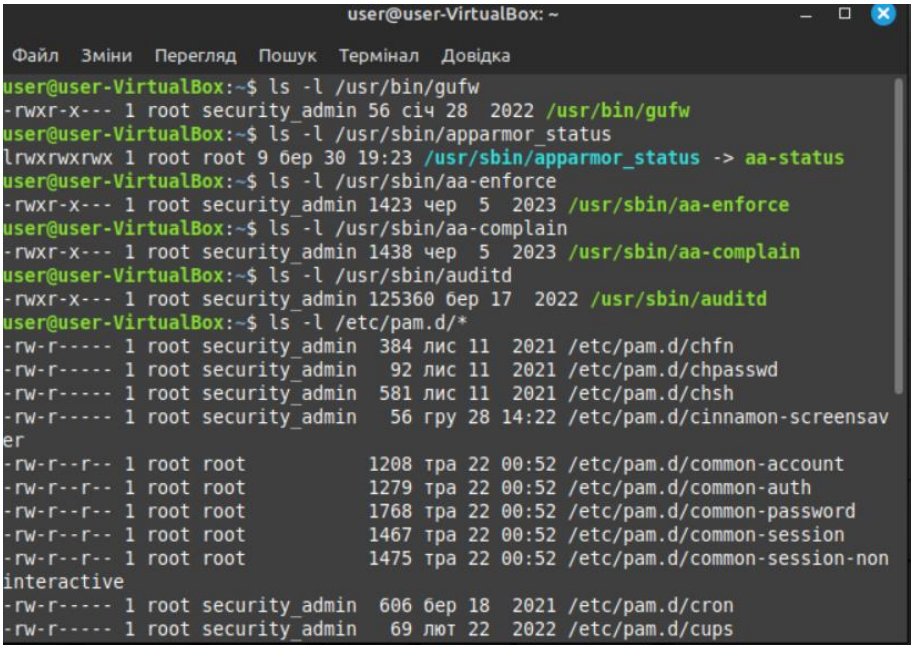


Рисунок 3.9 – Статус налаштування доступів і дозволів для групи security_admin

Головним інструментом адміністрування будь-якої операційної системи на базі Linux є термінал (консоль). Для попередження подальшого впливу на систему захисту інформації потрібно виконати ізоляцію терміналу від користувачів, та розподілити доступ між такими групами як security_admin та sys_admin. Базовими налаштуваннями безпеки ОС Linux Mint з-під облікових записів, що є звичайними користувачами, виключено можливість використання терміналу, а при спробі його використання ними запитується пароль адміністратора безпеки. Проте, для більш строгої ізоляції терміналу, варто виконати обмеження на рівні ядра операційної системи.

У Linux Mint можна обмежити доступ до терміналу для декількох груп, але це потребує певної організації, оскільки файлова система Linux Mint дозволяє встановлювати права лише для одного власника та однієї групи на файл чи директорію. Є кілька способів, як це можна зробити, зокрема використання ACL (Access Control Lists) або створення проміжного скрипта. В нашому випадку доцільним є використання скрипта. Створення скрипта може бути простішим для налаштування та підтримки.

Для створення відповідного скрипта необхідно визначити які термінальні емулятори встановлені в системі (рис.3.10). Для цього необхідно виконати команду ls /usr/bin | grep terminal.

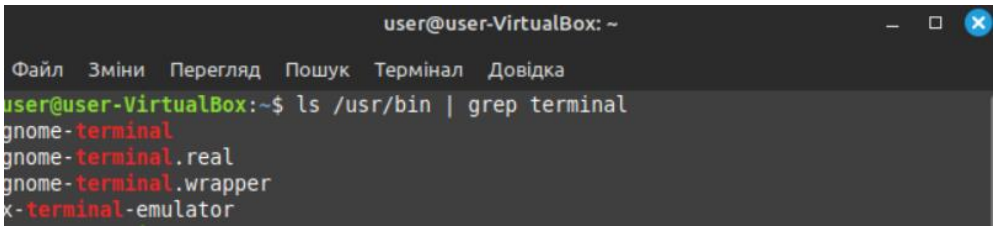


Рисунок 3.10 – Список встановлених емуляторів терміналу в поточній системі

Отриману інформацію можна використати для побудови наступного універсального скрипта який перевірятиме наявність в системі потрібного терміналу і запускатиме його для відповідних груп:

sudo nano /usr/local/bin/secure-terminal

```
# Перевірка, чи користувач належить до однієї з дозволених груп
if groups $USER | grep -q '\bsecurity_admin\b' || groups $USER | grep -q '\bsys_admin\b'; then
   # Список терміналів для перевірки
   TERMINALS=("gnome-terminal" "gnome-terminal.real" "gnome-terminal.wrapper" "x-terminal-emulator")
   # Перебір терміналів і запуск першого доступного
   for TERM in "${TERMINALS[@]}"; do
      if command -v $TERM &> /dev/null; then
         exec $TERM
         exit 0
      fi
   done
   # Якщо жоден з терміналів не знайдено
   echo "No terminal emulator found."
   exit 1
else
   # Повідомлення про відмову у доступі
   echo "Access denied. You must be in the security_admin or sys_admin group to run this program."
   exit 1
fi
```

Виконання роботи даного скрипта забезпечується тим, що команда запуску шортката терміналу (консолі) змінюється на /usr/local/bin/secure-terminal замість стандартного /usr/bin/gnome-terminal.

Отже, виконавши попереднє конфігурування ОС таким чином, систему можна вважати готовою для впровадження наступного етапу, що полягає у реалізації мандатного керування доступом.

3.2 Реалізація мандатного керування доступом

Реалізація моделі мандатного керування доступом (MAC) в обраній для побудови системи захисту інформації від несанкціонованого доступу ОС, виконуватиметься за допомогою AppArmor.

Налаштувавши розмежування моделлю мандатного керування доступом на рівні програм за допомогою модуля AppArmor, дозволить додатково обмежити дії, які вони можуть виконувати, та забезпечити більш високий рівень безпеки.

Для конфігурування профілів використовується власний формат конфігураційних файлів, який є специфічним для AppArmor. Цей формат не є повноцінною мовою програмування, як Python чи C++, але є декларативною мовою конфігурації, яка дозволяє визначати правила безпеки для застосунків.

Конфігураційні файли AppArmor зазвичай мають розширення .profile і містять правила, що описують, до яких ресурсів (файли, мережеві порти тощо) додаток має або не має доступу.

Як зазначалося раніше, AppArmor застосовується до конкретних програм або процесів, а не безпосередньо до облікових записів користувачів. Однак, можна створити профілі для критичних програм, які будуть використовувати ці користувачі.

В Розділі 2 даної бакалаврської дипломної роботи було визначено перелік програм включених в ОС для яких необхідно реалізувати відповідні профілі доступу.

3.2.1 Реалізація та тестування профілів доступу програм безпеки

Відповідно до програм які були визначені як ті, що мають відношення до управління безпекою в ОС, для розмежування доступу по моделі мандатного керування доступом потрібно створити профілі. Профілі включатимуть в себе правила обмеження доступу. Також, варто зауважити, що деякі з цих програм можуть бути скриптами або виконуваними файлами, розташованими в різних каталогах.

Першим етапом в реалізації профілів є перевірка статусу AppArmor та його активація в разі необхідності (рис.3.11).

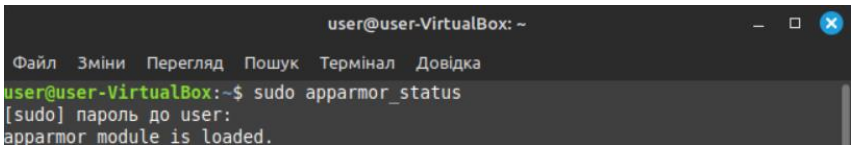


Рисунок 3.11 – Перевірка статусу роботи модуля AppArmor

Далі виконаємо безпосереднє створення і впровадження створених профілів.

Створимо профіль usr.sbin.apparmor для самої програми AppArmor, впровадимо його (рис.3.12-3.13).

Створення нового профілю відбувається шляхом створення нового текстового файлу в директорії модуля за допомогою команди sudo nano /etc/apparmor.d/usr.sbin.apparmor. Рис.3.12 демонструє вікно створеного файлу профілю, в який вписано розроблені правила.

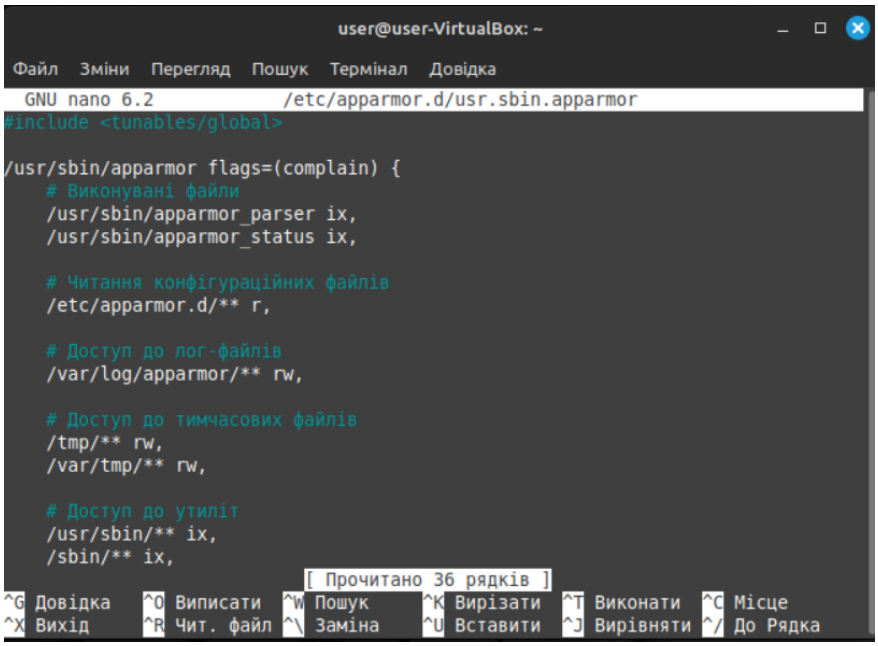


Рисунок 3.12 – Вікно редагування профілю usr.sbin.apparmor

Лістинг профілю usr.sbin.apparmor :

```
#include <tunables/global>
/usr/sbin/apparmor {
  # Виконувані файли
  /usr/sbin/apparmor_parser ix,
  /usr/sbin/apparmor_status ix,
  # Читання конфігураційних файлів
```

```
    /etc/apparmor.d/** r,
    # Доступ до лог-файлів
    /var/log/apparmor/** rw,
    # Доступ до тимчасових файлів
    /tmp/** rw,
    /var/tmp/** rw,

    # Доступ до утиліт
    /usr/sbin/** ix,
    /sbin/** ix,
    /usr/bin/** ix,
    /bin/** ix,
    # Доступ до системних ресурсів
    capability sys_admin,
    capability audit_control,
    capability dac_read_search,
    # Доступ до мережевих інтерфейсів
    network inet stream,
    network inet6 stream,
    # Обмеження доступу до конфіденційних файлів
    deny /etc/shadow r,
    deny /etc/passwd r,
}
```

Розглянемо детально значення прапорів (flag) та директив реалізованого профілю для самої програми AppArmor.

1) Виконувані файли:

- ix (inherit execute) дозволяє виконання програми та наслідування її контексту.

2) Читання конфігураційних файлів:

- r (read) дозволяє читання конфігураційних файлів, необхідних для роботи програми.

3) Доступ до лог-файлів:

- rw (read/write) дозволяє читання та запис у лог-файли.

4) Доступ до тимчасових файлів:

- rw (read/write) дозволяє створення, читання та запис тимчасових файлів.

5) Доступ до системних утиліт:

- ix (inherit execute) дозволяє виконання системних утиліт.

6) Доступ до сокетів та мережевих інтерфейсів:

- rw (read/write) для сокетів.

- network inet stream та network inet6 stream дозволяють доступ до мережевих інтерфейсів.

7) Доступ до системних ресурсів:

- capability дозволяє виконання певних системних функцій.

8) Обмеження доступу до конфіденційних файлів:

- deny забороняє доступ до конфіденційних файлів, таких як /etc/shadow та /etc/passwd.

Провівши необхідне конфігурування файлу профілю, наступним кроком є його завантаження (рис.3.13). Якщо синтаксичних чи інших помилок в файлі профілю не було допущено, команда завантаження виконуватиметься одразу.

```
user@user-VirtualBox:~$ sudo nano /etc/apparmor.d/usr.sbin.apparmor
user@user-VirtualBox:~$ sudo apparmor_parser -r /etc/apparmor.d/usr.sbin.apparmo
r
```

Рисунок 3.13 – Створення та завантаження профілю в AppArmor

Після збереження профілю usr.sbin.apparmor необхідно його активувати (рис.3.14) та перевірити статус (рис.3.15). Так як система захисту інформації знаходиться на етапі розробки доцільним є використання режиму complain.

Після повного створення всіх необхідних профілів та виконання реалізації всіх необхідних послуг, створені профілі можна перевести в режим enforce.

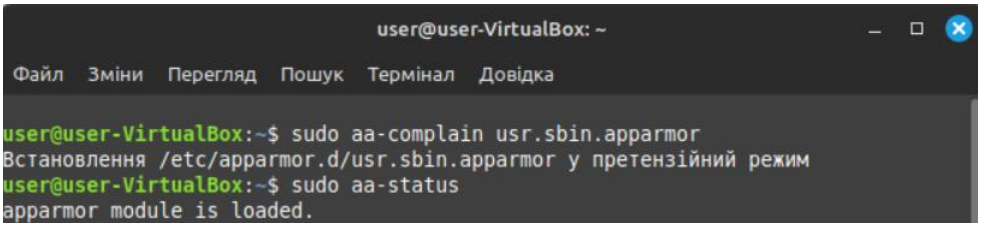


Рисунок 3.14 – Активування створеного профілю usr.sbin.apparmor

Рисунок 3.15 – Перевірка статусу профілю usr.sbin.apparmor

За таким самим принципом створимо профіль usr.lib.security.pam для модуля аутентифікації PAM (рис.3.16). Для створення профілю AppArmor для програми PAM, потрібно врахувати специфічні файли та директорії, до яких PAM

повинна мати доступ. Оскільки PAM зазвичай працює на рівні системи і забезпечує автентифікацію, профіль повинен враховувати доступ до конфігураційних файлів, бібліотек та системних утиліт.

Лістинг профілю usr.lib.security.pam:

```
#include <tunables/global>
/usr/lib/security/pam {
  # Читання конфігураційних файлів PAM
  /etc/pam.d/** r,
  /etc/security/** r,
  # Доступ до необхідних бібліотек
  /lib/x86_64-linux-gnu/** r,
  /usr/lib/x86_64-linux-gnu/** r,
  # Доступ до системних утиліт
  /bin/** ix,
  /usr/bin/** ix,
  /sbin/** ix,
  /usr/sbin/** ix,
  # Читання лог-файлів
  /var/log/auth.log r,
  /var/log/secure r,
  /var/log/faillog rw,
  /var/log/btmp rw,
  # Доступ до тимчасових файлів
  /tmp/** rw,
  /var/tmp/** rw,
  # Виконання необхідних команд
  capability dac_override,
  capability sys_admin,
  capability net_admin,
  capability chown,
  capability setuid,
  capability setgid,
  # Запобігання доступу до конфіденційних файлів
  deny /etc/shadow r,
  deny /etc/passwd r,
  # Дозволити запис у /var/log/audit
  /var/log/audit/** rw,
  # Доступ до домашніх директорій
  /home/*/.pam_environment r,
   # Запуск профілю на read-only файлових системах
  mount options=(ro),
  # Дозвіл на створення та видалення власних тимчасових файлів
  /run/user/*/pam-*.tmp rwk,
  # Запобігання доступу до інших користувачів
  deny /home/*/ r,
  # Підтримка зовнішніх модулів автентифікації
  /etc/pam_ldap.conf r,
  /etc/nsswitch.conf r,
  # Доступ до мережевих інтерфейсів для автентифікації
```

```
  network inet stream,
  network inet6 stream,
  # Додаткові права доступу (якщо необхідно)
  # /etc/ldap.conf r,
  # /etc/krb5.conf r,
}
```

Реалізований профіль для PAM:

1) дозволяє PAM читати свої конфігураційні файли;
2) дозволяє доступ до системних бібліотек, необхідних для роботи PAM;
3) дозволяє виконувати системні утиліти, необхідні для автентифікації;
4) дозволяє PAM читати та записувати лог-файли автентифікації;
5) дозволяє доступ до тимчасових файлів;
6) директиви capability дозволяють PAM виконувати системні функції, необхідні для автентифікації;
7) запобігає читанню конфіденційних файлів;
8) дозволяє записувати в лог-файли аудиту;
9) дозволяє читати файли конфігурації користувачів;
10) забезпечує роботу профілю на файлових системах, змонтованих лише для читання;
11) дозволяє створення та видалення тимчасових файлів PAM;
12) запобігає доступу до домашніх директорій інших користувачів;
13) дозволяє читати конфігураційні файли зовнішніх модулів;
14) дозволяє мережевий доступ для автентифікації.

```
user@user-VirtualBox:~$ sudo nano /etc/apparmor.d/usr.lib.security.pam
user@user-VirtualBox:~$ sudo apparmor_parser -r /etc/apparmor.d/usr.lib.security
.pam
```

Рисунок 3.16 – Створення та завантаження профіля PAM

Так як і в попередньому випадку, створений і завантажений профіль потрібно активувати і перевірити статус (рис.3.17).

```
3 profiles are in complain mode.
   /usr/lib/security/pam
   libreoffice-oosplash
   libreoffice-soffice
```

Рисунок 3.17 – Статус профілю usr.lib.security.pam

Наступними реалізуємо профілі sbin.auditd та usr.bin.gufw які відноситимуться до програми аудиту auditd та брандмауера Gufw (рис.3.18-3.19).

```
user@user-VirtualBox:~$ sudo nano /etc/apparmor.d/sbin.auditd
[sudo] пароль до user:
user@user-VirtualBox:~$ sudo apparmor_parser -r /etc/apparmor.d/sbin.auditd
Skipping profile in /etc/apparmor.d/disable: sbin.auditd
user@user-VirtualBox:~$ sudo nano /etc/apparmor.d/usr.bin.gufw
user@user-VirtualBox:~$ sudo apparmor_parser -r /etc/apparmor.d/usr.bin.gufw
Skipping profile in /etc/apparmor.d/disable: usr.bin.gufw
```

Рисунок 3.18 – Створення та завантаження профілів sbin.auditd та usr.bin.gufw

```
6 profiles are in complain mode.
   /usr/bin/gufw
   /usr/lib/security/pam
   /usr/sbin/apparmor
   /usr/sbin/auditd
   libreoffice-oosplash
   libreoffice-soffice
```

Рисунок 3.19 – Статус завантажених профілів

Лістинги профілів sbin.auditd та usr.bin.gufw наведено в Додатку В.

Створивши всі необхідні профілі для визначених програм протестуємо їх роботу. Тестування профілів AppArmor важливе для перевірки належного функціонування і перевірки забезпечення необхідного рівня безпеки без надмірних обмежень.

Показником коректності налаштування правил профілів в нашому випадку є те, чи програми і відповідні їм служби перезапустяться та виконуватимуть команди після того як профіль буде активовано.

Перезапустимо службу auditd:

sudo systemctl restart auditd

sudo systemctl status auditd

Після виконання цих команд служба аудиту повинна перезапуститись і працювати без проблем. На рис.3.20 продемонстровано результат перезапуску служби аудиту.

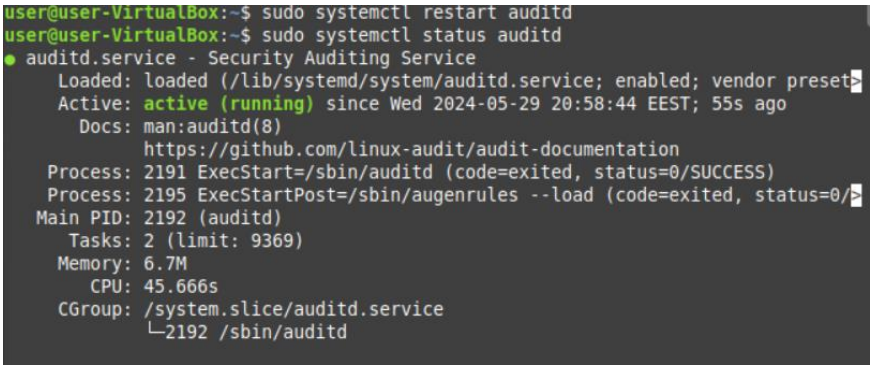


Рисунок 3.20 – Перезапуск auditd

Перезапустимо службу AppArmor:

sudo systemctl restart apparmor

sudo systemctl status apparmor

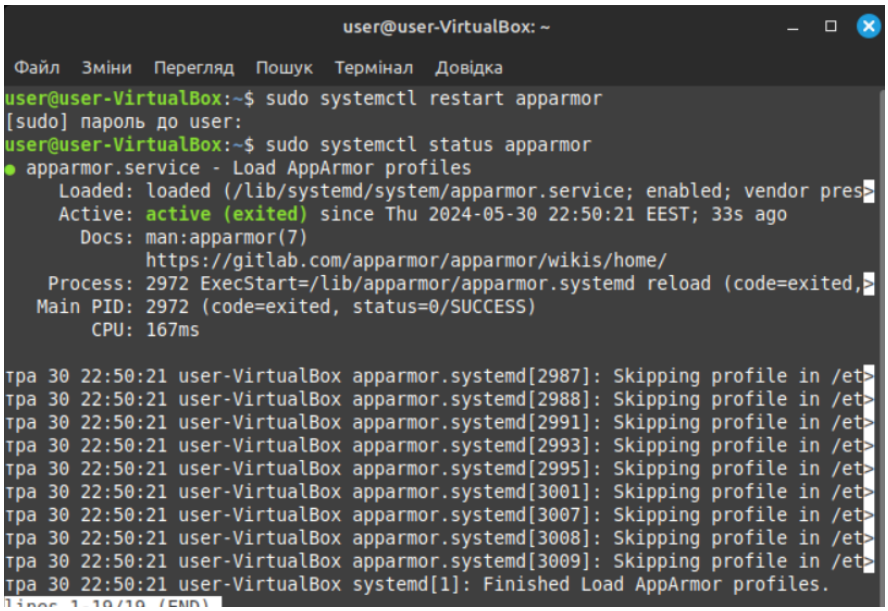


Рисунок 3.21 – Перезапуск AppArmor

На рис.3.21 видно, що служба AppArmor перезапустилась успішно і без проблем.

Запуск брандмауера Gufw можна виконати двома шляхами: через термінал або через меню графічного інтерфейсу ОС. Процес запуску Gufw та його вікно зображено на рис.3.22-3.23.

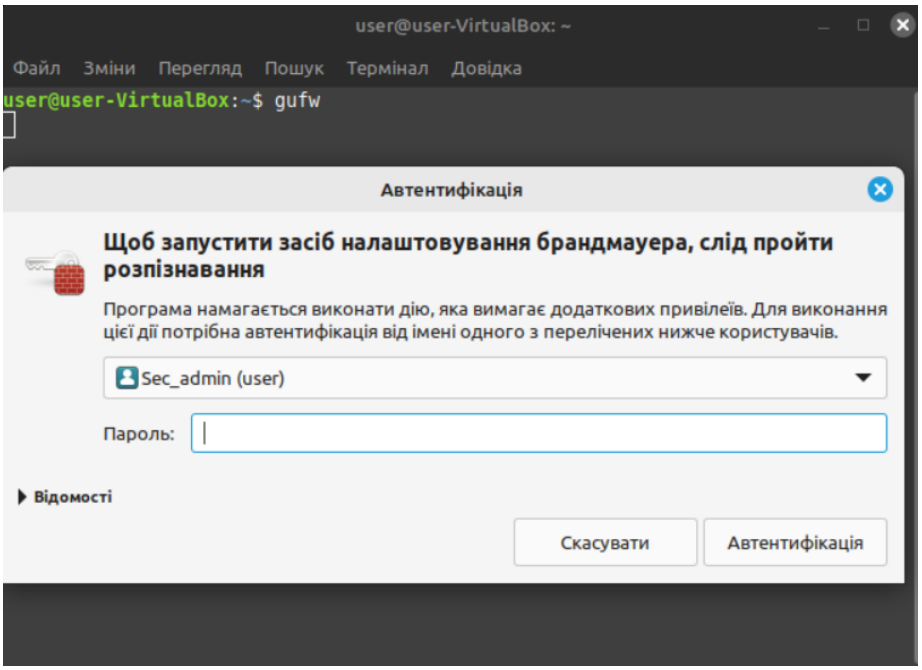


Рисунок 3.22 – Запуск Gufw

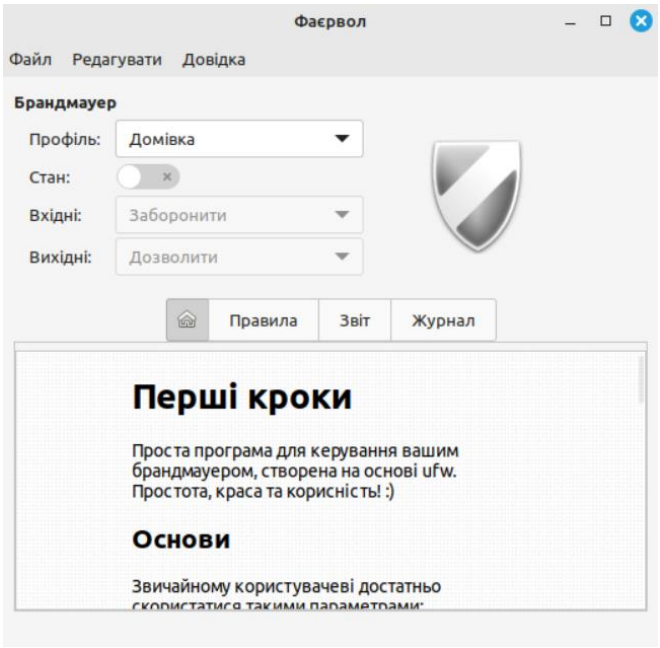


Рисунок 3.23 – Вікно Gufw

Як бачимо, початково профілі працюють як очікується, можна вважати їх успішно налаштованими та готовими для подальшого використання.

Результат виконання команд визначеними програма буде продемонстровано в наступних пунктах, оскільки подальша реалізація компонентів системи захисту інформації відбуватиметься з увімкненими реалізованими профілями.

3.2.2 Реалізація та тестування профілів доступу програм системного адміністрування

Аналогічно крокам які були виконані для п.3.2.1 виконаємо реалізацію профілів для програм системного адміністрування визначеного в Розділі 2, що базуються на їх типовій функціональності (рис.3.24). Лістинги всіх конфігураційних файлів профілів наведено в Додатку Г.

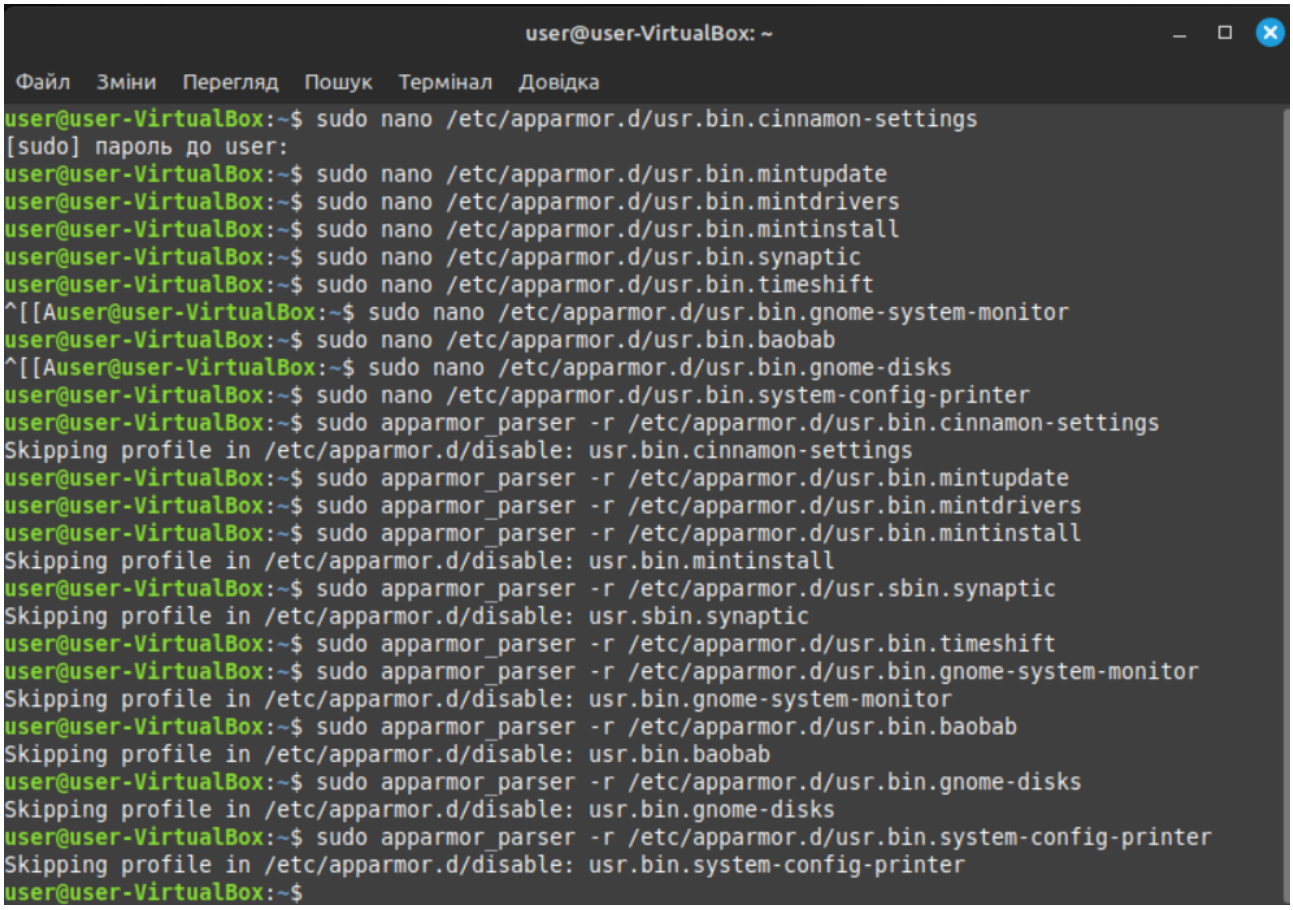


Рисунок 3.24 – Створення та завантаження профілів програм системного адміністрування

Як і у випадках описаних в п.3.2.1 всі розроблені профілі до кінцевого введення в робочій режим знаходитимуться в режимі complain. Результат активації профілів зображено на рис. 3.25.

```
16 profiles are in complain mode.
   /usr/bin/baobab
   /usr/bin/cinnamon-settings
   /usr/bin/gnome-disks
   /usr/bin/gnome-system-monitor
   /usr/bin/gufw
   /usr/bin/mintdrivers
   /usr/bin/mintinstall
   /usr/bin/mintupdate
   /usr/bin/system-config-printer
   /usr/bin/timeshift
   /usr/lib/security/pam
   /usr/sbin/apparmor
   /usr/sbin/auditd
   /usr/sbin/synaptic
   libreoffice-oosplash
   libreoffice-soffice
```

Рисунок 3.25 – Статус активації профілів програм системного адміністрування

Оскільки, зі списку визначених програм утиліта cinnamon-settings є багатокомпонентною і містить близько десятка основних категорій налаштувань робочого середовища, конфігурування і тестування профіля виконаємо для цього випадку.

Лістинг профіля usr.bin.cinnamon-settings:

```
#include <tunables/global>
profile /usr/bin/cinnamon-settings {
  # Include the standard policy
  #include <abstractions/base>
  # Доступ до конфігураційних директорій
  /etc/cinnamon/** r,
  /usr/share/cinnamon/** r,
  /usr/lib/cinnamon-settings/** r,
  # Виконання бінарних файлів
  /usr/bin/cinnamon-settings Pix,
  # Доступ до налаштувань користувача
  owner @{HOME}/.config/cinnamon/** rw,
  # Обмеження доступу до критичних системних файлів
  /etc/shadow r,
  /etc/passwd r,
    # Доступ до системного D-Bus
  /var/run/dbus/system_bus_socket rw,
    # Доступ до необхідних файлів для налаштувань системи
  /usr/bin/gsettings rix,
  /usr/bin/python3 rix,
  /usr/lib/python3.8/** r,
  # Заборона всього іншого
  deny / rw,
}
```

Розглянемо профіль usr.bin.cinnamon-settings детальніше. Даний профіль:

1) включає базові політики AppArmor, що забезпечують стандартні дозволи для програм;

2) дозволяє читання всіх файлів у вказаних директоріях, що необхідно для нормальної роботи cinnamon-settings;

3) дозволяє виконання бінарного файлу cinnamon-settings з обмеженням доступу до інших файлів та директорій;

4) дозволяє користувачеві читати та записувати налаштування у своїй домашній директорії, де зберігаються конфігураційні файли Cinnamon;

5) дозволяє тільки читання файлів /etc/shadow та /etc/passwd. Це потрібно для доступу до інформації про користувачів, але забороняє зміну цих файлів;

6) дозволяє читання і запис до системного D-Bus, що необхідно для комунікації з іншими компонентами системи;

7) дозволяє виконання та читання файлів, пов'язаних з налаштуваннями системи та скриптами на Python, що використовуються cinnamon-settings;

8) забороняє доступ до всіх інших файлів та директорій, які не вказані у профілі. Це забезпечує обмеження доступу програми тільки до необхідних ресурсів.

Для тестування профілю можна спробувати виконати дії, які порушують правила, визначені в профілі. Це дозволить перевірити, чи працюють ці правила належним чином.

Щоб перевірити, чи працюють правила профілю, спробуємо отримати доступ до системних файлів, які не вказані в профілі. Ця дія повинна бути заблокована профілем (рис.3.26).

```
user@user-VirtualBox:~$ sudo -u user1 cat /etc/sudoers
cat: /etc/sudoers: Відмовлено у доступі
```

Рис.3.26 – Спроба доступу до системних файлів

Перевіримо дії, які передбачені профілем як дозволені, а саме спробуємо виконати запис у користувацькі налаштування (рис.3.27).

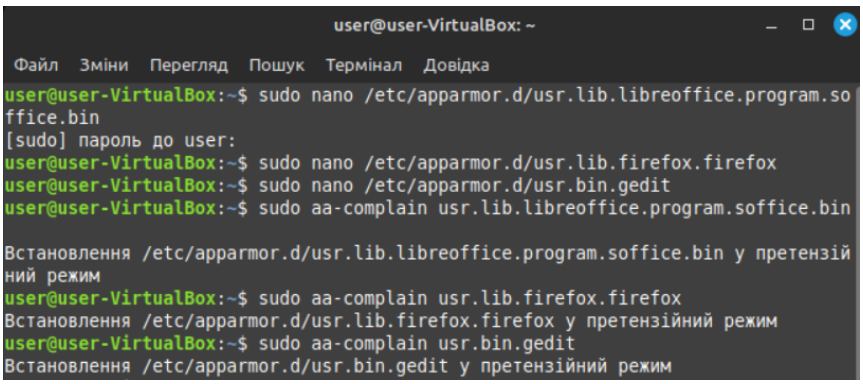
```
user@user-VirtualBox:~$ echo "test" > ~/.config/cinnamon/testfile
cat ~/.config/cinnamon/testfile
test
```

Рисунок 3.27 – Спроба запису в користувацькі налаштування

Таким чином, бачимо, що розроблений профіль коректний, виконується та готовий для впровадження в продуктивний режим.

3.2.3 Реалізація та тестування профілів доступу програм загального користування

Відповідно до визначеного списку програм загального користування, що є стандартними в ОС Linux Mint, створимо і активуємо наступні профілі: профіль офісного пакету LibreOffice - usr.lib.libreoffice.program.soffice.bin; браузера Firefox - usr.lib.firefox.firefox; текстового редактора Gedit - usr.bin.gedit (рис.3.28). Лістинги конфігурації профілів програм загального користування наведено в Додатку Д.



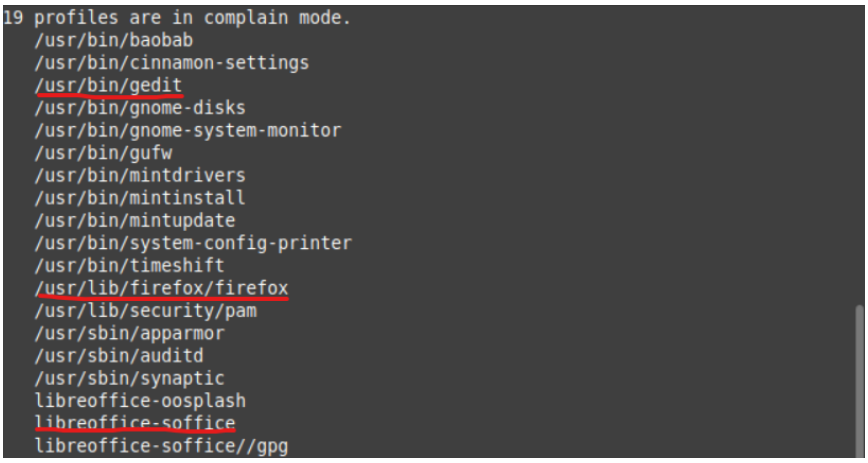


Рисунок 3.28 – Створення та активація профілів програм загального користування

В більшості випадків робота звичайних користувачів автоматизованих систем зводиться до роботи з документами, а отже і прив’язка цих користувачів в роботі зміщена в бік використання застосунків офісного пакету LibreOffice. Тому важливим є реалізація конфігурування профілю для нього.

Розглянемо детальніше створений профіль usr.lib.libreoffice.program.soffice.bin. Лістинг конфігураційного файлу профілю наступний:

```
#include <tunables/global>

/usr/lib/libreoffice/program/soffice.bin {
  # Включення стандартних політик
  #include <abstractions/base>
  # Виконання бінарного файлу
  /usr/lib/libreoffice/program/soffice.bin Pix,
  # Доступ до конфігураційних файлів
  /etc/libreoffice/** r,
  # Доступ до налаштувань і документів користувача
  owner @{HOME}/.config/libreoffice/** rw,
  owner @{HOME}/Documents/** rw,
  # Доступ до тимчасових файлів
  /tmp/** rw,
  # Доступ до шрифтів
  /usr/share/fonts/** r,
  /usr/local/share/fonts/** r,
  # Мережевий доступ
  network inet,
  # Доступ до принтерів
  /var/spool/cups/** rw,
  /usr/lib/cups/backend/** x,
  # Обмеження доступу до чутливих файлі
```

```
  deny /etc/shadow r,
  deny /etc/gshadow r,
  deny /etc/sudoers r,
  # Default deny all other access
  deny / rw,
}
```

Роботою профілю usr.lib.libreoffice.program.soffice.bin передбачено:

- дозвіл виконання самого бінарного файлу soffice.bin;
- дозвіл читання конфігураційних файлів LibreOffice;
- дозвіл читання і запис налаштувань користувача і документів у домашній директорії;
- дозвіл читання і запис у тимчасову директорію;
- дозвіл читання системних і локальних шрифтів;
- дозвіл мережевого доступу для онлайн-функцій (за потреби, адміністратором безпеки можна вимкнути);
- дозвіл доступу до файлів і виконання команд, пов'язаних з друком;
- заборону доступу до файлів з паролями і конфігурацій sudo.

Щоб перевірити роботу профілю потрібно виконати базові дії, які виконуватимуться при стандартній роботі з офісним пакетом, а саме запуск однієї з програм пакету, робота з ним, збереження файлу в робочу директорію та відкриття збереженого файлу (рис.3.29). Збереження тестового файлу відбувалась в директорію Документи ОС Linux Mint, до якої, створеним профілем, передбачено доступ.

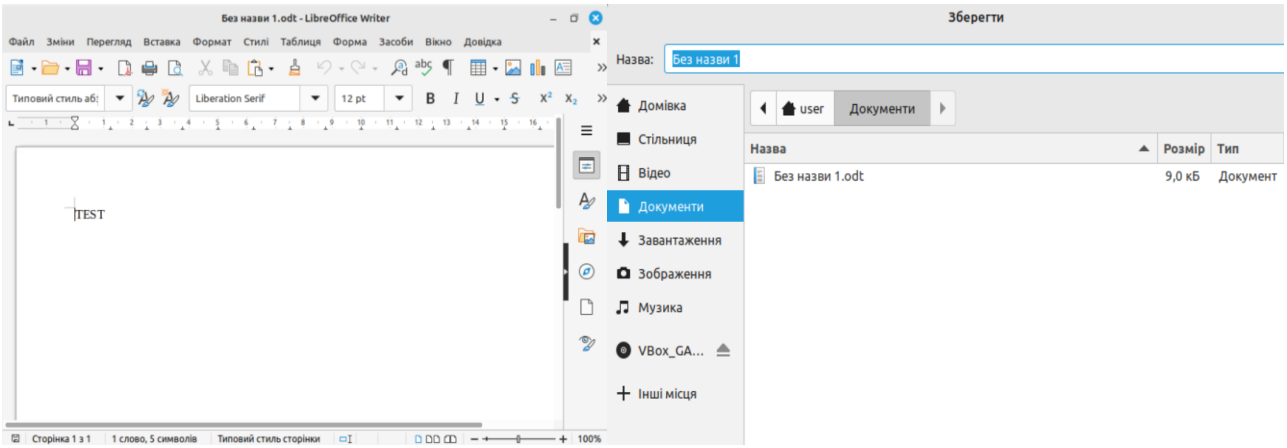


Рисунок 3.29 – Створення та збереження документу

Після виконання цих дій, в системному журналі даної програми повинні відобразитися вказані події (рис.3.30).

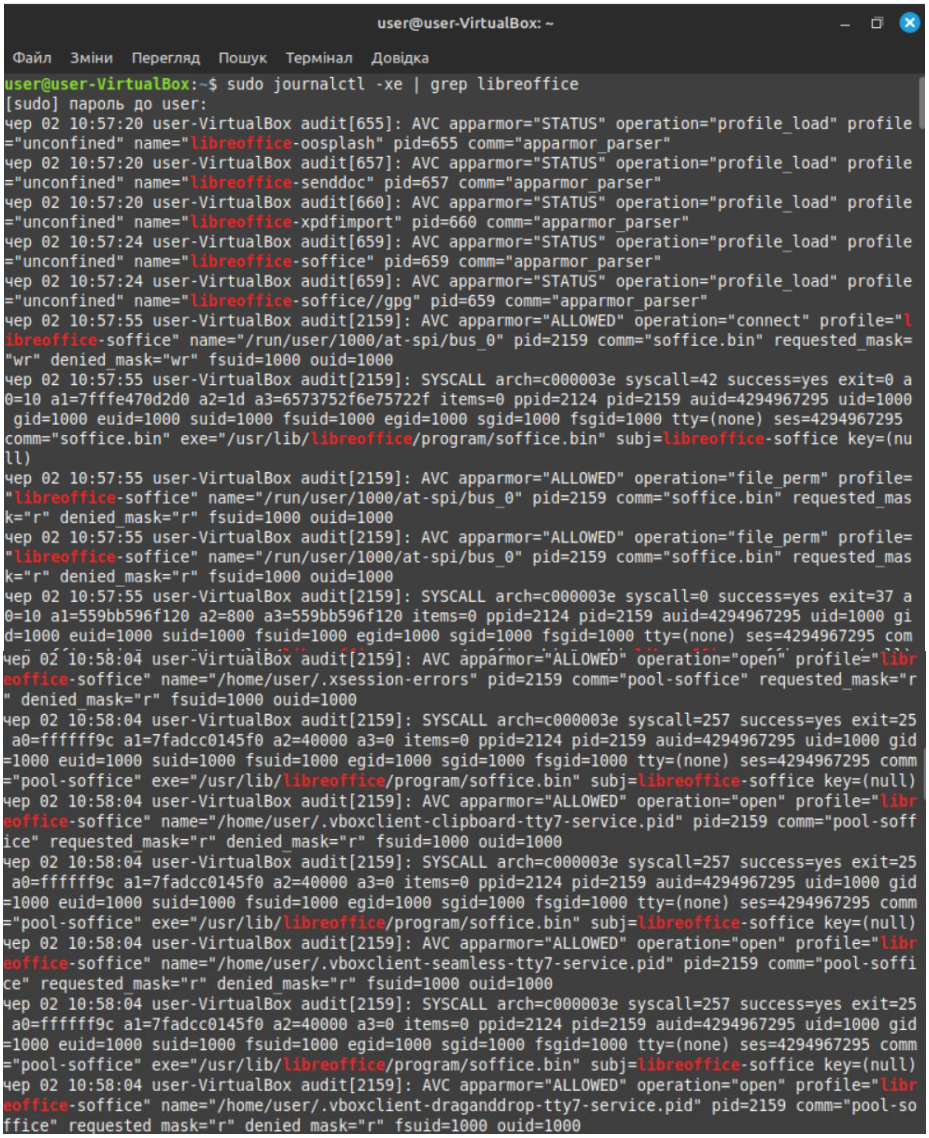


Рисунок 3.30 – Журнал подій

Як можна побачити з журналу подій, при спробі відкриття створеного та збереженого раніше тестового файлу, функціонал передбачений роботою самої програми має відповідні дозволи декларовані реалізованим профілем AppArmor usr.lib.libreoffice.program.soffice.bin.

3.3 Реалізація правил аудиту системи

При налаштуванні auditd для операційної системи, яка буде використовуватися як база для побудови системи захисту інформації, необхідно

забезпечити максимальну безпеку та надійність журналів аудиту. За безпеку відповідатиме розроблений профіль для програми аудиту. Для підвищення надійності потрібно змінити конфігураційні налаштування для auditd, що містяться в /etc/audit/auditd.conf.

Лістинг оновленого конфігураційного файлу:

```
local_events = yes
write_logs = yes
log_file = /var/log/audit/audit.log
log_group = adm
log_format = ENRICHED
flush = INCREMENTAL_ASYNC
freq = 50
max_log_file = 20
num_logs = 10
priority_boost = 4
name_format = NONE
##name = mydomain
max_log_file_action = SUSPEND
space_left = 200
space_left_action = EMAIL
verify_email = yes
action_mail_acct = security_admin
admin_space_left = 100
admin_space_left_action = SUSPEND
disk_full_action = HALT
disk_error_action = HALT
use_libwrap = yes
tcp_listen_port = 60
tcp_listen_queue = 5
tcp_max_per_addr = 2
tcp_client_ports = 1024-65535
tcp_client_max_idle = 600
transport = TCP
krb5_principal = auditd
##krb5_key_file = /etc/audit/audit.key
distribute_network = no
q_depth = 4000
overflow_action = SUSPEND
max_restarts = 5
plugin_dir = /etc/audit/plugins.d
end_of_event_timeout = 2
```

В порівнянні зі стандартними налаштуваннями, запропонована конфігурація орієнтована на підвищення рівня безпеки та надійності системи аудиту. Вони забезпечують кращий моніторинг та інформування адміністраторів,

збільшують обсяг збережених історичних даних, а також включають більш жорсткі дії при критичних ситуаціях для запобігання втраті даних.

Конфігурувавши програму аудиту, наступним етапом є імплементація розроблених правил аудиту в неї і в систему захисту інформації в цілому. Впровадження власних правил аудиту передбачає конфігурування файлу /etc/audit/rules.d/audit.rules. Конфігурація і реалізація розроблених правил аудиту зображено на рис.3.31.

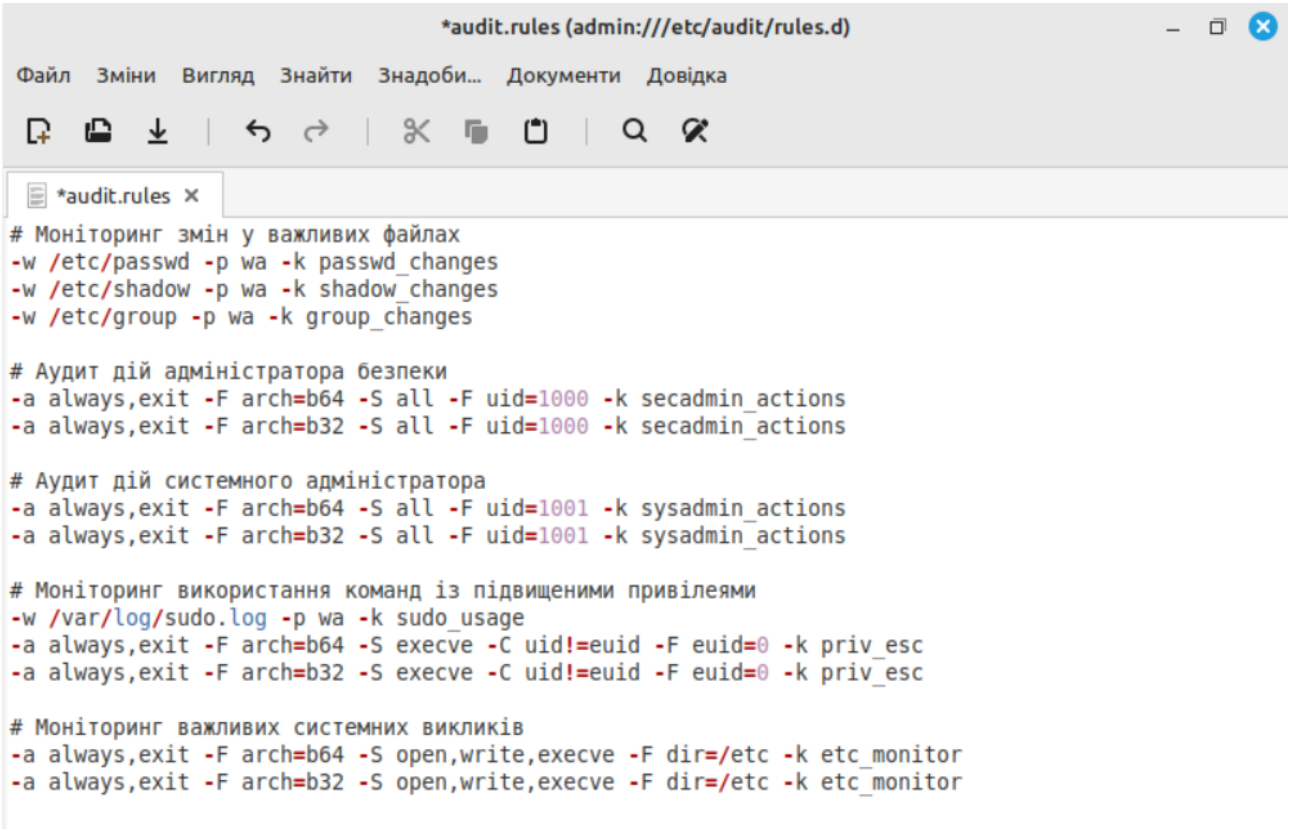
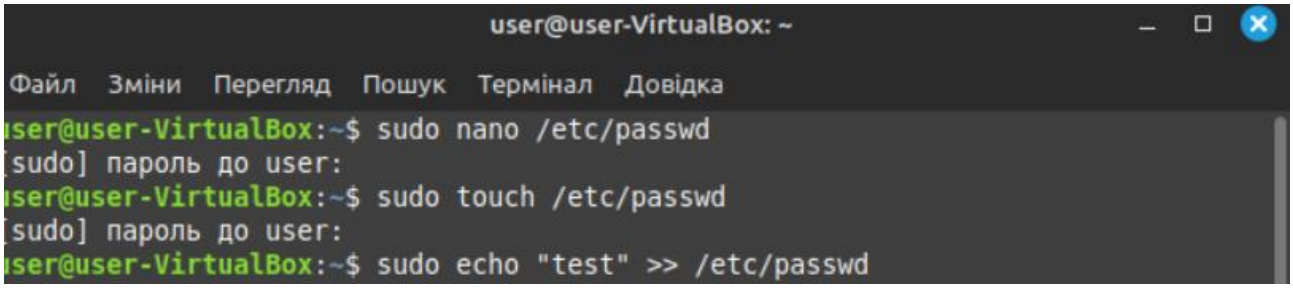


Рисунок 3.31 – Конфігурація правил аудиту

Для перевірки роботи правил, виконаємо дії, які повинні створювати записи в журналах (рис.3.32).



```
user@user-VirtualBox:~$ touch /tmp/testfile
user@user-VirtualBox:~$ ls -l /tmp/testfile
-rw-rw-r-- 1 user user 0 чер  2 17:53 /tmp/testfile
user@user-VirtualBox:~$ rm /tmp/testfile
```

Рисунок 3.32 – Виконання команд дій за обліковим записом

Після виконання команд тестових подій, відповідні записи повинні з’явитись в журналі подій. Здійснимо перевірку по трьом критеріям: перевірку дій адміністратора безпеки, перевірку виконання команди sudo та перевірку системних викликів (рис.3.33)

```
user@user-VirtualBox:~$ sudo ausearch -ua 1000
----
```

```
time->Sun Jun  2 15:55:19 2024
type=PROCTITLE msg=audit(1717332919.095:5463): proctitle=2F7573722F62696E2F56426
F78436C69656E74002D2D64726167616E6464726F70
type=SYSCALL msg=audit(1717332919.095:5463): arch=c000003e syscall=47 success=no
 exit=-11 a0=6 a1=7f1c76694b10 a2=0 a3=7f1c76694c80 items=0 ppid=1573 pid=1574 a
uid=4294967295 uid=1000 gid=1000 euid=1000 suid=1000 fsuid=1000 egid=1000 sgid=1
000 fsgid=1000 tty=(none) ses=4294967295 comm="VBoxClient" exe="/opt/VBoxGuestAd
ditions-7.0.14/bin/VBoxClient" subj=unconfined key="secadmin_actions"
----
time->Sun Jun  2 15:55:19 2024
type=PROCTITLE msg=audit(1717332919.095:8360): proctitle=2F7573722F62696E2F56426
F78436C69656E74002D2D64726167616E6464726F70
type=SYSCALL msg=audit(1717332919.095:8360): arch=c000003e syscall=230 success=y
es exit=0 a0=0 a1=0 a2=7f1c76694c90 a3=7f1c76694c80 items=0 ppid=1573 pid=1574 a
uid=4294967295 uid=1000 gid=1000 euid=1000 suid=1000 fsuid=1000 egid=1000 sgid=1
000 fsgid=1000 tty=(none) ses=4294967295 comm="VBoxClient" exe="/opt/VBoxGuestAd
ditions-7.0.14/bin/VBoxClient" subj=unconfined key="secadmin_actions"
----
user@user-VirtualBox:~$ sudo ausearch -k sudo_usage
----
time->Sun Jun  2 15:55:18 2024
type=PROCTITLE msg=audit(1717332918.991:2219): proctitle=2F7362696E2F61756469746
3746C002D52002F6574632F61756469742F61756469742E72756C6573
type=PATH msg=audit(1717332918.991:2219): item=0 name="/var/log/" inode=341688 d
ev=08:03 mode=040775 ouid=0 ogid=111 rdev=00:00 nametype=PARENT cap_fp=0 cap_fi=
0 cap_fe=0 cap_fver=0 cap_frootid=0
type=CWD msg=audit(1717332918.991:2219): cwd="/"
type=SOCKADDR msg=audit(1717332918.991:2219): saddr=100000000000000000000000
type=SYSCALL msg=audit(1717332918.991:2219): arch=c000003e syscall=44 success=ye
s exit=1084 a0=3 a1=7ffd66d54e20 a2=43c a3=0 items=1 ppid=4401 pid=4415 auid=429
4967295 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4
294967295 comm="auditctl" exe="/usr/sbin/auditctl" subj=unconfined key=(null)
type=CONFIG_CHANGE msg=audit(1717332918.991:2219): auid=4294967295 ses=429496729
5 subj=unconfined op=add_rule key="sudo_usage" list=4 res=1
user@user-VirtualBox:~$ sudo ausearch -k etc_monitor
----
time->Sun Jun  2 15:55:18 2024
type=PROCTITLE msg=audit(1717332918.991:2243): proctitle=2F7362696E2F61756469746
3746C002D52002F6574632F61756469742F61756469742E72756C6573
type=PATH msg=audit(1717332918.991:2243): item=0 name="/etc" inode=917505 dev=08
:03 mode=040755 ouid=0 ogid=0 rdev=00:00 nametype=NORMAL cap_fp=0 cap_fi=0 cap_f
e=0 cap_fver=0 cap_frootid=0
type=CWD msg=audit(1717332918.991:2243): cwd="/"
type=SOCKADDR msg=audit(1717332918.991:2243): saddr=100000000000000000000000
type=SYSCALL msg=audit(1717332918.991:2243): arch=c000003e syscall=44 success=ye
s exit=1072 a0=3 a1=7ffd66d54e20 a2=430 a3=0 items=1 ppid=4401 pid=4415 auid=429
4967295 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4
294967295 comm="auditctl" exe="/usr/sbin/auditctl" subj=unconfined key=(null)
type=CONFIG_CHANGE msg=audit(1717332918.991:2243): auid=4294967295 ses=429496729
5 subj=unconfined op=add_rule key="etc_monitor" list=4 res=1
----
```

Рисунок 3.33 – Журнал подій

Використавши команду ausearch, можна відстежити події, які повинні відповідати налаштованим правилам аудиту. Як бачимо, правила аудиту

працюють належним чином. Виконані дії відображаються в журналі подій. Також, можемо побачити, що програма auditd працює коректно, а це означає, що безпековий профіль для даної програми правильно налаштований і працює.

3.4 Реалізація розширеної аутентифікації до системи

Для реалізації розробленої політики паролів для адміністраторів та користувачів системи за допомогою модуля PAM потрібно виконати конфігурування файлу /etc/pam.d/common-password (рис.3.34).

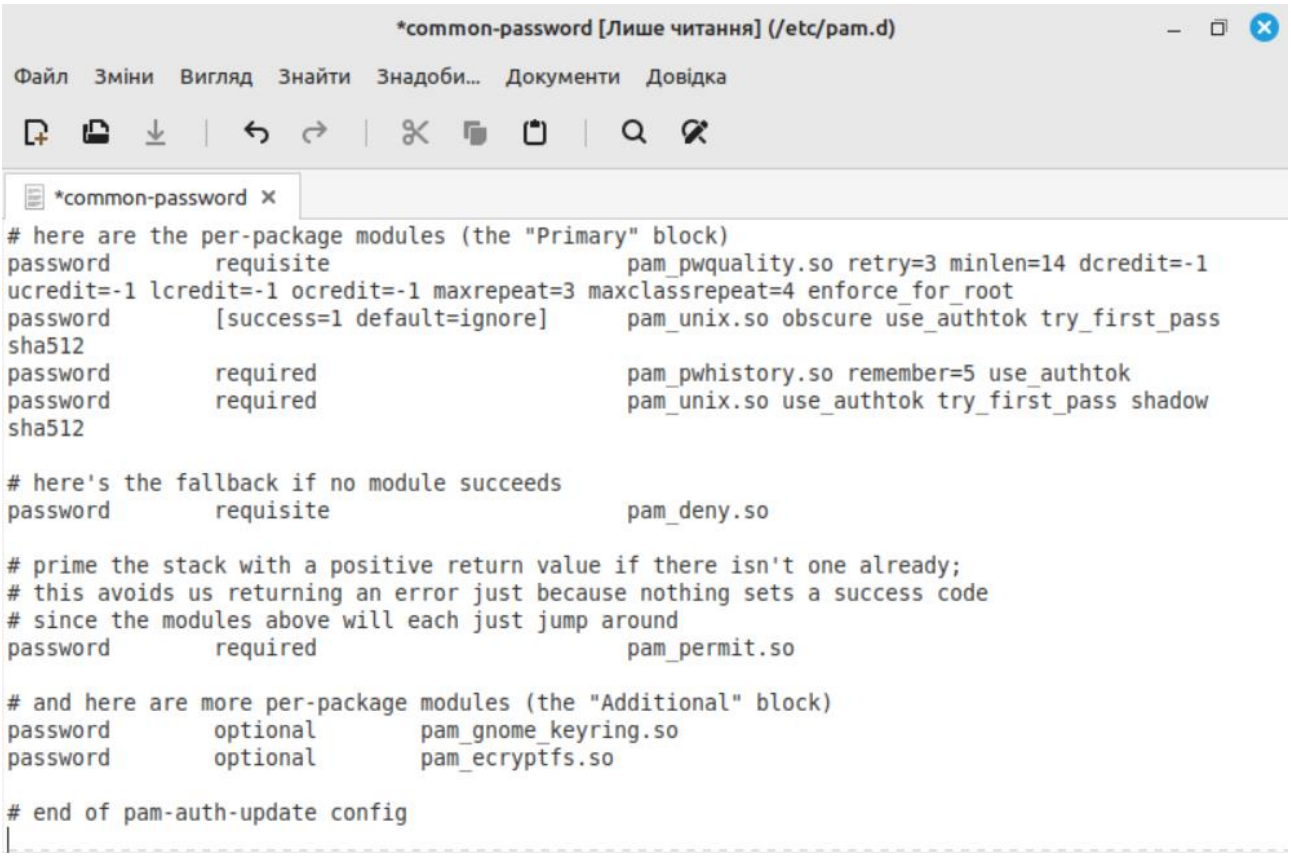


Рисунок 3.34 – Конфігурація визначеної політики паролів

Відповідно до файлу конфігурації встановлюються наступні параметри:

- retry=3. Дозволяє користувачеві зробити три спроби введення пароля.
- minlen=14. Мінімальна довжина пароля повинна бути 14 символів.
- dcredit=-1. Пароль повинен містити принаймні одну цифру.
- ucredit=-1. Пароль повинен містити принаймні одну велику літеру.
- lcredit=-1. Пароль повинен містити принаймні одну малу літеру.

- ocredit=-1. Пароль повинен містити принаймні один спеціальний символ.
- maxrepeat=3. Не більше 3 повторюваних символів підряд.
- maxclassrepeat=4. Не більше 4 однакових символів з однієї категорії підряд.
- enforce_for_root. Застосування правил також для користувача root.
- remember=5. Запам'ятовує останні 5 паролів, щоб уникнути їх повторного використання.
- use_authtok. Використання введеного пароля з попередніх модулів PAM.
- shadow. Використання shadow-utils для зберігання паролів.
- sha512. Використання алгоритму хешування SHA-512 для паролів.

Розробленою політикою паролів передбачено встановлення примусової зміни паролів. Для її налаштування виконаємо зміну конфігураційного файлу /etc/login.defs (рис.3.35).

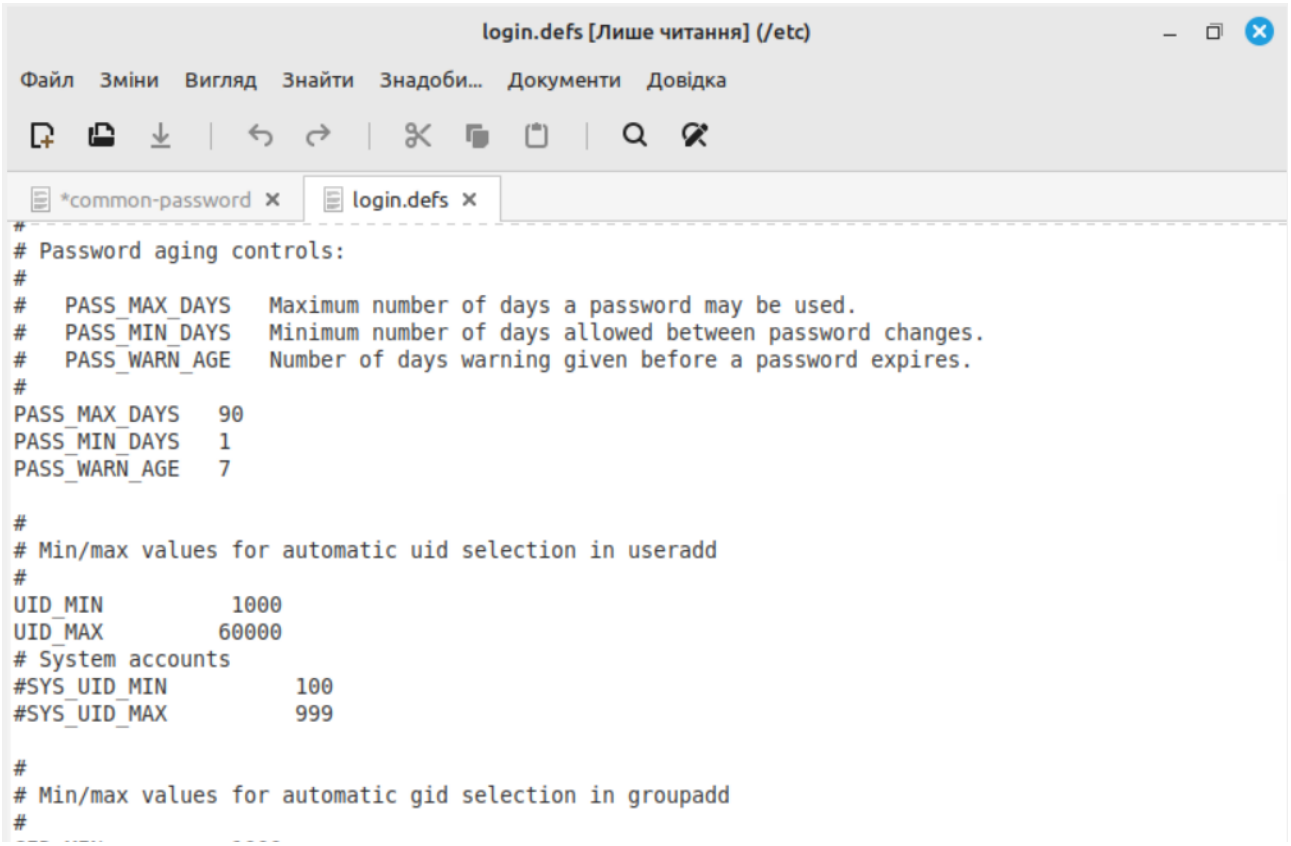


Рисунок 3.35 – Налаштування примусової зміни паролів

Встановлення примусової зміни виконано зміною наступних параметрів:

- PASS_MAX_DAYS 90 - максимальна кількість днів, на яку може бути встановлений пароль;
- PASS_MIN_DAYS 1 - мінімальна кількість днів між зміною паролів.
- PASS_WARN_AGE 7 - кількість днів, протягом яких буде виконано попередження користувача про закінчення терміну дії пароля.

Після конфігурування і внесення відповідних змін виконаємо перевірку роботи реалізованої політики паролів. Для цього виконаємо зміну паролю для поточного користувача і протестуємо його на відповідність встановленим вимогам за допомогою утиліти pamtester модуля PAM. Якщо модуль PAM встановлено правильно, коректно налаштований профіль безпеки AppArmor результатом має бути успішна зміна поточного паролю. Процес зміни паролю і результат зображено на рис.3.36.

Рисунок 3.36 – Зміна паролю для поточного користувача

Як бачимо, результат зміни успішний. Протестуємо встановлений пароль на відповідність реалізований політиці. Процес виконання і результат зображено на рис.3.37.

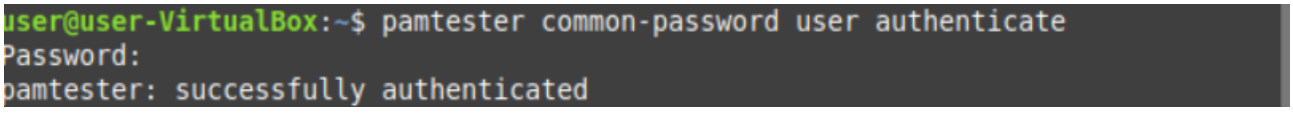


Рисунок 3.37 – Перевірка відповідності паролю встановленій політиці

Виходячи з результатів перевірки, розроблену політику паролів успішно реалізовано і впроваджено.

3.5 Збереження та розгортання конфігурацій захищеності ОС

Виконання необхідних попередні налаштування обраної операційної системи, реалізація і впровадження мандатного керування доступом, розширених

правил аудиту та політики аутентифікації разом складають систему захисту інформації яку можна використовувати для захисту автоматизованих систем від несанкціонованого доступу, що працюють на об’єктах інформаційної діяльності.

Конфігурування та впровадження частин описаних в попередніх пунктах потребує певного часу. Для його скорочення і зручності потрібно передбачити механізм зручного і розгортання на інших автоматизованих системах. Операційна система Linux Mint в повній мірі, як і будь-яка інша операційна система з відкритим вихідним кодом, може забезпечити реалізацію даного механізму шляхом резервного копіювання поточних конфігурації і їх відновлення.

Для виконання резервного копіювання системних конфігурацій та конфігурацій AppArmor, auditd та PAM необхідно визначити їх розміщення.

Розміщення системних конфігурацій:

- /etc/passwd;
- /etc/shadow;
- /etc/group;
- /etc/gshadow.

Профілі AppArmor розташовуються в /etc/apparmor.d/, файли конфігурації аудиту - в /etc/audit/, а файли конфігурації PAM - в /etc/pam.d/.

Виконаємо резервне копіювання системних конфігурацій (рис.3.38), в попередньо підготовлену директорію /home/user/backups.

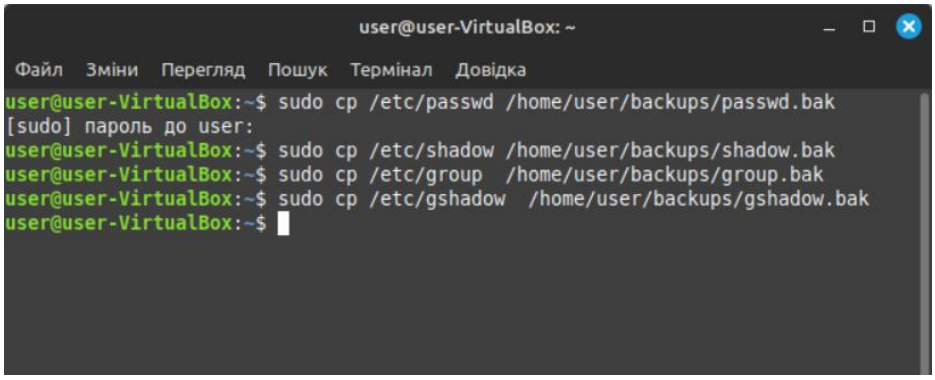


Рисунок 3.38 – Резервне копіювання системних конфігурацій

Так само виконаємо резервне копіювання директорії з якими працюють AppArmor, auditd та PAM. Загальний результат успішності резервного копіювання в визначену директорію зображена рис.3.39.

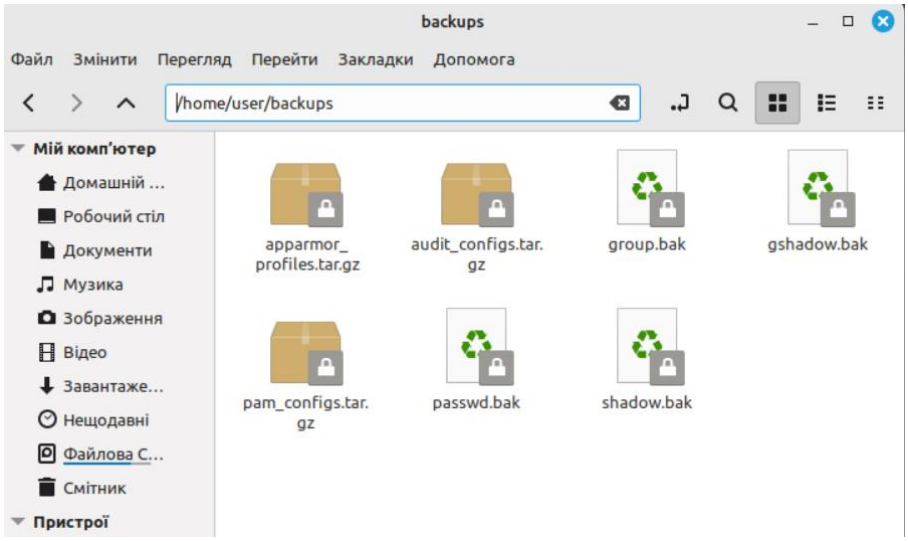


Рисунок 3.39 – Результат повного резервного копіювання

Створені резервні копії готові для розгортання на інших системах. Щоб автоматизувати процес розгортання конфігурацій на нових системах, можна створити скрипт, який буде виконувати всі необхідні дії. Лістинг скрипта розгортання конфігурації наступний:

```
#!/bin/deployment
# Відновлення користувачів і груп
sudo cp passwd.bak /etc/passwd
sudo cp shadow.bak /etc/shadow
sudo cp group.bak /etc/group
sudo cp gshadow.bak /etc/gshadow
# Відновлення профілів AppArmor
sudo tar xzvf apparmor_profiles.tar.gz -C /
# Відновлення конфігурацій аудиту
sudo tar xzvf audit_configs.tar.gz -C /
# Відновлення конфігурацій PAM
sudo tar xzvf pam_configs.tar.gz -C /
echo "Конфігурація успішно розгорнута."
```

За допомогою представленого механізму, можна легко зберегти поточну конфігурацію системи Linux Mint і розгортати її на інших таких же системах, що значно дозволить значно скоротити затрачений час та спростить процес впровадження.

3.6 Висновки

Третій розділ даної дипломної роботи присвячений реалізації компонентів системи захисту інформації, в тому числі профілів мандатного керування доступом на базі дистрибутиву Linux Mint 21.3 з робочим середовищем Cinnamon.

В результаті реалізовано низку налаштувань та конфігурацій для забезпечення безпеки системи, зокрема:

– виконано встановлення базових компонентів системи та створено облікові записи для різних груп користувачів з налаштуванням відповідних дозволів доступу;

– реалізовано профілі мандатного керування доступом (MAC) за допомогою програмного пакета AppArmor, що включає створення та налаштування профілів безпеки для критичних програм;

– впроваджено детальний аудит системи за допомогою налаштування сценаріїв журналювання подій;

– налаштовано політику паролів через встановлення та конфігурацію утиліт модуля PAM.

У даному розділі здійснено опис головних та унікальних частин налаштувань системи, що може бути використаний в якості інструкції для використання налаштованих компонентів безпеки або їх переконфігурації за потреби. Продемонстровано результат їх роботи.

Важливо, що реалізовані заходи забезпечують базовий рівень захисту інформації, а всі дії щодо налаштування безпеки спрямовані на збереження конфіденційності та цілісності даних під час їх обробки та зберігання.

Тестування реалізованих складових системи показав, що реалізовані заходи працюють коректно та ефективно.

ВИСНОВКИ

Дана бакалаврська дипломна робота присвячена розробці системи захисту інформації від несанкціонованого доступу з використанням мандатного керування доступом на основі операційної системи Linux Mint.

В ході роботи над першим розділом було визначено поняття системи захисту інформації, які складові вона включають. Встановлено, що основною частиною системи захисту інформації є технічні заходи спрямовані на перешкоджання несанкціонованому доступу до інформації. Основою технічних засобів є системи керування доступом, що реалізують різні моделі керування доступом. В ході аналізу існуючих моделей керування доступом, зокрема тих які використовуються в операційних системах з відкритим вихідним кодом, було встановлено, що використання систем які реалізують модель мандатного керування доступом дозволяють значно підвищити рівень захисту інформації.

Розглянуті в першому розділі існуючі системи захисту є потужними інструментами захисту інформації від несанкціонованого доступу, які пропонують обширний функціонал. Проте, вони не позбавлені недоліків. Детальний аналіз існуючих рішень висвітлює потребу у нових рішеннях, які будуть позбавлені цих недоліків та забезпечуватимуть належний рівень захищенності інформації.

В результаті проведеного аналізу загроз системи захисту інформації на базі операційної системи з відкритим вихідним кодом в другому розділі дипломної роботи було розроблено і представлено модель загроз, яка стала основою розроблюваної системи захисту інформації.

Було обґрунтовано вибір складових системи захисту інформації, включаючи дистрибутив ОС, середовище реалізації мандатного керування доступом, а також програми для політики розширеної аутентифікації та аудиту системи.

Особлива увага в другому розділі приділена розробці профілів мандатного керування доступом, що реалізуватиметься в обраному середовищі. Розробка профілів включала визначення основних груп користувачів, програм якими вони можуть користуватися та визначення критичних директорій операційної системи до яких ці програми матимуть або не матимуть.

В третьому розділі здійснено реалізацію основних складових, що складаються в систему захисту інформації на базі дистрибутиву операційної системи Linux Mint 21.3 з робочим середовищем Cinnamon. Визначені складові системи захисту реалізовувались відповідно до обраних середовищ та розроблених політик та правил.

Реалізація розроблених профілів мандатного керування доступом виконана за допомогою модуля AppArmor, який дозволяє керувати доступом програмам, що використовуються користувачами. Реалізовані профілі дозволяють детально контролювати дії визначених програм відповідно ролям, що використовуються у системі. Впровадження профілів мандатного керування доступом дасть змогу підвисити надійність системи, забезпечуючи високий рівень безпеки для користувачів та їх даних.

За допомогою модуля PAM виконано конфігурування розробленої розширеної, більш жорсткої політики аутентифікації, ніж та яка встановлена за замовчуванням і яка її замінятиме.

Реалізована конфігурація правил системи журналювання та моніторингу дій користувачів за допомогою модуля auditd дозволяє ефективно відстежувати аномальну активність та запобігати несанкціонованому доступу до даних.

Проведене тестування на кожному етапі реалізації складових системи показало, що розроблені рішення забезпечують необхідний рівень захисту та працюють коректно, як окремо так і в сукупності.

Щоб реалізовані компоненти функціонували як єдина система захисту на інших автоматизованих системах, можливостями ОС Linux Mint передбачено механізм, за яким можна легко зберегти поточну конфігурацію системи, а за допомогою реалізованого скрипта розгортати її на інших таких же системах, що значно дозволить значно скоротити затрачений час та спростить процес впровадження, адже в такому випадку для роботи з даною системою достатньо мати базові навички адміністрування Linux та розуміння принципів роботи систем захисту інформації.

Таким чином, розроблена конфігурація, що складається в систему захисту інформації на базі операційної системи Linux Mint, в якій реалізовано та впроваджено мандатне керування доступом, відповідає вимогам щодо забезпечення конфіденційності та цілісності даних, що підтверджено результатами тестування та аналізу. Дана система захисту може бути доцільною для організацій, які мають специфічні вимоги до безпеки та обмежені ресурси для налаштування і підтримки. Розроблена система створює гнучке та адаптоване рішення, яке забезпечує високий рівень безпеки, одночасно знижуючи витрати на ліцензії і отримуючи повний контроль над системою.

ПЕРЕЛІК ПОСИЛАНЬ

1. Антонюк А. О. Основи захисту інформації в автоматизованих системах : Навч. посіб. Київ : КМ Акад., 2003. 244 с.

2. Остапов С. Е. Технології захисту інформації : навчальний посібник / С. Е. Остапов, С. П. Євсеєв, О. Г. Король. – Х. : Вид. ХНЕУ, 2013. – 476 с.

3. Захист інформації в автоматизованих системах управління : навч. посібник/ Уклад. І.А. Пількевич, Н.М. Лобанчикова, К.В. Молодецька. – Житомир: Вид-во ЖДУ ім. І. Франка, 2015. – 226 с.

4. НД ТЗІ 2.5-005-99: Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної формації від несанкціонованого доступу. Затверджено наказом ДСТСЗІ СБУ № 22 від 28.04.1999. ДСТСЗІ СБУ. – К., 1999. – 34 с.

5. Основи інформаційної безпеки / С. В. Кавун, О. А. Смірнов, В. Ф. Столбов – Кіровоград : Вид. КНТУ, 2012. – 414 с.

6. Жилін А. В. Технології захисту інформації в інформаційно-телекомунікаційних системах : навч. посіб. / А. В. Жилін, О. М. Шаповал, О. А. Успенський ; ІСЗЗІ КПІ ім. Ігоря Сікорського. – Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2021. – 213 с.

7. Про захист персональних даних : Закон України від 01.06.2010 р. № 2297-VI : станом на 27 квіт. 2024 р. URL: https://zakon.rada.gov.ua/laws/show/2297-17#Text.

8. Про захист інформації в інформаційно-телекомунікаційних системах : Закон України від 05.07.1994 р. № 80/94-ВР : станом на 4 квіт. 2024 р. URL: https://zakon.rada.gov.ua/laws/show/80/94-вр#Text

9. Кримінальний кодекс України : Кодекс України від 05.04.2001 р. № 2341-III : станом на 19 трав. 2024 р. URL: https://zakon.rada.gov.ua/laws/show/2341-14#Text

10. НД ТЗІ 2.5-005-99: Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу. Затверджено наказом ДСТСЗІ СБУ № 22 від 28.04.1999. ДСТСЗІ СБУ. – К., 1999. – 21 с.

11. Питання забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах : Постанова Каб. Міністрів України від 08.02.2021 р. № 92. URL: https://zakon.rada.gov.ua/laws/show/92-2021-п#Text

12. НД ТЗІ 2.5-008-2002. Вимоги із захисту конфіденційної інформації від несанкціонованого доступу під час обробки в автоматизованих системах класу 2. Затверджено наказом ДСТСЗІ СБУ від 13.12.2002 № 84 (Зміна № 1 наказ Адміністрації Держспецзв'язку від 28.12.2012 № 806). – К., 2002. – 35 с.

13. Degtyareva L., Miroshnykova M., Voloshko S. АНАЛІЗ СТРУКТУРИ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ. Системи управління, навігації та зв'язку. Збірник наукових праць. 2019. Т. 2, № 54. С. 78–82. URL: https://doi.org/10.26906/sunz.2019.2.078

14. Єгоров Ф. І., Тискіна Є. О., Хорошко В. О. Задачі захисту інформації. Ukrainian Information Security Research Journal. 2009. Т. 11, № 1(42). URL: https://doi.org/10.18372/2410-7840.11.5368

15. Комплексні системи захисту інформації : навчальний посібник / [Яремчук Ю. Є., Павловський П. В., Катаєв В. С., Сінюгін В. В.] – Вінниця : ВНТУ, 2018. – 118 с.

16. Mukkamala P. P., Rajendran S. A SURVEY ON THE DIFFERENT FIREWALL TECHNOLOGIES. International Journal of Engineering Applied

Sciences and Technology. 2020. Vol. 5, no. 1. P. 363–365. URL: https://doi.org/10.33564/ijeast.2020.v05i01.059

17. Mahboub S. A., Sayed Ali Ahmed E., Saeed R. A. Smart IDS and IPS for Cyber-Physical Systems. Advances in Systems Analysis, Software Engineering, and High Performance Computing. 2021. P. 109–136. URL: https://doi.org/10.4018/978-1-7998-5101-1.ch006

18. Краліна Г., Баков Н. ШИФРУВАННЯ: ТИПИ ТА АЛГОРИТМИ. SPECIALIZED AND MULTIDISCIPLINARY SCIENTIFIC RESEARCHES. 2020. URL: https://doi.org/10.36074/11.12.2020.v2.36

19. Що таке СКУД і як це працює, різновиди систем контролю доступу. idcard.com.ua. URL: https://idcard.com.ua/ua/blog/chto-takoe-skud-i-kak-eto-rabotaet/

20. Якименко Н. М. Важливість організаційних заходів захисту інформації: thesis. 2014. URL: http://dspace.kntu.kr.ua/jspui/handle/123456789/2983

21. Горячий Н.В. Методи і засоби підвищення надійності та захищеності комп'ютерних мереж: автореферат дипломної роботи магістра за спеціальністю «123» – комп'ютерна інженерія/ Н.В. Горячий – Тернопільський національний технічний університет імені Івана Пулюя – Тернопіль, ТНТУ, 2018. – 9 с.

22. Discretionary-Access Control and the Access-Matrix Model. Access Control Systems. Boston. P. 147–167. URL: https://doi.org/10.1007/0-387-27716-1_5

23. De Capitani di Vimercati S., Samarati P. Mandatory Access Control (MAC) Policies. Encyclopedia of Cryptography, Security and Privacy. Berlin, Heidelberg, 2021. P. 1–2. URL: https://doi.org/10.1007/978-3-642-27739-9_822-2

24. Osborn S. L. Role-Based Access Control. Security, Privacy, and Trust in Modern Data Management. Berlin, Heidelberg, 2007. P. 55–70. URL: https://doi.org/10.1007/978-3-540-69861-6_5

25. Hu V. C., Kuhn D. R., Ferraiolo D. F. Attribute-Based Access Control. Computer. 2015. Vol. 48, no. 2. P. 85–88. URL: https://doi.org/10.1109/mc.2015.33

26. Panende M. F. Comparison of Attribute Based Access Control (ABAC) Model and Rule Based Access (RBAC) to Digital Evidence Storage (DES).

International Journal of Cyber-Security and Digital Forensics. 2018. Vol. 7, no. 3. P. 275–282. URL: https://doi.org/10.17781/p002451

27. Karale A., Poulkov V., Lazarova M. Attribute-Based Encryption. Functional Encryption. Cham, 2021. P. 225–242. URL: https://doi.org/10.1007/978-3-030-60890-3_13

28. Securing the Operating System A Software Based Approach in Linux / P. av S et al. International Journal of Computer Trends and Technology. 2017. Vol. 50, no. 3. P. 133–136. URL: https://doi.org/10.14445/22312803/ijctt-v50p123

29. Засоби ТЗІ, які мають експертний висновок про відповідність до вимог технічного захисту інформації. Державна служба спеціального зв'язку та захисту інформації України. URL: https://cip.gov.ua/ua/news/zasobi-tzi-yaki-mayut-ekspertnii-visnovok-pro-vidpovidnist-do-vimog-tekhnichnogo-zakhistu-informaciyi.

30. Комплекс засобів захисту операційної системи Microsoft Windows 10 Professional - Інформаційна безпека та захист інформації. ТЗІ - інформаційна безпека та захист інформації - Інформаційна безпека та захист інформації. URL: https://tzi.ua/ua/kompleks_zasobv_zahistu_operacjno_sistemi_microsoft_windows_10_professional.html

31. Secured*Pack (SSSCIP). UALinux - технології майбутнього вже зараз. URL: https://ualinux.com/en/uapack

32. СІЧ - Захищена операційна система створена на базі. SiCh. URL: https://sich.help/

33. Інститут комп'ютерних технологій / Захист інформації від НСД в АС класу 1. Інститут комп'ютерних технологій / Новини. URL: https://www.ict.com.ua/?lng=1&sec=8&art=41

34. Система захисту інформації ЛОЗА™-1, версія 4 | ТОВ НДІ "АВТОПРОМ". ТОВ НДІ "АВТОПРОМ" | засоби захисту інформації від несанкціонованого доступу. URL: http://avtoprom.kiev.ua/avtoprom/ua/content/Система-захисту-інформації-ЛОЗА™-1-версія-4

35. Gideon P. Windows 10 Pro User Guide: Beginner's Guide to Master the New Windows 10 Pro. Independently Published, 2021.

36. Гапон А. О., Федорченко В. М., Поляков А. О. Підходи до побудови моделі загроз для аналізу безпеки відкритого програмного кода. Системи обробки інформації. 2020. № 1(160),. С. 128–135.

37. Sulaiman, N. S., & Ahmad Shafiq Hilmi Ahmad Raffi. (2021). Comparison of Operating System Performance Between Windows 10 and Linux Mint . International Journal of Synergy in Engineering and Technology, 2(1), 92-102.

38. Brodschelm L., Gelderie M. Using ILP to Learn AppArmor Policies. 10th International Conference on Information Systems Security and Privacy, Rome, Italy, 26–28 February 2024. 2024

39. Vermeulen S. Selinux Policy Administration. Packt Publishing, Limited, 2013.

40. Bertin P., Touati H. PAM programming environments: practice and experience. IEEE Workshop on FPGA's for Custom Computing Machines, Napa Valley, CA, USA. URL: https://doi.org/10.1109/fpga.1994.315599 (date of access: 09.06.2024).

41. Threat Detection and Response in Linux Endpoints / S. Agarwal et al. 2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS), Bangalore, India, 4–8 January 2022. 2022. URL: https://doi.org/10.1109/comsnets53615.2022.9668567

42. Mint C. Linux Beginners Guide: A Comprehensive and Updated Guide for Beginners to Learn Linux Operating System, Easy Installation and Configuration Including Tips and Essentials Principles. Independently Published, 2019.

43. Shotts W. E. The Linux command line. San Francisco : No Starch Press, 2011.

44. van Vugt S. Installing Linux. Beginning the Linux Command Line. Berkeley, CA, 2015. P. 361–382. URL: https://doi.org/10.1007/978-1-4302-6829-1_15

45. Linux.org. Linux.org. URL: https://linux.org/

46. Negus C. Linux Bible. Wiley & Sons, Limited, John, 2019. 950 p.

47. Home | Linux Journal. Home | Linux Journal. URL: https://www.linuxjournal.com/

48. Home - Linux Mint. Home - Linux Mint. URL: https://linuxmint.com/ (date of access: 09.06.2024).

ДОДАТКИ

Додаток А. Технічне завдання

Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

ЗАТВЕРДЖУЮ

Голова секції "Управління інформаційною безпекою" кафедри МБІС

д.т.н., професор

_____Ю.Є. Яремчук

«22» березня 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

до бакалаврської дипломної роботи на тему:

«Розробка системи захисту інформації від несанкціонованого доступу з використанням мандатного керування на основі ОС Linux Mint»

08-72.БДР.009.00.000.ТЗ

Керівник бакалаврської дипломної роботи

д.ф., доцент каф. МБІС

_____Салієва О.В.

«_____»_____

Вінниця – 2024р.

Область застосування: обробка та зберігання інформації в автоматизованих системах.

2.Підстава для розробки

Розробка виконується на основі наказу ректора ВНТУ № 80 від 11.03.2024 р.

3. Мета та призначення розробки

3.1 Мета розробки: розроблення системи захисту інформації на базі операційної системи Linux Mint в якій реалізоване мандатне керування доступом.

3.2 Призначення: розроблена система захисту інформації представляє собою гнучке та адаптоване рішення, яке забезпечує високий рівень безпеки, одночасно знижуючи витрати на ліцензії і отримуючи повний контроль над системою.

4. Джерела розробки

4.1. Остапов С. Е. Технології захисту інформації : навчальний посібник / С. Е. Остапов, С. П. Євсеєв, О. Г. Король. – Х. : Вид. ХНЕУ, 2013. – 476 с.

4.2. Degtyareva L., Miroshnykova M., Voloshko S. АНАЛІЗ СТРУКТУРИ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ. Системи управління, навігації та зв’язку. Збірник наукових праць. 2019. Т. 2, № 54. С. 78–82.

4.3. Securing the Operating System A Software Based Approach in Linux / P. av S et al. International Journal of Computer Trends and Technology. 2017. Vol. 50, no. 3. P. 133–136.

4.4. Brodschelm L., Gelderie M. Using ILP to Learn AppArmor Policies. 10th International Conference on Information Systems Security and Privacy, Rome, Italy, 26–28 February 2024. 2024

4.5. Mint C. Linux Beginners Guide: A Comprehensive and Updated Guide for Beginners to Learn Linux Operating System, Easy Installation and Configuration Including Tips and Essentials Principles. Independently Published, 2019.

5. Вимоги до розробки

5.1 Вимоги до функціональних характеристик:

5.1.1 Розроблюваний засіб повинен мати зручний, легкий та зрозумілий у використанні інтерфейс користувача;

5.1.2 Реалізація моделі керування доступом не повинна вимагати спеціальних ліцензійних програмних додатків.

5.2 Вимоги до надійності:

5.2.1 Система повинна працювати без помилок, у випадку виникнення критичних ситуацій необхідно передбачити виведення відповідних повідомлень;

5.2.2 Програмна частина повинна виконувати свої функції.

5.3 Вимоги до складу і параметрів технічних засобів:

– процесор – x86_64 (64-бітний) з підтримкою PAE, NX, і SSE2;

– оперативна пам’ять – мінімум 2 ГБ;

– місце на жорсткому диску - мінімум 20 ГБ;

– графічна карта – будь-яка здатна до роздільної здатності 1024x768;

– середовище функціонування – операційна система Linux Mint;

– вимоги до техніки безпеки при роботі з системою повинні відповідати існуючим вимогам та стандартам з техніки безпеки при користуванні комп’ютерною технікою.

6. Вимоги до технічного захисту інформації

6.1 Неможливість отримання доступу сторонніх користувачів до відкритих даних.

6.2. Обробка даних без можливості перегляду їх вмісту третіми сторонами.

7. Техніко-економічні показники

7.1 Цінність результатів використання даного проекту повинна перевищувати витрати на його реалізацію.

7.2 Має бути реалізований таким чином, щоб підходити для використання широкого загалу.

8. Стадії та етапи розробки

7. Техніко-економічні показники

7.1 Цінність результатів використання даного проекту повинна перевищувати витрати на його реалізацію.

7.2 Має бути реалізований таким чином, щоб підходити для використання широкого загалу.

8. Стадії та етапи розробки

№ з/п	Назва етапів бакалаврської дипломної роботи	Початок	Закінчення
1.	Визначення напрямку бакалаврської роботи, формулювання теми	23.03.2024	30.03.2024
2.	Аналіз предметної області обраної теми	1.04.2024	11.04.2024
3.	Розробка	14.04.2024	04.05.2024
4.	Написання бакалаврської роботи на основі розробленої теми	08.05.2024	27.05.2024
5.	Передзахист бакалаврської дипломної роботи	30.05.2024	10.06.2024
6.	Виправлення, уточнення, корегування бакалаврської дипломної роботи	12.06.2024	16.06.2024
7.	Захист бакалаврської дипломної роботи	17.06.2024	18.06.2024

10. Порядок контролю та прийому

10.1 До приймання бакалаврської дипломної роботи надається:

– ПЗ до бакалаврської дипломної роботи;
– демонстрація результату бакалаврської дипломної роботи;
– презентація;
– відгук керівника роботи;
– відгук рецензента.

Технічне завдання до виконання прийняв _____ Дячук О.О.

```
else
    # Повідомлення про відмову у доступі
    echo "Access denied. You must be in the security_admin or sys_admin group to run this program."
    exit 1
fi
```

Додаток В. Лістинг конфігурації профілів AppArmor програм безпеки

```
#include <tunables/global>
/usr/sbin/apparmor {
  # Виконувані файли
  /usr/sbin/apparmor_parser ix,
  /usr/sbin/apparmor_status ix,
  # Читання конфігураційних файлів
  /etc/apparmor.d/** r,
  # Доступ до лог-файлів
  /var/log/apparmor/** rw,
  # Доступ до тимчасових файлів
  /tmp/** rw,
  /var/tmp/** rw,

  # Доступ до утиліт
  /usr/sbin/** ix,
  /sbin/** ix,
  /usr/bin/** ix,
  /bin/** ix,
  # Доступ до системних ресурсів
  capability sys_admin,
  capability audit_control,
  capability dac_read_search,
  # Доступ до мережевих інтерфейсів
  network inet stream,
  network inet6 stream,
  # Обмеження доступу до конфіденційних файлів
  deny /etc/shadow r,
  deny /etc/passwd r,
}

#include <tunables/global>
/usr/lib/security/pam {
  # Читання конфігураційних файлів PAM
  /etc/pam.d/** r,
  /etc/security/** r,
  # Доступ до необхідних бібліотек
  /lib/x86_64-linux-gnu/** r,
  /usr/lib/x86_64-linux-gnu/** r,
  # Доступ до системних утиліт
  /bin/** ix,
  /usr/bin/** ix,
  /sbin/** ix,
  /usr/sbin/** ix,
  # Читання лог-файлів
  /var/log/auth.log r,
  /var/log/secure r,
  /var/log/faillog rw,
```

```
    /var/log/btmp rw,
    # Доступ до тимчасових файлів
    /tmp/** rw,
    /var/tmp/** rw,
    # Виконання необхідних команд
    capability dac_override,
    capability sys_admin,
    capability net_admin,
    capability chown,
    capability setuid,
    capability setgid,
    # Запобігання доступу до конфіденційних файлів
    deny /etc/shadow r,
    deny /etc/passwd r,
    # Дозволити запис у /var/log/audit
    /var/log/audit/** rw,
    # Доступ до домашніх директорій
    /home/*/.pam_environment r,
     # Запуск профілю на read-only файлових системах
    mount options=(ro),
    # Дозвіл на створення та видалення власних тимчасових файлів
    /run/user/*/pam-*.tmp rwk,
    # Запобігання доступу до інших користувачів
    deny /home/*/ r,
    # Підтримка зовнішніх модулів автентифікації
    /etc/pam_ldap.conf r,
    /etc/nsswitch.conf r,
    # Доступ до мережевих інтерфейсів для автентифікації
    network inet stream,
    network inet6 stream,
    # Додаткові права доступу (якщо необхідно)
    # /etc/ldap.conf r,
    # /etc/krb5.conf r,
}

#include <tunables/global>

/usr/sbin/auditd {
    # Виконувані файли
    /usr/sbin/auditd ix,
    /usr/libexec/** ix,
    # Читання конфігураційних файлів
    /etc/audit/** r,
    /etc/auditd.conf r,
    # Доступ до лог-файлів
    /var/log/audit/** rw,

    # Доступ до тимчасових файлів
    /tmp/** rw,
```

```
    /var/tmp/** rw,
    # Доступ до системних файлів та утиліт
    /usr/sbin/** ix,
    /sbin/** ix,
    /usr/bin/** ix,
    /bin/** ix,
    # Доступ до сокетів
    /run/auditd.socket rw,
    # Доступ до мережевих інтерфейсів для аудиту
    network inet stream,
    network inet6 stream,
    # Доступ до системних ресурсів
    capability sys_admin,
    capability audit_control,
    capability dac_read_search,
    # Обмеження доступу до конфіденційних файлів
    deny /etc/shadow r,
    deny /etc/passwd r,
}

#include <tunables/global>

/usr/bin/gufw {
    # Виконувані файли
    /usr/bin/gufw ix,
    # Читання конфігураційних файлів
    /etc/gufw/** r,
    # Доступ до утиліт
    /sbin/iptables ix,
    /usr/sbin/iptables ix,
    /sbin/ip6tables ix,
    /usr/sbin/ip6tables ix,
    # Доступ до лог-файлів
    /var/log/gufw/** rw,
    # Доступ до тимчасових файлів
    /tmp/** rw,
    /var/tmp/** rw,
    # Доступ до графічних ресурсів
    /usr/share/icons/** r,
    /usr/share/gufw/** r,
    # Доступ до системних ресурсів
    capability net_admin,
    # Обмеження доступу до конфіденційних файлів
    deny /etc/shadow r,
    deny /etc/passwd r,
}
```

Додаток Г. Лістинг конфігурації профілів AppArmor програм системного адміністрування

```
#include <tunables/global>
profile /usr/bin/cinnamon-settings {
  # Include the standard policy
  #include <abstractions/base>
  # Доступ до конфігураційних директорій
  /etc/cinnamon/** r,
  /usr/share/cinnamon/** r,
  /usr/lib/cinnamon-settings/** r,
  # Виконання бінарних файлів
  /usr/bin/cinnamon-settings Pix,
  # Доступ до налаштувань користувача
  owner @{HOME}/.config/cinnamon/** rw,
  # Обмеження доступу до критичних системних файлів
  /etc/shadow r,
  /etc/passwd r,
    # Доступ до системного D-Bus
  /var/run/dbus/system_bus_socket rw,
    # Доступ до необхідних файлів для налаштувань системи
  /usr/bin/gsettings rix,
  /usr/bin/python3 rix,
  /usr/lib/python3.8/** r,
  # Заборона всього іншого
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/cinnamon-settings {
  # Include the standard policy
  #include <abstractions/base>
  # Allow all read access to the configuration directories
  /etc/cinnamon/** r,
  /usr/share/cinnamon/** r,
  /usr/lib/cinnamon-settings/** r,
    # Allow executing binaries in the path
  /usr/bin/cinnamon-settings Pix,
  # Allow read and write access to user-specific settings
  owner @{HOME}/.config/cinnamon/** rw,
  # Deny access to shadow and password files
  /etc/shadow r,
  /etc/passwd r,
   # Allow access to system D-Bus
  /var/run/dbus/system_bus_socket rw,
   # Allow access to necessary files for general system settings
  /usr/bin/gsettings rix,
  /usr/bin/python3 rix,
```

```
  /usr/lib/python3.8/** r,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/mintupdate {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/mintupdate Pix,
  /usr/bin/apt-get Pix,
  /usr/bin/dpkg Pix,
  # Allow read access to the update manager configuration
  /etc/linuxmint/mintupdate.conf r,
  # Allow read and write access to apt directories and files
  /var/lib/apt/lists/** rw,
  /var/cache/apt/archives/** rw,
  /var/log/apt/** rw,
    # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>
profile /usr/bin/mintdrivers {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/mintdrivers Pix,
  /usr/bin/apt-get Pix,
  /usr/bin/dpkg Pix,
  # Allow read access to the driver manager configuration
  /etc/linuxmint/mintdrivers.conf r,
  # Allow read and write access to apt directories and files
  /var/lib/apt/lists/** rw,
  /var/cache/apt/archives/** rw,
  /var/log/apt/** rw,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}


#include <tunables/global>
```

```
profile /usr/bin/mintinstall {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/mintinstall Pix,
  /usr/bin/apt-get Pix,
  /usr/bin/dpkg Pix,
  # Allow read access to the software manager configuration
  /etc/linuxmint/mintinstall.conf r,
  # Allow read and write access to apt directories and files
  /var/lib/apt/lists/** rw,
  /var/cache/apt/archives/** rw,
  /var/log/apt/** rw,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>
profile /usr/sbin/synaptic {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/sbin/synaptic Pix,
  /usr/bin/apt-get Pix,
  /usr/bin/dpkg Pix,
  # Allow read and write access to apt directories and files
  /var/lib/apt/lists/** rw,
  /var/cache/apt/archives/** rw,
  /var/log/apt/** rw,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/timeshift {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/timeshift Pix,
  # Allow read and write access to the Timeshift configuration and snapshot directories
  /etc/timeshift/** rw,
  /var/log/timeshift/** rw,
```

```
  /timeshift/** rw,
    # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/gnome-system-monitor {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/gnome-system-monitor Pix,
  # Allow access to proc filesystem for monitoring system resources
  /proc/** r,
  /sys/** r,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/baobab {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/baobab Pix,
  # Allow access to user directories for analyzing disk usage
  @{HOME}/** r,
  /media/** r,
  /mnt/** r,
  /srv/** r,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/gnome-disks {
  # Include the standard policy
  #include <abstractions/base>

  # Allow read and execute access to the necessary binaries
```

```
  /usr/bin/gnome-disks Pix,
  # Allow access to system directories for disk management
  /dev/** rw,
  /sys/** r,
  /proc/** r,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}

#include <tunables/global>

profile /usr/bin/system-config-printer {
  # Include the standard policy
  #include <abstractions/base>
  # Allow read and execute access to the necessary binaries
  /usr/bin/system-config-printer Pix,
  # Allow access to printer configuration files and directories
  /etc/cups/** r,
  /var/spool/cups/** rw,
  /usr/share/cups/** r,
  # Allow access to D-Bus for system notifications
  /var/run/dbus/system_bus_socket rw,
  # Deny everything else
  deny / rw,
}
```

Додаток Д. Лістинг конфігурації профілів AppArmor програм загального користування

```
#include <tunables/global>

/usr/lib/libreoffice/program/soffice.bin {
  # Включення стандартних політик
  #include <abstractions/base>
  # Виконання бінарного файлу
  /usr/lib/libreoffice/program/soffice.bin Pix,
  # Доступ до конфігураційних файлів
  /etc/libreoffice/** r,
  # Доступ до налаштувань і документів користувача
  owner @{HOME}/.config/libreoffice/** rw,
  owner @{HOME}/Documents/** rw,
  # Доступ до тимчасових файлів
  /tmp/** rw,
  # Доступ до шрифтів
  /usr/share/fonts/** r,
  /usr/local/share/fonts/** r,
  # Мережевий доступ
  network inet,
  # Доступ до принтерів
  /var/spool/cups/** rw,
  /usr/lib/cups/backend/** x,
  # Обмеження доступу до чутливих файлі
  deny /etc/shadow r,
  deny /etc/gshadow r,
  deny /etc/sudoers r,
  # Default deny all other access
  deny / rw,
}

#include <tunables/global>

/usr/lib/firefox/firefox {
  # Include the standard policy
  #include <abstractions/base>
  # Allow execution of the binary
  /usr/lib/firefox/firefox Pix,
  # Allow read access to system-wide configuration files
  /etc/firefox/** r,
  # Allow read and write access to user-specific settings and data
  owner @{HOME}/.mozilla/** rw,
  owner @{HOME}/Downloads/** rw,
  owner @{HOME}/Pictures/** rw,
  owner @{HOME}/Documents/** rw,
```

```
  # Allow access to temporary files
  /tmp/** rw,
  # Allow access to fonts
  /usr/share/fonts/** r,
  /usr/local/share/fonts/** r,
  # Allow network access
  network inet,
  # Allow access to sound devices
  /dev/snd/** rw,
  # Allow access to video devices
  /dev/video0 rw,
  # Allow access to shared libraries
  /usr/lib/** mr,
  /usr/lib/firefox/** mr,
  # Deny access to shadow and other sensitive files
  deny /etc/shadow r,
  deny /etc/gshadow r,
  deny /etc/sudoers r,
  # Default deny all other access
  deny / rw,
}

#include <tunables/global>

/usr/bin/gedit {
  # Include the standard policy
  #include <abstractions/base>
  # Allow execution of the binary
  /usr/bin/gedit Pix,
  # Allow read access to system-wide configuration files
  /etc/gedit/** r,
  # Allow read and write access to user-specific settings and documents
  owner @{HOME}/.config/gedit/** rw,
  owner @{HOME}/Documents/** rw,
  # Allow access to temporary files
  /tmp/** rw,
  # Allow access to fonts
  /usr/share/fonts/** r,
  /usr/local/share/fonts/** r,
  # Allow access to shared libraries
  /usr/lib/** mr,
  # Deny access to shadow and other sensitive files
  deny /etc/shadow r,
  deny /etc/gshadow r,
  deny /etc/sudoers r,
  # Log denied access attempts
  audit deny,
  # Default deny all other access
```

```
  deny / rw,
}
```

Додаток Е. Лістинг скрипта розгортання системи

```
#!/bin/deployment
# Відновлення користувачів і груп
sudo cp passwd.bak /etc/passwd
sudo cp shadow.bak /etc/shadow
sudo cp group.bak /etc/group
sudo cp gshadow.bak /etc/gshadow
# Відновлення профілів AppArmor
sudo tar xzvf apparmor_profiles.tar.gz -C /
# Відновлення конфігурацій аудиту
sudo tar xzvf audit_configs.tar.gz -C /
# Відновлення конфігурацій PAM
sudo tar xzvf pam_configs.tar.gz -C /
echo "Конфігурація успішно розгорнута."
```

Додаток Є. Ілюстративний матеріал

Розробка системи захисту інформації від несанкціонованого доступу з використанням мандатного керування на основі ОС Linux Mint

КЕРІВНИК: Д.Ф. ДОЦ.КАФ.МБІС
САЛІЄВА О.В.
РОЗРОБИВ: СТ.ГР.КІТС-20б
ДЯЧУК ОЛЕКСАНДР

Актуальність і мета роботи

Актуальність:

- Інформатизація охоплює всі сфери діяльності, зростання значущості інформації є невід'ємною частиною розвитку технологій. Розголошення даних може призвести до збитків, тому захист інформації є актуальною проблемою.
- Для запобігання витоку використовуються комплекси захисту та операційні системи з вбудованими засобами безпеки, що реалізують політику безпеки.
- ОС важливі для обробки конфіденційної інформації, але стандартні налаштування часто не відповідають критеріям безпеки. Неправильні налаштування можуть призвести до порушень безпеки.
- Відкритий код дозволяє створювати спеціалізовані інструменти для захисту інформації без значних витрат, налаштовуючи їх для конкретних потреб у різних галузях.

Метою є розробка системи захисту інформації яка буде використовувати сучасні рішення, що надаються системами з відкритим вихідним кодом, яка буде доступною, гнучкою, налаштовуваною та багатофункціональною.

Задачі

- визначити загрози безпеки та на їх основі розробити модель загроз, яка ляже в основу розроблюваної системи;
- вибрати середовища реалізації складових системи захисту інформації;
- розробити та реалізувати конфігурацію обраної моделі керування доступом;
- розробити та реалізувати конфігурації елементів системи захисту;
- реалізувати механізм розгортання реалізованої конфігурації в нових системах.

Модель мандатного керування доступом (MAC)

Строга модель контролю доступу, в якій права доступу встановлюються системним адміністратором або політиком безпеки, а не власником ресурсу. У системах MAC кожен об'єкт та суб'єкт мають мітки безпеки, які визначають рівень дозволу або відмови в доступі. Приклади систем, які використовують MAC, включають SELinux та AppArmor

Порівняльна характеристика моделей керування доступом

Характеристика	DAC (Discretionary Access Control)	MAC (Mandatory Access Control)	RBAC (Role-Based Access Control)	ABAC (Attribute-Based Access Control)	Role-Based Access Control	ABE (Attribute-based Encryption)
Основний принцип	Контроль доступу на розсуд власника об'єкта	Контроль доступу на основі політик, встановлених системою	Контроль доступу на основі ролей користувачів	Контроль доступу на основі атрибутів користувача та об'єкта	Контроль доступу на основі правил	Шифрування даних на основі атрибутів
Гнучкість	Високо	Низько	Середньо	Високо	Високо	Високо
Безпека	Низька	Висока	Висока	Висока	Висока	Висока
Простота використання	Високо	Низько	Середньо	Низько	Середньо	Низько
Контроль адміністратором	Низький	Високий	Високий	Високий	Високий	Середній
Підходить для малих організацій	Середньо	Так	Так	Так	Так	Так
Підходить для великих організацій	Так	Ні	Можливо	Можливо	Можливо	Ні
Приклади використання	Типові файлові системи Linux	SELinux, AppArmor	Багато корпоративних систем	Складні інформаційні системи	Мережеві брандмауери	Захист даних у хмарних системах
Вимагають до знання користувача	Високо	Низько	Середньо	Низько	Середньо	Низько
Складність підтримки	Низька	Висока	Середньо	Висока	Середньо	Висока

Існуючі системи захисту інформації

ОС Windows 10 Professional

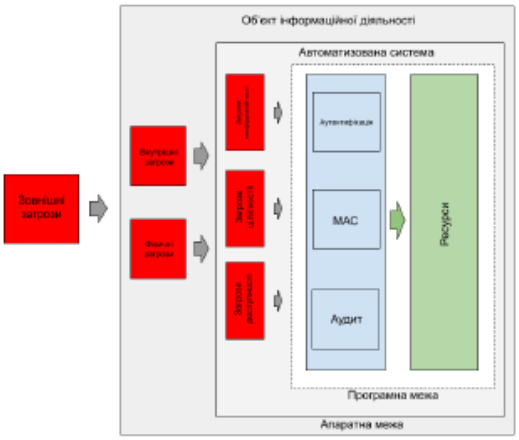
ОС Ubuntu*Pack

ОС «Сігма»

Порівняльна характеристика існуючих засобів

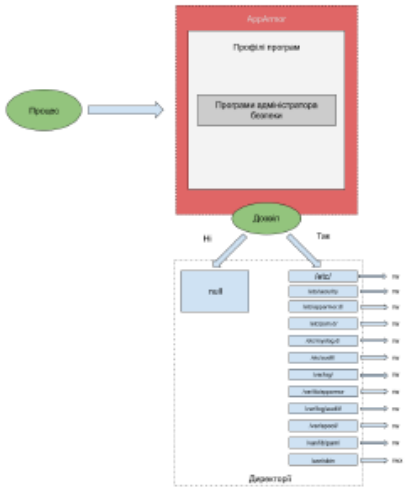
Критерій	Ubuntu/Pinck 20.04	ОС "Сін"	Microsoft Windows 10 Professional
Основний фокус	Безпека і конфіденційність	Безпека для державних установ	Комплексний захист для бізнесу
Модель керування доступом	MAC (SELinux або AppArmor)	MAC (SELinux)	DAC, RBAC, частково MAC (Device Guard, Credential Guard)
Шифрування	LUKS	LUKS	BitLocker
Антивірусний захист	Відкриті рішення, ClamAV	Відкриті рішення	Windows Defender Antivirus
Мережевий захист	Підтримка VPN і Tor	Підтримка VPN і Tor	Windows Firewall, VPN підтримка
Контроль доступу	Багаторівневий контроль	Багаторівневий контроль	Багаторівневий контроль, Credential Guard
Підтримка комерційного ПЗ	Обмежена	Обмежена	Широка підтримка
Сумісність обладнання	Можливі проблеми з деяким обладнанням	Можливі проблеми з деяким обладнанням	Широка підтримка обладнання
Ліцензійна вартість	Відкритий код, комерційний продукт	Відкритий код, комерційний продукт	Висока вартість ліцензій
Вимоги до ресурсів	Середні	Середні	Високі
Зручність використання	Складне налаштування	Складне налаштування	Інтуїтивний інтерфейс
Документація та підтримка	Обмежена підтримка	Обмежена підтримка	Розширена документація, технічна підтримка
Регулярність оновлень	Регулярні оновлення	Регулярні оновлення від розробників	Централізовані оновлення від Microsoft

Модель загроз

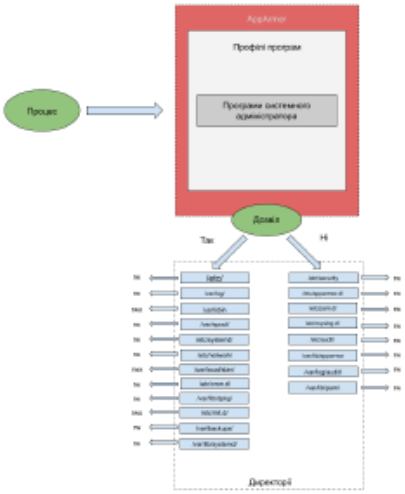


Алгоритм створення профілів мандатного керування AppArmor

Загальна структура профілю мандатного керування AppArmor



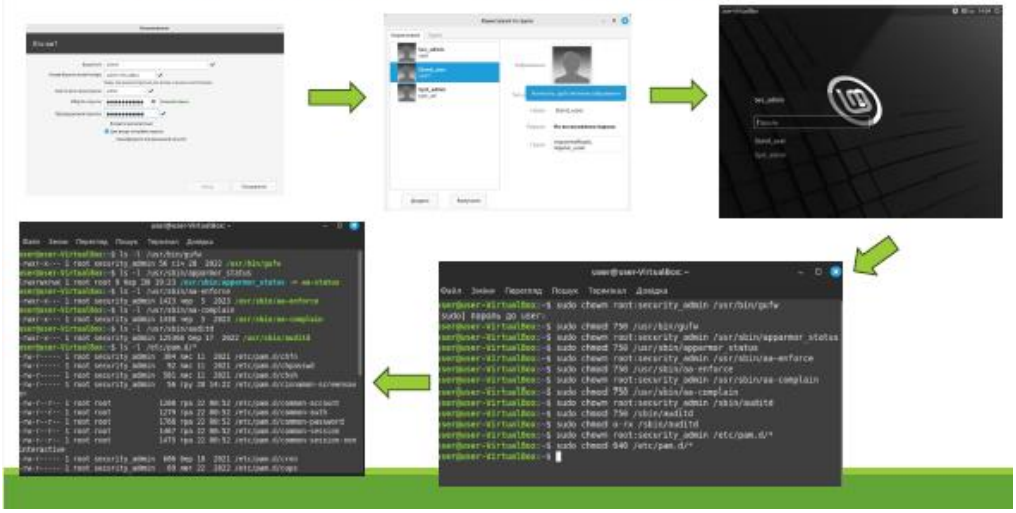
Базовий профіль для програм адміністратора безпеки



Базовий профіль для програм системного адміністратора

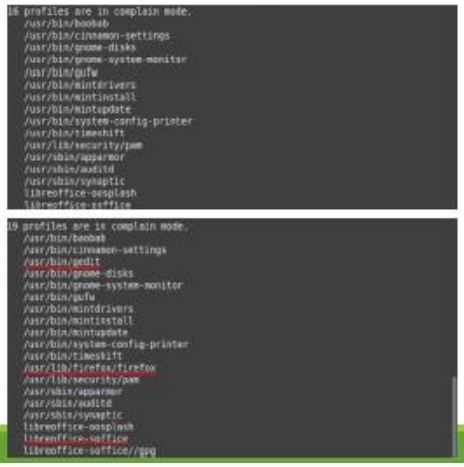
Базовий профіль для програм звичайного користувача

Встановлення та попередня конфігурація Linux Mint



Процес створення, редагування, завантаження і активування профілю AppArmor

Загальний статус активації профілів AppArmor для всіх визначених програм



Тестування профілів AppArmor

Створивши всі необхідні профілі для визначених програм протестуємо їх роботу. Тестування профілів AppArmor важливе для перевірки належного функціонування і перевірки забезпечення необхідного рівня безпеки без надмірних обмежень. Показником коректності налаштування правил профілів в нашому випадку є те, чи програми і відповідні їм служби перезапустяться та виконуватимуть команди після того як профіль буде активовано.

Перезапуск служби auditd:

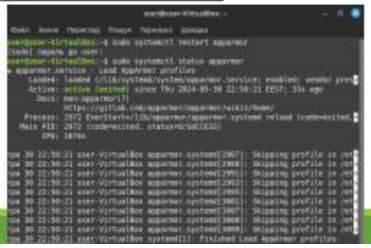
sudo systemctl restart auditd

sudo systemctl status auditd

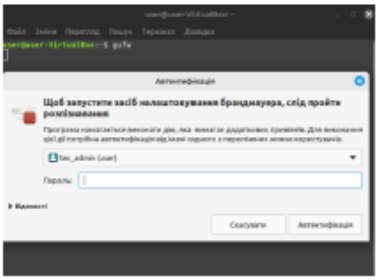
Перезапуск служби AppArmor:

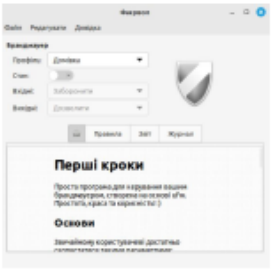
sudo systemctl restart apparmor

sudo systemctl status apparmor



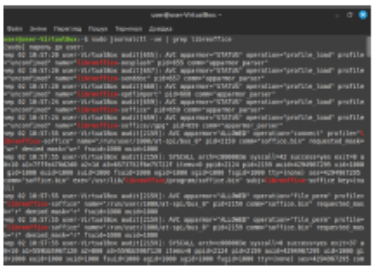
Тестування профілів AppArmor



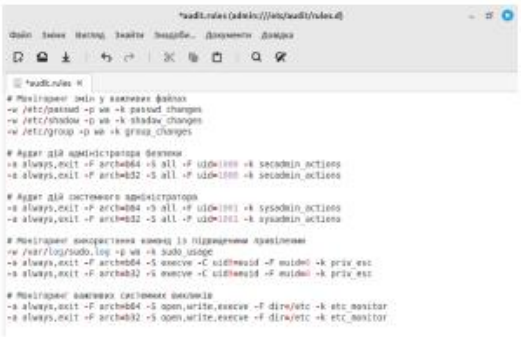


Тестування профілів AppArmor

Щоб перевірити роботу профілю потрібно виконати базові дії, які виконуватимуться при стандартній роботі з офісним пакетом, а саме запуск однієї з програм пакету, робота з ним, збереження файлу в робочу директорію та відкриття збереженого файлу. Збереження тестового файлу відбувалась в директорію Документи ОС Linux Mint, до якої, створеним профілем, передбачено доступ.

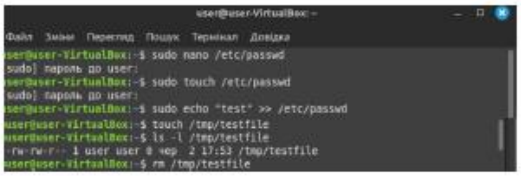


Конфігурація правил аудиту



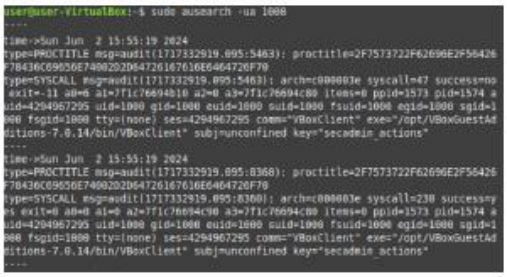
Тестування розроблених і реалізованих правил аудиту

Для перевірки роботи правил, виконаємо дії, які повинні створювати записи в журналах

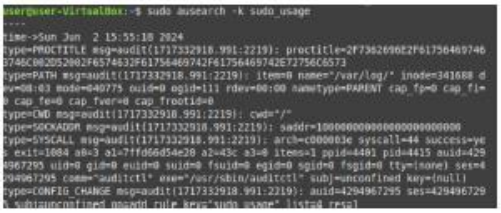


Тестування розроблених і реалізованих правил аудиту

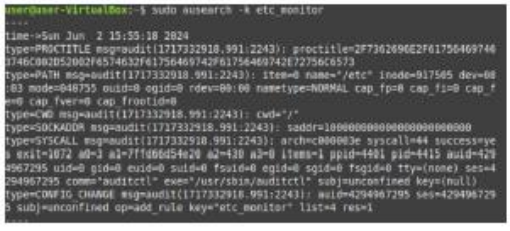
Після виконання команд тестових подій, відповідні записи повинні з'явитись в журналі подій. Здійснимо перевірку по трьом критеріям: перевірку дій адміністратора безпеки, перевірку виконання команди sudo та перевірку системних викликів



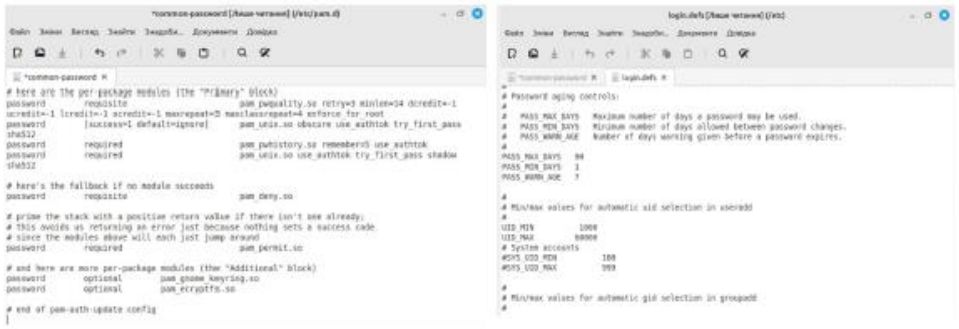
Тестування розроблених і реалізованих правил аудиту



Тестування розроблених і реалізованих правил аудиту



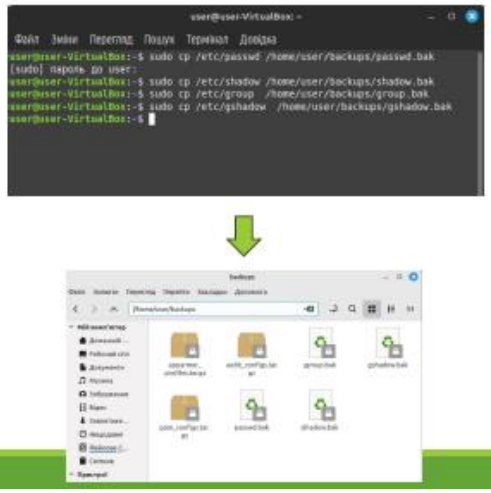
Конфігурація розробленої політики аутентифікації



Тестування реалізованої політики аутентифікації

Виконаємо зміну паролю для поточного користувача і протестуємо його на відповідність встановленим вимогам за допомогою утиліти pamtester модуля PAM. Якщо модуль PAM встановлено правильно, коректно налаштований профіль безпеки AppArmor результатом має бути успішна зміна поточного паролю.

Збереження реалізованої конфігурації



Висновки

- ✓ Реалізовано профілі мандатного керування доступом для детального контролю дій програм, підвищуючи надійність і безпеку системи.
- ✓ Конфігурація жорсткої політики аутентифікації замінює стандартну політику, забезпечуючи більшу безпеку.
- ✓ Налаштовані правила журналювання та моніторингу дій користувачів дозволяють ефективно відстежувати аномалії та запобігати несанкціонованому доступу.
- ✓ Тестування, показало, що рішення забезпечують необхідний рівень захисту та працюють коректно.
- ✓ Використовуючи можливості ОС Linux Mint, конфігурацію системи можна легко зберігати та розгортати на інших системах, спрощуючи процес впровадження.
- ✓ Система забезпечує конфіденційність та цілісність даних, що підтверджено результатами тестування.
- ✓ Система підходить для організацій зі специфічними вимогами до безпеки та обмеженими ресурсами, забезпечуючи високий рівень безпеки та знижуючи витрати на ліцензії.

Дякую за увагу!

Додаток Ж. Протокол перевірки на антиплагіат

ПРОТОКОЛ
ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ
ЗАПОЗИЧЕНЬ

Назва роботи: Розробка системи захисту інформації від несанкціонованого доступу з використанням мандатного керування на основі ОС Linux Mint

Тип роботи: бакалаврська дипломна робота
(БДР, МКР)

Підрозділ: Кафедра менеджменту та безпеки інформаційних систем
Факультет менеджменту та інформаційної безпеки
(кафедра, факультет)

Показники звіту подібності Unicheck

Оригінальність 95 % Схожість 5 %

Аналіз звіту подібності (відмітити потрібне):

1. Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

2. Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.

3. Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Коваль Н.П.
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи _____ Дячук О.О.
(підпис) (прізвище, ініціали)

Керівник роботи _____ Салієва О.В.
(підпис) (прізвище, ініціали)