DOI: https://doi.org/10.17721/ISTS.2019.1.42-51
УДК 004.056

ДО КОНЦЕПЦІЇ ЗАХИЩЕНОЇ ОПЕРАЦІЙНОЇ СИСТЕМИ

Бичков Олексій
orcid.org/0000-0002-9378-9535

м. Київ, Київський національний університет імені Тараса Шевченко, bos.knu@gmail.com

Історія статті:
Надійшла до редакції 02.10.2019
Прийнято 11.10.2019

Ключові слова:
захищеність;
операційна система;
вбудовані системи;
мікроконтролери;
інтернет речей

***Анотація**: На сучасному етапі використання інформаційних технологій у суспільстві, важливим стає питання захисту інформації. Основну роль в цьому відіграють операційні системи. На них покладається роль захисників усіх даних користувача та прав доступу до інформації.*

Перед авторами статті ставилась задача запропонувати класифікацію використання операційних систем та надати вимоги до механізмів захисту інформації за цією класифікацією. В статті автором:

- проаналізовано існуючі стандарти забезпечення безпеки, які реалізовано у сучасних операційних системах. Наведено існуючі стандарти забезпечення безпеки (Критерії оцінки комп'ютерних систем "Помаранчева книга», міжнародний стандарт – ISO 17799). У "Помаранчевій книзі" довірена система визначається як "система, що використовує достатні апаратні і програмні засоби, щоб забезпечити одночасну обробку інформації різного ступеня секретності групою користувачів без порушення прав доступу". Також розглянуто механізми безпеки і класи безпеки сучасних операційних систем і модель управління безпекою BS 7799. Цей стандарт містить систематичний, повний, універсальний перелік регулювальників безпеки, корисний для організації практично будь-якого розміру, структури і сфери діяльності системи управління інформаційною безпекою. В стандарті під системою управління інформаційною безпекою (СУІБ) (Information Security Management System, ISMS) розуміється частка загальної системи управління, що базується на аналізі ризику і призначена для проектування, реалізації, контролю, супроводу і вдосконалення заходів в області інформаційної безпеки. Цю систему складають організаційні структури, політика, дії з планування, обов'язки, процедури, процеси і ресурси;

- проведено аналіз механізмів комплексної системи захисту інформації (КСЗІ) та забезпечення безпеки, які реалізовані в сучасних операційних системах;

- запропоновано класифікацію варіантів використання операційних систем у інформаційно-телекомунікаційних системах. Визначено вимоги до механізмів захисту інформації для операційних систем за класифікацією, що запропонована;

- запропоновано вимоги до стандарту захисту інформації операційної системи та вимоги до механізмів захищеності ОС в рамках класу використання.

***Abstract**: At the present stage of the use of information technologies in society, the issue of information protection becomes important. Operating systems play a major role in this. They are assigned the role of protectors of all user data and access rights.*

The authors of the article were tasked with proposing a classification of the use of operating systems and with the requirements for mechanisms of protection of information under this classification. In the article:

- the existing security standards that are implemented in modern operating systems are analyzed. Existing security standards are outlined (Trusted Computer System Evaluation Criteria «Orange Book», TCSEC, ISO 17799). In the Orange Book, a trusted system is defined as "a system that uses sufficient hardware and software to provide simultaneous processing of information of varying secrecy by a group of users without violating access rights." Security mechanisms and security classes of modern operating systems and BS 7799 security management model are also considered;

of safety regulators, useful for the organization of almost any size, structure and scope information security management system. The standard Information Security Management System (ISMS) refers to the proportion of the overall risk-based management system designed to design, implement, control, maintain and improve information security activities. This system consists of organizational structures, policies, planning actions, responsibilities, procedures, processes and resources;

- the analysis of the mechanisms of the complex system of information security (CSIS) and security, which are implemented in modern operating systems;

- classification of operating system usage variants in information and telecommunication systems is offered. Requirements for information security mechanisms for operating systems according to the proposed classification are defined;

- requirements for operating system information security standard and requirements for OS security mechanisms within the usage class are proposed.

this standard contains a systematic, complete, universal list

1. ВСТУП

Кількість атак на сервери і системи, що як локальні, без підключення, так і такі, що знаходяться в локальній мережі або в мережі Інтернет росте з кожним роком. Тому необхідно розробити правила, за допомогою яких ми визначатимемо, - наскільки можна назвати операційну систему захищеною, а також розробити стандартні правила, на основі яких кваліфікуватимемо встановлюване або таке, що розробляється ПО як захищене, або як таке, що порушує безпеку операційної системи [1-4].

1.1 Існуючі стандарти забезпечення безпеки. Проведення аналізу механізмів забезпечення безпеки, які реалізовано у сучасних операційних системах.

Критерії оцінки довірених комп'ютерних систем було запропоновано в “Помаранчевій книзі» [5], яка згодом стала міжнародним стандартом – ISO 17799.

Історично першим стандартом, що здобув широке розповсюдження і такий, що зробив величезний вплив на базу стандартизації в багатьох країнах, став стандарт Міністерства оборони США "Критерії оцінки довірених комп'ютерних систем". Він був вперше опублікований в серпні 1983 року. Мова йде не про безпечні, а про довірені системи, тобто системи, яким можна надати певний ступінь довіри. Дану працю називають найчастіше за кольором обкладинки "Помаранчевою книгою. "Помаранчева книга" пояснює поняття безпечної системи, яка "управляє, за допомогою відповідних засобів, доступом до інформації, так що тільки належним чином авторизовані особи або процеси, що діють від їх імені, отримують право читати, записувати, створювати і видаляти інформацію". Очевидно, проте, що абсолютно безпечних систем не існує, це абстракція. Є сенс оцінювати лише ступінь довіри, яку можна надати тій або іншій системі. У "Помаранчевій книзі" довірена система визначається як "система, що використовує достатні апаратні і програмні засоби, щоб забезпечити одночасну обробку інформації різного ступеня секретності групою користувачів без порушення прав доступу".

Британський стандарт BS 7799 [6] фактично теж має статус міжнародного (ISO/IEC 17799).

Перша частина стандарту, "Управління інформаційною безпекою. Практичні правила", містить систематичний, вельми повний, універсальний перелік регулювальників безпеки, корисний для організації практично будь-якого розміру, структури і сфери діяльності.

Ця частина призначена для використання як довідковий документ керівниками і рядовими співробітниками, що відповідають за планування, реалізацію і підтримку внутрішньої системи інформаційної безпеки.

Згідно стандарту, мета інформаційної безпеки - забезпечити безперебійну роботу організації, по можливості запобігти і/або мінімізувати збиток від порушень безпеки.

Управління інформаційною безпекою дозволяє колективно використовувати дані, одночасно забезпечуючи їх захист і захист обчислювальних ресурсів.

Підкреслюється, що захисні заходи виявляються значно дешевшими і ефективнішими, якщо вони закладені в інформаційні системи і сервіси на стадіях завдання вимог і проектування.

У стандарті виділяється десять ключових регулювальників, які або є обов'язковими відповідно до чинного законодавства, або вважаються за основні структурні елементи інформаційної безпеки.

У другій частині стандарту BS 7799-2:2002 "Системи управління інформаційною безпекою -

специфікація з керівництвом по використанню" предметом розгляду, як випливає з назви, є система управління інформаційною безпекою.

Також слід звернути увагу на захищеність операційних систем для мобільних приладів та вбудованих систем [7,8].

2. АНАЛІЗ СУЧАСНИХ СТАНДАРТІВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

2.1 КЛАСИ БЕЗПЕКИ СУЧАСНИХ ОПЕРАЦІЙНИХ СИСТЕМ І МОДЕЛЬ УПРАВЛІННЯ БЕЗПЕКОЮ BS 7799

Проведений аналіз стандартів показав, що "Критерії ..." Міністерства оборони США відкрили шлях до ранжування інформаційних систем по ступеню довіри безпеки.

У "Помаранчевій книзі" визначається чотири рівні довіри - D, C, B і A. Рівень D призначений для систем, визнаних незадовільними. У міру переходу від рівня C до A до систем пред'являються все більш жорсткі вимоги. Рівні C і B підрозділяються на класи (C1, C2, B1, B2, B3) з поступовим зростанням ступеня довіри.

Всього є шість класів безпеки - C1, C2, B1, B2, B3, A1. Щоб в результаті процедури сертифікації систему можна було віднести до деякого класу, її політика безпеки і рівень гарантованості повинні задовольняти заданим вимогам, з яких ми згадаємо лише найважливіші.

Клас C1:

- довірена обчислювальна база повинна управляти доступом іменованих користувачів до іменованих об'єктів;

- користувачі повинні ідентифікувати себе, перш ніж виконувати які-небудь інші дії, контрольовані довіреною обчислювальною базою. Для аутентифікації повинен використовуватися який-небудь захисний механізм, наприклад пароль. Аутентифікаційна інформація має бути захищена від несанкціонованого доступу;

- довірена обчислювальна база повинна підтримувати область для власного виконання, захищену від зовнішніх дій (зокрема, від зміни команд і/або даних) і від спроб стеження за ходом роботи;

- захисні механізми мають бути протестовані на предмет відповідності їх поведінки системної документації.

Клас C2 (на додаток до C1):

- права доступу повинні гранулюватися з точністю до користувача. Всі об'єкти повинні піддаватися контролю доступу;

- при виділенні об'єкту, що зберігається, з пулу ресурсів довіреної обчислювальної бази необхідно ліквідовувати всі сліди його використання;

- кожен користувач системи повинен унікальним чином ідентифікуватися.

Клас B1 (на додаток до C2):

- довірена обчислювальна база повинна управляти мітками безпеки асоційованими з кожним суб'єктом і об'єктом, що зберігається;

- довірена обчислювальна база повинна забезпечити реалізацію примусового управління доступом всіх суб'єктів до всіх об'єктів, що зберігаються;

- група фахівців, що повністю розуміють реалізацію довіреної обчислювальної бази, повинна піддати опис архітектури, початкових і об'єктних кодів ретельному аналізу і тестуванню;

Клас B2 (на додаток до B1):

- забезпечуватися мітками повинні всі ресурси системи (наприклад, ПЗП), прямо або побічно доступні суб'єктам;

- до довіреної обчислювальної бази повинен підтримуватися довірений комунікаційний шлях для користувача, що виконує операції початковій ідентифікації і аутентифікації;

- має бути передбачена можливість реєстрації подій, пов'язаних з організацією таємних каналів обміну з пам'яттю;

- має бути продемонстрована відносна стійкість довіреної обчислювальної бази до спроб проникнення;

- модель політики безпеки має бути формальною. Для довіреної обчислювальної бази повинні існувати описові специфікації верхнього рівня, точно і що повно визначають її інтерфейс;

Клас B3 (на додаток до B2):

- для довільного управління доступом повинні обов'язково використовуватися списки управління доступом з вказівкою дозволених режимів;

- має бути передбачена можливість реєстрації появи або накопичення подій, що несуть загрозу політиці безпеки системи. Адміністратор безпеки має негайно сповіщатися про спроби порушення політики безпеки, а система, в разі продовження спроб має присікати їх найменш хворобливим способом;

- процедура аналізу має бути виконана для тимчасових таємних каналів;

- має бути специфікована роль адміністратора безпеки. Здобути права адміністратора безпеки можна тільки після виконання явних протокольованих дій;

- має бути продемонстрована стійкість довіреної обчислювальної бази до спроб проникнення.

Клас A1 (на додаток до B3):

- тестування повинне продемонструвати, що реалізація довіреної обчислювальної бази відповідає формальним специфікаціям верхнього рівня;

- окрім описових, мають бути представлені формальні специфікації верхнього рівня. Необхідно використовувати сучасні методи формальної специфікації і верифікації систем;

- механізм конфігураційного управління повинен розповсюджуватися на весь життєвий цикл і всі компоненти системи, що мають відношення до забезпечення безпеки;

- має бути описана відповідність між формальними специфікаціями верхнього рівня і початковими текстами.

Коротко класифікацію, що введена в "Помаранчевій книзі" можна сформулювати так:

рівень С - довільне управління доступом;

рівень В - примусове управління доступом;

рівень А – безпека, що верифікується.

З недоліків даної концепції варто відзначити, наприклад, повне ігнорування проблем, що виникають в розподілених системах.

У стандарті BS 7799-2:2002 застосована чотирьох фазна модель процесу управління інформаційною безпекою: планування – реалізація – оцінка – коректування (ПРОК).

Процес управління має циклічний характер; на фазі первинного планування здійснюється вхід в цикл. Як перший крок має бути визначена і документована політика безпеки організації.

Потім визначається зона дії системи управління інформаційною безпекою. Вона може охоплювати всю організацію або її частки. Слід специфікувати залежності, інтерфейси і припущення, пов'язані з межею між СУІБ і її оточенням; це особливо важливо, якщо в зону дії потрапляє лише частка організації. Велику область доцільно поділити на підобласті управління.

Ключовим елементом фази планування є аналіз ризику.

Результат аналізу ризику - вибір регулювальників безпеки. План повинен включати графік і пріоритети, детальний робочий план і розподіл обов'язків по реалізації цих регулювальників.

В стандарті наголошується, що систему управління інформаційною безпекою треба постійно удосконалювати, щоб вона залишалася ефективною . Цю мету переслідує четверта фаза розглядуваного в стандарті циклу - коректування. Вона може зажадати як щодо незначних дій, так і повернення до фаз планування (наприклад, якщо з'явилися нові погрози) або реалізації (якщо слід здійснити намічене раніше).

Завдання фази оцінки - виявити проблеми. На фазі коректування необхідно докопатися до їх коріння і усунути першопричини невідповідностей, щоб уникнути повторної появи. З цією метою можуть робитися як реактивні, так і превентивні дії, розраховані на середньострокову або довгострокову перспективу.

2.2. АНАЛІЗ МЕХАНІЗМІВ КОМПЛЕКСНОЇ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ (КСЗІ) ТА ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ, ЯКІ РЕАЛІЗОВАНІ В СУЧАСНИХ ОПЕРАЦІЙНИХ СИСТЕМАХ.

Завдання ідентифікації і аутентифікації користувачів є ключовим завданням захисту інформації від несанкціонованого доступу, вирішуваною системою захисту інформації в будь-якій операційній системі, оскільки розмежувальна політика доступу до ресурсів формується за допомогою завдання прав доступу для користувачів операційної системи.

1. Дискреційний механізм управління доступом користувачів до файлових об'єктів.

Проаналізуємо вимоги до механізму.

КСЗІ повинна контролювати доступ суб'єктів (користувачів), що найменовують, до об'єктів, що найменовують (файлам, програмам, томам і так далі).

Для кожної пари (суб'єкт - об'єкт) має бути задане явне і недвозначне перерахування допустимих типів доступу (читати, писати і так далі), тобто тих типів доступу, які є санкціонованими для даного суб'єкта (індивіда або групи індивідів) до даного ресурсу (об'єкту).

КСЗІ повинна містити механізм, що запроваджує в життя дискреційні правила розмежування доступу.

Контроль доступу має бути застосовний до кожного об'єкту і кожного суб'єкта (індивідові або групі рівноправних індивідів).

Механізм, що реалізовує дискреційний принцип контролю доступу, повинен передбачати можливості санкціонованої зміни правил розмежування доступу (ПРД), зокрема, можливість санкціонованої зміни списку користувачів і списку об'єктів, що захищаються.

Право змінювати ПРД повинне надаватися виділеним суб'єктам (адміністрації, службі безпеці і так далі).

Наведемо реалізацію вимог:

У систему в ланцюжок обробки запитів на доступ до файлової системи встановлюється драйвер системи захисту, який здійснює контроль доступу користувачів до об'єктів згідно заздалегідь визначеним правилам.

Кожен користувач визначається такими параметрами: реальний ідентифікатор користувача (RUID), ефективний ідентифікатор користувача (EUID). Це дозволяє розмежовувати доступ до об'єктів для користувача (групи користувачів), здійснюючи при цьому контроль коректності запозичення прав, що принципово підвищує рівень захищеності системи.

Право змінювати ПРД надається виділеним суб'єктам (адміністрації, службі безпеці і так далі). Це забезпечується власне реалізацією механізму розмежування прав доступу, права на завдання (зміна) ПРД надаються тільки привілейованому користувачеві - адміністраторові безпеки з інтерфейсу КСЗІ.

2. Дискреційний механізм управління доступом користувачів до пристроїв.

Проаналізуємо вимоги до механізму.

Ті ж, що і для управління доступом до файлових об'єктів.

Наведемо реалізацію вимог.

Звернення до пристроїв в ОС Unix реалізується через файлову систему, при цьому як об'єкт виступають файли пристроїв, які можуть бути блокові (дискові пристрої - дисковод, CD-ROM і так далі), або символьні (клавіатура, миша і так далі). Драйвером КСЗІ здійснюється розмежування доступу до файлів пристрою (пристроям) для користувачів.

3. Дискреційний механізм управління доступом до видалених робочих станцій і серверів ЛВС по протоколу TCP(UDP) /IP.

Проаналізуємо вимоги до механізму.

Ті ж, що і для управління доступом до файлових об'єктів.

Наведемо реалізацію механізму.

До складу КСЗІ входить драйвер контролю доступу до мережевих ресурсів. Як об'єкти доступу виступають IP-адреса і значення портів доступу. Драйвер КСЗІ здійснює фільтрацію повідомлень дозволяючи взаємодіяти робочій станції в складі ЛВС тільки з комп'ютерами, що мають задані IP, - адреси, причому взаємодіяти тільки по заданих номерах портів.

4. Механізм очищення зовнішньої пам'яті.

Проаналізуємо вимоги до механізму.

При первинному призначенні або при перерозподілі зовнішньої пам'яті КСЗІ повинен запобігати доступу суб'єктові до залишкової інформації.

Наведемо реалізацію механізму.

Перерозподіл зовнішньої пам'яті можливий тільки в разі звільнення частки пам'яті, раніше займаної файлом. Це звільнення відбувається у момент видалення (або модифікації) файлу. Для запобігання доступу до залишкової інформації в КСЗІ використовується прикладна програма очищення зовнішньої пам'яті, яка перед видаленням (модифікацією) файлу автоматично затирає його (записує в нього нульові дані, або "всі одиниці" - кількість циклів перезапису "нулями і "одиницями" задається при налаштуванні КСЗІ), затираючи цим його початкову інформацію.

5. Механізм ідентифікації і аутентифікації.

Проаналізуємо вимоги до механізму.

КСЗІ повинна вимагати від користувачів ідентифікувати себе при запитах на доступ. КСЗІ повинна піддавати перевірці достовірність ідентифікації - здійснювати аутентифікацію. КСЗІ повинна мати в своєму розпорядженні необхідні дані для ідентифікації і аутентифікації. КСЗІ повинна перешкоджати доступу до ресурсів неідентифікованих користувачів і користувачів, що захищаються, достовірність ідентифікації яких при аутентифікації не підтвердилася.

Наведемо реалізацію механізму.

- заміна стандартної програми ОС аутентифікації користувачів (у ОС Linux вся ідентифікація і аутентифікація користувачів проводиться за допомогою такої програми) - login - на програму, що входить до складу КСЗІ, а також заміна стандартної програми ОС призначення паролів (завдання обмежень на пароль) користувачам passwd.

- ідентифікація і аутентифікація операторів при вході в ОС по ідентифікатору (коду) і паролю умовно-постійної дії;

- перевірка складності пароля (пароль повинен містити не менше двох букв і не менш однієї цифри, пароль не повинен повторювати імені користувача або бути якою-небудь модифікацією імені користувача) і терміну його дії;

- для кожного облікового запису передбачена можливість ведення історії паролів;

- блокування облікових записів операторів при введенні невірного пароля з розблокуванням через службу адміністратора безпеки.

6. Механізм реєстрації (аудиту) подій.

Проаналізуємо вимоги до механізму.

КСЗІ повинна здійснювати реєстрацію наступних подій:

- використання ідентифікаційного і аутентифікаційного механізму;

- запит на доступ до ресурсу, що захищається (відкриття файлу, запуск програми і так далі);

- створення і знищення об'єкту;

- дії із зміни ПРД.

Для кожної з цих подій повинна реєструватися наступна інформація:

- суб'єкт, що здійснює реєстровану дію;

- дата і час;

- тип події (якщо реєструється запит на доступ, то слід відзначати об'єкт і тип доступу);

- чи вдало здійснилася подія (обслужений запит на доступ чи ні).

КСЗІ повинна містити засоби вибіркового ознайомлення з реєстраційною інформацією.

Наведемо реалізацію механізму.

- реєстрація входу (виходу) операторів до ОС, або реєстрації завантаження і ініціалізації операційної системи і її програмного останову;

- реєстрація запуску/завершення програм і процесів;

- реєстрація спроб доступу користувачів і програмних засобів (програм процесів) до томів, що захищаються, каталогів і файлів;

- дії із зміни ПРД і так далі.

Кожен власний механізм захисту КСЗІ веде реєстрацію заданих адміністратором безпеки подій, реєструючи потрібні вище параметри.

7. Механізм контролю цілісності КСЗІ.

Проаналізуємо вимоги до механізму.

Мають бути передбачені засоби періодичного контролю цілісності програмної і інформаційної частки КСЗІ.

Наведемо реалізацію механізму.

Цілісність КСЗІ контролюється підсистемою контролю цілісності файлової системи. Вона реалізована у вигляді додаткової програми, яка через заданий інтервал часу перевіряє набір файлів, що задається адміністратором, на предмет зміни їх контрольної суми, розміру, часу останньої модифікації. При виявленні невідповідності (при завданні відповідній опції адміністратором) проводиться автоматичне відновлення файлів з резервної копії.

КСЗІ здійснює періодичний контроль цілісності не лише власних файлів, але і будь-яких інших файлів (по вибору адміністратором), які розташовані на робочій станції, що захищається, задаються для контролю адміністратором безпеки.

8. Дискреційний механізм управління доступом процесів до файлових об'єктів і до пристроїв.

Проаналізуємо вимоги до механізму.

КСЗІ повинна забезпечувати дискреційний доступ до об'єктів і пристроїв

Наведемо реалізацію механізму.

У ланцюжок обробки запитів на доступ до файлової системи і до пристроїв встановлюється драйвер системи захисту, який здійснює контроль доступу процесів до об'єктів згідно заздалегідь визначеним правилам.

Кожен суб'єкт доступу визначається не лише параметрами користувача: реальний ідентифікатор користувача (RUID), ефективний ідентифікатор користувача (EUID), але і параметром процесу - процес, що виконує дію. Це дозволяє розмежовувати доступ до об'єкту не лише для користувача (групи користувачів), але і для процесу. КСЗІ реалізує розмежування доступу, як тільки для користувача (доступ до об'єкту дозволений, якщо він дозволений користувачеві, що звертається до об'єкту) або тільки для процесу (доступ до об'єкту дозволений, якщо він дозволений процесу, що звертається до об'єкту), так для користувача і процесу разом (доступ до об'єкту дозволений, якщо він дозволений користувачеві і процесу одночасно).

КСЗІ надає адміністраторові вельми сильний механізм протидії погрозам, проте адміністратор повинен відповідним чином набудувати механізм з урахуванням особливостей прояву погроз, від яких слід захищати комп'ютер.

9. Забезпечення замкнутості програмного середовища.

Проаналізуємо вимоги до механізму.

Це найважливіший механізм захисту, який є основним механізмом протидії прихованим погрозам. Коректне його налаштування не дозволить зловмисникові запускати власні програми, як наслідок, позбавить його якого-небудь інструментарію протидії системі захисту. Крім того, даний механізм є одним з основних механізмів антивірусної протидії. Даний механізм захисту обов'язково має бути активізований і настроєний.

Наведемо реалізацію механізму.

Для кожного користувача можна встановити розмежування на запуск програм, тобто для кожного користувача можна задати список дозволених йому для запуску процесів. Дозволені для запуску програми задаються їх повнопутніми іменами (або каталогами, з яких можуть запускатися файли, або масками). При забороні модифікації відповідних файлів (засобами розмежування доступу до файлової системи), користувач не має можливості підмінити дозволені йому на запуск програми. Оскільки при запуску програми аналізується її повний шлях розташування у файловій системі, то користувач не має можливості запустити власний процес під ім'ям дозволеного до запуску.

Даний механізм реалізується власним драйвером. Перед запуском процесу, драйвер аналізує повнопутнє ім'я процесу, на предмет входження даного імені в список дозволених для запуску процесів для користувача. У випадку, якщо процесу з даним повнопутнім ім'ям в списку не існує, КСЗІ не дозволить здійснити його запуск.

10. Механізм контролю цілісності програм перед запуском.

Проаналізуємо вимоги до механізму.

У КСЗІ має бути присутнім механізм відстежування несанкціонованої модифікації системних файлів при їх використанні, а також наявність можливості періодичної перевірки і перевірка цілісності системних файлів при запуску операційної системи. При виявленні несанкціонованої модифікації системних файлів (при завданні відповідній опції адміністратором) повинно проводиться автоматичне відновлення файлів з резервної копії.

Наведемо реалізацію механізму.

Цілісність програм перед запуском контролюється підсистемою, виконаною у вигляді окремої прикладної програми. Адміністратором безпеки задаються програми, які повинні контролюватися перед запуском на предмет зміни їх контрольної суми, розміру, часу останньої модифікації. Перед запуском контрольованої програми драйвером КСЗІ передається управління даній прикладній програмі, що здійснює контроль. При виявленні невідповідності (при завданні відповідній опції адміністратором) проводиться автоматичне відновлення контрольованої програми з резервної копії, після чого вона запускається, або забороняється запуск модифікованої програми.

11. Механізм управління доступом до монтування пристроїв.

Проаналізумо вимоги до механізму.

Механізм повинен реалізовувати права доступу для кожного користувача по відношенню до кожного пристрою

Наведемо реалізацію механізму.

Для підключення до системи пристрою (дисковода, CD-ROM приводу і ін.) його необхідно змонтувати - "підключити" до файлової системи (до каталога). "Точкою" підключення пристрою до файлової системи визначається повнопутнє ім'я пристрою. КСЗІ дозволяє розмежовувати доступ до монтування пристроїв, за допомогою завдання до яких каталогів, які пристрої і з якою файловою системою можуть вмонтовуватися.

12. Механізм аутентифікації при завантаженні в режимі Single-User Mode.

Проаналізуємо вимоги до механізму.

Ідентифікація і аутентифікація перед запуском даного режиму, причому ця можливість має бути доступна тільки адміністраторові системи.

Наведемо реалізацію механізму.

У процес початкового завантаження системи вбудовується механізм (власними засобами КСЗІ змінена стартова процедура завантаження системи) що запобігає неаутентифікованому завантаженню системи в режимі Single-User Mode (цей аварійний режим дозволяє здійснити вибіркове завантаження системи, зокрема без драйверів КСЗІ). За умовчанням вхід в даний режим доступний будь-якому користувачеві системи. КСЗІ вимагає проведення ідентифікації і аутентифікації перед запуском даного режиму, при цьому "за умовчанням" дозволяє доступ до запуску режиму тільки адміністраторові (після його авторизації).

13. Механізм надання привілеїв адміністраторові безпеки.

Проаналізуємо вимоги до механізму.

Даний механізм призначений для підвищення прав адміністраторові безпеки в порівнянні з "root", що дозволяє функціонально розмежовувати завдання системного адміністратора, якому належать права "root" і адміністратора безпеки.

Наведемо реалізацію механізму.

Механізм забезпечує неможливість проводити які-небудь дії, щодо процесів КСЗІ користувачеві з правами "root". При цьому, оскільки КСЗІ розмежовує доступ до об'єктів, зокрема до налаштувань КСЗІ власними засобами, користувач з правами "root" не має можливості впливати на функціонування КСЗІ.

3. МЕТА СТАТТІ

Метою статті є висвітлення існуючих стандартів безпеки інформаційних систем і механізмів комплексної системи захисту інформації та запропонувати класифікацію варіантів використання операційних систем у інформаційно-телекомунікаційних системах.

4. ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

Будемо вважати операційну систему захищеною, якщо вона забезпечує збереженість інформації та цілісність даних власними засобами без використання допоміжного програмного забезпечення.

Основною задачею захищеної операційної системи є запобігання несанкціонованому доступу до даних із зовнішніх (по відношенню до неї) джерел. Тому ми будемо класифікувати операційні системи, по особливостях алгоритмів управління ресурсами – локальні і мережеві ОС.

Локальні ОС управляють ресурсами окремого комп'ютера. Мережеві ОС беруть участь в управлінні ресурсами мережі.

4.1. КЛАСИФІКАЦІЯ ОПЕРАЦІЙНИХ СИСТЕМ ЗА ВАРІАНТАМИ ВИКОРИСТАННЯ

Наведемо таку класифікацію операційних систем за варіантами використання.

1. ОС для використання на робочій станції, не

підключеній до мережі передачі даних.

2. ОС для використання на робочій станції, підключеній до внутрішньої мережі інтранет.

3. ОС для використання на робочій станції, підключеній до зовнішньої мережі інтернет.

4. ОС для використання на серверах, підключених у внутрішній мережі інтранет.

5. ОС для використання на серверах, підключених в зовнішній мережі інтернет.

Ми називатимемо сторонніми компонентами операційної системи, компоненти, не використання яких не є критичним для нормального функціонування системи локально або в мережі передачі даних

Часткою системи ми називатимемо ті компоненти, які необхідні для безвідмовного захищеного функціонування системи локально і при підключенні мережі передачі даних, а також механізми захисту доступу до інформації і контролю цілісності.

Не часткою системи ми вважатимемо компоненти, які вносять корективи до системних налаштувань і у яких є функція отримання інформації із-за зовнішнього периметра захисту операційної системи.

Будемо вважати операційну систему такою, що задовольняє стандарту захищеної операційної системи України, якщо вона включає такі компоненти (механізми): пакетний фільтр, механізм розмежування доступу до інформації, механізми очищення; механізм реєстрації системних подій, механізм управління і розподілу пам'яті, механізм шифрування і криптоалгоритмів, механізм резервного копіювання, механізм захисту від збоїв, механізм підключення до мережі і управління налаштуваннями підключення, механізм трансляції мережевих адрес, механізм підключення додаткових пристроїв до системи і їх управління, механізм контролю цілісності системи, механізм захисту файлової системи, механізм відновлення системи, механізм управління системними сервісами і драйверами, механізм запуску завдань за розкладом;

4.2. ВИМОГИ ДО СТАНДАРТУ ЗАХИСТУ ІНФОРМАЦІЇ ОПЕРАЦІЙНОЇ СИСТЕМИ

Сформулюємо певні вимоги до захищеної операційної системи.

1. Пакетний фільтр має забезпечувати:

- можливість створення окремих правил для всіх мережевих інтерфейсів;

- можливість створення окремих правил управління доступом для вихідного і вхідного трафіку;

- можливість створення правила «Що явно не дозволене – заборонено»;

- правила, що допускають вільне безперешкодне переміщення внутрішніх пакетів на локальні інтерфейси і адресу 127.0.0.1;

- можливість заносити в лог пакети для кожного правила;

- правила, для яких частіше виконується відповідність, мають бути написані вище за правила, для яких відповідність виконується рідше. Останнє правило в розділі повинно блокувати і заносити в лог всі пакети даного інтерфейсу і напряму;

- не повинна відправлятися відповідь на небажані пакети, що блоковані засобами операційної системи, тобто щоб той, хто атакує не знав чи досягли його пакети мети;

- можливість створення динамічного правила для віддзеркалення атак типу «відмова у обслуговуванні». Мається на увазі обмеження кількості одночасно встановлюваних сесій за допомогою огляду полів джерела або одержувача в правилі, яке виконується, якщо встановлений ліміт одночасних сесій. Якщо значення лічильника більше чим значення, що вказане в ліміті, – пакет відкидається;

- можливість текстової фільтрації трафіку (відкидати або пропускати пакети, в яких зустрічаються символи або рядки, що задовольняють встановленим правилам).

2. Засоби розмежування доступу до інформації

- має бути задане явне і недвозначне перерахування допустимих типів доступу (читання, запис, виконання, виконання з більш вищими привілеями) для кожного суб'єкта або групи по відношенню до об'єкту (ресурсу);

- контроль доступу має бути застосовний до кожного об'єкту і кожному суб'єктові;

- можливість змінювати правила доступу повинно надаватися виділеним суб'єктам (адміністраторам, довіреним групам);

3. Механізми очищення

- повинне здійснюватися очищення тих, що звільняються областей оперативної пам'яті ЕОМ і зовнішніх накопичувачів. Очищення здійснюється довільним записом в область пам'яті, що звільняється, раніше використану для зберігання даних, що захищаються (файлів);

- має здійснюватися очищення тимчасових системних файлів;

- архівація результатів моніторингу і відправка їх на сервер зберігання резервних копій, з подальшим видаленням незапакованих файлів з метою зменшити об'єм займаного місця;

4. Механізм реєстрації системних подій

- всі події в системі, які мають відношення до роботи системи безпосередньо, до безпеки і до

призначених для користувача застосувань винні фіксуватися і записуватися;

5. Критерій витривалості атаки: операційна система повинна встигнути записати повний моніторинг дій зловмисника

- підтримка стабільної роботи системного моніторингу на протязі моніторингу атаки з повною фіксацією дій зловмисника і паралельною відправкою на сервер резервних копій;

6. Вимоги до паролів

- механізм встановлення паролів не повинен допускати установки слабих паролів менше 8 символів і що мають осмислене значення;

7. Механізм розподіленої пам'яті для процесів, об’єктів ядра, файлових і мережевих дескрипторів

- процеси і об'єкти ядра не повинні мати можливості по власній ініціативі міняти дані інших процесів або об'єктів ядра.

8. Механізм шифрування і криптоалгоритмів

- у системі мають бути інтегрована можливість здійснювати криптографічний стійкий захист даних від несанкціонованого доступу;

- система повинна використовувати інтегроване шифрування файлової системи при якому неможливо зчитати дані при знятті накопичувача і підключення його до іншого комп'ютера;

9. Відомості про операційну систему і сервіси

- система не повинна давати інформацію про своє найменування, версію і версії працюючих сервісів неавторизованому користувачеві;

10. Механізм резервного копіювання даних

- у систему має бути вбудований механізм, що забезпечує передачу даних на видалений сервер резервного копіювання;

11. Механізм захисту від помилок переповнювання буфера

- наявність механізму, який би на рівні ядра забороняв виконання коду з адресного сегменту даних;

12. Механізм сповіщення про успішні і невдалі спроби авторизації

- механізм повинен в реальному часі повідомляти адміністратора або групу довірених осіб про вироблювані спроби авторизації або підвищення привілеїв у системі;

- всі спроби повинні записуватися у файл і передаватися на резервний сервер зберігання даних;

13. Механізм контролю доступу пристроїв, що підключаються

- наявність системних таблиць з описом прав доступу для підключення пристрої для кожного користувача зі встановленими дозволами використання (читання, запис, виконання);

14. Механізм управління процесами

- має бути реалізований механізм управління процесами в системі;

- кожен користувач може управляти процесами його групи, але не має доступу до більш вищих процесів;

15. Механізм підтримки цілісності системи

- наявність механізму відстежування несанкціонованої модифікації системних файлів при їх використанні, а також наявність можливості періодичної перевірки і перевірка цілісності системних файлів при запуску операційної системи;

- при виявленні несанкціонованої модифікації системних файлів (при завданні відповідній опції адміністратором) проводиться автоматичне відновлення файлів з резервної копії;

16. Механізм відновлення системи

- можливість відновлення системи після збою, пошкодження системні файлів, можливість створення контрольної точки відновлення;

17. Локальне блокування системи у відсутності користувача, що авторизувався

- можливість локально заблокувати доступ до системи на якийсь час відсутність користувача;

18. Механізм налагодження мережевих з'єднань

- можливість управляти налаштуванням підключення до мережі і параметрами підключення тільки адміністраторові або групі довірених осіб можливість управління правами доступу до мережі для різних користувачів;

19. Механізм управління системними сервісами

- управління системними сервісами і драйверами можливо тільки адміністраторові або групі довірених осіб;

20. Загальні критерії

- здобути права адміністратора безпеки можна тільки після виконання явних, протокольованих дій;

- вхід в систему під single-user повинен вимагати авторизації;

- користувачі повинні пройти ідентифікацію і аутентифікацію перш ніж виконувати які-небудь дії в системі

- аутентифікаційна інформація має бути захищена від несанкціонованого доступу;

- кожна реєстрована дія повинна асоціюватися з конкретним користувачем.

Зазначені вимоги мають скласти ядро захищеної операційної системи реального часу.

6. ВИСНОВОК

В статті проведено аналіз існуючих стандартів забезпечення безпеки, які реалізовано у сучасних операційних системах. Також проведено аналіз механізмів комплексної системи захисту інформації (КСЗІ) та забезпечення безпеки, які реалізовані в сучасних операційних системах.

Автором запропоновано класифікацію варіантів використання операційних систем у інформаційно-телекомунікаційних системах. На основі отриманих результатів пропонується в подальшому визначити вимоги до механізмів захисту інформації для операційних систем за класифікацією, що запропонована та запропонувати вимоги до стандарту захисту інформації операційної системи та вимоги до механізмів захищеності ОС в рамках класу використання.

7. СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] Multilevel Secure Operating Systems // Journal Of Information Science And Engineering 15, 91-106 (1999).

[2] Bell E., LaPaduIa L. Secure Computer Systems: A Mathematical Model // MITRE Technical Report 2547, Volume II, 1973.

[3] McLean J. Security models // Encyclopedia of software engineering, 1994

[4] Bell E., LaPaduIa L. Secure Computer Systems: Mathematical Foundations // MITRE Technical Report 2547, Volume I, 1973.

[5] Trusted Computer System Evaluation Criteria [Online]. Available: https://en.wikipedia.org/wiki/Trusted_Computer_System_Evaluation_Criteria

[6] Code of Practice for Information Security Management [Online]. Available: https://ru.wikipedia.org/wiki/BS_7799-1

[7] Фроимсон Л.И., Кутепов С.В., Тараканов О.В., Шереметов А.В. Основные принципы построения защищенной операционной системы для мобильных устройств. Спецтехника и связь. – 2013.- №1.-С.43-47

[8] SeLinux documentation. National security agency, 2011- [Online]. Available: http://www.nsa.gov/research//selinux/docs.html

***Бичков Олексій Сергійович**, доктор технічних наук, доцент, завідувач кафедри Програмних систем і технологій Київського національного університету імені Тараса Шевченка. Область наукових інтересів: математичні моделі систем захисту інформації, теорія гібридних автоматів, теорія можливості.*