Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики
Факультет інформаційних технологій
Кафедра безпеки інформації та телекомунікацій

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеню магістра

студента *Пуркара Сергія Олександровича*

академічної групи *125м-17-1*

спеціальності *125 Кібербезпека*

спеціалізації[1]

за освітньо-професійною програмою *Кібербезпека*

на тему *Розробка рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі*

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	к.т.н., доц. Флоров С.В.			
розділів:				
спеціальний	ас. Рибальченко Ю.П.			
економічний	к.е.н., доц. Пілова Д.П.			

Рецензент				

Нормоконтролер	ст. викл. Мєшков В.І.			

Дніпро
2018

ЗАТВЕРДЖЕНО:
завідувач кафедри
безпеки інформації та телекомунікацій
_____ д.т.н., проф. Корнієнко В.І.

«_____»_____20___року

ЗАВДАННЯ
на кваліфікаційну роботу ступеня магістра

студенту ***Пуркару С.О.*** **академічної групи** ***125м-17-1***
(прізвище та ініціали) (шифр)

спеціальності ***125 Кібербезпека***

спеціалізації[1]

за освітньо-професійною програмою ***Кібербезпека***

на тему ***Розробка рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі***

1 ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Наказ ректора НТУ «Дніпровська політехніка» від 29.11.2018 № 2025-л_____

2 МЕТА ТА ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Об'єкт досліджень ***процес налаштування реплікацій в Windows та FreeBSD***

Предмет досліджень ***є моделі реплікацій в Windows та FreeBSD, а також системи безпеки служб каталогів цих ОС***

Мета ***Забезпечити можливість захищеної реплікацій даних в гетерогенному середовищі***

Вихідні дані для проведення роботи ***законодавство України та міжнародні стандарти у сфері кібербезпеки***

3 ОЧІКУВАНІ РЕЗУЛЬТАТИ

Наукова новизна ***розробити рекомендацій для захищеного реплікаційного з'єднання серверів, які працюють зі службами каталогів на базі протоколу X.500***

Практична цінність ***полягає в розробці способу проводити реплікації між ОС Windows та FreeBSD, а також створено систему захисту, яка за допомогою SSL з'єднання допомагає шифрувати канал передачі інформації***

4 ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Закону України «Про інформацію», НД ТЗІ 1.1-002-99, НД ТЗІ 2.5-004-99, НД ТЗІ 2.5-005-99, НД ТЗІ 3.7-001-99

5 ЕТАПИ ВИКОНАННЯ РОБІТ

Найменування етапів робіт	**Строки виконання робіт** (початок-кінець)
Огляд джерел за темою та напрям досліджень	03.09.18-06.10.18
Методи досліджень	07.10.18-31.10.18
Результати досліджень	01.11.18-24.11.18
Виконання економічного розділу	25.11.18-04.12.18
Оформлення пояснювальної записки	05.12.18-10.12.18

6 РЕАЛІЗАЦІЯ РЕЗУЛЬТАТІВ ТА ЕФЕКТИВНІСТЬ

Економічний ефект ***очікується позитивним завдяки підвищенню захищеності ІзОД і зниженню ймовірності збитку від несанкціонованих дій зловмисників внаслідок залучення рекомендованих мір та засобів захисту ІзОД***

Соціальний ефект ***полягає в забезпеченні захисту корпоративної інформації від витоків.***

7 ДОДАТКОВІ ВИМОГИ

Завдання видано _____ Флоров С.В.
(підпис керівника) (прізвище, ініціали)

Дата видачі: 03.09.18р.

Дата подання до екзаменаційної комісії: 14.12.18р.

Прийнято до виконання _____ Пуркар С.О.
(підпис студента) (прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: ___ с., __ рис., __ табл., 4 додатки, 14 джерел.

Мета магістерської дипломної роботи: розробка списку рекомендацій по безпечному налаштуванню реплікацій в гетерогенній системі, порушення захищеності якої може загрожувати конфіденційності, цілісності чи доступності інформації.

У розділі «Стан питання. Постановка задачі» були розглянуті особливості обчислювальної мережі підприємства, поставлені основні задачі для виконання в дипломній роботі, була створена модель порушника, розроблені модель загроз та профіль захищеності.

В спеціальній частині по існуючому профілю були налаштовані захищені реплікації між ОС Windows Server 2008 R2 та FreeBSD 9.0. На основі всіх налаштувань створено список рекомендацій, які в майбутньому допоможуть спеціалістам з інформаційної безпеки без перешкод і зайвих пошуків потрібного матеріалу створювати захищені реплікаційні з’єднання між цими двома ОС, які основані на різних архітектурних платформах.

Практична цінність полягає в розробці способу проводити реплікації між ОС Windows та FreeBSD, а також створено систему захисту, яка за допомогою SSL з’єднання допомагає шифрувати канал передачі інформації.

СИНХРОНА РЕПЛІКАЦІЯ, АСИНХРОНА РЕПЛІКАЦІЯ, X.500, ACTIVE DIRECTORY, LDAP, КОРЕНЕВИЙ ЦЕНТР СЕРТИФІКАЦІЇ.

РЕФЕРАТ

Пояснительная записка: ___ с., __ рис., __ табл., 4 приложений, 14 источников.

Цель магистерской дипломной работы: разработка списка рекомендаций по безопасной настройке репликации в гетерогенной системе, нарушения защищенности которой может угрожать конфиденциальности, целостности или доступности информации.

В разделе «Состояние вопроса. Постановка задачи» были рассмотрены особенности вычислительной сети предприятия, поставлены основные задачи для выполнения в дипломной работе, была создана модель нарушителя, разработанные модель угроз и профиль защищенности.

В специальной части по существующему профилю были настроены защищенные репликации между Windows Server 2008 R2 и FreeBSD 9.0. На основе всех настроек создан список рекомендаций, которые в будущем помогут специалистам по информационной безопасности без помех и лишних поисков нужного материала создавать защищенные Репликационный соединение между этими двумя ОС, основанные на различных архитектурных платформах.

Практическая ценность заключается в разработке способа проводить репликации между ОС Windows и FreeBSD, а также создана система защиты, которая с помощью SSL соединение помогает шифровать канал передачи информации.

СИНХРОННЫХ РЕПЛИКАЦИИ, АСИНХРОННЫХ РЕПЛИКАЦИИ, X.500, ACTIVE DIRECTORY, LDAP, КОРНЕВАЯ ЦЕНТР СЕРТИФИКАЦИИ.

ABSTRACT

Explanatory note: ___ p., __ fig., __ tab., 4 application, 14 sources.

The purpose of the master's thesis is to develop a list of recommendations for the safe configuration of replication in a heterogeneous system, the security breach of which may threaten the confidentiality, integrity or availability of information.

In the section Question Status. Task setting, the features of the enterprise's computing network were considered, the main tasks were set for performance in the thesis work, the intruder model was created, the threat model and security profile were developed.

In the special section on the existing profile, secure replication between Windows Server 2008 R2 and FreeBSD 9.0 was configured. Based on all settings, a list of recommendations has been created that in the future will help information security specialists to create secure Replication connection between these two operating systems based on various architectural platforms without interference and unnecessary searches for the necessary material.

The practical value is to develop a way to replicate between Windows and FreeBSD, and a security system has been created that, using an SSL connection, helps to encrypt the communication channel.

SYNCHRONOUS REPLICATIONS, ASYNCHRONOUS REPLICATIONS, X.500, ACTIVE DIRECTORY, LDAP, ROOT CENTER FOR CERTIFICATION.

СПИСОК УМОВНИХ СКОРОЧЕНЬ

АС	–	автоматизована система;
ДСК	–	для службового користування;
ІБ	–	інформаційна безпека;
ІзОД	–	інформація з обмеженим доступом;
ІТ	–	інформаційні технології;
ІТС	–	інформаційно-телекомунікаційна система;
КЗЗ	–	комплекс засобів захисту;
ЗІ	–	захист інформації;
КСЗІ	–	комплексна система захисту інформації;
НСД	–	несанкціонований доступ;
НД ТЗІ	–	нормативний документ технічного захисту інформації;
ОС	–	обчислювальна система;
ПБ	–	політика безпеки;
ПЗ	–	програмне забезпечення;
AD	–	Active Directory;
HTTP	–	Hyper Text Transfer Protocol;
HTTPS	–	HTTP incapsulated SSL;
LDAP	–	Lightweight Directory Access Protocol;
SSL	–	Secure Sockets Layer;
XML	–	eXtensible Markup Language.

ЗМІСТ

с.

ВСТУП

За останні кілька років наша країна зробила величезний крок у розвитку й популярності Internet-технологій. За період до 2018-го року робота цілих підприємств через глобальну мережу вже нікого не дивувала і не була прерогативою великих корпорацій. З часом почав розширюватись і список сервісів, які стали необхідними через глобальну мережу.

Однією з самих популярних послуг, яку прагнуть отримати працівники підприємств, котрі працюють через Internet – це можливість отримувати доступ до свого облікового запису в будь-якій точці земного шару.

Коли користувач вносить зміни в свій обліковий запис, то вони зберігаються на сервері, до якого підключений термінал. Після цього через якийсь фіксований проміжок часу відбувається синхронізація між серверами компанії, офіси якої є географічно-віддаленими один від одного. Даний вид синхронізації називається реплікацією. Але, нажаль, на сьогоднішній день багато підприємств використовує гетерогенні автоматизовані системи(АС, в яких є операційні системи, які базуються на різних архітектурних платформах). Статистика корпорації IBM свідчить, що майже 42% комерційних та державних оргазінацій використовують в своїй роботі такі операційні системи, як Microsoft Windows 7, Linux Red Hat, Linux Fedora, MacOS та інші. А дослідження Themeware, яке проводилось в 2016 році показало, що в наш час набирають оберти технологічні можливості використовувати сервери з архітектурно різними ОС, та обмінюватись між ними інформацією.

Користувачів навіть не цікавить скільки і яке програмне забезпечення спрацьовує, коли вони, скажімо, хочуть змінити ім’я облікового запису, чи модернізувати свій пароль більш захищеним. Вони проводять десятки тисяч подібних операцій кожного дня, а в цей час на серверах проходять зміни і реплікації. Але проблема в іншому: якщо існує декілька офісів однієї організації працює на базі серверів з архітектурно різними ОС, то питання

налаштування реплікацій між ними буде супроводжуватись багатьма незручностями, використанням сторонніх технологій.

В той же час, облікові записи користувачів є персональними даними, які потребують захисту. Але чи можливо сконфігурувати між собою системи аутентифікації та авторизації різних ОС. Кожен інженер з захисту інформації отримавши подібне завдання пройде довгий шлях вивчення технічної документації та інших джерел доки зможе отримати відповідь на це питання.

Проблематика в тому, що розробники ОС навмисне не створюють спільних серверних протоколів взаємодії. Так деякі рішення є, але їх можна перелічити на пальцях однієї руки. Тож головна мета даного дипломного проекту – дати інженерам з захисту інформації список рекомендацій за допомогою якого можна отримати захищені реплікації в автоматизованих системах класу 3, коли в мережі є не захищений канал передачі інформації.

Для виконання дослідів та налаштувань було використано дві найпопулярніші серверні ОС: Windows Server 2008 R2 та FreeBSD 9.0.

1 СТАН ПИТАННЯ. ПОСТАНОВКА ЗАДАЧІ

1.1 Актуальність обраної теми

Тема є актуальною тому, що в наш час все більше і більше стає тенденцій до використання різних ОС за потребами. А також стає популярним безкоштовне ПЗ. Більшість ОС на ринку є безкоштовним саме тому вивчення методів захищених комунікацій в даних системах відіграє велику роль, адже фінансові активи фірм малого і середнього бізнесу не є наскільки великими, щоб купувати кожні 2 роки абонемент на технічну підтримку.

Також можна сказати, що для вирішення багатьох задач для різних ОС написано гарне ПЗ, використання якого дає певні результати при роботі підприємства, саме тому використання гетерогенних АС в наш час все більше набуває популярності, а такі АС також потребують захисту.

1.2 Мета дослідження

Розробка рекомендацій для захищеного реплікаційного з’єднання серверів, які працюють зі службами каталогів на базі протоколу X.500. Даний список рекомендацій послугує спеціалістам в сфері інформаційної безпеки та системним адміністраторам для реалізації одного з шляхів налаштування реплікацій між Windows та FreeBSD.

1.3 Об’єкт дослідження

Об’єктом дослідження процес налаштування реплікацій в Windows та FreeBSD. Спроба синхронізації служб каталогів цих ОС та можливість взаємодії серверів через різні протоколи, текстові, чи гіпертекстові формати даних, технології віддаленого виклику функціоналу ОС. Також досліджується спроба налаштування взаємодії систем захисту даних ОС та можливості взаємодій в них.

1.4 Предмет дослідження

Предметом дослідження є моделі реплікацій в Windows та FreeBSD а також системи безпеки служб каталогів цих ОС.

1.5 Задачі дослідження

1) Вивчити технологічні можливості для того, щоб зробити можливим передачу реплікаційних повідомлень в гетерогенній мережі;

2) Розробити список рекомендацій, які допоможуть фахівцям в сфері захисту інформації працювати в гетерогенній мережі;

3) Забезпечети захист каналу передачі, по якому будуть передаватись реплікаційні повідомлення.

1.6 Основні відомості про типове підприємство

Підприємство є аутсорсінговим підприємством, яке займається розробкою веб-систем. В ході роботи на підприємстві розроблено модель загроз і модель порушника. Підприємство складається з двох підрозділів: один з них використовує сервер на базі Windows Server 2008 R2, інший – FreeBSD 9.0

Персонал:

На підприємстві працює:

1 Директор підприємства (1 людина);

2 Заступник директора (1 людина);

3 Секретар (1 людина);

4 Бухгалтери (2 людини);

5 Програмісти (12 чоловік);

6 Відділ кадрів (2 людини);

7 Системний адміністратор (1 людина)

8 Тестувальники (4 чоловік)

9 Дизайнери (2 людини)

10 Охорона (3 людини)

11 Прибиральниці (2 людини)

Інформація з обмеженим доступом, циркулююча на ОІД: інформація про співробітників і клієнтів. Цей вид інформації охороняється законами України про інформацію та про захист персональних даних та іншими.

Вигляд циркулюючої інформації: електронна, паперова.

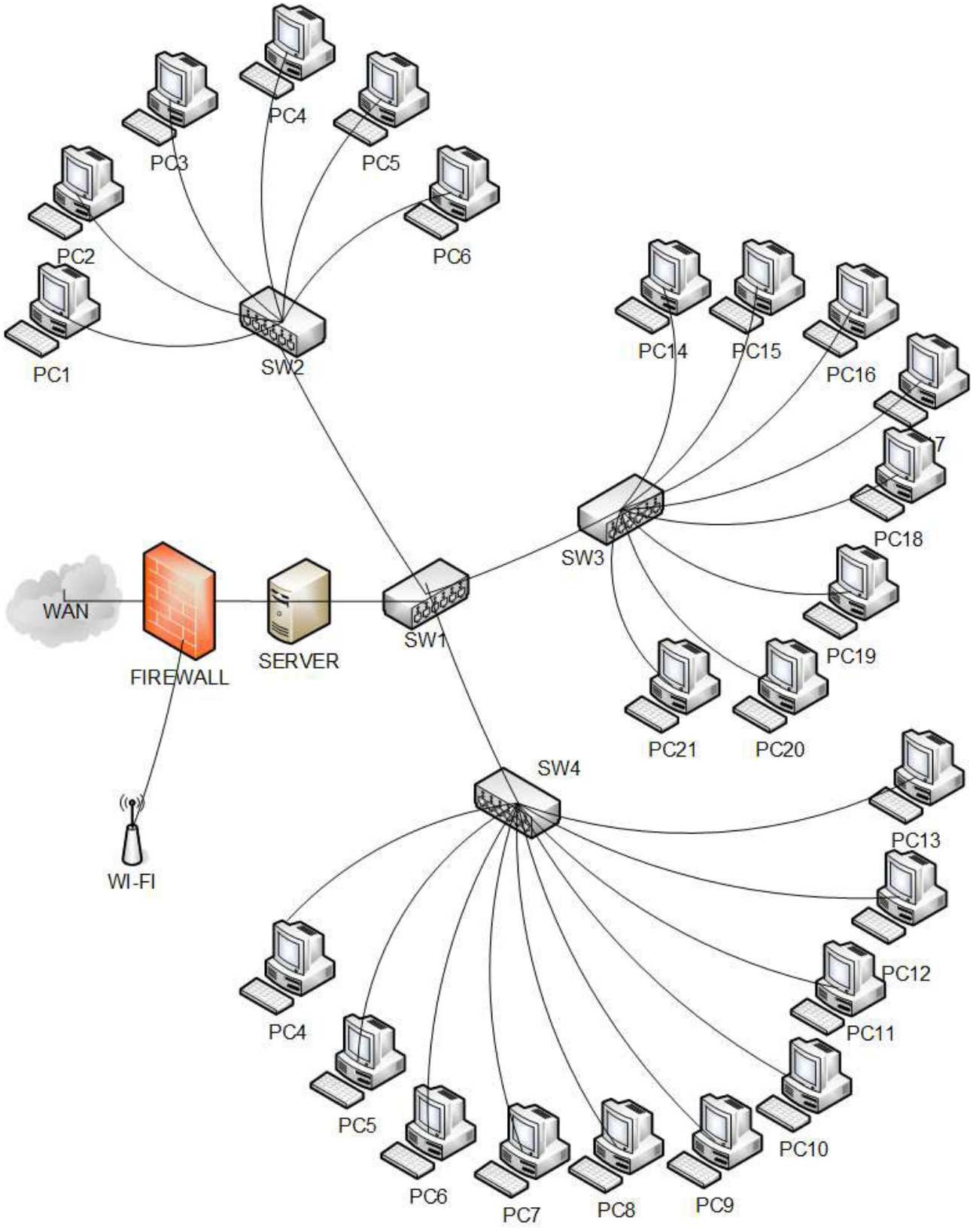

Рисунок 1.1 - Схема мережевих підключень підприємства (частина 1)

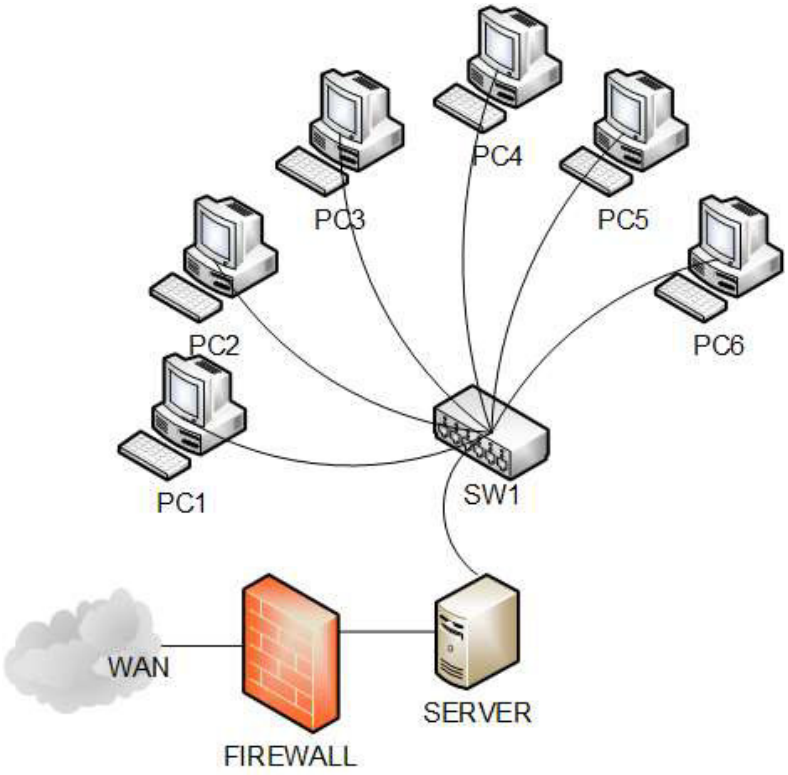

Рисунок 1.2 - Схема мережевих підключень підприємства (частина 2)

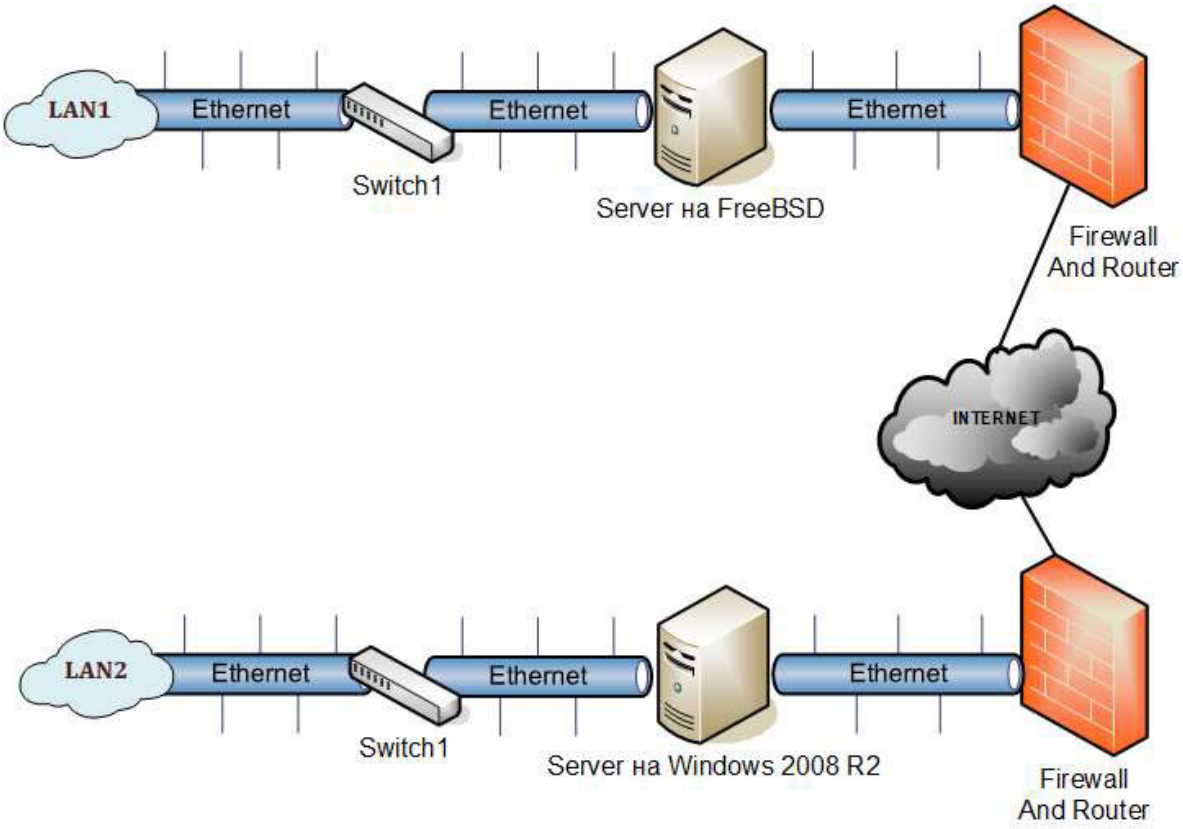

Рисунок 1.3 - Підключення через глобальну мережу

В ході проходження практики потрібно було створити захищене реплікацій не з’єднання між серверами, які базувались на архітектурно різних ОС. В ході досліджень виявилось неможливим передавати інформацію якимось іншим чином, крім передачі даних через Internet. Так як служби каталогів обох ОС створенні на основі серії стандартів X.500, то під час вивчення протоколу виявилось, що обидві ОС підтримують розгортання реплікацій них з’єднань, коли дані знаходяться в форматі XML. Є також інші варіанти створення синхронізацію серверів (Наприклад: використання розподіленої файлової системи Windows). Всі можливі варіанти проведення реплікацій мають бути розглянуті. Серед слід вибрати той, який дає найвищу швидкість передачі реплікаційних повідомлень, та найбільші можливості для захищення каналу передачі інформації між серверами.

Так як реплікації бувають синхронні та асинхронні, то в ході досліджень буде протестовано обидва варіанти реплікаційного з’єднання та дано оцінку захищеності видам реплікаційних з’єднань, коли процес синхронізації проходить між архітектурно різними ОС.

1.7 Модель порушника

Порушником може бути особа з наступних категорій персоналу підприємства:

- зареєстровані користувачі системи (співробітники фірми);

- співробітники відділів розробки;

- технічний персонал, що обслуговує будинки (прибиральниці, електрики, сантехники й інші співробітники, що мають доступ у будинки й приміщення, де розташовані компоненти АС);

Сторонні особи, які можуть бути порушниками:

1 клієнти (представники різних організацій та громадяни);

2 відвідувачі (запрошені по якому-небудь приводі) представники керуючих організацій (злочинних організацій, іноземних спецслужб) або особи, що діють по їхньому завданню;

3 представники організацій, взаємодіючих з питань забезпечення життєдіяльності організації (энерго-, водо-, теплопостачання й т.п.).

Приймаються наступні обмеження й припущення про характер дій можливих порушників:

1 робота з підбора кадрів і спеціальні заходи виключають можливість створення коаліцій порушників, тобто об'єднання (змови) і цілеспрямованих дій по подоланню підсистеми захисту двох і більше порушників.

Таблиця 1.1 – Категорії порушників

Позначеуня	Визначення категорії	Рівень загроз
Внутрішні		
ВН1	Технічний персонал, що обслуговує будівлю й приміщення (електрики, сантехники, прибиральниці й т.п)	1
ВН2	Користувачі програм (працівники організації).	3
Зовнішні		
ЗП1	Будь-які особи, які перебувають за межами контрольованої зони	1
ЗП2	Відвідувачі, клієнти	2
ЗПЗ	Представники організацій, які взаємодіють із питань технічного забезпечення й експлуатації будинків органів ГПС України (энерго-, водо-, теплопостачання тощо)	3
ЗП4	Хакери	4

Примітка. ВН1…2 – внутрішні порушники, ЗП1…3 – зовнішні порушники.

Самий високий рівень загрози становлять хакери (таблицю 2.1). Це пов’язано з тим, що в роботі мережі присутній незахищений канал передачі

інформації(інформаційна система є представником АС 3-ого класу), тож через даний канал передається різного роду інформація, яка має відкритий та конфіденційний характер.

Моє завдання – захистити канал передачі інформації. І якщо самими небезпечними порушниками в моїй системі є хакери, то особливу увагу потрібно приділити саме захисту від хакерських атак (скануванню портів, сніфінгу та ін.). Особливу увагу потрібно приділити захисту конфіденційних даних користувачів. Для цього необхідно ввести в роботу криптографічні протоколи захисту каналів передачі інформації (SSL, SSH). Для того, щоб забезпечити неможливість реалізації інших загроз в системі потрібно розробити модель загроз та профіль захищеності, які дозволять виявити основні загрози в системі та ідентифікувати їх, а також визначити основні критерії захищеності, які необхідно реалізувати в даній системі.

1.8 Модель загроз

По відношенню до інформації, що обробляється на серверах, та до користувачів серверів існує ряд загроз, відображених в таблиці 2.3.

Таблиця 1.2 - Загрози безпеки інформації серверів

Загроза	Вплив
Антропогенні загрози	
Порушення властивостей цілісності інформації в наслідок несанкціонованої модифікації даних зловмисником.	На інформацію, до якої отримає доступ зловмисник.
Порушення властивостей конфіденційності інформації в наслідок захоплення облікових записів користувачів серверів.	На інформацію, до якої отримає доступ зловмисник.
Порушення властивостей доступності інформації в наслідок підвищення навантаження на сервери.	На інформацію, яка призначена до розповсюдження.
Розповсюдження незапрошеної інформації.	На користувачів серверів.

Продовження таблиці 1.2

Загроза	Вплив
Техногенні загрози	
Відмови і збої обладнання, яке приймає, обробляє, зберігає й передає інформацію.	На інформацію, яка обробляється обладнанням, що відмовило.
Відмови і збої активного мережевого обладнання.	На інформацію, яка передавалася обладнанням, що відмовило.
Саботаж роботи інформаційної системи за допомогою програмних засобів (віруси, шкідливі скрипти та ін.).	На інформацію, до якої вдасться отримати доступ шкідливій програмі.
Стихійні загрози	
Руйнування технічних засобів, що обробляють інформацію, внаслідок стихійних впливів (землетруси, пожежі, електромагнітні поля та блискавки).	На інформацію, що обробляється засобами, які піддаються зовнішнім впливам.

1.9 Вибір профілю захищеності серверів з встановленою службою каталогів

Згідно з рекомендаціями НД ТЗІ 2.5-010-03 та з урахуванням особливостей надання доступу до інформації, яку необхідно синхронізувати між двома серверами, типових характеристик середовища функціонування та особливостей технологічних процесів оброблення інформації, а також зважаючи на те, що АС класу «3» розміщується як з боку однієї сторони , а сервер, на який дані будуть реплікуватись – на іншій території.

Взаємодія серверів здійснюється з використанням мереж передачі даних, був обраний наступний профіль захищеності:

{КА-2, КВ-1, КО-1, ЦА-1, ЦО-1, ЦВ-1, ДВ-1, ДР-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1, НВ-1}

Критерій конфіденційності

КЗЗ оцінюваної КС надає послуги з захисту об'єктів від несанкціонованого ознайомлення з їх змістом. Конфіденційність забезпечується такими послугами: базова адміністративна конфіденційність, конфіденційність при обміні, повторне використання об'єктів.

КА-2. Базова адміністративна конфіденційність

Ця послуга дозволяє адміністратору безпеки керувати потоками інформації від захищених об'єктів до користувачів. Політика адміністративної конфіденційності стосується: користувачів усіх категорій, крім, визначених згідно з 6.3.1 "а" НД ТЗІ 2.5-010-03, об'єктів, що містять технологічну інформацію КСЗІ та технологічну інформацію щодо управління АС; системного та функціонального програмного забезпечення, що використовується для актуалізації, захисту загальнодоступної інформації та супроводження серверу; доступу користувачів до окремих видів периферійних пристроїв (принтерів, накопичувачів інформації тощо), використання яких передбачено технологією обробки інформації.

КЗЗ здійснює розмежування доступу на підставі атрибутів доступу користувача і захищеного об’єкта. Доступ до загальнодоступної інформації встановлюється для користувачів усіх категорій.

Призначення атрибутів доступу користувачам і процесам до захищених об'єктів здійснюється адміністратором безпеки на основі аналізу функціональних та службових обов'язків окремих користувачів.

Права доступу до кожного захищеного об'єкта, визначеного політикою безпеки послуги, встановлюються в момент його створення або ініціалізації.

КВ-1. Конфіденційність при обміні

Ця послуга дозволяє забезпечити захист об’єктів від несанкціонованого ознайомлення з інформацією, що міститься в них, під час їх експорту/імпорту через незахищене середовище. Політика мінімальної конфіденційності при обміні стосується користувачів, яким надано право супроводження КСЗІ та

управління АС та об’єктів, які містять технологічну інформацію КСЗІ та технологічну інформацію щодо управління АС під час її передавання між віддаленими компонентами АС. КЗЗ забезпечує захист від безпосереднього ознайомлення з інформацією, що міститься в об’єкті, який передається. КЗЗ забезпечує можливість реєстрації подій, які призвели або можуть призвести до порушення конфіденційності інформації, що міститься в об’єктах, які передаються. Ця послуга забезпечується використанням модифікації базового протоку передачі даних HTTP – HTTPS, який використовує криптографічні протоколи TSL и SSL та TCP-порт 443.

КО-1. Повторне використання об’єктів

Ця послуга дозволяє забезпечити коректність повторного використання розділюваних об'єктів, гарантуючи, що в разі, якщо розділюваний об'єкт виділяється новому користувачу або процесу, то він не містить інформації, яка залишилась від попереднього користувача або процесу.

Політика повторного використання об'єктів, що реалізується КЗЗ, відноситься до всіх об'єктів КС. Перш ніж користувач або процес зможе одержати в своє розпорядження звільнений іншим користувачем або процесом об'єкт, встановлені для попереднього користувача або процесу права доступу до даного об'єкта будуть скасовані. Перш ніж користувач або процес зможе одержати в своє розпорядження звільнений іншим користувачем або процесом об'єкт, вся інформація, що міститься в даному об'єкті, стає недосяжною.

Ця послуга реалізується завдяки вилученню створених сесій відразу після кінцевої стадії її використання передбаченою КЗЗ.

Критерій цілісності

Цілісність забезпечується дотриманням вимог політики безпеки щодо переміщення інформації до серверу, який буде зберігати мої данні. Правильне (допустиме) переміщення визначається як переміщення інформації до об’єкта від авторизованого серверу.

ЦА-1. Мінімальна адміністративна цілісність.

Ця послуга дозволяє керувати потоками інформації від одного серверу через захищений канал до авторизованого іншого серверу. Політика мінімальної адміністративної цілісності стосується: користувачів усіх категорій; загальнодоступної інформації, яка зберігається на сервері; файлової системи та функціонального ПЗ, що використовується для актуалізації, захисту загальнодоступної інформації та супроводження кожного з серверів.

КЗЗ здійснює розмежування доступу на підставі атрибутів доступу користувачів і захищених об’єктів. Розмежування доступу здійснюється на рівні надання (встановлення заборони) користувачеві(службі) прав проводити реплікації.

Право визначати створювати реплікаційне з’єднання надається тільки одній службі в системі(не зважаючи на архітектуру). В Windows це «Адміністратор реплікацій», а в Unix – «Демон реплікації».

ЦО-1. Відкат

Ця послуга забезпечує можливість відмінити окрему операцію або послідовність операцій і повернути захищений об'єкт після внесення до нього змін до попереднього наперед визначеного стану. Політика обмеженого відкату стосується користувачів, яким надано право супроводження КСЗІ та управління АС; об’єктів, які містять публічну інформацію; функціонального програмного забезпечення, що використовується для актуалізації, Якщо стосовно якогось з об’єктів зазначених категорій в процесі обробки не передбачається можливості його модифікації, політика послуги на нього не розповсюджується.

Ця послуга забезпечується завдяки використанню резервних копій даних. Відкат в системі можуть виконувати:

– системний адміністратор(за власними потребами, якщо його не влаштовує стан тієї чи іншої інформації в даний момент часу);

– служба проведення реплікацій в автоматичному режимі(якщо та чи інша транзакція не виконалась).

ЦВ-1. Цілісність при обміні

Ця послуга дозволяє забезпечити захист потоку репликаційних від несанкціонованої модифікації інформації, яка передається між двома серверами в момент синхронізації через незахищене середовище. Політика послуги стосується всіх об’єктів, що передаються.

КЗЗ забезпечує контроль за цілісністю інформації в повідомленнях, які передаються, а також виявляє факти їх несанкціонованого видалення або дублювання. КЗЗ забезпечує можливість реєстрації подій, які призвели до порушення цілісності повідомлень, їх несанкціонованого видалення або дублювання.

Ця послуга забезпечується використанням протоколу HTTPS, що є модифікацією базового протоку передачі даних HTTP, який використовує криптографічні протоколи TSL и SSL та TCP-порт 443. А також використанням протоколу Kerberos.

Критерій доступності

КЗЗ оцінюваної КС надає послуги щодо забезпечення можливості використання КС в цілому, окремих функцій або оброблюваної інформації на певному проміжку часу і гарантує спроможність КС функціонувати у випадку відмови її компонентів. Доступність забезпечується в КС такими послугами: використання ресурсів, відновлення після збоїв.

ДР-1. Використання ресурсів

Ця послуга дозволяє керувати використанням користувачами послуг та ресурсів. Політика використання ресурсів, що реалізується КЗЗ, стосується: користу вачів загальнодоступної інформації; адміністратора безпеки та користувачів, яким надано повноваження щодо управління АС; файлової системи; системного та функціонального програмного забезпечення; технологічної інформації щодо управління АС; окремих периферійних пристроїв (принтерів, накопичувачів інформації та ін.); обчислювальних ресурсів АС і передбачає можливість встановлення обмежень на їх використання.

Обмеження щодо використання окремим користувачем або процесом обсягів обчислювальних ресурсів АС або кількості об’єктів встановлюються адміністратором безпеки або користувачами, яким надано повноваження щодо управління АС. Запити на зміну встановлених обмежень повинні оброблятися КЗЗ тільки в тому випадку, якщо вони надходять від зазначених користувачів.

Ця послуга забезпечується завдяки можливосям операційних систем щодо розмежування прав доступу до ресурсів та об’єктів системи.

ДР-1. Відновлення після збоїв

Політика відновлення після збоїв, що реалізується КЗЗ, стосується: системного та функціонального програмного забезпечення; засобів захисту інформації та засобів управління КСЗІ; засобів адміністрування та управління обчислювальною системою АС; служб реплікації на обох серверах; транзакцій – і гарантує повернення АС у відомий захищений стан після відмов або переривання обслуговування, спричинених помилковими діями користувачів, неврахованою функціональною недостатністю програмного та апаратного забезпечення (наприклад, можливою наявністю не виявлених під час проектування програмного забезпечення, неполадок), іншими непередбачуваними ситуаціями.

Політика відновлення, яка реалізується КЗЗ, визначає множину типів відмов серверів і переривань обслуговування, після яких можливе повернення у відомий захищений стан без порушення політики безпеки. Для кожної з відмов чітко визначені і задокументовані рівні відмов, у разі перевищення яких необхідна повторне налаштування серверу.

Ця послуга реалізується безпосередньо налаштованою системою проведення реплікацій.

Критерій спостереженості

КЗЗ надає послуги щодо забезпечення відповідальності користувача за свої дії і щодо підтримки спроможності КЗЗ виконувати свої функції. Спостереженість забезпечується в КС такими послугами: реєстрація, ідентифікація і аутентифікація, достовірний канал, розподіл обов'язків,

цілісність КЗЗ, самотестування, ідентифікація і аутентифікація при обміні, аутентифікація відправника, аутентифікація отримувача.

НР-2. Реєстрація

Ця послуга дозволяє контролювати небезпечні відповідно до політики безпеки серверів дії користувачів всіх категорій із захищеними об’єктами.

Політика реєстрації стосується: користувачів усіх категорій; системного та функціонального програмного забезпечення, що використовується для актуалізації, захисту інформації та супроводження технологічної інформації КСЗІ та технологічної інформації щодо управління АС. КЗЗ повинен забезпечувати реєстрацію всіх подій, які мають безпосереднє відношення до безпеки.

Реєстрація всіх подій, що мають безпосереднє відношення до безпеки, здійснюється в журналі реєстрації, який містить інформацію стосовно дати, часу, місця, типу і наслідків зареєстрованої події (успішність/неуспішність), ім’я (IP-адресу) та ідентифікатор причетного до цієї події користувача. Реєстраційна інформація повинна бути достатньою для однозначної ідентифікації користувача, процесу або об'єкта, що мали відношення до кожної зареєстрованої події.

Реалізується за допомогою технологій операційних систем.

НИ-2. Ідентифікація і аутентифікація

Ідентифікація і аутентифікація дозволяють КЗЗ визначити і перевірити особу суб’єкта, що намагається одержати доступ до захищених об’єктів, пов’язаних з процесами реплікації.

Реалізується за допомогою модулів операційної системи, які відповідають за аутентифікацію та ідентифікацію користувачів і служб.

НО-1. Розподіл обов’язків

Ця послуга дозволяє розмежувати повноважень користувачів, визначивши категорії користувачів з певними і притаманними для кожної з категорій функціями (ролі). Послуга призначена для зменшення потенційних

збитків від навмисних або помилкових дій користувачів і обмеження авторитарності керування АС.

Політика розподілу обов’язків, що реалізується КЗЗ, стосується користувачів усіх категорій і визначає такі ролі:

– адміністратор;

– адміністратор реплікацій;

– користувач, якому надано право доступу до певних видів інформації (публічної, технологічної, системного та функціонального ПЗ).

Кількість користувачів, які мають доступ до технологічної інформації та системного і функціонального ПЗ мінімізована, щоб обмежити їх коло тільки тими, кому необхідний такий доступ для виконання функціональних обов'язків.

Розмежування повноважень користувачів здійснюється на підставі атрибутів повноважень і захищених об’єктів.

НЦ-1. Цілісність комплексу засобів захисту

Ця послуга визначає міру здатності КЗЗ серверу захищати себе і гарантувати свою спроможність керувати захищеними об’єктами.

Політика цілісності КЗЗ визначає склад КЗЗ, механізми контролю цілісності його компонентів та порядок їх використання. Політика цілісності КЗЗ стосується: адміністратора безпеки; окремих компонентів системного та функціонального програмного забезпечення, які задіяні для реалізації механізмів КЗЗ; засобів захисту інформації, а також технологічної інформації КСЗІ – і забезпечує взаємодію зазначених об’єктів. Політика реалізації послуги гарантує, що всі послуги безпеки доступні тільки через інтерфейс КЗЗ і всі запити на доступ до захищених об'єктів контролюються КЗЗ. Обмеження, недотримання яких може призвести до надання послуг в обхід інтерфейсу КЗЗ і порушення цілісності КЗЗ, описуються і документуються.

НТ-1. Самотестування

Самотестування дозволяє КЗЗ перевірити і на підставі цього гарантувати правильність функціонування і цілісність певної множини функцій захисту серверів.

НВ-1. Ідентифікація і аутентифікація при обміні

Ця послуга дозволяє компонентам КЗЗ серверів здійснити взаємну ідентифікацію, перш ніж розпочати взаємодію. Це досягається за допомогою використання кореневого серверу сертифікації на базі протоколу Kerberos.

1.10 Висновок

В даному розділі було розглянуте питання захищеності реплікацій в гетерогенній системі, а також сформовані основні задачі для можливості проведення реплікацій між різноархітектурними ОС та способи їх захисту.

РОЗДІЛ 2. СПЕЦІАЛЬНИЙ РОЗДІЛ

2.1 Визначення необхідного формату повідомлень для правильного обміну повідомленнями між двома серверами

Проблематика в тому, що потік байтів для репликацій, який передається через TCP/IP при має різні формати в різних операційних системах. Windows формує спеціальну форму пакетів (рисунок 2.1), через які передає репликаційні повідомлення. База даних користувачів зберігається в спеціальному файлі.

ID користувача, який проводить реплікацію	Ім’я користувача, який проводить реплікацію
Чи потрібен сервер сертифікатів?	Місце знаходження серверу(IP-адреса)
Наявність сертифікованого серверу, який передає інформацію	ID користувача, для якого впроваджуються зміни.
Ім’я користувача, для якого впроваджуються зміни.	Пароль користувача, для якого впроваджуються зміни.
Інформація, що підлягає зміні, видаленню, чи іншим діям з боку служби каталогів.	

Рисунок 2.1 – Формат пакету реплікаційного повідомлення Windows

Система FreeBSD використовує службу Replication Demon для передачі інформації про реплікації. Цю службу можливо конфігурувати за допомогою зміни спеціального конфігураційного файлу, але вона не підтримує пакети реплікацій Windows. Обслуговують систему реплікацій в системах Unix спеціальні сервери реплікацій, які є вмонтованими в ОС. Зберігається база користувачів в СУБД PostgreeSQL, яка є вмонтованою в систему. Тож розглянемо можливості з’єднання операційних систем між собою.

Таблиця 2.1 – Можливості з’єднання операційних систем, які базуються на різних платформах між собою

Формат передачі реплікаційних повідомлень	Можливе використання для обох операційних систем	Неможливе використання, так як одна з ОС не підтримує його.
Стандартні пакети реплікації в Windows	Не можливе	Неможливе використання тому, що в ОС FreeBSD не вміє обробляти дані пакети на рівні ядра операційної системи. Можна скачати вихідні коди операційної системи і переписати модулі так, щоб пакети з Windows могли розпізнаватись в FreeBSD. Але для цього необхідно отримати ліцензію GNU Security Licence.
Передача XML-повідомлень, які несуть реплікаційні дані	Можливе налаштування модуля проведення реплікацій операційної системи Windows для передачі реплікаційних повідомлень, інкапсульованих в XML. Такі повідомлення можна передавати через HTTP. З іншого боку FreeBSD можна конфігурувати для прийому XML потрібного формату	Не можливе
Можливість роботи з Distributed File System	Не можливе	Розподілена файлова система від Microsoft – це дуже гарне вирішення проблема реплікацій, але вона не підходить для вирішення проблеми тому, що FreeBSD не зберігає в файлах дані про користувачів, а використовує СУБД.

Тож для синхронізації двох операційних систем використаємо XML формат, при цьому модернізуємо сервер реплікацій на BSD-системі. Безпеку незахищеного каналу організуємо за допомогою кореневого серверу сертифікатів Windows системи.

2.2 Питання захищеності реплікацій

Для забезпечення безпечного обміну реплікаційними даними можна використати сервер сертифікації Windows. Центр сертифікації – це компонент глобальної служби каталогів, що відповідає за управління криптографічними ключами користувачів.

Відкриті ключі та інша інформація про користувачів зберігається центрами сертифікації у вигляді цифрових сертифікатів, що мають наступну структуру: серійний номер сертифіката; об’єктний ідентифікатор алгоритму електронного підпису; термін дії сертифіката; ім’я власника сертифіката (ім’я користувача, якому належить сертифікат); відкриті ключі власника сертифіката (ключів може бути декілька); об’єктні ідентифікатори алгоритмів, асоційованих з відкритими ключами власника сертифіката; електронний підпис, згенерована з використанням секретного ключа посвідчує центру (підписується результат хешування всієї інформації, що зберігається в сертифікаті).

В основі серверу сертифікації лежить протокол Kerberos. Це мережевий протокол аутентифікації, що дозволяє передавати дані через незахищені мережі для безпечної ідентифікації. Орієнтований, в першу чергу, на взаємну аутентифікацію - обидва користувача через сервер підтверджують особистості один одного.

Варіанти налаштування безпечного з’єднання між серверами на базі кореневого серверу сертифікації приведено далі.

2.3 Тестування безпеки з’єднання

Виконується за допомогою потужного сніферу EffeTech, фірмою Sun для тестування захищеності власних мережевих з’єднань. Вибір даного сніферу ґрунтується на дослідженнях авторитетного інтернет-порталу, присвяченому інформаційній безпеці SecurityLab.

В ході дослідження було проаналізовано 100 різних сніферів на предмет ефективного сканування мережі та можливості працювати з даними, які зашифровані за допомогою хеш-функції md5. EffeTech зайняв перше місце після проведення досліджень. Було виявлено, що даний сніфер для кожного потоку TCP-з'єднання виділяє окремий потік, який аналізується за допомогою потужної системи сканування. Таким чином жоден пакет, який проходе в мережі від вузла, який сканується, не пропускається сніфером.

Щодо даних, які закриті md5-хешем, то EffeTech може за допомогою серверів, на яких зберігаються деякі розшифровані md5-повідомлення.

Список серверів наведений нижче.

Також сніфер може виконувати розшифрування md5 за допомогою райдужних таблиць (RainbowCrack), по словарям та брутфорс методом.

2.4 Алгоритм налаштування серверів

1 Налаштовуємо передачу реплікаційних повідомлень на ОС Windows Server 2008 R2 за допомогою методики, яка була приведена на сайті розробника – Microsoft.

2 При налаштуванні Windows Server 2008 R2 вказуємо, що використовується XML формат передачі повідомлень.

3 Налаштовуємо передачу реплікаційних повідомлень на FreeBSD 9.0 за допомогою методики фірми IBM.

4 При налаштуванні використаємо формат приймання реплікаційних повідомлень XML.

5 Модернізуємо сервер реплікацій FreeBSD 9.0 за допомогою знання в мовах програмування.

6 Налаштуємо в Windows Server 2008 R2 кореневий центр сертифікації.

7 Налаштуємо в FreeBSD прийом сертифіків.

2.4.1 Налаштування захищених реплікацій в Windows Server 2008

Вперше з'явившись в Windows Server 2000, технологія Active Directory (це насамперед транзакційна база даних, що містить інформацію про об'єкти вашої мережі, глибоко інтегрована з системою безпеки Windows) більше ніж за

дев’ять років зазнала деяких змін. Але навіть в Windows Server 2008 R2 працює на добре зарекомендувавшому себе движку Extensible Storage Engine (ESE). Якщо вірити розрахункам масштабованості, даного рішення повинно вистачити навіть мережам з кількома мільйонами об’єктів, а вирости база може до 16 терабайт. Серцем Active Directory є файл NTDS.DIT (рисунок 2.2), в якому, власне кажучи, вся інформація і зберігається. При додаванні другого і наступних контролерів домену відбувається створення копії даного файлу і розміщення її на введеному в дію новому контролері. Можна зробити чіткий висновок: кожен контролер домену зберігає свою версію файлу NTDS.DIT.

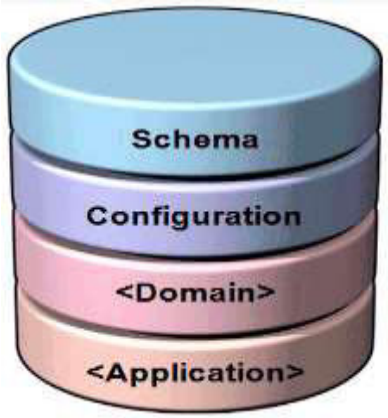

Рисунок 2.2 – Схема зберігання даних в Active Directory

Фізично NTDS.DIT – це просто один файл, але логічно він складається з декількох розділів (іноді їх називають контекстами іменування або контекстами реплікації), кожен з яких містить певну інформацію і реплікується по-своєму.

Розділ Schema зберігає в собі схему Active Directory, яка описує, які об’єкти можуть бути створені і що вони собою будуть представляти.

Змінюється найрідше, як правило, при переході контролерів на нову операційну систему або при установці в організації поштової системи Exchange. Процес зміни бази даних в контексті схеми найчастіше називають розширенням схеми, і це не випадково, тому скасувати дані зміни (наприклад, видалити створені в цьому контексті об’єкти) неможливо. Реплікація розділу здійснюється на всі контролери домену в лісі Active Directory. Єдиний розділ,

чия реплікація не є мультімастерной. Реплікація розділу Schema завжди одностороння і виконується від контролера домену, який володіє роллю FSMO «Господар Схеми», на все залишилися контролери домену. (Згодом залишилися контролери домену реплікується інформацію своїм репликационной партнерам, але джерелом цього ланцюжка реплікації завжди буде «Господар схеми».)

Розділ Configuration – містить інформацію про конфігурацію Active Directory. Він описує, які і скільки доменів створено, як вони між собою пов'язані, скільки існує сайтів, які сервіси доступні в організації і просто системні налаштування служби каталогів, такі як квоти, політики LDAP-запитів, правила неточного пошуку, розв'язання імен об'єктів. Розділ реплікується між усіма контролерами доменів в лісі, може бути змінений на будь-якому контроллері домена. Зміни даного розділу пов'язані з конфігуруванням і настроюванням самої ActiveDirectory.

Розділ Domain - або доменний. Усі облікові записи користувачів, групи безпеки, організаційні підрозділи, об'єкти комп'ютерів і принтерів створюються і зберігаються в даному розділі. Реплікується розділ тільки між контролерами домену в тому домені, до якого належить контролер. Виходить, що в організаціях, що мають декілька доменів, в кожному з них буде свій розділ Domain, контролерами домену в тому домені, до якого належить контролер. Виходить, що в організаціях, що мають декілька доменів, в кожному з них буде свій розділ Domain.

Розділи Application - опціональні розділи. Зона реплікації залежить від налаштування. Як правило, створюється два Application-розділу, це ForestDNSZones і DomainDNSZones.

ForestDNSZones - зберігає SRV і CNAME записи для лісу AD і реплікується на всі контролери домену в лісі, що є DNS-серверами.

DomainDNSZones - містить DNS-записи для зони домену. Реплікується на всі контролери домену з встановленим на них DNS-сервером.

Налаштування реплікації виконується за допомогою оснастки «Сайти та служби» Active Directory (рисунок 2.3).

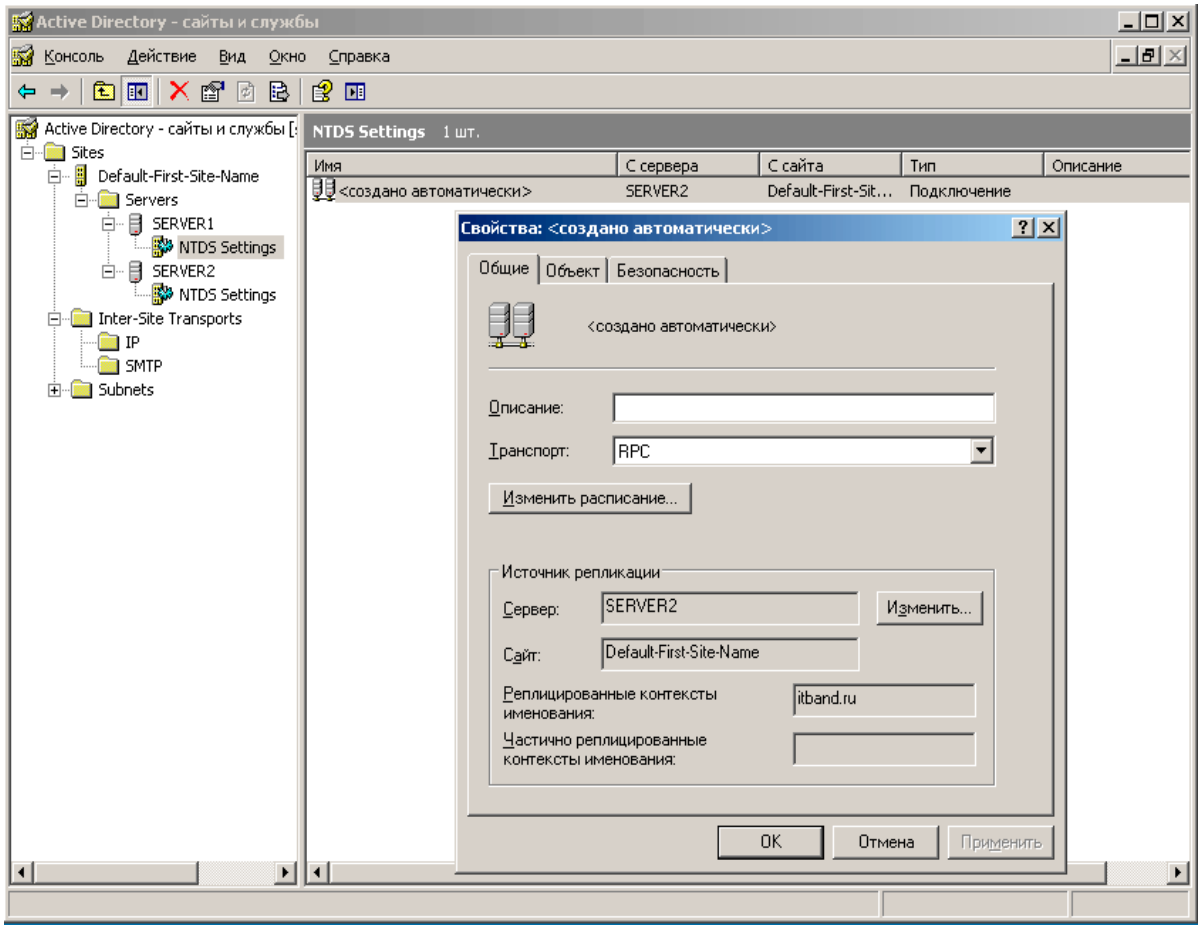

Рисунок 2.3 – оснастка «Сайти та служби»

На поверхні NTDS Settings правою клавішею мишки і обераємо пункт «Створити реплікацію». Отримаємо вікно, яке зображене на рисунку 2.4. Тут задаємо час проведення реплікації:

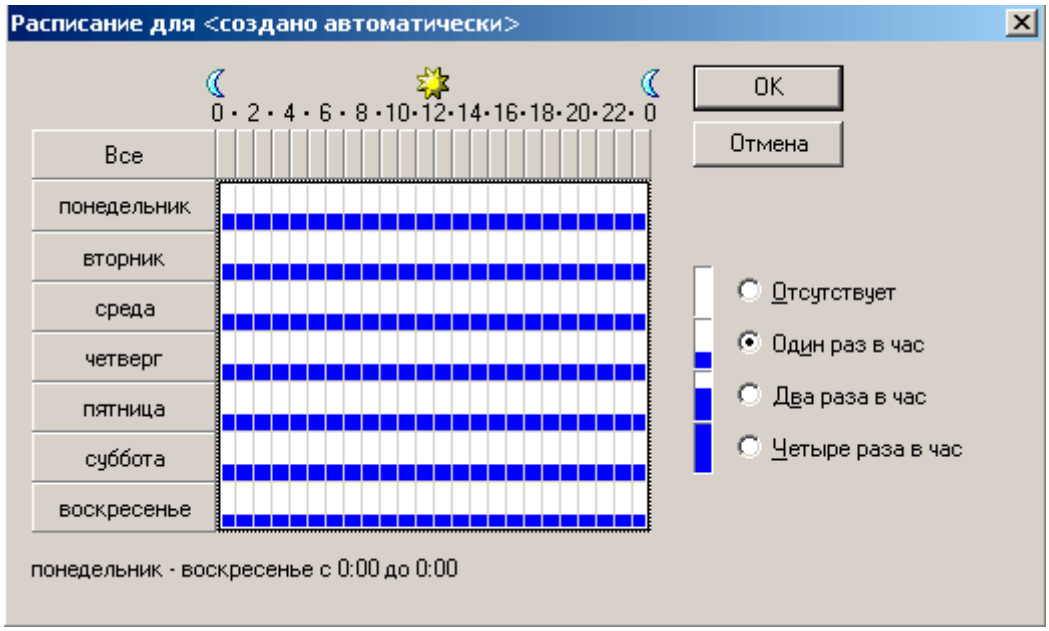

Рисунок 2.4 – Задання часу для проведення реплікацій

В наступному вікну налаштування запросить синхронні чи асинхронні реплікації потрібно проводити, а також формат даних, в який будуть запаковуватись реплікаційні повідомлення. Потрібно обрати XML. Обравши потрібне тиснемо «ОК». Процес реплікації налаштовано і готово для використання. Є інший спосіб ручного запуску реплікації в будь-який момент часу. Для цього слугує вмонтована в ОС утиліта Repadmin.

Запуск реплікації за домогою Repadmin здійснюється командою «Repadmin /syncall /Serv1/Serv2», де Serv1, Serv2 – це IP-адреси потрібних серверів.

За допомогою Repadmin можна:

– викликати інформацію про конкретного користувача (рисунок 2.5);

– продивитись статистику реплікацій;

– продивитись і корегувати список серверів, які задіяні в реплікації (рисунок 2.6).

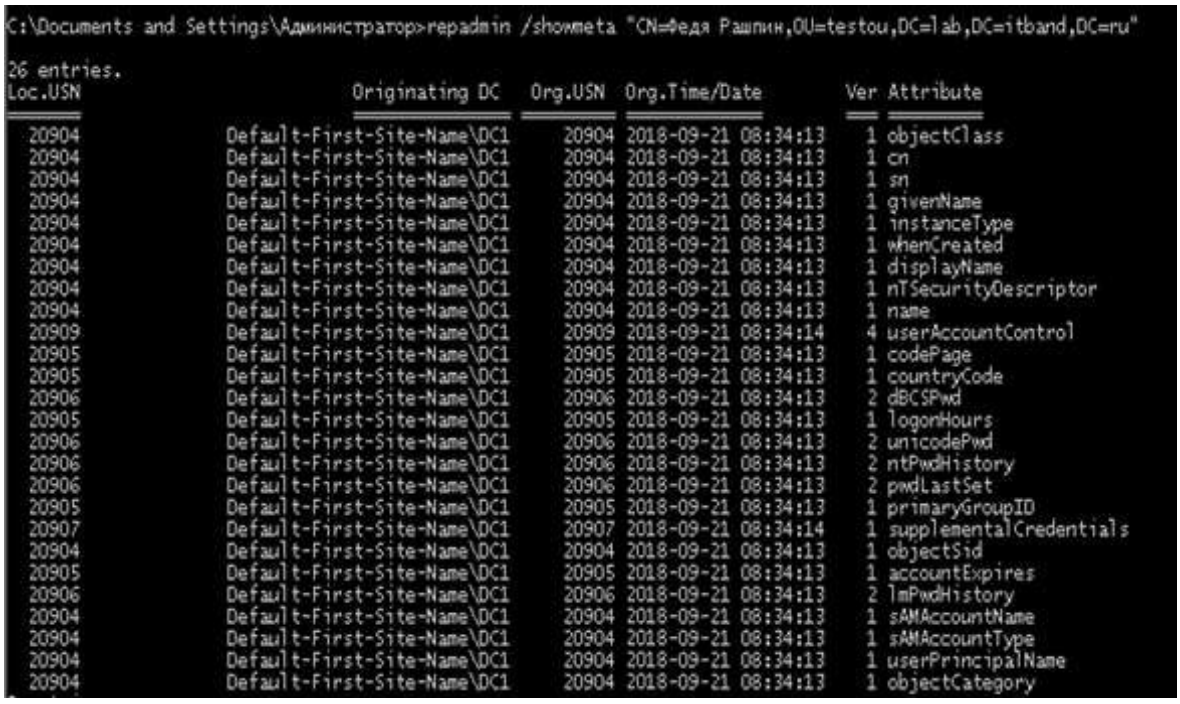

Рисунок 2.5 – Інформація про користувача, яка отримана за допомогою програми Repadmin

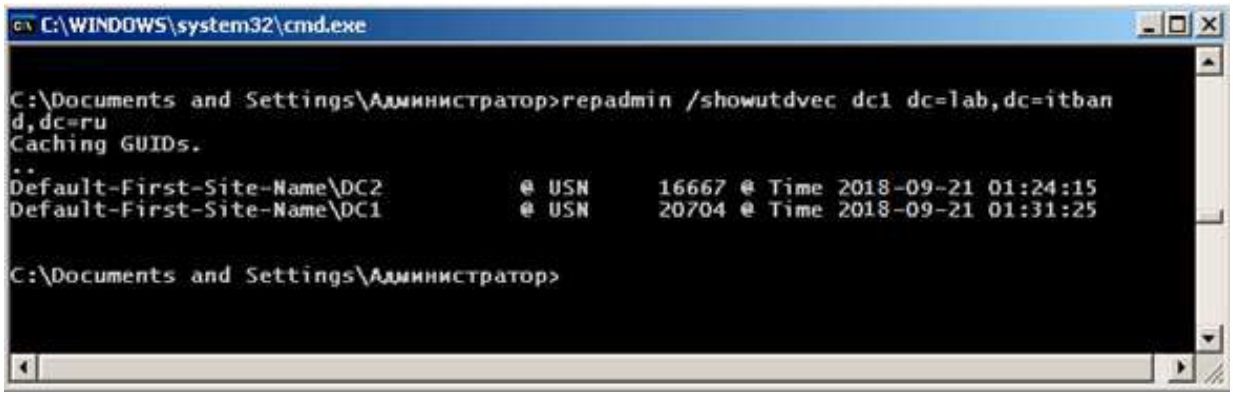

Рисунок 2.6 – Інформація про сервери, що задіяні в реплікації, яка отримана за допомогою програми Repadmin

Примітка. Для забезпечення захищеності серверу необхідно видалити програму Repadmin з пакету ПЗ серверу. На даний час існує багато вразливостей в ОС від фірми Microsoft, які дозволяють отримати доступ до командної консолі з правами адміністратора в Windows.

Базові міри, які приймаються для забезпечення захищеності реплікацій з боку Windows Server 2008 R2 (їх налаштування виконується по методиці, запропонованій фірмою Microsoft):

– При запуску кожного сеансу реплікації необхідно використовувати новий обліковий запис Windows (це можна зафіксувати в налаштуваннях політики безпеки), а у відношенні всіх з’єднань реплікації застосовувати перевірку автентичності Windows;

– Налаштувати всі облікові записи на сервері (за допомогою яких проводиться реплікація) знаходяться в списку доступу до публікації (PAL-«білий список» серверів, що є учасниками реплікації);

– Налаштувати файервол, щоб запобігти атакам інсайдерів;

– Організувати NAT (для того, щоб весь трафік проходив через сервер).

Для того, щоб реплікації були справді захищеними необхідно позбавити всі облікові записи, що приймають участь в процесі реплікацій всіх прав доступу і залишити їм права «Адміністратор реплікацій».

Це можна зробити або налаштувавши для всіх користувачів з облікових записів яких проводяться реплікації в оснастці «Користувачі», або ж за допомогою команди програми Repadmin «repadmin –I».

2.4.2 Налаштування захищених репликацій в ОС FreeBSD

Дана серверна операційна система використовує для створення реплікаційного середовища різні RDBMS, які може використовувати як безпосередньо на сервері, на якому вона встановлена, так і використовувати розмежоване середовище для проведення реплікації(наприклад: за допомогою веб-сервісів, які є інтегрованими в клієнт до бази даних).

В BSD системах використовується потокова реплікація –вид реплікації , який використовує багато потоковий пул з’єднань для того, щоб підвищити швидкість та надійність проведення реплікацій.

Для базових налаштувань реплікацій необхідно ви 2 сервери (c1 – основний, або master і c2 – допоміжний, або ж slave). Тут будемо оперувати поняттям WAL – це одинична частина інформації, яка має бути реплікованою (в BSD системі має представлення в виді лог-файлу, який буде записуватись в конкретну базу даних). Розглядаємо файл repl.conf, який знаходиться в папкі /etc/ на системному диску.

Для даного випадку, коли сервер на основі FreeBSD має бути прийомною стороною розглянемо тільки Slave налаштування.

– max_wal_senders (ціле) – Визначає максимальну кількість одночасних підключень від серверів потокового резервного копіювання (тобто максимальна кількість одночасно запущених процесів відправників WAL). Значення за замовчуванням - 0. Цей параметр може бути заданий тільки при старті сервера.

– wal_sender_delay (ціле) – Визначає затримку між циклами активності процесів відправників WAL. У кожному циклі відправник посилає всі WAL WAL, зібрані з останнього циклу, на резернвий сервер. Після цього процес засинає на wal_sender_delay і цикл починається заново. Сон переривається підтвердженням транзакції, так що підтверджені транзакції відправляються на резервні сервера відразу після підтвердження, незалежно від цієї настройки.

Значення за замовчуванням – 1 секунда (1 сек.) Зверніть увагу, що не багатьох системах різниця між значеннями цього параметра повинна бути не менше 10 мілісекунд; завдання значення не кратного 10 буде мати той же ефект, що і найближчим більше значення, кратне 10. Цей параметр може бути заданий або в repl.conf, або в командному рядку.

– wal_keep_segments (ціле) – Визначає мінімальне число файлів сегментів логу, збережених в каталозі pg_xlog, у разі якщо резервний сервер повинен отримати їх для потокової реплікації. Кожен сегмент звичайно дорівнює 16 мегабайт. Якщо резервний сервер відстає від майстра більше ніж на wal_keep_segments сегментів, то майстер видалить потрібні сегменти WAL і в такому випадку підключення реплікації буде перервано. (Проте, резервний сервер може потім відновити ці сегменти з архіву, якщо відбувається архівування WAL.) Дана настройка задає тільки мінімальна кількість сегментів, що зберігаються в pg_xlog; система може зберігати і більше кількість сегментів для архівування WAL або відновлення з певної точки, яка зберігається в базі даних. Якщо wal_keep_segments = 0 (значення за замовчуванням), то система на зберігає додаткові сегменти для резервних серверів, так що кількість доступних сегментів WAL є функцією місця попереднього Чекпойнт і статусу архівування WAL. Цей параметр не має впливу на restartpoints. Цей параметр може бути заданий або в repl.conf, або в командному рядку.

– vacuum_defer_cleanup_age (ціле) - Визначає число транзакцій, за якими VACUUM і HOT (перше – дані рідко використовуються, HOT - навпаки) оновлення будуть визначати очистку версій мертвих рядків. Значення за замовчуванням - 0 транзакцій, що означає, що версії мертвих рядків будуть видалені як тільки з’явиться можливість, тобто як тільки вони не видно жодної відкритої транзакції. Ви можете захотіти встановити нульове значення на майстра, який підтримує "гарячі" резервні сервера, як це описано в технічній документації по даній ОС.

Це надає більше часу для запитів до резервних серверів, щоб при цьому не виникли конфлікти через раннього видалення рядків. Однак, оскільки це значення вимірюється в термінах кількості операцій запису транзакцій на основному сервері, важко передбачити, скільки часу це надасть для резервного сервера. Цей параметр може бути заданий або в repl.conf, або в командному рядку.

– replication_timeout (ціле) – Сполуки реплікації, неактивні більше ніж заданий тут час у мілісекундах, розриваються. Це корисно для майстра для виявлення проблем в мережі і збою резервного сервера. Значення, рівне 0, відключає цей механізм. Цей параметр може бути заданий або в repl.conf, або в командному рядку. Значення за замовчуванням - 60 секунд.

– replication_db_url – Так як BSD використовує для зберігання бази даних користувачів сторонні ресурси(наприклад технології віддаленого користування функціями з сучасних мов програмування, або за допомою прямого доступу до бази данних). В це поле користувач може записати шлях до бази даних, яка знаходиться або на станції серверу, або ж віддалена від нього.

– Wsdl_url – В нових версіях BSD з’явилась можливість використовувати сучасні технології віддаленого користування функціями. Хоча в мануалах і дуже слабо описаний даний підхід, я вважаю, що він є безпечнішим , ніж використання просто віддаленої БД. Він дозволяє використовувати протокол https для обміну трафіком. А також централізувати підхід до обробки інформації. Для роботи з даним методом проведення реплікацій також потрібно відредагувати файл rep_wsdl_prop(потрібно вказати назву функції , яка буде отримана парсером із xml документу).

– Use_JVM_for_compile – так, як для отримання XML від Windows нам потрібно його обробити в формат, необхідний для роботи в FreeBSD, то його необхідно скомпілювати. FreeBSD сама вміє компілювати необхідні коди і використовувати їх потім. Даний параметр дозволяє використовувати коди на мовах програмування Java і Scala для написання процедур, виконання яких приводить до робот з технологіями віддаленого виклику функцій.

FreeBSD 9.0 використовує в якості серверу реплікацій Oracle Glassfish. Тож для того, щоб отримувати реплікацій ні пакети від Windows нам необхідно створити для цього необхідні умови. Тобто створити парсер, який перетворить XML формат на потрібні дані. Дане ПЗ встановлюється на Glassfish і зберігає дані з HTTP в базу даних.

Код, який демонструє створення таблиці в базі даних для обробки запитів служби реплікації FreeBSD на мові Java:

```
package entities;

import java.io.Serializable;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedQuery;
import javax.persistence.Table;

@Entity
@Table(name = "users")
@NamedQuery(name = "User.getAllUsers", query = "select u
from User u")
public class User implements Serializable {

private static final long serialVersionUID = 1L;

@Id
@Column(name = "USER_ID", unique = true)
@GeneratedValue(strategy=GenerationType.SEQUENCE)
private int id;

@Column(name = "USERNAME")
private String name;
```

```
@Column(name = "PASSWORD")
private String passwd;

@Column(name = "ENABLED")
private boolean enabled;
 public User() {
  }
 public int getId() {
  return id;
  }
 public void setId(int id) {
  this.id = id;
  }
 public String getName() {
  return name;
  }
 public void setName(String name) {
  this.name = name;
  }
 public String getPasswd() {
  return passwd;
  }
  public void setPasswd(String passwd) {
  this.passwd = passwd;
  }
  public boolean isEnabled()
  {
  return enabled;
  }
  public void setEnabled(boolean enabled)
  {
  this.enabled = enabled;
  }}
```

Код, який демонструє вибірки, заміну, створення, видалення користувачів з бази для обробки запитів служби реплікації FreeBSD на мові Java:

```
package beans;
import java.util.List;
import javax.ejb.Stateless;
import javax.persistence.EntityManager;
import javax.persistence.PersistenceContext;
import entities.User;
@Stateless(mappedName = "UserBean")
public class UserBean implements UserActions{
@PersistenceContext (unitName = "EJBProj")
 EntityManager em;
public UserBean() {
}
@Override
public List<User> getAllUsers() {
List<User>     users=em.createNamedQuery("User.getAllUsers",
User.class).getResultList();
return users;
   }
@Override
public User getSingleUser(Integer id) {
return em.find(User.class, id);
}
@Override
public void modifyUserData(User user) {
em.merge(user);
}
@Override
public void addNewUser(User user) {
em.persist(user);
}
@Override
public void removeUser(User user) {
em.remove(user);
```

```
}
public void setEm(EntityManager em)
{
this.em = em;
}}
```

Код, який демонструє приймання XML для обробки запитів служби реплікації FreeBSD на мові Java:

```
package userservice;
import java.util.List;
import javax.ejb.EJB;
import javax.jws.WebService;
import beans.UserActions;
import entities.User;
@WebService
public class UserService
{
@EJB
UserActions userActions;
public List<User> getAllUsers(){
return userActions.getAllUsers();
}
public User getSingleUser(Integer id)
{
return userActions.getSingleUser(id);
}
public void modifyUserData(User user)
{
userActions.modifyUserData(user);
}
public void addNewUser(User user){
userActions.addNewUser(user);
}
public void removeUser(User user){
userActions.removeUser(user);
}}
```

Після налаштування захищеного з’єднання протестуємо на сервері, чи отримані повідомлення з створеними користувачами (рисунок 2.7).

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
    <S:Body>
        <ns2:getAllUsersResponse xmlns:ns2="http://userservice/">
            <return>
                <enabled>true</enabled>
                <id>1</id>
                <name>user</name>
                <passwd>1</passwd>
            </return>
            <return>
                </ns2:getAllUsersResponse>
            </S:Body>
        </S:Envelope>
```

Рисунок 2.7 – Прийняте повідомлення в форматі XML з обліковими даними створеного користувача

2.4.3 Windows Server 2008 R2. Створення кореневого центру сертифікації

Інфраструктура відкритих ключів Windows Server дозволяє використовувати цифрові сертифікати для підтвердження автентичності власника і дозволяє надійно і ефективно захищати трафік передається по відкритих мережах зв'язку, а також здійснювати з їх допомогою аутентифікацію користувачів. Основою інфраструктури відкритих ключів є центр сертифікації, який здійснює видачу та відкликання сертифікатів, а також забезпечує перевірку їх достовірності.

Цифрові сертифікати дозволяють використовувати шифрування на рівні додатків (SSL/TLS) для захисту веб-сторінок, електронної пошти, служб терміналів і т.п., реєстрацію в домені, міжсерверну аутентифікацію, аутентифікацію користувачів віртуальних приватних мереж (VPN), шифрування даних на жорсткому диску (EFS), а також у ряді випадків обійтися без використання паролів.

Для створення центру сертифікації нам знадобиться сервер, що працює під управлінням Windows Server, який може бути як виділеним, так і

поєднувати роль центру сертифікації з іншими ролями. Після розгортання центру сертифікації можна поміняти ім'я комп'ютера і його приналежність до домену (робочій групі).

В даній дипломній роботі буде використано сервер підприємства з ОС Windows Server, який виконує роль контролеру домена.

Методика розгортання центру сертифікації використана з сайту MSDN.

1 Додаємо роль серверу, як кореневому серверу сертифікації. Обираємо встановлення серверу служб сертифікації Active Directory (рисунок 2.8);

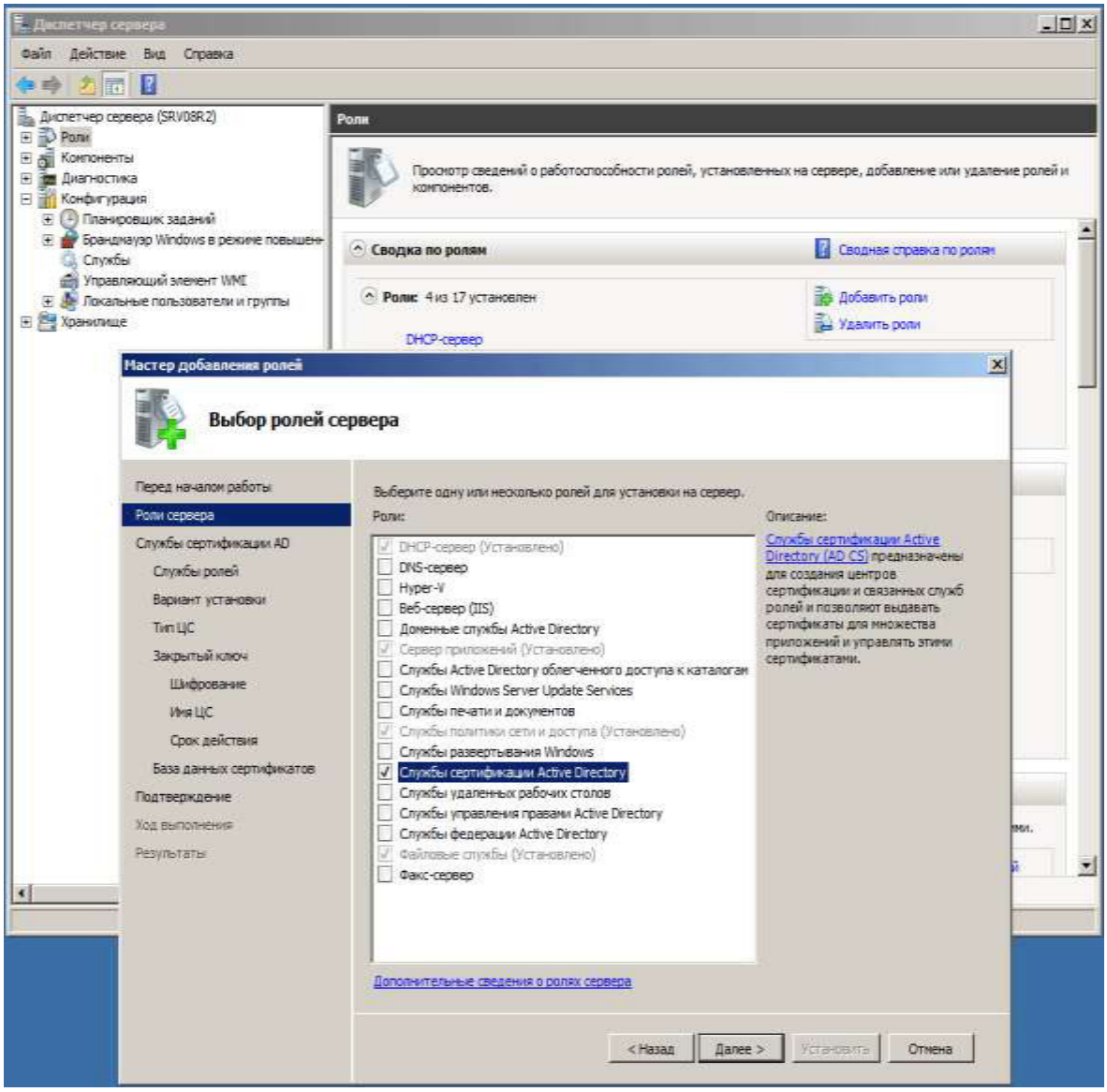

Рисунок 2.8 – Надання серверу ролі кореневого центру сертифікації

2 Після цього в слідуючому вікні вкажімо основні параметри сертифікату:

– Організація: ПП;

– Країна: UA;

– Місто: Dnipro.

В додаткових параметрах налаштування вказуємо сервер якому довірено видавати сертифікат:

– IP: 86.44.13.79;

– Організація: ПП;

– Країна: US;

– Місто: Connecticut;

2.4.4 Налаштування приймання сертифікатів в ОС FreeBSD від Windows Server

Спочатку відредагуємо конфігураційний файл openssl_recive.cnf, який відповідає за налаштування сертифікатів, що приймаються сервером і знаходиться в папкі /etc/ssl/ (рисунок 2.9):

```
[ req_distinguished_name ]
countryName                     = Country Name (2 letter code)
countryName_default             = US
countryName_min                 = 2
countryName_max                 = 2

stateOrProvinceName             = State or Province Name (full name)
stateOrProvinceName_default     = Connecticut

localityName                    = Locality Name (eg, city)
localityName_default            = Connecticut

0.organizationName              = Organization Name (eg, company)
0.organizationName_default      = PP

# configurations for recive certificates from some servers
#1.organizationName             = Second Organization Name (eg, company)
#1.organizationName_default     = World Wide Web Pty Ltd

organizationalUnitName          = Organizational Unit Name (eg, section)
#organizationalUnitName_default =

commonReciveCertServName                        = Common Name (eg, YOUR name)
commonReciveCertServName_default                = Win_Srv
IPAReciveCertddr_srv              = 94.56.13.25
commonName_max                  = 64
```

Рисунок 2.9 – Конфігураційний файл openssl_recive.cnf

В файлі openssl_recive.cnf налаштовуються параметри сертифікатів, які приймаються даним вузлом. Спочатку налаштовується локальні дані ОС, а потом можна налаштувати сервер, від якого можна приймати сертифікати. За замовчуванням поля файлу налаштувань commonReciveCertServName_default, IPAReciveCertddr_srv ідентифікують сервер, з яким буде проводитись обмін сертифікатами.

Далі – сконфігуруємо за допомогою файлу /usr/lib/ssl/misc/CA.conf роботу ОС з прийомом сертифікатів від Windows Server. Для цього вкажемо шлях до конфігураційного файлу openssl_recive.cnf (рисунок 2.10):

```
$SSLEAY_CONFIG="-config /etc/ssl/openssl_recive.cnf";
$CADAYS="-days 365";
$CATOP=".";
```

Рисунок 2.10 – Конфігурація файлу CA.conf

Спробуємо налаштувати з'єднання з сервером на Windows Server і отримати перший сертифікат (Рисунок 2.10):

```
/usr/lib/ssl/misc/CA.conf -newca -serv Win_Srv -ip 94.56.13.25
```

Рисунок 2.11 – Спроба за допомогою команди newca отримати сертифікат

В результаті налаштувань отримаємо файл /usr/lib/ssl/certs/ca.crt, який і є сертифікатом:

```
Certificate: Data: Version: 3 (0x2)
Serial Number: d9:98:4f:55:e0:bb:b3:3c
Signature Algorithm: sha1WithRSAEncryption
Issuer: C=UA, ST=Dnipro, O=PP,
CN=root Certificate Validity Not Before: Oct 11 12:16:26 2018
GMT Not After: Oct 8 12:16:26 2021 GMT Subject: C=US,
ST=Connecticut, O=PP, CN=root Certificate Subject Public Key
Info: Public Key Algorithm: rsaEncryption Public-Key: (1024 bit)
Modulus: 00:d1:d1:0a:11:a3:1e:67:2b:d2:39:3e:ea:bf:44:
04:f9:2a:ae:c4:37:a2:76:8b:fc:de:6c:04:5a:56:
35:0b:12:8e:e6:31:62:5a:88:b4:53:a5:bf:9f:63:
ea:6d:33:f9:4a:84:a5:8b:b1:f3:0b:9e:56:f8:27:
0d:8c:be:1d:76:be:6c:e5:c9:f3:f1:b0:cd:df:79:
b4:0b:05:db:25:15:c1:e5:b3:08:17:af:67:e9:be:
44:a2:ce:ed:9a:6c:16:bb:f6:8c:73:ab:dd:86:5a:
82:73:6c:5e:03:fa:6e:8e:06:07:dd:8e:fb:95:51:
16:4b:91:87:be:15:9e:12:21
```

```
    Exponent: 65537 (0x10001) X509v3 extensions: X509v3 Subject Key
Identifier:1F:1C:A6:43:A2:49:E7:16:49:EC:FD:73:71:72:D7:7F:24:6D:AC:1
7 X509v3
    AuthorityKeyIdentifier:
keyid:1F:1C:A6:43:A2:49:E7:16:49:EC:FD:73:71:72:D7:7F:24:6D:AC:17
X509v3 Basic Constraints:
    CA:TRUE Signature
    Algorithm:sha1WithRSAEncryption
60:f2:cf:c7:52:11:83:c7:ea:b7:ae:68:8a:63:7a:89:5b:d6:
4e:ae:ba:0d:9d:a3:e6:86:07:db:54:77:59:b1:f9:dc:38:bd:
a2:31:b6:18:80:80:3d:e1:20:13:23:28:26:b8:b0:aa:4f:a8:
f7:92:89:13:2a:48:62:29:fb:3c:b7:ab:23:cb:97:ae:7c:21:
15:8e:23:e3:13:a1:e1:0d:85:dc:d0:8d:f7:fc:a5:60:0e:bc:
5d:ea:31:d1:b4:ac:f6:24:b2:7e:4e:27:88:67:16:94:6e:5d:
4b:b4:ef:fa:8a:49:71:23:62:81:78:2c:03:a3:3d:ae:c9:7b: 5c:f8
```

Примітка. ключі для обміну будуть згенеровані серверами автоматично.

2.4.5 Список серверів, які опрацьовують md5 хеші

Наведемо список серверів, які опрацьовують md5 хеші з можливістю розкриття деяких з них:

```
https://hashcracking.info
http://passcracking.ru
http://crackfor.me
http://gdataonline.com/seekhash.php
http://milw0rm.com/md5/info.php
http://us.md5.crysm.net
http://www.plain-text.info
http://www.securitystats.com/tools
http://md5.rednoize.com
http://md5crack.it-helpnet.de
http://ivdb.org/search/md5
http://www.tmto.org
http://www.xmd5.org/index_en.htm
http://ice.breaker.free.fr
http://md5.benramsey.com
http://www.csthis.com/md5/index.php
```

```
http://md5.geeks.li/
http://www.md5database.net/
http://www.md5decrypter.com/
http://www.hashreverse.com/
http://rainbowtables.net/services/results.php
http://www.md5this.com/reverse.php
http://www.cmd5.com/english.aspx
http://www.md5encryption.com/
http://www.thepanicroom.org/index.php?view=cracker
http://www.md5hashes.com/
http://md5pass.info/
http://md5.fastpic.de/crack.php
```

2.4.6 Програма для підрахунку процесорного часу

Лістинг функції, яка рахує час на мові програмування Java приведений нижче:

```
Public int f(CpuTimer ct)
long time;
if(ct.delayStatus > 50){
time = currentTimeMills();
}
return time;
}
```

2.5 Етапи проведення випробувань

Після налаштування з’єднання між серверами перевіримо швидкість з’єднання. Проведемо 15 випробування асинхронних з’єднань.

В тому та іншому випадку заміряймо час, за який формується в Windows та обробляються в FreeBSD пакети (за допомогою власноруч написаного ПЗ, яке рахує процесорний час виконання операцій).

Даними замірами ми зможемо перевірити наскільки доступною є інформація на сервері, якщо в момент синхронізації 2-ох серверів якийсь користувач захоче отримати доступ до інформації (для випадку з асинхронними реплікаціями). В випадку з синхронними – який час займає зупинка серверів для передачі реплікацій них повідомлень.

Проведемо 100 випробувань для того, щоб перевірити можливість порушення конфіденційності інформації, яка передається по каналу, який захищений кореневим сервером сертифікації Windows Server 2008 R2.

На основі проведених досліджень перевіримо доступність інформації та її захищеність.

2.6 Проведення випробувань щодо швидкості та захищеності реплікацій

Випробування, проведені за допомогою програми, функція підрахунку часу та сніферу Sun EffeTech, приведені нижче. Для зручності час проходження реплікації поданий в хвилинах. На швидкість з’єднання тестувались лише синхроні реплікації, як більш надійний спосіб синхронізації:

Таблиця 2.2 – Швидкість проходження синхронних реплікацій між серверами

№ випробування	Кількість записів, які необхідно реплікувати, шт.	Час простою серверу, хв
1	1	0.0004
2	10	0.001
3	100	0.05
4	1000	0.12
5	3000	0.5
6	5000	1.1

Продовження таблиці 2.2

№ випробування	Кількість записів, які необхідно реплікувати, шт.	Час простою серверу, хв
7	8000	1.25
9	10000	1.54
10	12000	2.01
11	15000	2.56
12	18000	3.27

13	21000	4.00
14	24000	4.47
15	27000	5.07

Графік, що наведений нижче демонструє, як з часом зростає час простою серверу коли кість пакетів зростає при синхронній реплікації(рисунок 2.12).

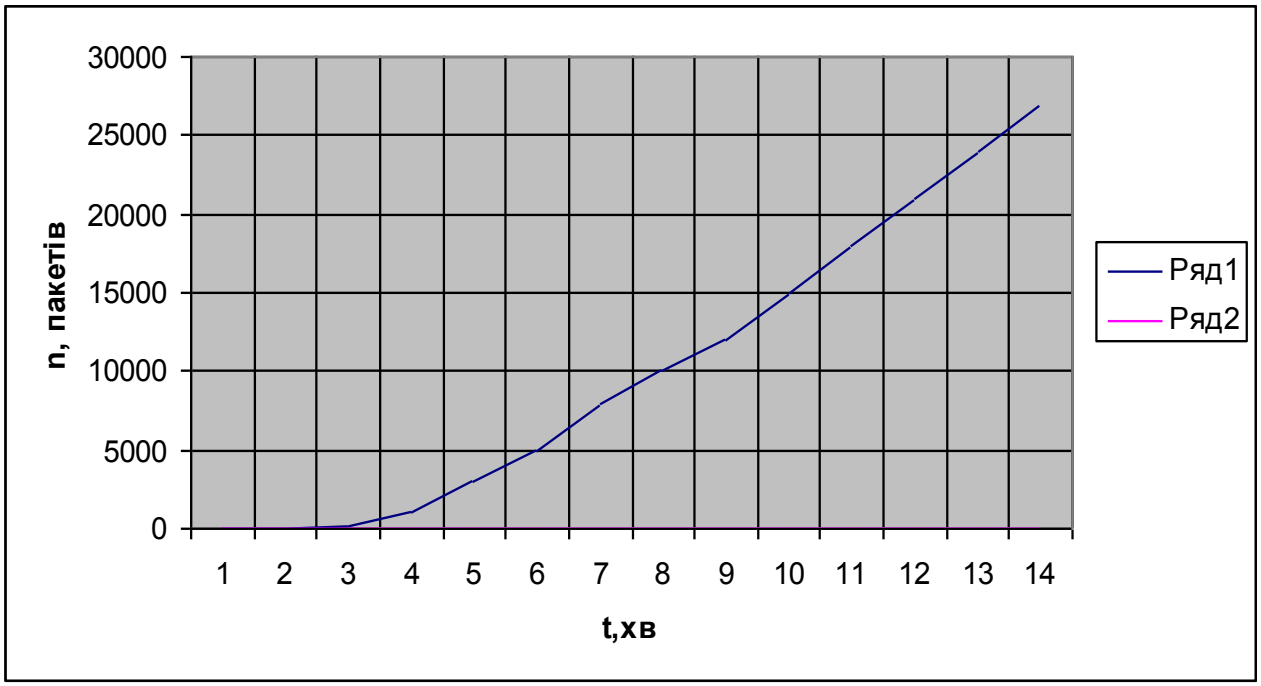

Рисунок 2.12 – Залежність часу простою серверу від кількості реплікаційних пакетів

Тож з цього можна зробити висновок, що даний список рекомендацій підходить лише для асинхронних реплікацій.

Далі – випробування захищеності каналу.

Таблиця 2.3 – захищеність каналу для проходження реплікацій між серверами

№	Випробування за допомогою програмного засобу	Позитивний результат	Негативний результат
1	EffeTech Sniffer	-	+
2	EffeTech Sniffer	-	+
3	EffeTech Sniffer	-	+

4	EffeTech Sniffer	-	+
5	EffeTech Sniffer	-	+
6	EffeTech Sniffer	-	+
7	EffeTech Sniffer	-	+
8	EffeTech Sniffer	-	+
9	EffeTech Sniffer	-	+
10	EffeTech Sniffer	-	+
11	EffeTech Sniffer	-	+
12	EffeTech Sniffer	-	+
13	EffeTech Sniffer	-	+
14	EffeTech Sniffer	-	+
15	EffeTech Sniffer	-	+

Примітка. Як свідчать результати досліджень канал є абсолютно захищеним за допомогою кореневого серверу сертифікації Windows Server 2008 R2.

2.7 Висновки

Результатом проведеної роботи в даному розділі були розглянуті можливі варіанти для синхронізації двох серверних систем, а також представлена можливість захисту каналу, через який будуть передаватись репликаційні повідомлення за допомогою кореневого центру сертифікації Windows.

РОЗДІЛ 3. ЕКОНОМІЧНИЙ РОЗДІЛ

Метою розділу є економічне обґрунтування розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі. Для цього необхідно здійснити розрахунок:

– капітальних витрат на придбання і налагодження складових системи інформаційної безпеки або витрат, що пов’язані з виготовленням апаратури, приладів, програмного забезпечення;

– річних експлуатаційних витрат на утримання і обслуговування об'єкта проектування;

– річного економічного ефекту;

– показників економічної ефективності розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі.

3.1 Розрахунок (фіксованих) капітальних витрат

Капітальні інвестиції – це кошти, призначені для створення і придбання основних фондів і нематеріальних активів, що підлягають амортизації.

Капітальні (фіксовані) витрати на проектування та впровадження проектного варіанта системи інформаційної безпеки складають:

$$К = К_{пр} + К_{зпз} + К_{пз} + К_{аз} + К_{навч} + К_{н}$$

де $К_{пр}$ – вартість розробки проекту інформаційної безпеки та залучення для цього зовнішніх консультантів;

$К_{зпз}$ – вартість закупівель ліцензійного основного й додаткового програмного забезпечення (ПЗ);

$К_{пз}$ – вартість створення основного й додаткового програмного забезпечення;

$К_{аз}$ – вартість закупівлі апаратного забезпечення та допоміжних матеріалів;

$К_{навч}$ – витрати на навчання технічних фахівців і обслуговуючого персоналу;

$К_{н}$ – витрати на встановлення обладнання та налагодження системи інформаційної безпеки, $К_{н}$=1400 грн.

Відповідно до запропонованих в даній роботі рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі формування в Windows та обробка в FreeBSD пакеті здійснюється за допомогою власноруч написаного ПЗ, яке рахує процесорний час виконання операцій.

3.1.1. Визначення витрат на створення програмного забезпечення з формування пакетів в Windows та обробка їх в FreeBSD

3.1.1.1. Визначення трудомісткості розробки та опрацювання програмного продукту

$$t = t_{mз} + t_{в} + t_{а} + t_{пр} + t_{опр} + t_{д}, \text{ годин}$$

де $t_{mз}$ – тривалість складання технічного завдання на розробку ПЗ, $t_{тз}$=10;

$t_{в}$ – тривалість вивчення ТЗ, літературних джерел за темою тощо;

$t_{а}$ – тривалість розробки блок-схеми алгоритму;

$t_{пр}$ – тривалість програмування за готовою блок-схемою;

$t_{опр}$ – тривалість опрацювання програми на ПК;

$t_{д}$ – тривалість підготовки технічної документації на ПЗ.

Умовна кількість операторів у програмі:

$$Q = q \cdot c\ (1 + p) = 10*1,5*(1+0,1) = 16,5$$

де q – очікувана кількість операторів;

c – коефіцієнт складності програми;

p – коефіцієнт корекції програми в процесі її опрацювання.

Тривалість вивчення технічного завдання, опрацювання довідкової літератури з урахуванням уточнення ТЗ і кваліфікації програміста:

$$t_в = \frac{Q \cdot B}{(75...85) \cdot k} = \frac{16{,}5 \cdot 1{,}4}{85 \cdot 1} = 0{,}27 \text{ годин},$$

де В – коефіцієнт збільшення тривалості етапу внаслідок недостатнього опису завдання, В = 1,2...1,5;

k – коефіцієнт, що враховує кваліфікацію програміста і визначається стажем роботи за фахом:

- до 2 років – 0,8;
- від 2 до 3 років – 1,0;
- від 3 до 5 років – 1,1...1,2;
- від 5 до 7 років – 1,3...1,4;
- понад 7 років – 1,5...1,6.

Тривалість розробки блок-схеми алгоритму:

$$t_a = \frac{Q}{(20...25) \cdot k} = \frac{16{,}5}{20 \cdot 1} = 0{,}825 \text{ годин.}$$

Тривалість складання програми за готовою блок-схемою:

$$t_{пр} = \frac{Q}{(20...25) \cdot k} = \frac{16{,}5}{20 \cdot 1} = 0{,}825, \text{ годин.}$$

Тривалість опрацювання програми на ПК:

$$tonp = \frac{1{,}5Q}{(4...5)\cdot k} = \frac{1{,}5\cdot 16{,}5}{5\cdot 1} = 4{,}95\text{, годин.}$$

Тривалість підготовки технічної документації на ПЗ:

$$t_{\text{Д}} = \frac{Q}{(15...20)\cdot k} + \frac{Q}{(15...20)}\cdot 0{,}75 = \frac{16{,}5}{20\cdot 1} + \frac{16{,}5}{20\cdot 1}\cdot 0{,}75 = 1{,}44$$

Отже,

$$t = 10 + 0{,}27 + 0{,}825 + 0{,}825 + 4{,}95 + 1{,}44 = 18{,}31 \text{ годин,}$$

3.1.1.2. Розрахунок витрат на створення програмного продукту

Витрати на створення програмного продукту Кпз складаються з витрат на заробітну плату виконавця програмного забезпечення Ззп і вартості витрат машинного часу, що необхідний для опрацювання програми на ПК Змч:

$$K_{пз} = З_{зп} + З_{мч} = 2197{,}2 + 168{,}44 = 2365{,}64 \text{ грн.}$$

$$З_{зп} = t\, З_{пр} = 18{,}31 * 120 = 2197{,}2 \text{ грн.}$$

де t – загальна тривалість створення ПЗ, годин;

Зпр – средньогодинна заробітна плата програміста з нарахуваннями, грн/годину.

Вартість машинного часу для налагодження програми на ПК визначається за формулою:

$$З_{мч} = (t_{опр} + t_{д})\cdot C_{мч} = (4{,}95 + 1{,}44)\cdot 26{,}36 = 168{,}44 \text{ грн.}$$

де $t_{опр}$ – трудомісткість налагодження програми на ПК, годин;

$t_д$ – трудомісткість підготовки документації на ПК, годин;

$C_{мч}$ – вартість 1 години машинного часу ПК, грн./година.

Вартість 1 години машинного часу ПК визначається за формулою:

$$C_{мч} = 1{,}5 \cdot 10 \cdot 1{,}64 + \frac{4800 \cdot 0{,}6}{1920} + \frac{2500 \cdot 0{,}2}{1920} = 26{,}36 \text{ грн.}$$

3.1.2. Визначення витрат на розробку рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі

3.1.2.1. Визначення трудомісткості розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі

Трудомісткість розробки визначається тривалістю кожної робочої операції:

$$t = t_{тз} + t_в + t_{фп} + t_{ан} + t_с + t_і + t_е + t_д, \text{ годин,}$$

де $t_{тз}$ – тривалість складання технічного завдання, $t_{тз}=6$;

$t_в$ – тривалість вивчення ТЗ, літературних джерел за темою тощо, $t_в=20$;

$t_{фп}$ – тривалість визначення необхідного формату повідомлень для правильного обміну повідомленнями між двома серверами, $t_{фп}=8;$

$t_{ан}$ – тривалість опрацювання алгоритму налаштування серверів, $t_{ан}=24;$

$t_с$ – тривалість виконання операцій на серверах, які опрацьовують md5 хеші, $t_с=8;$

$t_і$ – тривалість підрахунку процесорного часу, $t_і=12;$

$t_е$ – тривалість проведення випробувань, $t_е=14$;

$t_{ер}$ – тривалість проведення випробувань щодо швидкості та захищеності реплікацій, $t_{ер}=7$

$t_д$ – тривалість підготовки технічної документації, $t_д=5$.

Таким чином,

$$t = 8 + 20 + 8 + 24 + 8 + 12 + 12 + 7 + 5 = 106 \text{ годин.}$$

3.1.2.2. Розрахунок витрат розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі

Витрати на розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі $К_{рр}$ складаються з витрат на заробітну плату виконавця $З_{зп}$ і вартості витрат машинного часу, що необхідний для опрацювання $З_{мч}$:

$$К_{рр} = З_{зп} + З_{мч} = 12720 + 719{,}74 = 13439{,}74 \text{ грн.}$$
$$З_{зп} = t\, З_{пр} = 106*120 = 12720 \text{ грн.}$$

де t – загальна тривалість розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі, годин;

$Зпр$ – средньогодинна заробітна плата спеціаліста з нарахуваннями, грн/годину.

Вартість машинного часу для розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі визначається за формулою:

$$З_{мч} = t \cdot С_{мч} = 106 \cdot 6{,}79 = 719{,}74 \text{ грн.}$$

де t – трудомісткість розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі, годин;

$C_{мч}$ – вартість 1 години машинного часу ПК, грн./година.

Вартість 1 години машинного часу ПК визначається за формулою:

$$C_{мч} = 0{,}9 \cdot 4 \cdot 1{,}64 + \frac{3200 \cdot 0{,}5}{1920} + \frac{1100 \cdot 0{,}1}{1920} = 6{,}79 \text{ грн.}$$

Відповідно до розроблених рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі тестування безпеки з'єднання виконується за допомогою потужного сніферу EffeTech, вартість якого складає близько 5600 грн.

Таким чином, капітальні (фіксовані) витрати на проектування та впровадження проектного варіанта системи інформаційної безпеки складають:

$$К = 5600 + 2365{,}64 + 13439{,}74 + 1400 = 22805{,}38 \text{ грн.}$$

3.1.1 Розрахунок поточних витрат

Річні поточні витрати на функціонування системи інформаційної безпеки складають:

$$С = С_в + С_к + С_{ак}, \quad \text{грн.}$$

де $С_в$ - вартість відновлення й модернізації системи ($С_в$=0);

$С_к$ - витрати на керування системою в цілому;

$С_{ак}$ - витрати, викликані активністю користувачів системи інформаційної безпеки ($С_{ак}$ = 0 грн.).

Витрати на керування системою інформаційної безпеки ($С_к$) складають:

$$С_к = С_н + С_а + С_з + С_{ел} + С_о + С_{тос}, \text{ грн.}$$

Річний фонд амортизаційних відрахувань ($С_а$) визначається прямолінійним методом нарахування амортизації відповідно до строків їх корисного використання.

$$С_а = 5600 / 4 = 1400 \text{ грн.}$$

Річний фонд заробітної плати інженерно-технічного персоналу, що обслуговує систему інформаційної безпеки ($С_з$), складає:

$$С_з = З_{осн} + З_{дод}, \text{ грн.}$$

Основна заробітна плата визначається, виходячи з місячного посадового окладу, а додаткова заробітна плата – в розмірі 8-10% від основної заробітної плати.

Основна заробітна плата одного спеціаліста з інформаційної безпеки на місяць складає 15000 грн. Додаткова заробітна плата – 8% від основної заробітної плати. Отже,

$$С_з = 15000*12 + 15000*12*0,08= 194400 \text{ грн.}$$

Ставка ЄСВ для всіх категорій платників з 01.01.2016 р. складає 22%.

$$С_{єв} = 181440 *0,22 = 42768 \text{ грн.}$$

Вартість електроенергії, що споживається апаратурою системою інформаційної безпеки протягом року ($С_{ел}$), визначається за формулою:

$$С_{ел} = Р \cdot F_р \cdot Ц_е, \text{ грн.,}$$

де Р – встановлена потужність апаратури інформаційної безпеки, (Р=1,4 кВт);

$F_р$ – річний фонд робочого часу системи інформаційної безпеки ($F_р$ = 1920 год.);

$Ц_е$ – тариф на електроенергію, ($Ц_е$ = 1,64 грн./кВт за годину).

Вартість електроенергії, що споживається апаратурою системою інформаційної безпеки протягом року:

$$С_{ел} = 1,4*1920*1,64 = 4408,32 \text{ грн.}$$

Витрати на технічне й організаційне адміністрування та сервіс системи інформаційної безпеки визначаються у відсотках від вартості капітальних витрат - 1% ($С_{тос}$ = 22805,38 *0,01=228,05 грн).

Витрати на керування системою інформаційної безпеки ($С_к$) визначаються:

$$С_к = 1400 + 194400 + 42768 + 4408,32 = 242976,32 \text{ грн.}$$

Отже, річні поточні витрати на функціонування системи інформаційної безпеки складають:

$$С = 242976,32 \text{ грн.}$$

3.2 Оцінка можливого збитку від атаки на вузол або сегмент мережі

3.2.1 Оцінка величини збитку

Для розрахунку вартості такого збитку можна застосувати наступну спрощену модель оцінки.

Необхідні *вихідні дані* для розрахунку:

$t_п$ – час простою вузла або сегмента корпоративної мережі внаслідок атаки, 4 години;

$t_в$ – час відновлення після атаки персоналом, що обслуговує корпоративну мережу, 2 години;

$t_{ви}$ – час повторного введення загубленої інформації співробітниками атакованого вузла або сегмента корпоративної мережі, 8 годин;

$З_о$ – заробітна плата обслуговуючого персоналу (адміністраторів та ін.), 7000 грн./міс.;

$З_с$ – заробітна плата співробітників атакованого вузла або сегмента корпоративної мережі, 9000 грн./міс.;

$Ч_о$ – чисельність обслуговуючого персоналу (адміністраторів та ін.), 3 особи;

$Ч_с$ – чисельність співробітників атакованого вузла або сегмента корпоративної мережі, 7 осіб.;

О – обсяг прибутку атакованого вузла або сегмента корпоративної мережі, 400 тис. грн. у рік;

$П_{зч}$ – вартість заміни встаткування або запасних частин, грн;

I – число атакованих сегментів корпоративної мережі, 1;

N – середнє число атак на рік, 50.

Упущена вигода від простою атакованого сегмента корпоративної мережі становить:

$$U = П_п + П_в + V,$$

де $П_п$ – оплачувані втрати робочого часу та простої співробітників атакованого вузла або сегмента корпоративної мережі, грн;

$П_в$ – вартість відновлення працездатності вузла або сегмента корпоративної мережі (переустановлення системи, зміна конфігурації та ін.), грн;

V – втрати від зниження обсягу продажів за час простою атакованого вузла або сегмента корпоративної мережі, грн.

Втрати від зниження продуктивності співробітників атакованого вузла або сегмента корпоративної мережі являють собою втрати їхньої заробітної плати (оплата непродуктивної праці) за час простою внаслідок атаки:

$$П_П = \frac{\sum Зс}{F} \cdot t_п = \frac{7000 \cdot 7}{176} \cdot 4 = 1113{,}64 \quad \text{грн},$$

де F – місячний фонд робочого часу (при 40-а годинному робочому тижні становить 176 ч).

Витрати на відновлення працездатності вузла або сегмента корпоративної мережі включають кілька складових:

$$П_в = П_{ви} + П_{пв} + П_{зч},$$

де $П_{ви}$ – витрати на повторне уведення інформації, грн.;

$П_{пв}$ – витрати на відновлення вузла або сегмента корпоративної мережі, грн;

$П_{зч}$ – вартість заміни устаткування або запасних частин, грн.

Витрати на повторне введення інформації $П_{ви}$ розраховуються виходячи з розміру заробітної плати співробітників атакованого вузла або сегмента корпоративної мережі $З_с$, які зайняті повторним введенням втраченої інформації, з урахуванням необхідного для цього часу $t_{ви}$:

$$П_{ВИ} = \frac{\sum Зс}{F} \cdot t_{ви} = \frac{7000 \cdot 10}{176} \cdot 8 = 3181{,}81 \text{ грн.}$$

Витрати на відновлення сегмента корпоративної мережі $П_{пв}$ визначаються часом відновлення після атаки $t_в$ і розміром середньогодинної заробітної плати обслуговуючого персоналу (адміністраторів):

$$П_{ПВ} = \frac{\sum Зо}{F} \cdot t_в = \frac{9000 \cdot 3}{176} \cdot 2 = 306{,}82 \text{ грн.}$$

$$П_в = 3181,81 + 306,82 = 3488,63 \text{ грн.}$$

Втрати від зниження очікуваного обсягу прибутків за час простою атакованого вузла або сегмента корпоративної мережі визначаються виходячи із средньогодинного обсягу прибутку і сумарного часу простою сегмента корпоративної мережі:

$$V = \frac{O}{Fг} \cdot (t_П + t_В + t_{ВИ})$$

$$V = \frac{400000}{2080} \cdot (4 + 2 + 8) = 2692,31 \text{ грн.}$$

де $F_г$ – річний фонд часу роботи філії (52 робочих тижні, 5-ти денний робочий тиждень, 8-ми годинний робочий день) становить близько 2080 год.

$$U = 1113,64 + 3488,63 + 2692,31 = 7294,58 \text{ грн.}$$

Таким чином, загальний збиток від атаки на сегмент корпоративної мережі організації складе:

$$B = \sum_i \sum_n U = \sum_1 \sum_{50} 7294,58 = 364728,88 \text{ грн.}$$

3.2.2 Загальний ефект від впровадження системи інформаційної безпеки

Загальний ефект від впровадження системи інформаційної безпеки визначається з урахуванням ризиків порушення інформаційної:

$$E = B \cdot R - C \text{ грн.},$$

де B – загальний збиток від атаки у разі перехоплення інформації, тис. грн.;

R – вірогідність успішної реалізації атаки на сегмент мережі, частки одиниці (85%);

С – щорічні витрати на експлуатацію системи інформаційної безпеки.

Загальний ефект від впровадження системи інформаційної безпеки визначається з урахуванням ризиків порушення інформаційної безпеки:

$$E = 364728{,}88*0{,}85 - 242976{,}32 = 67043{,}23 \text{ грн.}$$

3.3 Визначення та аналіз показників економічної ефективності системи інформаційної безпеки

Коефіцієнт повернення інвестицій ROSI показує, скільки гривень додаткового прибутку приносить одна гривня капітальних інвестицій на впровадження системи інформаційної безпеки:

$$ROSI = \frac{\text{E}}{K}, \quad \text{частки одиниці,}$$

де E – загальний ефект від впровадження системи інформаційної безпеки грн.;

К – капітальні інвестиції за варіантами, що забезпечили цей ефект, грн.

Коефіцієнт повернення інвестицій ROSI:

$$ROSI = \frac{67043{,}23}{22805{,}38} = 2{,}94, \quad \text{частки одиниці,}$$

Проект визнається економічно доцільним, якщо розрахункове значення коефіцієнта повернення інвестицій перевищує величину річної депозитної ставки з урахуванням інфляції:

$$\text{ROSI} > (\text{N}_{\text{деп}} - \text{N}_{\text{інф}})/100),$$

де $N_{деп}$ – річна депозитна ставка, (18 %);

$N_{інф}$ – річний рівень інфляції, (12%).

Розрахункове значення коефіцієнта повернення інвестицій:

$$2{,}94 > (18 - 12)/100 = 2{,}94 > 0{,}06.$$

Термін окупності капітальних інвестицій T_o показує, за скільки років капітальні інвестиції окупляться за рахунок загального ефекту від впровадження системи інформаційної безпеки:

$$T_O = \frac{K}{E} = \frac{1}{ROSI} = \frac{1}{2{,}94} = 0{,}34, \quad \text{років.}$$

3.4 Висновок

Розробки рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі є економічно доцільною згідно із розрахунками економічної ефективності. Коефіцієнт повернення інвестицій ROSI складає 2,94, що означає отримання додаткового прибутку в розмірі 2,94 грн. з кожної гривні, вкладеної в розробку рекомендацій. Термін окупності складає 0,34 роки (124 дні).

ВИСНОВКИ

В ході виконання дипломної роботи було проаналізовано існуючі методи захищених з'єднань між FreeBSD та Windows Server 2008 R2, знайдено спосіб проводити реплікації між цими ОС, а також створено систему захисту, яка за допомогою SSL з'єднання допомагає шифрувати канал передачі інформації. Реплікаційні повідомлення передаються через протокол HTTP в форматі XML.

Сервер реплікацій ОС FreeBSD був модернізований для правильного оброблення записів користувачів під формат Windows Server. Була створена реляційна БД для зберігання облікових записів користувачів, що відповідає файлу NTDAT ОС Windows.

На основі цих налаштувань було проведено випробування захищеності за допомогою сніферу Sun EffeTech та швидкості проходження реплікацій. Перевірка захищеності показала, що за допомогою сніферу можна перехопити реплікацій ні пакети, але ввесь трафік HTTP зашифровано за допомогою криптоалгоритму RSA, який лежить в основі SSL. Тож прочитати дані, інкапсульовані в пакети HTTP неможливо.

Перевірка швидкості при синхронних реплікаціях показала, що передача пакетів з понад 20 тисячами змін в базі продовжується на протязі чотирьох хвилин при синхронних реплікаціях, що само по собі не дає можливості використовувати дані рекомендації по налаштуванню для синхронних реплікацій в високонавантажених системах, та системах реального часу відгуку. Але для середніх і малих підприємств (менше 10 тисяч абонентів) швидкість проходження реплікацій суттєво не знижується.

ПЕРЛІК ПОСИЛАНЬ

1 Низамунтдинов М.Ф. Волшебный мир Active Directory. – СПб.: БХВ-Петербург, 2005. – 432 с.: ил.;

2 Деккер А. FreeBsd для системных администраторов. // В сб.: Хофштадер Д., Деннет Д. Глаз разума. – Самара: Бахрах-М, 2003.

3 Хакер 04/135/12. Взлом LDAP: теория и практика. Разбираемся, как ломают капчи. Службы каталогов, Литва, 2012. – 44-49 с.;

4 Закону України «Про інформацію».

5 Закону України «Про захист персональних даних».

6 ДСТУ 3396.2-97 «Захист інформації. Технічний захист інформації. Терміни та визначення».

7 ДСТУ 3396.1-96 «Захист інформації. Технічний захист інформації. Порядок проведення робіт»

8 НД ТЗІ 1.4-001-2000 «Типове положення про службу Захисту інформації в автоматизованій системі».

9 НД ТЗІ 3.7-003-2005 «Порядок проведення робіт із створення комплексної системи захисту інформації в інформаційно-телекомунікаційній системі»

10 НД ТЗІ 1.1-002-99 «Загальні положення про захист інформації в комп’ютерних системах від несанкціонованого доступу».

11 НД ТЗІ 2.5-004-99. «Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу».

12 НД ТЗІ 2.5-005-99. «Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу».

13 Грайворонський М.В., Новіков О.М. Безпека інформаційно-комунікаційних систем. – К.: Видавнича група BHV, 2009. – 608 с.

14 В.Г. Олифер, Н.А. Олифер Компьютерные сети. Принципы, технологии, протоколы – Спб.: Питер, 2001. – 672 с.

15 Microsoft Windows Server 2008 R2. Полное руководство, Вильямс, 2011, - 1456с.

16 Современные операционные системы. Книга, Эндрю Таненбаум, 2010, - 1 101с.

17 Операционная система Linux. Курс лекций. Книга, Георгий Курячий и Кирилл Маслинский, 2011, - 803с.

18 Самоучитель системного администратора Linux. Книга, Денис Колисниченко, 2011, - 982с.

ДОДАТОК А. Відомість матеріалів дипломного проекту

№	Формат	Найменування	Кількість листів	Примітка
1	А4	Реферат	3	
2	А4	Список умовних скорочень	1	
3	А4	Зміст	2	
4	А4	Вступ	2	
5	А4	1 Розділ	15	
6	А4	2 Розділ	25	
7	А4	3 Розділ	14	
8	А4	Висновки	1	
9	А4	Перелік посилань	2	
10	А4	Додаток А	1	
11	А4	Додаток Б	1	
12	А4	Додаток В	1	
13	А4	Додаток Г	1	

ДОДАТОК Б. Перелік документів на оптичному носії

1 Титульна сторінка.doc

2 Завдання.doc

3 Реферат.doc

4 Список умовних скорочень.doc

5 Зміст.doc

6 Вступ.doc

7 Розділ 1.doc

8 Розділ 2.doc

9 Розділ 3.doc

10 Висновки.doc

11 Перелік посилань.doc

12 Додаток А.doc

13 Додаток Б.doc

14 Додаток В.doc

15 Додаток Г.doc

Презентація.pptx

ДОДАТОК В. Відгуки керівників розділів

Відгук керівника економічного розділу:

Керівник розділу _____ _____

(підпис) (ініціали, прізвище)

ДОДАТОК Г. ВІДГУК

на дипломну роботу магістра на тему:
Розробка рекомендацій щодо забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі
студента групи 125м-17-1
Пуркара Сергія Олександровича

Пояснювальна записка складається з титульного аркуша, завдання, реферату, списку умовних скорочень, змісту, вступу, трьох розділів, висновків, переліку посилань та додатків, розташованих на __ сторінках та містить __ рисунків, __ таблиць, __ джерела та __ додатка.

Актуальність теми полягає в необхідності забезпечення інформаційної безпеки служби каталогів в гетерогенному середовищі.

Зміст та структура дипломної роботи дозволяють розкрити поставлену тему повністю.

Студент показав достатній рівень володіння теоретичними положеннями з обраної теми, показав здатність формувати власну точку зору (теоретичну позицію).

Робота виконана самостійно. У роботі було проаналізовано існуючі методи захищених з’єднань між FreeBSD та Windows Server 2008 R2, знайдено спосіб проводити реплікації між цими ОС, а також створено систему захисту, яка за допомогою SSL з’єднання допомагає шифрувати канал передачі інформації. Реплікаційні повідомлення передаються через протокол HTTP в форматі XML.

Це підтверджує самостійність обробки даних, практичні рекомендації та висновки.

Робота оформлена та написана грамотною мовою. Містить необхідний ілюстрований матеріал. Автор добре знає проблему, уміє формулювати наукові та практичні завдання і знаходить адекватні засоби для їх вирішення.

В цілому дипломна робота задовольняє усім вимогам і може бути допущена до захисту, а його автор Пуркар Сергій Олександрович заслуговує на оцінку «_____».

Керівник дипломної роботи,
к.т.н., доц. С.В. Флоров

Керівник спец. част.,
ас. кафедри БІТ Ю.П. Рибальченко