МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

_____ С.В. Казмірчук

«_____»_____ 20__ р.

На правах рукопису
УДК 004.056.5:510.22(043.3)

КВАЛІФІКАЦІЙНА РОБОТА

ЗДОБУВАЧА ВИЩОЇ ОСВІТИ ОСВІТНЬОГО СТУПЕНЯ «МАГІСТР»

Тема: Система захищеного розподіленого доступу до конфіденційної інформації на базі хмарних технологій

Виконавець: Ткаченко Б. В.

Науковий керівник: к.т.н., доцент Карловський С. Є.

Нормоконтролер: к.т.н., доцент Карловський С. Є.

Київ 2020

Вступ

Розвиток глобальної мережі Інтернет та всеохоплююче її використання у різних сферах життя та комерційної діяльності людей призводить до збільшення обсягів інформації, що циркулює та оброблюється.

Хмарні технології створюють базу в інфраструктурі нового покоління, яка дозволяє створити потужну інформаційно-телекомунікаційну систему з новою архітектурою та можливостями. Доволі великий відсоток коштів підприємства та організації використовують на експлуатацію та технічне обслуговування локальних інформаційних систем, хмарні технології дозволяють пришвидшити процес розробки та випуску програмних продуктів на ринок та збільшити ефективності праці підприємства шляхом організації прозорого документообігу та спільної праці над проектами.

Швидке зростання хмарних технологій надає величезний потенціал для підвищення ефективності функціонування інформаційної системи, скорочення витрат на її обслуговування, технічне забезпечення та швидке розгортання філіалів для підприємств різної форми власності. Основними особливостями хмарних технологій є можливість масштабування інфраструктури для зберігання даних та динамічне керування потужностями, що звільнить користувача від управління складною технологією.

Використання хмарних технологій має широкий спектр переваг, однак досі залишається відкритим процес формування нормативно-правової платформи врегулювання взаємодії постачальника послуг та користувача.

Сьогодні хмарні технології знаходять активне застосування у всіх розвинених країнах, забезпечуючи принципово нові, економічно ефективні можливості для бізнесу, управління, освіти і наукових досліджень. Вони можуть бути використані в управлінні вузом, навчальному процесі, створення ефективних інструментів організації науково-дослідницької діяльності як в масштабах вузу, так і на міжвузівському рівні.

Хмарні технології – це парадигма, що передбачає віддалену обробку та зберігання даних. Ця технологія надає користувачам мережі Інтернет, доступ до комп'ютерних ресурсів сервера і використання програмного забезпечення як онлайн-сервісу. Тобто якщо є підключення до Інтернету то можна виконувати складні обчислення, опрацьовувати дані використовуючи потужності віддалених серверів [1].

Хмарні сервіси, що дозволяють перенести обчислювальні ресурси й дані на віддалені інтернет-сервери, в останні роки стали одним з основних трендів розвитку IT-технологій.

Актуальність

На підприємстві де циркулює конфіденційна інформація незадоволені рівнем надійності існуючих інструментів віддаленного зберігання інформації тому було вирішено розробити власний проект де будуть усунені усі потенційні вразливості. Програма буде призначатися для обмеженої кількості осіб у корпоративній мережі.

Мета

Метою роботи є аналіз сучасних сервісів віддаленого зберігання інформації та розробка інфраструктури що забезпечує конфедінційність та цілістність персональних даних користувача від сучасних загроз.

Задачі

1. Провести аналіз сучасних сервісів та дослідження кріптоалгоритмів.
2. Розробити інраструктуру з усуненням знайдених вразливостей.
3. Провести експерементальне дослідження розробленої системи.

РОЗДІЛ 1

ОГЛЯД СТАНУ ПРОБЛЕМИ.ОСНОВНІ ПОНЯТТЯ

1.1 Історія розвитку хмарних технологій

Хмарні обчислення мають досить довгу історію (концепція зародилася ще в 1960 році), однак сам термін утвердився тільки кілька років тому. Незважаючи на широке поширення і часте вживання, у цього терміна до теперішнього часу немає чіткого і однозначного визначення, так як в процесі розвитку хмарних технологій формулювання піддається все новим і новим змінам і доповненням. Тому замість строгого визначення наведемо його найбільш поширену версію: «Хмарні обчислення (англ. cloud computing) – технологія розподіленої обробки даних, в якій комп'ютерні ресурси і потужності надаються користувачеві як Інтернет-сервіс.

Ідея доступу до ресурсів за розподілом часу пронизувала IT галузь від самого початку – з часів перших мейнфреймів, коли час роботи з комп'ютером був розвподілений за графіком.[1] Джон Маккарті засновник ідеї комп'ютерного розподілу часу, саме під його керівнитством уперше в світі було розроблено мережу SAGE, яка дозволяла декільком користувачам, одночасно, отримувати доступ до системи. Інший вчений Лестер Еарнест висловлював: "Без поділу часу, не було б сучасного Інтернету".
Інший амерекаській вчений Джозеф Ліклайдер один із засновників мережі ARPANET, у своїй публікації “Міжгалактична Комп'ютерна мережа” висловлював ідею: “У майбутньому я зможу користуватися певними мережевими функціями, здійснюючи вибірку потрібних мені даних за допомогою системи, яка підбере необхідні мені програми. Для цього вона буде використовувати запропоновані їй описи, які з часом можна буде робити природною мовою. Між запозиченими програмами і моїми власними можна буде встановлювати зв'язок ... виконання завдань може відбуватися де завгодно”. Отже ідея облачних обчислень була запропонвана ще на зорі комп'ютерної ери, на той час не було достатньо технічних засобів для її втілення.

Першим же кроком до втілення облачних обчислень можна вважати появу ASP (Application service provider - провайдери послуг доступу до додатків) у другій половині 1990х років. ASP можна вважати одними із перших SaaS сервісів. Пальма першості належить сервісу електроної пошти від компанії Hotmail. Але за відсутності швидких та стабілних каналів інтернет користувачі не могли отримати якісні послуги, а без технологій віртуалізації неможливо було ефективно та гнучно розподіляти ресурси та масштабувати сервіси. Також слід зазначити що лавиноподібний ріст користувачів інтернет, що сформували попит на послуги SaaS, відбувся лише у 2000х роках, тому можна лише на пальцях рук порахувати ASP провайдерів що дожили до наших днів, серед них найбільш відомий - Salesforce.[2] Розвиток IT галузі зображений на малюнку 1.1.

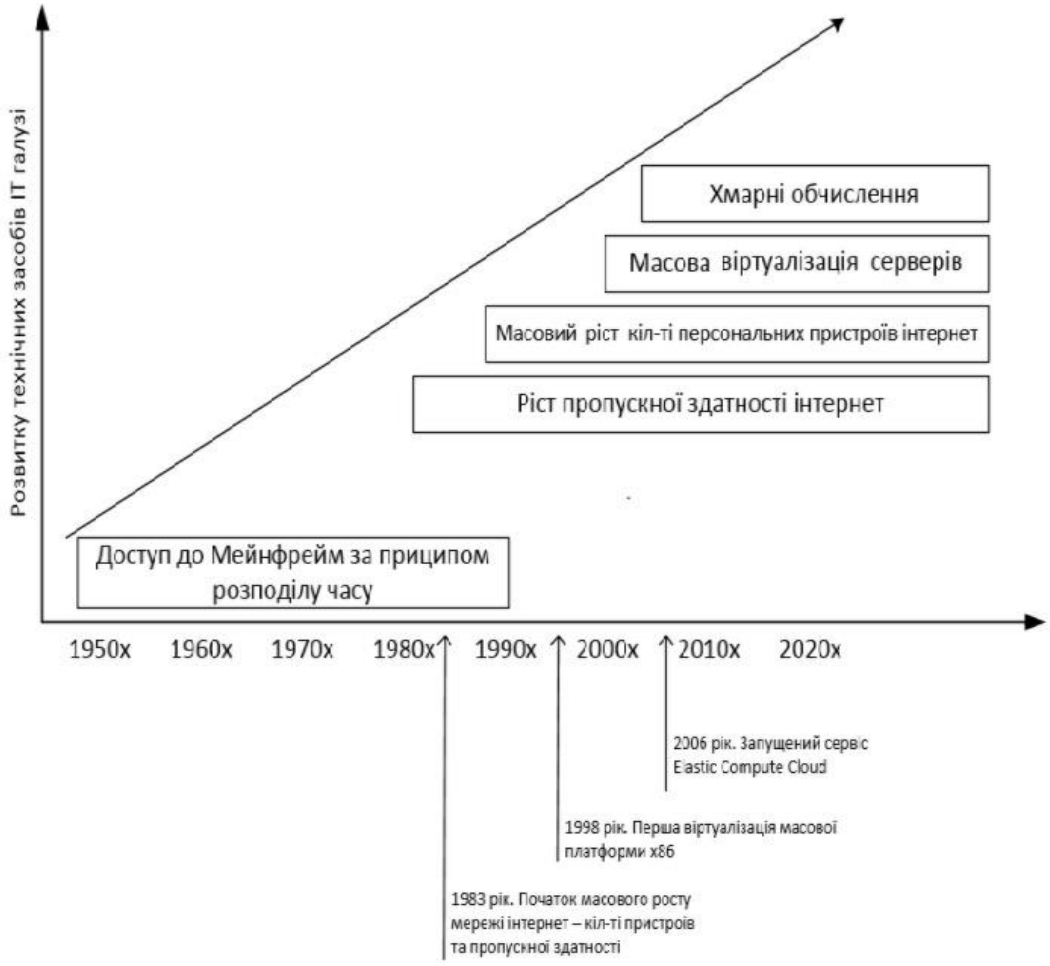

Рис. 1.1 Розвиток IT галузі

Модель обслуговування визначає рівень автоматизації IT процесів інфраструктури. Виділяють наступні моделі надання послуг за допомогою хмари:

Програмне забезпечення як послуга (SaaS) Прикладами програмного забезпечення як послуги, що працює на основі обчислювальної хмари, є сервіси Gmail та Google docs.

Платформа як послуга (PaaS) Наприклад, Google Apps надає застосунки для бізнесу в режимі онлайн, доступ до яких відбувається за допомогою Інтернет-браузера тоді як ПЗ і дані зберігаються на серверах Google.[3] Інфраструктура як послуга (IaaS) Найбільшими гравцями на ринку інфраструктури, як послуги є Amazon, Microsoft, VMWare, Rackspace та Red Hat. Хоча деякі з них пропонують більше, ніж просто інфраструктуру, їх об’єднує мета продавати базові обчислювальні ресурси. Моделі обсуговування наведені на малюнку 1.2.

Традиційний підхід	Сервісний підхід до управління IT інфраструктурою		
	IaaS	PaaS	SaaS
Установка патчів і оновлень протягом життєвого циклу програм	Установка патчів і оновлень протягом життєвого циклу програм	Установка патчів і оновлень протягом життєвого циклу програм	Установка патчів і оновлень протягом життєвого циклу програм
Завантаження Даних Користувача	Завантаження Даних Користувача	Завантаження Даних Користувача	Завантаження Даних Користувача
Установка складних додатків з багаторівневою архітектурою	Установка складних додатків з багаторівневою архітектурою	Установка складних додатків з багаторівневою архітектурою	Установка складних додатків з багаторівневою архітектурою
Установка патчів і оновлень протягом життєвого циклу Middleware and Runtime	Установка патчів і оновлень протягом життєвого циклу Middleware and Runtime	Установка патчів і оновлень протягом життєвого циклу Middleware and Runtime	Установка патчів і оновлень протягом життєвого циклу Middleware and Runtime
Установка додаткового ПЗ, бібліотеки, виконувані середовища: JAVA, .Net (Middleware and Runtime)	Установка додаткового ПЗ, бібліотеки, виконувані середовища: JAVA, .Net (Middleware and Runtime)	Установка додаткового ПЗ, бібліотеки, виконувані середовища: JAVA, .Net (Middleware and Runtime)	Установка додаткового ПЗ, бібліотеки, виконувані середовища: JAVA, .Net (Middleware and Runtime)
Установка патчів і оновлення в перебігу життєвого циклу ОС	Установка патчів і оновлення в перебігу життєвого циклу ОС	Установка патчів и оновлення в перебігу життєвого циклу ОС	Установка патчів и оновлення в перебігу життєвого циклу ОС
Установка і настройка ОС	Установка і настройка ОС	Установка і настройка ОС	Установка і настройка ОС
Установка гипервизора і настройка віртуалізації (опціонально)	Установка гипервизора і настройка віртуалізації (опціонально)	Установка гипервизора и настройка віртуалізації (опціонально)	Установка гипервизора и настройка віртуалізації (опціонально)
Виділення мережевих ресурсів (Фізичні порти, VLAN, IP адресації)	Виділення мережевих ресурсів (Фізичні порти, VLAN, IP адресації)	Виділення мережевих ресурсів (Фізичні порти, VLAN, IP адресації)	Виділення мережевих ресурсів (Фізичні порти, VLAN, IP адресації)
Виділення ресурсів Системи Зберігання Даних	Виділення ресурсів Системи Зберігання Даних	Виділення ресурсів Системи Зберігання Даних	Виділення ресурсів Системи Зберігання Даних
Виділення фізичного сервера	Виділення фізичного сервера	Виділення фізичного сервера	Виділення фізичного сервера

Рис.1.2 Моделі обслуговування

За моделлю розгортання можна розділити хмари на приватні, публічні та гібридні. Приватна хмара (private cloud) – це IT-інфраструктура, контрольована і експлуатована в інтересах однієї-єдиної організації. Приватне хмара може

перебувати у власності, управлінні та експлуатації у самої організації (замовника), або у зовнішнього оператора, або частково у замовника і частково у оператора. Приклад приватної хмари зображений на малюнку 1.3.

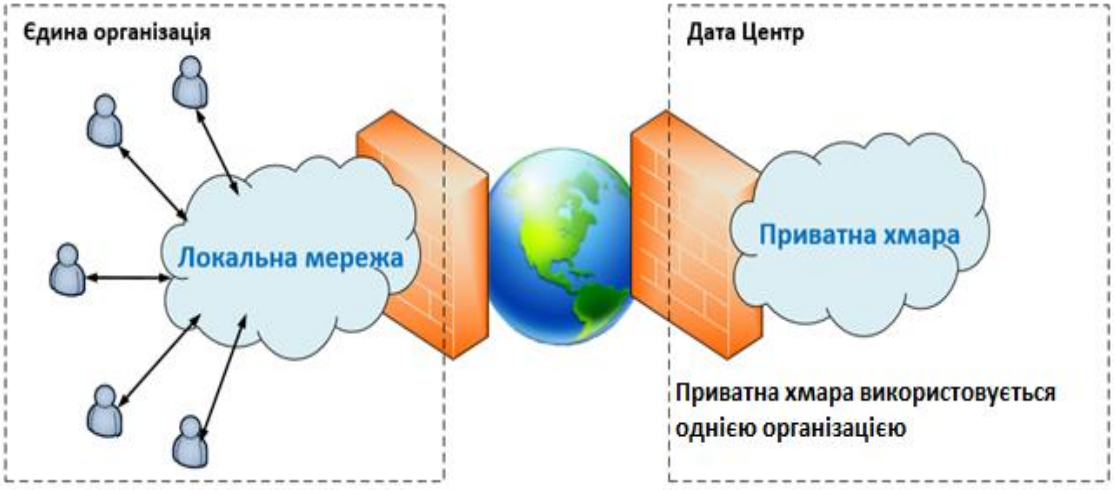

Рис. 1.3 Приклад приватної хмари

Публічна хмара (public cloud) – це IT-інфраструктура, призначена для вільного використання широкою публікою. Користувачі даних, розміщених в хмарі, не мають можливості управляти і обслуговувати дане хмара, вся відповідальність за цих питань покладено на його власника. Публічна хмара може перебувати у власності, управлінні та експлуатації комерційних, наукових і урядових організацій (або їх комбінацій). Публічна хмара фізично існує в юрисдикції власника - постачальника послуг (хмарного провайдера).[4] Приклад публічної хмари зображений на малюнку 1.4.

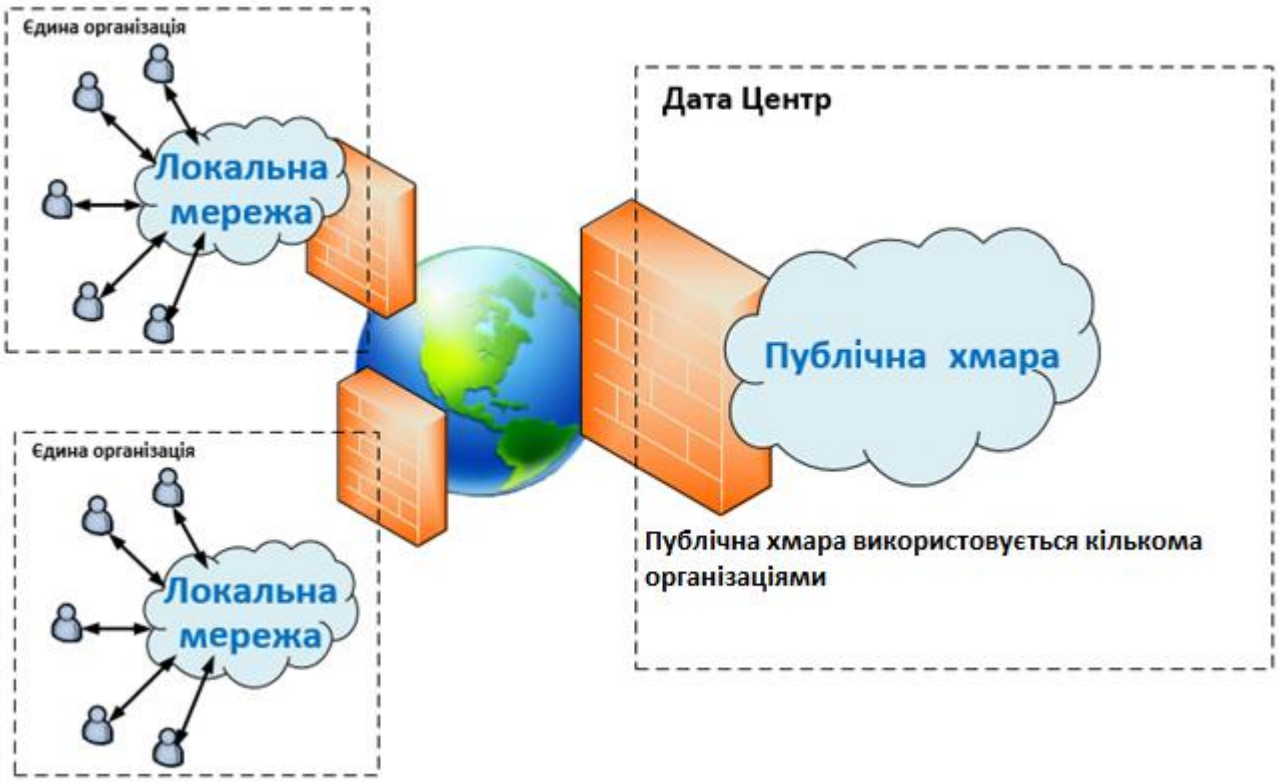

Рис. 1.4 Приклад публічної хмари

Гібридна хмара (hybrid cloud) – це IT-інфраструктура, що представляє собою комбінацію приватних і публічних хмар, пов'язаних між собою стандартизованими або власними технологіями передачі даних і додатків. Відповідальність за управління хмарними сервісами розподіляється між постачальником послуг публічного хмари і організацією замовником. Приклад гібридної хмари забражений на малюнку 1.5.

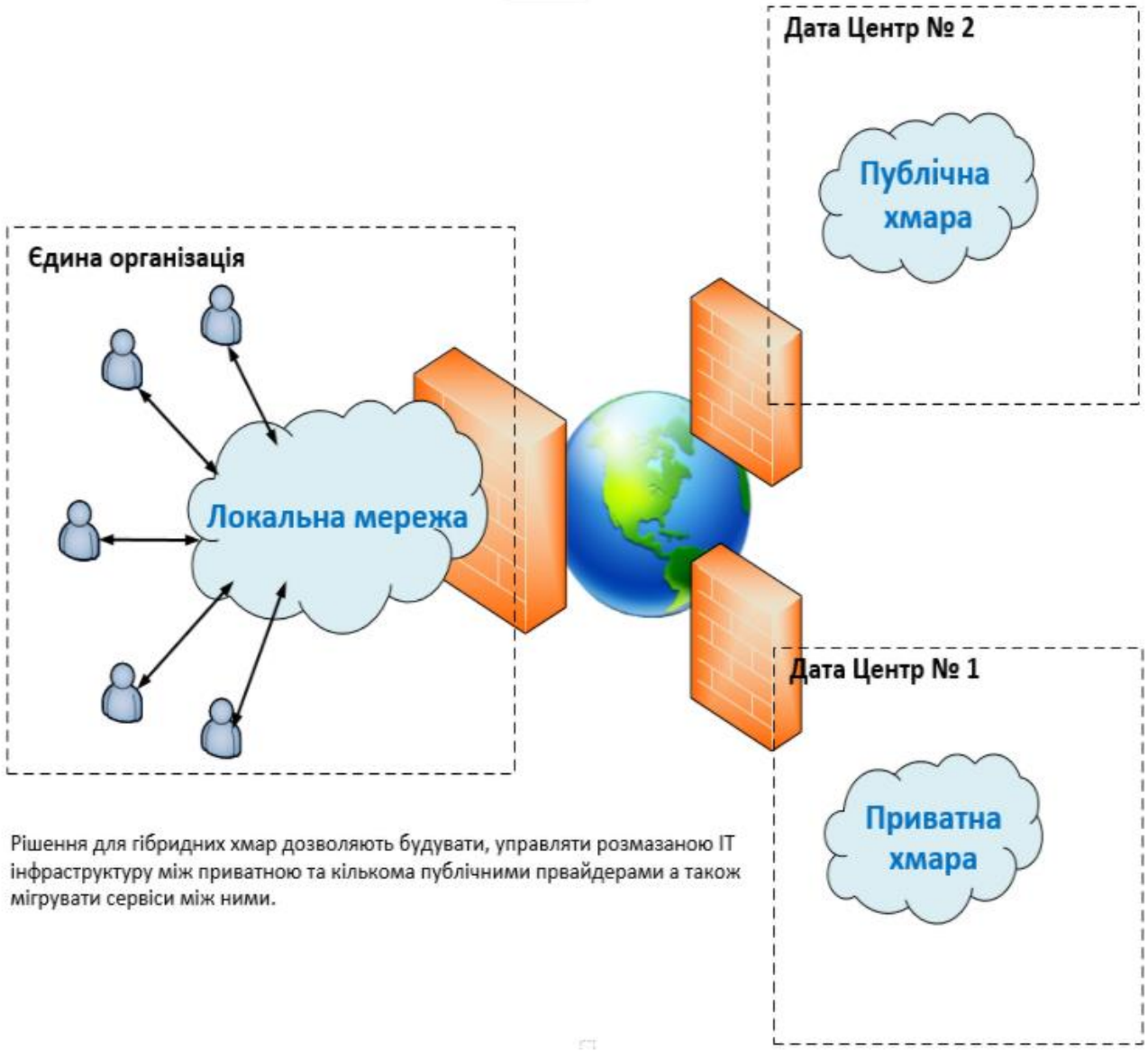

Рис. 1.5 Приклад гібридної хмари

По суті гібридне хмара не є самостійним типом хмарних впроваджень, а лише вказує на тісну інтеграцію публічних і приватних (приватних) хмарних систем. Повної ясності в питанні про те, де пролягає межа між приватними і гібридними хмарними впровадженнями, поки ще немає, тому сьогодні важко знайти достовірні прогнози, що дозволяють оцінити перспективи гібридної, публічної і приватної моделей в найближчому майбутньому. Водночас накопичилося достатньо підстав, щоб передбачити зростання популярності гібридної моделі. Можна виділити наступні переваги даної моделі: G гібридні хмари дозволяють контролювати ключові дані завдяки можливості залишити їх під внутрішньої захищеної мережі компанії; G гібридна модель дає можливість інтегрувати і комбінувати публічні хмарні сервіси від різних постачальників. За

моделлю обслуговування нині хмарні технології прийнято ділити на наступні: G SaaS (Software as a Service - програмне забезпечення як послуга) – надання додатки для кінцевих користувачів з доступом через Інтернет; G IaaS (Infrastructure as a Service – інфраструктура як послуга) – надання апаратної IT-инфра- структури, що включає сервери, мережі та пристрої зберігання інформації (нерідко, говорячи про IaaS, ми маємо на увазі оренду віртуальних серверів на чужому обладнанні. G PaaS (Platform as a Service - платформа як послуга) – це IaaS плюс опера- ционная система і її API (Application Programming Interface - інтерфейс програмування додатків).[5] Якщо в основі хмарних додатків класу IaaS лежать технології віртуалізації, то рішення класу PaaS крім віртуалізації вимагають додаткових інструментів, що дозволяють розробляти мережеві додатки з більшою ефективністю і меншими затратами. У міру розвитку технологій прийняте в даний час поділ хмарних обчислень на SaaS, IaaS і PaaS в найближчому майбутньому піде в минуле. Вернер Вогельс – технічний директор компанії Amazon, найбільшої в світі за обсягом продажів товарів і послуг через Інтернет, ще в 2011 констатував, що поділ на IaaS і PaaS встаріло. В хмарних додатках майбутнього не тільки поєднуватимуться інфраструктурні та платформні елементи від одного постачальника, але і різні сервіси, зібрані від різних постачальників. Що саме прийде на зміну моделі, поки важко сказати, і різні компанії висувають найрізноманітніші варіанти. Так, аналітики консалтингової компанії Gartner, що спеціалізується на ринках інформаційних технологій, вважають, що в кінцевому рахунку хмарні обчислення приведуть до появи концепції EaaS (Everything as a Service - все як послуга).[6]

1.2. Огляд сучасних хмарових сховищ. Їх недоліки та переваги.

1.2.1 Google Drive

Google Drive - хмарне сховище даних від Google. Google Drive дозволяє користувачам зберігати свої дані на серверах в хмарі і ділитися ними з іншими

користувачами в інтернеті. Хмарне сховище ділить простір між Google Drive, Gmail і Google Photo. У сервісі можна зберігати не тільки документи, а й фотографії, музику, відео та багато інших файлів - всього 30 типів.[7]

Таблиця 1.1

Тарифні плани Google Drive

Об’єм	Місячна плата
15 Гбайт	Бескоштовно
100 Гбайт	$1.99
1 Тбайт	$9.99
10 Тбайт	$99.99
20 Тбайт	$199.99
30 Тбайт	$299.99

Максимальний обсяг файлу 5 Гбайт. Можна використувати в веб-браузерах, Windows, Mac OS, Android, iOS і ін.

1.2.2 Microsoft OneDrive

OneDrive - перейменований в лютому 2014 Microsoft SkyDrive, базується на хмарнії організації інтернет-сервіс зберігання файлів з функціями файлообміну. SkyDrive створений в серпні 2007 року компанією Microsoft. Зараз OneDrive один з флагманів хмарних сховищ даних.

Перевага сервісу OneDrive в тому, що він відразу інтегрований з Office 365, тому безпосередньо з програми можна створювати, редагувати, зберігати файли Excel, OneNote, PowerPoint і Word в службі Windows Live OneDrive.

Сервіс OneDrive дозволяє зберігати на даний момент безкоштовно 5 Гбайт (раніше пропонувалося 15 Гбайт) інформації в упорядкованому за допомогою стандартних папок вигляді. Для зображень передбачений предпросмотр у вигляді ескізів, а також можливість їх перегляду в вигляді слайдів.[8]

Весь контент шифрується і розподіляється в центрі даних. Кожен файл що зберігається, в залежності від розміру, розбивається на один або кілька блоків. Потім кожен блок шифрується з використанням свого унікального ключа. Всі ці блоки зберігаються як великі двійкові об'єкти в сховище великих двійкових об'єктів. Вони також розподіляються між декількома контейнерами великих двійкових об'єктів. "Карта", яка використовується для повторного складання файлу з компонентів, зберігається в базі даних контенту. Кожен контейнер великих двійкових об'єктів використовує власні унікальні облікові дані для кожного типу доступу (на читання, запис, перерахування та видалення). Кожен набір облікових даних міститься в безпечному сховищі ключів і регулярно оновлюється.

OneDrive надає такі пакети:

OneDrive storage:

- Free: 5 Гбайт - безкоштовно;
- Basic: 50 Гбайт - $ 1,99 на місяць;

OneDrive + Office 365:

• Personal: 1 TB для 1 користувача - $ 6,99 на місяць;

• Home: 1 TB для кожного з 5 користувачів - $ 9,99 на місяць;

1.2.3 Dropbox

Dropbox - хмарне сховище даних, що дозволяє користувачам зберігати свої дані на серверах в хмарі і розділяти їх з іншими користувачами в інтернеті. Його робота побудована на синхронізації даних.

Особливості Dropbox:

• 256-бітове шифрування AES і шифрування SSL;

• Краща в своєму класі технологія синхронізації;

• Інтеграція з Microsoft Office 365;

• Необмежене відновлення файлів і журнал версій;

• Посилання доступу з паролем і терміном дії;

• Настроювані рівні доступу;

• Управління рівнями доступу та ін.

Таблиця 1.2

Тарифні плани Dropbox

Пакет	Місячна плата	Інфо
Базовий аккаунт	Бескоштовно	2 Гб
Dropbox Pro	9.99 €	1Тбайт
Dropbox Business	10 € / користувач / місяць	Стільки місця, скільки потрібно, необмежена кількість відновлення файлів, управління доступом до файлу, конфігуровані рівні доступу, пріоритетна технічна підтримка, 14 днів тріал безкоштовно
Dropbox Enterprise	Ціна залежить від необхідного функціоналу	Все те ж саме, що й в Dropbox Business, але на корпоративному рівні

Додаток Dropbox можна скачати і інсталювати на PC, Mac, Linux або мобільний пристрій. На відміну від основних конкурентів, при роботі з Dropbox редаговані файли не копіюються повністю на сервер - здійснюється передача тільки зміненої частини, попередньо стиснутої. Це забезпечує відому оперативність роботи Dropbox, у порівнянні з аналогами.[9]

У Dropbox розроблено кілька рівнів захисту, включаючи безпечну передачу даних, сховище зашифрованих блоків і засоби управління на рівні додатків, і розподілену інфраструктуру.

Головним недоліком Dropbox можна вважати підхід до вибору папок для синхронізації. Фактично додаток стежить за вмістом тільки однієї папки - Dropbox.

У липні 2014 в своєму інтерв'ю виданню The Guardian відомий Едвард Сноуден сказав, що Dropbox не в повній мірі піклується про конфіденційність даних користувача і навіть безпосередньо бере участь у глобальній системі стеження PRISM.

1.2.4 Mega

Mega - (MEGA Encrypted Global Access) - хмарний файлообмінник створений Кімом доткомів (Kim Dotcom)

Mega шифрує весь контент прямо в браузері за допомогою алгоритму AES; користувачі можуть передавати один одному файли в зашифрованому вигляді, при цьому всі дані зберігаються в «хмарі»; ключі доступу до файлів не публікуються у відкритому доступі, а поширюються по схемі Friend-to-Friend, між довіренними користувачами.

Таблиця 1.3

Тарифні плани Mega

Пакет	Місячна плата	Інфо
Базовий	Безкоштовно	50 Гбайт
Lite	4,99 €	200 Гбайт
Pro I	9,99 €	500 Гбайт
Pro II	19,99 €	2 Тб
Pro III	29,99 €	4 Тб

За наданим дисковим простором та по його вартості, Mega, безсумнівно можна назвати одним з найбільш вигідних хмарних сервісів, крім того, важлива відмінність Mega від інших подібних сервісів - конфіденційність, адже Mega позиціонується як сервіс, який захищає особисті дані користувача. Це досягається за рахунок наскрізного шифрування.

Все шифрування, розшифрування і генерація ключів реалізовані в JavaScript, який обмежує пропускну здатність до декількох MB / C і викликає значне навантаження CPU.[10]

Всі симетричні криптографічні операції засновані на AES-128. Він працює в режимі ланцюжка блоків шифру для файлу і блоки атрибутів папки і в режимі лічильника для фактичних даних файлу. Кожен файл і кожен вузол папки використовує свій власний випадково згенерований 128 біт ключ. Файлові вузли використовують один і той же ключ для блоку атрибутів і файлу дані, плюс 64-

бітове випадкове початкове значення лічильника і 64-бітний meta MAC щоб перевірити цілісність файлу. Кожна обліковий запис користувача використовує симетричний майстер-ключ до ECB-шифрування всіх ключів вузлів, які він зберігає в своєму власному дерева. Цей майстер-ключ зберігається на серверах MEGA, зашифрований за допомогою hash, отриманим з пароля користувача. Цілісність файлу-це перевірено за допомогою chunked CBC-MAC. Розміри блоків починаються від 128 KB і збільшення до 1 MB, що є розумним балансом між необхідним простором для зберігання фрагмента MACs і середніх накладних витрат для часткове читання з перевіркою цілісності. На додаток до симетричного ключа, кожна обліковий запис користувача має пару ключів 2048 біт RSA для безпечного отримання даних наприклад, ключі загального доступу або ключі файлів / папок. Його приватний компонент зберігається в зашифрованому вигляді за допомогою симетричного головного ключа користувача.

1.2.5 Яндекс.Диск

Яндекс.Диск - російський хмарний сервіс від Яндекс, що дозволяє користувачам зберігати свої дані на серверах в хмарі і передавати їх іншим користувачам в інтернеті. Робота побудована на синхронізації даних між різними пристроями.

Таблиця 1.4

Платні пакети хмарного сервісу Яндекс.Диск

Пакет	Місячна плата
10 Гб	Бескоштовно. Назавжди.
+10 Гбайт	30 руб.
+100 Гбайт	80 руб.
+1 Тбайт	200 руб.

Яндекс.Диск може виступати в якості служби хмарного сервісу, інтегруючись в офісний пакет Microsoft Office, також з'явилася можливість автоматичного завантаження фото і відеофайлів з цифрових камер і зовнішніх носіїв інформації на Яндекс. Диск.

Присутня двухфакторна аутентифікація, в тому числі з використанням пін-коду, QR-коду і TouchID. При завантаженні файли перевіряються на віруси, а дані передаються по шифрованому каналу.[11]

1.2.6 Облако@mail.ru

Облако@mail.ru - хмарне сховище даних від компанії Mail.Ru Group, що дозволяє користувачам зберігати свої дані в хмарі і синхронізувати дані на різних пристроях, а також ділитися ними з іншими користувачами.

Після реєстрації користувачі отримують безкоштовно 16 Гбайт хмарного сховища.

Користуватися хмарою можна не тільки через веб-інтерфейс, але через десктопні (для Windows і Mac OS) і мобільні додатки для Android і iOS.

Функція, з самого початку доступна в мобільних додатках - автозавантаження фотографій з телефону. Якщо включена ця функція, все фото, зроблені за допомогою пристрою, миттєво виявляються в «Хмарі».

У Облако@mail.ru декілька блоків тарифів:

• Мобільні тарифи (iOS, Android): 6 тарифних планів, від +8 Гбайт за 29 рублів на місяць до 256 Гбайт за 229 рублів в місяці.

• Веб-тарифи: 4 тарифних плани, від 512 Гбайт за 379 рублів на місяць до 4 Тбайт за 2790 рублів на місяць.

• Тарифи сервісу «Хмара для бізнесу». Фіксована плата - 3 рубля / ГБ в місяць, платите тільки за займане місце.

1.2.7 Amazon Web Services

Amazon Web Services (AWS) - це не просто хмарне сховище даних, а цілий набір глобальних сервісів. Таким чином, Amazon Web Services включає в себе широкий спектр фундаментальних сервісів хмарної інфраструктури:

• Зберігання даних і доставка контенту: об'єктне сховище, CDN, блочне сховище, сховище файлових систем, архівне сховище, перенесення даних, інтегроване сховище.

• Обчислення: віртуальні сервери, контейнери, розгортання веб-додатків методом 1-Click, керовані подіями обчислювальні функції, auto scaling, load balancing.

• Бази даних: реляційні БД, міграція баз даних, NoSQL, кешування, сховище даних.

• Мережеві рішення: віртуальна приватна хмара, прямі підключення, балансування навантаження, DNS.

Крім того, AWS має багатий вибір сервісів для підвищення ефективності вашого хмари: аналітика, корпоративні програми, мобільні сервіси, інтернет речей.

Вартість AWS гнучка, це означає, що виплатите тільки за те, чим користуєтеся. Мінімальний збір не стягується. Оцінити свій щомісячний рахунок можна за допомогою калькулятору матеріалів AWS. При цьому ціни залежать від місця розташування вашої кошика Amazon S3.

12 місяців можна тестувати AWS безкоштовно, при цьому можна користуватися сховищем об'ємом 5 ГБ, 20 000 запитів Get і 2 000 запитів Put при використанні сервісу Amazon S3.[12]

1.2.8 pCloud

pCloud — безпечний хмарний сервіс для зберігання файлів. Безпечнічність забезпечується завдяки двум факторам: по перше вся інформація передається на сервер pCloud через TLS/SSL протокол і розміщується на 3-х серверах з різним місцезнаходженням у високозахищенних дата-центрах; по друге при підписці на pCloud Crypto ($3.99 в місяц) можливо скористатися перевагою у вигляді шифрування на стороні кліента з спрощенним інтерфейсом шифрування.

Таблиця 1.5

Тарифи pCloud

Пакет	Місячная плата	Інфо
Базовий аккаунт	Бескоштовно	20 Гбайт
Преміум	$3,99	500 Гбайт
Преміум Плюс	$7,99	2 Тбайта

1.2.9 iCloud Drive

iCloud Drive - хмарне сховище даних від Apple. iCloud Drive встановлений на пристрої на всіх пристроях Apple за замовчуванням. Для того щоб використовувати сервіс потрібно мати обліковий запис iCloud. Для безпеки даних iCloud Drive шифрує дані за допомогою 128-бітного AES шифрування. Він також використовує 128-бітове SSL-шифрування для передачі.[13]

Таблиця 1.6

Тарифні плани iCloud Drive

Об’єм	Місячная плата
5 Гб	При реєстрації, бескоштовно
50 Гб	59 руб
200 Гб	149 руб
1 Тб	599 руб
2 Тб	1490 руб

До недоліків можна віднести:

- Швидкість завантаження була дуже повільної на деяких тестах.
- Є питання до безпеки: ви не отримаєте ключ опції приватного шифрування, що означає, що служба зберігає ключі шифрування на серверах і може отримати доступ до файлів без вашої згоди, якщо ви не використовуєте додаток шифрування.

1.2.10 4shared

4shared - хмарний файлообмінний хостинг.

Базова безкоштовна реєстрація дає можливість завантажувати до 10 Гбайт в свій обліковий запис. Після підтвердження реєстрації електронною поштою обсяг збільшується до 15 Гбайт.

Преміум-користувачі отримують у своє розпорядження 100 Гбайт:

• 1 місяць - $ 9,95 на місяць;

• 1 рік - $ 6,50 на місяць;

Після успішного завантаження файлу користувач отримує унікальну посилання, по якій інші згодом зможуть завантажити цей файл. Всі завантажені файли зберігаються протягом 180 днів з моменту останнього відвідування аккаунта. Файли преміум-користувачів зберігаються на весь термін дії облікового запису. Інтерфейс сервісу зовні схожий на провідник Windows.

1.2.11 SugarSync

SugarSync - хмарне сховище даних, яке зберігає документи, файли, фотографії і музику. Має простив і зручний інтерфейс можливо використовувати для різних пристроїв. Щоб оцінити переваги SugarSync є 30-денна безкоштовна тріал-версія для всіх пакетів. Для взаємодії з сервісом використовується зручна програма-клієнт, версії якої існують не тільки для Windows і Mac, але і для Android, iPhone, Symbian. Кількість синхронізуються пристроїв не обмежена.[14]

З лютого 2014 перейшла виключно на платні пакети сховищ даних:

Таблиця 1.7

Тарифи для фізичних осіб SugarSync

Об’єм	Місячная плата
100 Гб	$7,49
250 Гб	$9,99
500 Гб	$18,95

Основна відмінність SugarSync від Dropbox, що він синхронізує ті папки, які вкаже користувач.

1.2.12 Box.net

Box.net - хмарне сховище даних, яке дозволяє зберігати ваші файли в мережі, а також спільно над ними працювати.

Box.net дає два типи пакетів для персонального використання і бізнесу:

Персональні пакети:

• Безкоштовно: для 1 користувача надається 10 Гб, ліміт на розмір файлу - 250 Мбайт;

• Personal Pro: 8 € в місяць, надається 100 Гб, ліміт на розмір файлу - 5 Гб;

Бізнес пакети:

• Starter: 4 € в місяць: від 3 до 10 користувачів, надається 100 Гб, ліміт на розмір файлу - 2 Гб;

• Business: 12 € на місяць: від 3 користувачів, необмежений обсяг сховища, ліміт на розмір файлу - 5 Гб;

Переваги Box.net - це можливість перегляду офісних документів власними силами, а також можливість розшарити файли або папки для колег прямо з мобільного. Крім того, розробникам вдалося інтегрувати в додаток нативний пошук Android за рахунок чого пошук файлів став швидше і точніше.

1.2.13 iDrive

iDrive є ідеальним інструментом для онлайнового резервного копіювання з високим рівнем приватності.

Створення резервних копій можна робити з ваших PC-комп'ютерів, Mac, iPhone, iPad і Android пристроїв на одному акаунті. Досить високий рівень безпеки даних. Трансфер і зберігання файлів з 256-бітовим AES шифруванням з використанням ключа визначається користувачем, який ніде не зберігаються на серверах. Безкоштовно надається 5 Гб дискової квоти зберігання.[15]

IDrive шифрує файли за допомогою 256-бітного AES шифрування до того, як файли будуть завантажені і передані на сервери. Дані також захищені 256-розрядним шифруванням SSL передачі.[16]

iDrive надає такі пакети:

• Basic: Безкоштовно - 5 Гбайт - назавжди;

• iDrive Personal:

1 Тб - $ 52,12 на рік;

10 Тб - $ 374,62 в рік;

• iDrive Team:

1 Тб - $ 74,62 на рік;

2 Тб - $ 149,62 в рік;

10 Тб - $ 749,62 в рік;

• iDrive Business:

250 Гб - $ 74,62 на рік;

500 Гб - $ 149,62 в рік;

1,25 Тб - $ 374,62 в рік;

1.2.14 OpenDrive

OpenDrive - безлімітне хмарне сховище даних, з більш 1 мільйона користувачів, яке поширюється в таких пакетах:

- Personal Free: безкоштовно 5 Гб;
- Personal Unlimited: $ 12,95 в місяць, необмежений простір;
- Business Unlimited: $ 29,95 в місяць;
- Enterprise: $ 59,95 в місяць;

1.2.15 Syncplicity

Syncplicity - програмний комплекс для синхронізації даних в кроссплатформенних середовищах. Безкоштовно надається 10 Гб дискової квоти для 1 користувача і двох комп'ютерів.[17]

У Syncplicity також організована тісна інтеграція з деякими онлайновими сервісами. Можна завантажувати фотографії в альбоми на Facebook, просто копіюючи їх в певні папки на ПК. Також доступна інтеграція з офісним пакетом Google: будь-який документ, створений в Google Docs, буде з'являтися в обраній папці на комп'ютері, і навпаки. Таким чином, файл редагується як локально за допомогою Microsoft Office, так і в онлайні.[18]

1.2.16 MediaFire

MediaFire - онлайн сховище даних, файлообмінник. В даному хмарному сервісі упор зроблений саме на зберігання і шарінг медіа-файлів (музика, відео, фото).[19]

MediaFire надає такі пакети:

- Безкоштовно: 10 Гбайт, присутня реклама.
- Pro $ 2,49 на місяць: 1 Тбайт, без реклами, обмеження 20 Гбайт в одному файлі, прямі посилання на файли і ін.
- Business $ 24.99 в місяць: понад 100 Тбайт, без реклами, обмеження 20 Гбайт в одному файлі, прямі посилання на файли і ін.

Зручно, що MediaFire є як для вебу, так і для компьютера і мобільних додатків з дружнім інтерфейсом. Можна сміливо рекомендувати MediaFire як гарну альтернативу іншим хмарних сервісів, зокрема, Dropbox.[20]

1.2.17 SpiderOak

SpiderOak - вважається одним з найбезпечніших хмарних сховищ даних. Розробники запевняють, що пароль для доступу до аккаунту нікуди не передається, а при першій авторизації створюється спеціальний ключ, який і відкриває доступ настільного клієнта до хмарного сховища даних.[21]

Таблиця 1.8

Тарифні плани SpiderOak:

Об’єм	Місячна плата
250 Гб	Безкоштовно 21 день.
100 Гб	$5
250 Гб	$9
1 Тб	$12

1.2.18 ADrive

ADrive - хмарне сховище даних, яке дозволяє централізовано працювати зі своїми файлами: надійно зберігати, управляти і обмінюватися з усіма музикою, відео, фотографії та ін.

ADrive дає тільки платні пакети з безкоштовним 60 денним тестовим періодом.

Персональний план: стартує від $ 2.50 в місяць або $ 25,00 на рік і 100 Гбайтами. На цьому плані можна розширити дисковий простір до 20+ Тбайт.

Бізнес план: стартує від $ 7 на місяць або $ 70 в рік і 200 Гбайтами. На цьому плані можна розширити дисковий простір до необмеженого обсягу.[22]

Таблиця 1.9

Порівняння хмарних сховищ даних

Сервіс	**Google Drive**	**SkyDrive**	**Яндекс.Диск**	**iCloud**	**DropBox**	**SendSpace**
Безкоштовний об'єм	15 Гбайт	7 Гбайт	10 Гбайт	5 Гбайт	2 Гбайт	300 Мбайт/файл
Максимальний об'єм, Гбайт	200	100	1024	50	100	300
Термін збереження	Необм.	Необм.	Необм.	30 днів	Необм	30 днів
Пряме посилання на загрузку	так	ні	так	ні	так	так
Десктопні версії	Windows, Mac OS	Windows, Mac OS	Windows, Mac OS, Linux	Mac OS	Windows, Mac OS, Linux	Windows, Mac OS, Linux
Мобільні версії	Android, iOS	Android, iOS, Windows Phone	Android, iOS, Windows Phone	iOS	Android, iOS, Windows Phone, Symbian, Bada	Android, iOS

РОЗДІЛ 2

ДОСЛІДЖЕННЯ ЗАСОБІВ ДЛЯ РОЗРОБКИ КРИПТОГРАФІЧНОГО МОДУЛЯ

2.1 Порівняння криптоалгоритмів

Сучасні криптографічні системи повинні задовольняти такі загальноприйняті вимоги:

- вихідний текст із зашифрованого тексту можна відтворити лише за допомогою ключа дешифрування;
- послідовне перебирання можливих ключів дешифрування з метою відтворення вихідного тексту потребує значного часу обчислень або великих затрат на реалізацію цих обчислень;
- інформація про алгоритм шифрування не повинна впливати на стійкість до зламування системи шифрування;
- незначна зміна ключа шифрування повинна призводити до істотних змін шифрограми одного і того самого тексту.

2.2 Шифрування з ключем

Алгоритм шифрування з ключем поділяють на дві великі групи – алгоритми симетричного шифрування й алгоритми асиметричного шифрування.

Методи симетричного шифрування/дешифрування – це метод, за яким ключі шифрування і дешифрування є або однаковими, або легко обчислюються один з одного, забезпечуючи спільний ключ, який є таємним.[23]

Методи асиметричного шифрування/дешифрування – набір методів криптографічного шифрування/дешифрування, в якому використовують два ключі – таємний (приватний) і відкритий; жодний із ключів не може бути

обчислений з іншого за визначений час. Таке шифрування/дешифрування ще називають шифруванням/дешифруванням з відкритим ключем.

До 70-х років минулого століття застосовували лише криптографію з симетричними криптоалгоритмами. Криптографія з асиметричними криптоалгоритмами значно молодша. Симетричні та асиметричні криптоалгоритми мають переваги та недоліки. Симетричні криптоалгоритми порівняно з асиметричними мають більшу швидкодію та меншу довжину ключа. Асиметричне шифрування застосовують за такої організації криптосистем, коли використання симетричних алгоритмів є неможливим. А загалом порівнювати характеристики цих криптоалгоритмів було б некоректно: вони створені для розв'язування різних задач шифрування. [24]

2.3 Метод симетричного шифрування

Симетричне шифрування ще називають шифруванням з таємним ключем, тобто з ключем, який обидві сторони обміну інформацією (таємно від інших користувачів) використовують для шифрування та дешифрування повідомлень. Основне призначення симетричних криптоалгоритмів – шифрування великих масивів даних із великою швидкістю. Разом із тим, через необхідність наявності захищеного каналу передавання таємного ключа ці криптоалгоритми під час створення сучасних криптосистем виявляють дуже низьку гнучкість. Розрізняють дві великі групи алгоритмів симетричного шифрування: потокове шифрування та блокове шифрування.

2.4 Метод асиметричного шифрування

Проблему зростання обсягів шифрованої інформації у криптографії вирішують підвищенням швидкодії традиційних методів шифрування з таємним ключем. Проте застосування цих методів в умовах постійного зростання кількості учасників спільної роботи (децентралізована структура управління) й ускладнення організації взаємодії між ними, зокрема попарного обміну

інформацією, виявляється неефективним. Це зумовлено тим, що зі збільшенням кількості учасників обміну інформацією квадратично зростає кількість таємних ключів. Можна показати, що для N учасників кількість таємних ключів у такій системі сягає N(N-1)/2. Крім того, у методах симетричної криптографії з таємним ключем ускладнене довірене узгодження таємного ключа. З метою зменшення цих недоліків було розроблено методи асиметричного шифрування з відкритим ключем. Шифрування з відкритим ключем – порівняно нова галузь криптографії. [25]

В асиметричних криптоалгоритмах для шифрування і дешифрування використовують різні ключі: для шифрування – відкриті, для дешифрування – таємні. Асиметрична криптографія основана на ідеях В. Діффі та М. Хеллмана про шифрування з двома ключами, що стали відомими у 1976 році. Але першим алгоритмом асиметричного шифрування, що набув практичного значення, став алгоритм, який запропонували Р. Рівест, А. Шамір і Л. Адлеман у 1978 році. Він дістав назву алгоритм RSA. В таб. 2 зображені алгоритми що використовують в криптографії. Структурна схема шифрування з відкритим ключем Математичним обґрунтуванням асиметричних криптоалгоритмів є важкооборотні (односторонні) функції. У теорії складності обчислень розглядають поняття, яке характеризує рівень складності обчислень (кількість операцій) залежно від розміру вхідних даних. Поширеними є поліноміальний та експоненційний характери залежності складності обчислень від кількості вхідних даних. В асиметричній криптографії В. Діффі та М. Хеллмана зашифроване повідомлення за наявності таємного ключа дешифровується за поліноміальний час роботи обчислювальної системи, а у разі його відсутності – за експоненційний час. Сучасна асиметрична криптографія основана на алгоритмах Ель-Гамаля та Міллера-Коблиця. Теоретичну основу стійкості алгоритму RSA становить проблема факторизації великих цілих чисел, а алгоритмів Ель-Гамаля та Міллера-Коблиця – проблема дискретного логарифмування. В таб. 3 зображені характеристики алгоритмів з блоковим шифруванням. Сьогодні відомі численні вразливості цих алгоритмів. Алгоритми шифрування на відкритому ключі

замінили стійкіші алгоритми шифрування на еліптичних кривих, які запропонували окремо В. Міллер і Н. Коблиць у 1986 р. [26].

Таблиця 2.1.

Алгоритми, що використовуються в криптографії

Симетричні	З відкритим ключем
DES, New DES, AES, Blowfish, RC2, CAST, ГОСТ 28147-89	RSA

Таблиця 2.2.

Характеристики алгоритмів з блоковим шифруванням

Назва	Довжина ключа, біт	Розмір оброблюваних блоків, біт	Число раундів
AES (Rijndael)	128, 192 або 256	128, 192 або 256	10, 12, 14
Blowfish	32-448	64	16
CAST-128	128	64	16
CAST-256	256	128	16
DES	56	64	16
NewDES	120	64	17
RC2	до 1024	64	16
RC5	до 2048	32, 64 або 128	0...255
ГОСТ 28147-89	256	64	16 або 32

В даний час найбільш поширені алгоритми DES, Triple DES, AES, а в Росії - ГОСТ 28147-89. Досить широко застосовуються також алгоритми Blowfish компанії Counterpane Systems; Safer, розроблений для компанії Cylink (придбана в 2003 р компанією SafeNet,); RC2 і RC5 корпорації RSA Data Security і CAST компанії Entrust.[27]

DES (Digital Encryption Standard) - це блоковий шифр, який використовує 56-розрядний ключ. Алгоритм був розроблений в кінці 70-х рр. минулого століття дослідниками з IBM і National Security Agency (NSA). У 80-х експерти вважали, що алгоритм не має слабких місць, але з появою швидкодіючих комп'ютерів в 90-х його репутація кілька постраждала - стала можливою атака методом перебору ключа (ключ DES був зламаний фахівцями компанії Electronic Frontier Foundation в 1999 р менше ніж за 24 ч).[30]

Triple DES - вдосконалений блоковий алгоритм DES. Принцип його роботи не відрізняється від застосовуваного в DES, а посилення досягається завдяки трикратному (triple) шифрування одного блоку алгоритмом DES. Три 56-розрядних ключа, використовуваних в даному процесі, об'єднуються алгоритмом в один 168-розрядний ключ. І хоча час атаки перебором при звичайній потужності комп'ютера становить кілька мільярдів років, що говорить про хорошу стійкості алгоритму, в деяких публікаціях описані способи скорочення часу атаки - до рівня перебору 108-розрядного ключа. Сьогодні існує також варіант Triple DES з "подвійним" DES-ключем розміром 112 біт, і він застосовується частіше.

AES (Advanced Encryption Standard) - ще один блоковий алгоритм, який був розроблений бельгійськими дослідниками Вінсентом Ріджменом (Vincent Rijmen) і Джоан Дімен (Joan Daemen) і прийнятий як стандарт Національним інститутом стандартів і технологій (NIST) 2. жовтня 2000 р Конкурс на новий стандарт був оголошений трьома роками раніше, причому серед його умов значився обов'язковий відмова розробників від права інтелектуальної власності, що дозволяло зробити новий стандарт відкритим і застосовувати його без відрахувань авторам.

ГОСТ 28147-89 в США часто називають російським аналогом DES. Але в порівнянні з DES ГОСТ 28147-89 значно більше криптостійкий і складніший. Він був розроблений в одному з інститутів КДБ наприкінці 1970-х рр., Статус офіційного стандарту шифрування СРСР отримав в 1989 р, після розпаду СРСР прийнятий в якості стандарту Російської Федерації. ГОСТ 28147-89

оптимізований для застосування в програмних реалізаціях, використовує вдвічі більше DES-раундів шифрування з набагато більш простими операціями, а довжина ключа у нього в п'ять разів більше.

2.5 RSA

RSA - абревіатура від прізвищ Rivest, Shamir і Adleman криптографічний алгоритм з відкритим ключем, який базується на обчислювальної складності задачі факторизації великих цілих чисел. Описано в 1977 році Рон Ривест, Аді Шамір і Леонард Адлеман з Массачусетського технологічного інституту.

Криптосистема RSA стала першою системою, придатною і для шифрування, і для цифрового підпису. Алгоритм використовується у великій кількості криптографічних додатків, включаючи PGP, S / MIME, TLS / SSL, IPSEC / IKE і інших. Алгоритм Rivest-Shamir-Adleman (RSA) є одним з найбільш популярних і безпечних методів шифрування з відкритим ключем. Алгоритм грунтується на тому, що немає ефективного способу обліку дуже великих (100-200 цифр) чисел.[36]

RSA-ключі генеруються наступним чином:

1. Вибираються два різних випадкових простих числа і заданого розміру (наприклад, 1024 біта кожне).

2. Обчислюється їх твір, яке називається модулем.

3. Обчислюється значення функції Ейлера від числа:

4. Вибирається ціле число, взаємно просте зі значенням функції φ (n). Зазвичай в якості беруть прості числа, що містять невелику кількість одиничних біт в двійковій запису, наприклад, прості числа Ферма 17, 257 або 65537.

- Число називається відкритою експонентою (англ. Public exponent)
- Час, необхідний для шифрування з використанням швидкого зведення в ступінь, пропорційно числу одиничних біт в.

– Занадто малі значення, наприклад 3, потенційно можуть послабити безпеку схеми RSA.

5. Обчислюється число, мультиплікативно зворотне до числа по модулю, тобто число, яке задовольняє порівнянні:

Число називається секретної експонентою. Зазвичай, воно обчислюється за допомогою розширеного алгоритму Евкліда.

6. Пара публікується в якості відкритого ключа RSA.

7. Пара грає роль закритого ключа RSA і тримається в секреті.

Безпека RSA залежить від обчислювальної складності факторизації великих цілих чисел. У міру збільшення обчислювальної потужності і виявлення більш ефективних алгоритмів факторингу збільшується і здатність до збільшення числа і великих чисел. Шифрування безпосередньо прив'язана до розміру ключа, а подвоєння довжини ключа забезпечує експоненціальне збільшення міцності, хоча і знижує продуктивність.

DH (Diffie-Hellman) був створений Уітфілд Діффі і Мартіном Хеллманом і опублікований в 1976 р По суті Діффі і Хеллмана запропонували схему, по якій шляхом обміну відкритою інформацією можна створити спільний секретний ключ. Фактично алгоритм DH - це не алгоритм шифрування, а схема розподілу ключів і алгоритм створення симетричного сеансового ключа. Згідно DH, кожна з взаємодіючих сторін має секретним і відкритим значеннями ключа. При об'єднанні секретного значення з іншим відкритим кожен користувач зможе створити один і той же секретний ключ.[42]

2.6 Вибір криптоалгоритму

Розглянемо детальніше процес шифрування / дешифрування. Криптосистема повинна відповідати таким вимогам:

– Зашифроване повідомлення повинно піддаватися читання тільки при наявності ключа;

- Число операцій, необхідних для визначення використаного ключа шифрування за фрагментом шифрованого повідомлення і відповідного йому відкритого тексту, має бути не менше загального числа можливих ключів;
- Число операцій, необхідних для розшифрування інформації шляхом перебору можливих ключів повинно мати строгу нижню оцінку і виходити за межі можливостей сучасних комп'ютерів (з урахуванням можливості використання мережевих розподілених обчислень);
- Знання алгоритму шифрування не повинно впливати на надійність захисту;
- Незначна зміна ключа повинно приводити до істотної зміни виду зашифрованого повідомлення;
- Структурні елементи алгоритму шифрування повинні бути незмінними;
- Додаткові біти, що вводяться в повідомлення в процесі шифрування, повинен бути повністю та надійно сховані в зашифрованому тексті;
- Довжина шифрованого тексту, повинна дорівнювати довжині вихідного тексту;
- Не повинно бути простих і легко встановлюваних залежністю між ключами, послідовно використовуваними в процесі шифрування;
- Будь-який ключ з безлічі можливих ключів повинен забезпечувати надійний захист інформації;
- Алгоритм повинен допускати як програмну, так і апаратну реалізацію, при цьому зміна довжини ключа не повинна вести до якісного погіршення алгоритму шифрування.

Відповідно до критеріїв оцінки алгоритмів криптографічного захисту інформації рекомендується вибирати алгоритм, який є найефективнішим. Критерій «безпека» має найбільший пріоритет і здійснює найбільший вплив , а

критерії «швидкість» і «характеристика алгоритму» є вторинними щодо «безпеки». Керуючись необхідністю забезпечення надійної безпеки алгоритмів від атак, можна сказати, що щодо безпеки MARS, Serpent і Twofish мають високий рівень захисту, але RC6, RSA і Rijndael мають вищий і надійніший захист. RC6 і Rijndael загалом демонструють швидкість шифрування і дешифрування, вищу за середню для 128-бітних ключів, але щодо 32-бітних платформ RC6 має найбільшу швидкість. MARS має середню швидкість виконання цих дій. Для Twofish час, затрачений на шифрування і дешифрування, відрізняється, але в обох випадках рівень є вищим за середній. RSA показав найнижчий показник порівняно з іншими алгоритмами. Rijndael потребує невеликих затрат оперативної пам'яті і відповідно є найкращим за обмежених можливостей. Serpent також забезпечує належний рівень шифрування та дешифрування за малої оперативної пам'яті. RC6 має невелику оперативну пам'ять, що є позитивним в обмеженому просторі, але має недолік при безперервній здатності обчислення підключів для дешифрування, – високу вимогу до оперативної пам'яті щодо інших алгоритмів.[44]

Serpent і Rijndael мають найкращу апаратну продуктивність для обох способів зворотного і незворотного зв'язку. Serpent має найвищу продуктивність в незворотному зв'язку, Rijndael пропонує найкращу ефективність роботи у зворотному зв'язку. RC6 і Twofish мають середню продуктивність, і обидва алгоритми можуть виконуватись компактно. RSA має високі вимоги і загалом його продуктивність є нижча від середнього рівня. Під час атак на виконання добре себе проявили алгоритми Rijndael, RSA і Serpent, швидко виявляючи і запобігаючи їм. Довше і з більшою складністю виконує Twofish, а RC6 і MARS з найбільшою затратою часу і труднощам протидіють атакам. Twofish, MARS і RC6 потребують мало додаткового простору, щоб здійснювати шифрування та дешифрування. Хоч Rijndael у цьому аспекті поступається за швидкістю, але може розділяти деякі технічні засоби. Twofish підтримує безперервне обчислення, підрахунок підключів як для шифрування, так і для дешифрування.

Serpent також підтримує безперервний підрахунок підключів як для шифрування, так і для дешифрування; проте процес дешифрування вимагає

одного додаткового обчислення підрахунку. Алгоритм Rijndael підтримує безперервне обчислення підключів для шифрування, але вимагає попереднього одноразового виконання повного ключового списку до ранішого дешифрування зі специфічним ключем. MARS має особливі характеристики, які є схожими до Rijndael, але додатково навантажує ресурс на MARS виконання. RC6 підтримує безперервне обчислення підключів тільки для шифрування. Кожен з алгоритмів забезпечує надійну захищеність і має певні переваги у деяких галузях порівняно з іншими.

Незважаючи на те що RSA поступається Rijndael в швидкості шифрування буде використовуватися RSA. Тому що, він потенційно дає більшу стійкість. Швидкість шифрування буде збільшена за рахунок паралельно шифрування в декількох потоках

У 2010 році групі вчених з Швейцарії, Японії, Франції, Нідерландів, Німеччини і США вдалося успішно обчислити дані, зашифровані за допомогою криптографічного ключа стандарту RSA довжиною 768 біт. Знаходження простих співмножників здійснювалося загальним методом решета числового поля. За словами дослідників, після їх роботи в якості надійної системи шифрування можна розглядати тільки RSA-ключі довжиною 1024 біта і більш. Причому від шифрування ключем довжиною в 1024 біт варто відмовитися в найближчі три-чотири роки. З 31 грудня 2013 року браузери Mozilla перестали підтримувати сертифікати центрів з ключами RSA менше 2048 біт.

З цих причин буде використовуватися алгоритм RSA довжиною ключа 2048 біт.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЯ ТА ЙОГО ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ

3.1. Клієнт-серверна архітектура

Додаток є розрахованих на багато користувачів і призначене для використання в мережі. Традиційно для такого роду додатків застосовуються два підходи: централізоване управління передачею повідомлень між користувачами або розподіл функцій синхронізації повідомлень між усіма учасниками мережевої взаємодії.

Перший підхід є, по суті, застосуванням технології «клієнт-сервер». При цьому підході створюється виділений сервер, який обслуговує запити клієнтів, зберігає дані для управління станом клієнтів і виконує синхронізацію даних між ними. Він відповідає за прийом даних від кожного клієнта і поширення цих даних між іншими. В цьому випадку всі клієнти є рівноправними щодо розподілу навантаження і взаємодіють виключно з сервером.

Іноді серверна частина може включатися в клієнтську програму. В цьому випадку відбувається взаємодія типу «клієнт-клієнт». Тоді один з клієнтів за попередньою домовленістю несе на собі функції сервера, в результаті будучи і клієнтом, і сервером. Така реалізація в разі великої кількості користувачів призводить до зайвого навантаження на одного з учасників. Це, безсумнівно, є вузьким місцем такого підходу. Ще одним з його недоліків є «обваження» клієнтської частини. На додаток до сказаного зазначимо, що поєднання функцій клієнта і сервера в одному додатку може бути виправдане при реалізації підключення «точка-точка», коли обидві сторони в різний час можуть мінятися ролями.

Другий підхід заснований на мережевій взаємодії без використання сервера, як такого. У цьому випадку кожен учасник комунікаційного процесу сам

здійснює розсилку даних всім іншим учасникам, наприклад, за допомогою широкомовлення. Це дозволяє заощадити необхідні на забезпечення підтримки сервера ресурси. Але така реалізація, в основному, підходить для комунікації в умовах локальної мережі.

Наш додаток має дозволяти користувачам спілкуватися за допомогою інтернету. У цих умовах для забезпечення комунікації безлічі користувачів застосування клієнт-серверної технології забезпечить наступні переваги:

- мінімізація навантаження на клієнта;
- відсутність необхідності встановлювати пряме з'єднання між клієнтами;
- централізація діяльності з обслуговування клієнтів в одному місці.

Дослідження сучасних сервісів зберігання інформації показало що найбільша уразливість конфідеціальності та цілістності даних не у передачі по мережі а у політиці конфіденційності та ліцензійному договіру згідно якому компанії аналізують данні користувачів і можуть видалити данні якщо компанія вирішить їх зміст неприйнятним в цьому випадку також можливе блокування акаунту що призведе до повної втрати власних даних. Компанії також можуть передавати дані користувачів третій стророні наприклад в разі запиту правохоронних органів. Забезбечити конфіденційність можна лише якщо організація що контролює сервер гарантує що ніколи не розкриватиме данні третій стороні але це мало ймовірно зважаючи на законодавство більшості крайн. Або якщо компанія користувач має власний сервер який фізично контролює тому в розробленній системі, було вирішено стровирити власну кліент-серверну систему.

3.2 Тестування додатку

Порядок дій:

1. Запускаємо сервер відкривши виконавчий файл.

2. Після запуску сервера можемо приступати до роботи з клієнтом.

3. При запуску клієнта відкривається вікно входу. (рис.3.1)

Рис. 3.1. вікно входу додатку

У першому вікні при запуску клієнта ми можемо зареєструвати нового користувача, кнопка addUser, або увійти під уже зареєстрованим обліковим записом, заповнивши поле ім'я та пароль і натиснувши кнопку увійти.

При додаванні користувача відкриється нове вікно яке попросить ввести ім'я нового користувача і пароль два рази що зображено на рисунку 3.2.

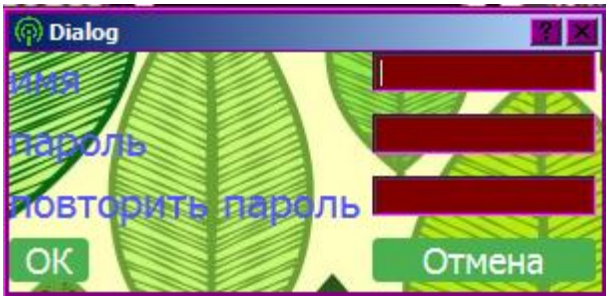

Рис 3.2. форма для нового користувача у додатку

Якщо пароль не співпадає або користувач з таким ім’ям вже існує, з’явиться інформаційне повідомлення.

При спробі входу перевіряється існування користувача і збіги пароля якщо ім'я користувача не існує або невірний пароль з'явиться повідомлення, що зображено на рисунку 3.3.

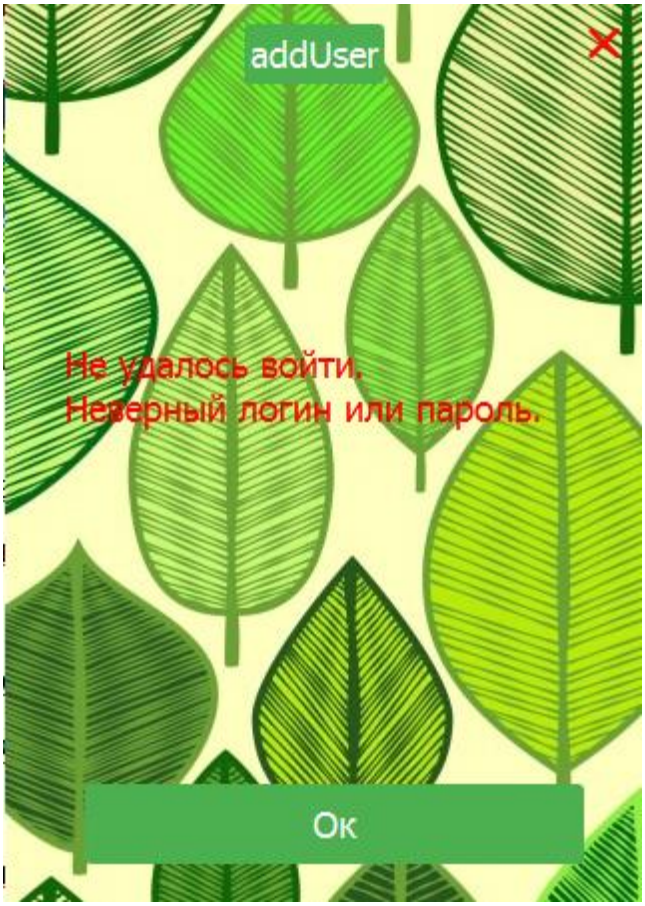

Рис 3.3. повідомлення про помилку у додатку

4. При вдалому вході у систему виконується відкриття діалогових вікон, що зображено на рисунку 3.4.

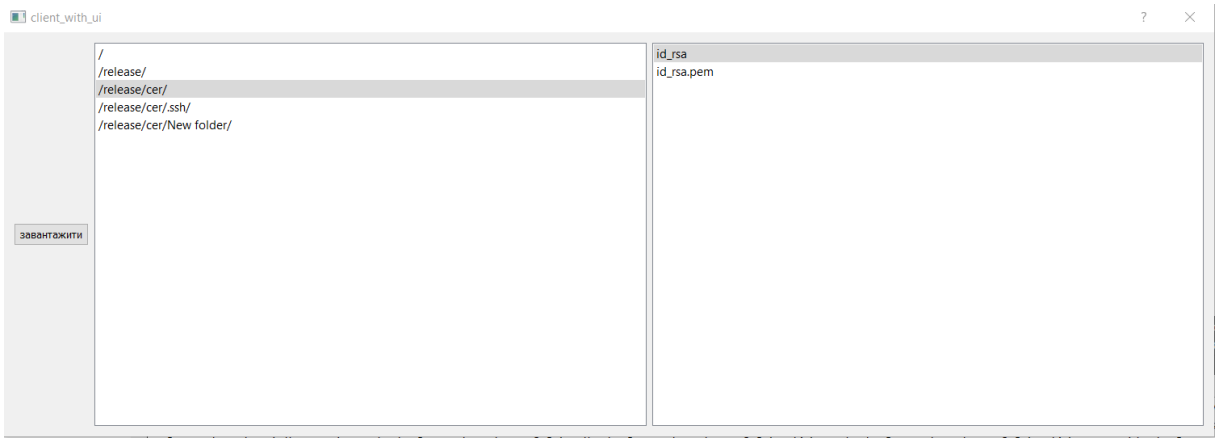

Рис 3.4. вдалий вхід у додаток

5. В лівій частині ми бачимо керуючі клавіши да діалогових вікна вибору папки та файла. Вибравши файл та натиснувши завантажити виконується завантаження файлу з сервера.

ВИСНОВКИ

У процесі аналізу сучасних сервісів зберігання інформації була виявлена загроза конфіденційності та цілісності інформації через політику використання згідно якої компанії аналізують данні користувачів і можуть видалити данні якщо компанія вирішить їх зміст неприйнятним в цьому випадку також можливе блокування акаунту що призведе до повної втрати власних даних. Компанії також можуть передавати дані користувачів третій стророні наприклад в разі запиту правохоронних органів. Забезбечити конфіденційність можна лише якщо компанія користувач має власний сервер який фізично контролює тому в розробленній системі, було вирішено стровирити власну кліент-серверну систему.

У процесі розробки було вирішено здійснювати шифрування за допомогою алгоритма RSA що забезпечує конфіденційність при передачі даних через відкриті канали зв’язку.

У процесі тестування розробки було виявлено що зловмисник не може отримати доступ до конфеденційних даних користувача як через відкриті канали зв’язку завдяки шифруванню алгоритмом RSA так і при доступі до машини користувача завдяки тому що було вирішено повністю відмовитися від збереження будь-яких даних користувача на стороні клієнта.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Шнайер, Б. Прикладная криптология Б. Шнайер. – М. : Триумф, 2002.

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

3. Зима Владимир, Молдовян Александр, Молдовян Николай. Безопасность глобальных сетевых технологий. СПб.:БХВ-Петербург, 2003.

4. Гордейчик С. В., Дубровин В. В. Безопасность беспроводных сетей. Москва.: Горячая линия - Телеком, 2008.

5. Леонтьев В. Безопасность в сети Интернет. Москва: ОЛМА Медиа Групп. 2008.

6. Коржов В. Многоуровневые системы клиент-сервер. Открытые системы, 1997.

7. Сударев, И. В. Криптографическая защита телефонных сообщений И. В. Сударев // Специальная техника. – 1998. – № 2. – С. 47 – 55.

8. Смит, Р. Разгадка тайны AES Р. Смит // LAN magazine / журнал сетевых решений. – 2001. – Том. 7, № 5. – С. 110-115.

9. Касаткин Ф. Связанные одной сетью // ИТ-диалог. 2014. № 6. С. 20–23.

10. ISO/IEC 19790 : 2012. Information technology – Security techniques – Security requirements for cryptographic modules. – International standard, Geneva, 2012. – 72p.

11. Terryn W. Fips 140-3. – International Book Marketing Service Limited, 2011. – 76p.

12. Standaert, F.-X. Introduction to Side-Channel Attacks / F.-X. Standaert // Secure Integrated Circuits and Systems. – 2009. – P. 27 – 44.

13. Kocher, P. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS and Other Systems / P. Kocher // Computer Science. – Santa-Barbara, California, USA, 1996. – vol 1109. – P. 104 – 113.

14. Boneh, D. DeMillo, R. A. Lipton, R. J. On the importance of checking cryptographic protocols for faults. / D. Boneh, R. A. DeMillo, R. J. Lipton. // EUROCRYPT '97. – 1997. – LNCS 1233. – P. 37 – 51.

15. Kocher, P. Jaffe, J. Jun, B. Differential Power Analysis / P. Kocher, J. Jaffe, B. Jun // Computer Science. – Santa-Barbara, California, USA, 1999. – vol 1666. – P. 398 – 412.

16. Ростовцев А.Г. Теоретическая криптография / А.Г. Ростовцев, Е.Б. Маховенко. – СПб: АНО НПО «Профессионал», 2004. – 480с.

17. Бернет С. Криптография. Официальное руководство RSA Security / С. Бернет, С. Пэйн. – М.: БиномПресс, 2002 г. – 384 с.

18. Жилин А.В., Корнейко А.В., Мохор В.В. Использование RSA алгоритма для обеспечения задач криптографической защиты информации в современных информационно-телекоммуникационных системах // Захист інформації. – 2013. – Том 15. – № 3. – С. 225-230.

19. Adleman L. A method for obtaining digital signatures and public-key cryptosystems / L. Adleman, R.L. Rivest, A. Shamir // Comm. ACM 21. – 1978. – P. 120-126.

20. RSA Cryptography Standard: PKCS #1 v 2.1 – RSA Laboratories, 2002. – P. 62.

21. Standard Specifications for Public Key Cryptography: IEEE Std 1363-2000. – IEEE, 2000.

22. Song Y. Yan. Cryptanalytic attacks on RSA / Song Y. Yan – Springer Science and Business Media, Inc. 2008. – P. 255

23. Коутинхо С. Введение в теорию чисел. Алгоритм RSA / С. Коутинхо. – М.: Постмаркет, 2001. – 328 с.

24. Мао Венбо. Современная криптография: теория и практика: пер. с англ./ Мао Венбо – М. : Изд. дом «Вильямс», 2005. – 768 с. 10 Информационные технологии. Методы и средства обеспечения безопасности. Алгоритмы шифрования. Часть 2. Асимметрические коды: ISO/IEC 18033-2:2006, 2006.

25. Нестеренко О.В. Теоретичні та методологічні основи побудови автоматизованих інформаційноаналітичних систем органів державної влади. Автореферат ... доктора техн. наук / НАН України. Національна бібліотека України ім. В. І. Вернадського. – К., 2006. – 41 с.

26. Джефф С. Документооборот по сети Internet // ComputerWorld Россия. – 1996. – 243 с.

27. Жовтенко В. В. Формування системи документальних ресурсів електронного урядування: досвід зарубіжних країн щодо її впровадження // Документознавство. Бібліотекознавство. Інформаційна діяльність: проблеми науки, освіти, практики: Зб. матеріалів III Міжнар. наук.-практ. конф., Київ, 16-18 травня 2006 р. / Редкол.: Г.В. Боряк, Г.В. Власова, Л.А. Дубровіна, М.С. Слободяник та ін. – К. : 2006. – С. 21-23

28. Колесов А. Автоматизация документооборота в Восточной Сибири // PCWeek/RE, Проекты. – 1997. – № 39 (113). – С. 71–81.

29. Гавриш С. ДІЛО – система автоматизації діловодства та електронного документообігу // Довідник секретаря та офіс-менеджера. – 2007. – № 2. – С. 25–31.

30. Правила посиленої сертифікації. Затверджені наказом ДСТСЗІ СБ України №3 від 13.01.2005 р. та зареєстрованих в Міністерстві юстиції України за №104/10384 від 27.01.2005 р.

31. Чирський Ю. Електронний цифровий підпис: правові аспекти застосування // Довідник секретаря та офіс-менеджера. – 2007. – № 1. – С. 17–22.

32. Мелащенко А.О., Перевозчикова О.Л. Організація кваліфікованої інфраструктури відкритих ключів. – К.: Наукова думка, 2010.

33. ISO/IEC 29500:2008 Information technology – Document description and processing languages – Office Open XML File Formats.

34. ISO/IEC 26300:2010 Information technology – Open Document Format for Office Applications (OpenDocument), v1.2.

35. ISO/TS 15000 Electronic business eXtensible Markup Language.

36. ISO/IEC 9594-8:2008 Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks.

37. ДСТУ EN 14890:2011 Прикладний інтерфейс для смарт-карток, використовуваних як безпечні засоби створення підписів.

38. Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures.

39. Куликова, А. С. Реализация многоверсионного поточного криптопреобразования данных с использованием бесключевых хеш-функций на программируемой логике А. С. Куликова, И. В. Лысенко // Системи обробки інформації. –2012. – № 7 (105). – С. 22 - 26.

40. Kulanov, V. Parameterized IP Infrastructures for Fault-Tolerant FPGA-Based Systems: Development, Assessment, Case-Study V. Kulanov, V. Kharchenko, A. Perepelitsyn // Proceedings of IEEE East-West Design & Test Symposium (EWDTS 2019), 2009. – P. 322–325.

42. Качанов, А. П. Анализ алгоритмов шифрования данных с открытым ключом А. П. Качанов, А. С. Дацько // Автоматика та приладобудування. Вестник НТУ "ХПИ". – 2011. – № 11. – С. 56-61.

43. Баркалов, А. А. Реализация алгоритма шифрования DES на базе FPGA А. А. Баркалов, А. А. Красичков, В. О. Кузьменко // Наукові праці ДонНТУ. – 2009. – Вип. 147. – С. 116-120.