**Форма N Н-9.02**

\_Національний університет «Полтавська політехніка імені Юрія Кондратюка»\_

(повне найменування вищого навчального закладу)

Навчально-науковий інститут інформаційних технологій та робототехніки

(повна назва факультету)

\_Кафедра комп’ютерних та інформаційних технологій і систем\_

(повна назва кафедри)

**Пояснювальна записка**

**до дипломного проекту (роботи)**

\_\_\_\_\_\_**бакалавра**\_\_\_\_\_\_\_\_\_\_

(рівень вищої освіти)

Аналіз стану та розробка системи захисту інформації на підприємстві “НЕТСО”

Виконав: студент 4 курсу, групи \_402\_ТН \_

спеціальності

\_\_\_\_122 Комп’ютерні науки\_\_\_\_\_\_\_\_

(шифр і назва напряму підготовки)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Саркісян А. С.\_\_

(прізвище та ініціали)

Керівник \_\_\_\_\_\_Беседін В. Ф.\_\_\_\_\_

(прізвище та ініціали)

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище та ініціали)

Полтава – 2021 року

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА»**

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ**

**КАФЕДРА КОМП’ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І**

**СИСТЕМ**

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВР**

**Спеціальність 122 «Комп’ютерні науки»**

**на тему**

**«АНАЛІЗ СТАНУ ТА РОЗРОБКА СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ НА ПІДПРИЄМСТВІ “НЕТСО” »**

**Студента групи 402-ТН Саркісяна Армана Сергійовича**

Керівник роботи

доктор економічних наук,

професор Беседін В. Ф.

Консультант

кандидат технічних наук, доцент Головко Г.В.

Завідувач кафедри

кандидат технічних наук, доцент Головко Г.В.

Полтава – 2021

**РЕФЕРАТ**

Пояснювальна записка містить: \_\_81\_ сторінок, \_16\_\_ малюнків, 7 таблиць, 2 додатки, 11 джерел.

**Об’єкт дослідження:** розробка та аналіз систем комплексного захисту інформації на підприємстві “НЕТСО”.

**Мета роботи:** проаналізувати стан та розробити рекомендації щодо систем комплексного захисту інформації на підприємстві “НЕТСО”.

**Методи:** Аналіз стану та розробка комплексного захисту інформації на підприємстві «Науково-практичний центр сприяння розвитку конкуренцій та захисту прав споживачів».

**Ключові слова:** захист інформації, інформаційна безпека, безпека інформації, комплексні методи захисту інформації, методи захисту інформації.

**ABSTRACT**

The explanatory note contains: \_\_81\_ pages, \_16\_\_ figures, 7 tables, 2 appendices, 11 sources.

**Object of research**: development and analysis of systems of complex information protection at the enterprise "NETSO".

**Purpose**: to analyze the state and develop recommendations for integrated information protection systems at the enterprise "NETSO".

**Methods**: Analysis of the state and development of comprehensive information protection at the enterprise "Scientific and Practical Center for Competition Development and Consumer Protection".

**Key words**: information protection, information security, information security, complex methods of information protection, methods of information protection.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

**ІБ –** Інформаційна безпека.

**ЗІ** **–** Захист інформації.

**СЗІ –** Системи захисту інформації.

**ІС –** Інформаційна система.

**ДТ** – Державна таємниця.

**СУБД** – Система управління базами даних.

**ПЗ** – Програмне забезпечення.

**C# (C Sharp)** – об’єктно орієнтована мова програмування.

**ПК** – персональний комп’ютер.

**Blowfish** – Алгоритм шифрування.

# [ВСТУП](https://mail.yandex.ua/neo2/#_Toc325035669)

Інформація є активом, котра, як і інші важливі бізнес активи, становить цінність для організацій та підприємств, отже, потребує відповідного захисту.

Окрім того, що інформацію можуть створювати, зберігати, змінювати, передавати, використовувати вона також може бути: ушкодженою, втраченою, викраденою або знищеною. Задля запобігання несанкціонованих операцій над інформацією дедалі компанії впроваджують методи захисту інформації.

При аналізі проблематики, пов'язаної з інформаційною безпекою, необхідно враховувати специфіку даного аспекту безпеки полягає в тому, що інформаційна безпека є складова частина інформаційних технологій – області, що розвивається безпрецедентно високими темпами. Тут важливі не стільки окремі рішення (закони, навчальні курси, програмно-технічні засоби), що знаходяться на сучасному рівні, скільки механізми генерації нових рішень, що дозволяють жити в темпі технічного прогресу [1].

Збільшення числа атак – ще не найбільша неприємність. Гірше те, що постійно виявляються нові вразливі місця в програмному забезпеченні і, як наслідок, виникають нові види атак.

В таких умовах системи інформаційної безпеки повинні вміти протистояти різноманітним атакам, як зовнішнім, так і внутрішнім, атакам автоматизованим і скоординованим. Іноді напад триває частки секунди; інколи знаходження вразливих місць ведеться повільно і продовжується годинами, так що підозріла активність практично непомітна. Метою зловмисників можливо порушення всіх складових ІБ – доступності, цілісності або конфіденційності.

Незалежний дослідник в галузі інформаційної безпеки Ponemon Institute спільно з IBM Security опитав 383 компанії з 12 країн і з'ясував, що за останній рік розмір збитку від витоків конфіденційних даних зросла з $ 3,8 млн до $ 4 млн.

Офіційні звіти повідомляють, що середній збиток від втрати кожного окремого запису виріс з $ 154 до $ 158. Експерти також встановили, що використання засобів інформаційного захисту та швидка реакція на інциденти допомагає компаніям знизити витрати на ліквідацію наслідків. Так, якщо витоку стоденної давності обходяться компаніям приблизно в $ 4,3 млн, то більш раннє виявлення мінімізує витрати до $ 3,23 млн.[2].

У першому півріччі 2020 року світі стався витік 7,78 млрд записів з персональної і платіжною інформацією. Про це йдеться в звіті компанії InfoWatch, що спеціалізується на інформаційній безпеці (документ є у РБК). Цей показник майже у вісім разів вище, ніж за аналогічний період 2016 роки (1,06 млрд записів), і більш ніж удвічі перевищує кількість даних, які потрапили в руки третіх осіб за весь 2016 рік (3 млрд записів).

Велика частина витоку даних (98%) сталася в результаті великих інцидентів, які InfoWatch називає «мегавитоками». Всього в компанії зафіксували 20 подібних випадків, коли в розпорядженні третіх осіб виявлялося більше 10 млн записів конфіденційних даних.

«Загальносвітовий тренд на збільшення числа витоків і обсягів скомпрометованих даних, на нашу думку, задають не особливості окремих регіонів, а нові можливості, які пов'язані з використанням інформації в цифровому світі, такі як переклад послуг в електронний вигляд, електронні гроші, об'єкти виняткових прав (інтелектуальна власність) в цифровому вигляді», – йдеться в звіті InfoWatch.

Всього за перше півріччя 2020 року відбулася 925 інцидентів, пов'язаних з втратою конфіденційної інформації, що на 10% більше, ніж за той же період 2016 року. Через зовнішнього втручання, наприклад з вини хакерів, відбулися 384 таких інциденту. Через внутрішні порушень (наприклад, з вини співробітників компаній, випадково або навмисно оприлюднили конфіденційну інформацію) – 520 витоків. Причини залишилася 21 витоку з'ясувати не вдалося[2].

Обсяг скомпрометованих даних в результаті атак ззовні виявився значніше (5,23 млрд записів), ніж через внутрішні порушень (2,32 млрд). У той же час, на думку InfoWatch, звіт охоплює не більше 1% всіх випадків витоку даних, так як заснований на аналізі публічної інформації, а випадки компрометації конфіденційних даних часто ховаються компаніями.

Збільшення кількості атак в останні два роки, є показником того що ставлення до максимального забезпечення безпеки інформації є неналежним, тому що більшість атак було здійснено через те, що зловмисники використали вади операційних систем застарілих версій. Використані недоліки були присутні лише на версіях ОС які були розроблені в кращому випадку 10 років тому, в останніх версіях ОС цього виробника дані недоліки були відсутні. Постраждали компаній та організації всіх рівнів, від аеропортів та банківських систем та Національної системи охорони здоров’я Великобританії, до малих компаній. Збитки за різними даними буле більше ніж $1 млрд.[2].

# РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

## 1.1 Поняття захисту інформації

1.1.1 Визначення захисту інформації та інформаційної безпеки. Захист інформації – сукупність методів та заходів , які впливають на збереження конфіденційності, цілісності у обміні та взаємодії з інформацією. Інформаційний захист і підтримка його належної інфраструктури для запобігання від випадкових або зловмисних дій, які можуть нанести збитки суб’єктам обміну інформацією (їх власникам і користувачам інфраструктури).

Якість комерційної інформації забезпечує необхідний економічний ефект для компанії, тому важливо охороняти критично важливі дані від неправомірних дій. Це дозволить компанії успішно конкурувати на ринку.

Інформаційна безпека – це стан, який визначає захищеність інформації у системах де вона зберігається, при якому головним є впровадження одного або комплексу методів задля забезпечення доступності, конфіденційності, цілісності. Слід враховувати, що інформаційна безпека є частиною інформаційних технологій галузі, що розвивається надто високими темпами. Важливі не стільки окремі рішення (закони, державні стандарти, програмні засоби), котрі знаходяться на високому сучасному технологічному рівні, скільки механізми генерації нових рішень та усвідомлення важливості їх впровадження на національному рівні, окремих галузей, корпоративних або персональних рівнів [3].

Безпека інформації, яка обробляється в організації, – це комплекс дій, спрямованих на вирішення проблеми захисту інформаційного середовища в рамках компанії. При цьому інформація не повинна бути обмежена у використанні і динамічний розвиток для уповноважених осіб.

Захист інформаційних ресурсів повинен бути:

* Постійним. Зловмисник в будь-який момент може спробувати обійти модулі захисту даних, які його цікавлять.
* Цільовим. Інформація повинна захищатися в рамках певної мети, яку ставить організація або власник даних.
* Плановим. Всі методи захисту повинні відповідати державним стандартам, законам і підзаконним актам, які регулюють питання захисту конфіденційних даних.
* Активним. Заходи для підтримки роботи та вдосконалення системи захисту повинні проводитися регулярно.
* Комплексним. Використання тільки окремих модулів захисту або технічних засобів неприпустимо. Необхідно застосовувати всі види захисту в повній мірі, інакше розроблена система буде позбавлена сенсу і економічного підґрунтя.
* Універсальним. Засоби захисту повинні бути обрані відповідно до існуючих в компанії каналами витоку.
* Надійним. Всі прийоми захисту повинні надійно перекривати можливі шляхи до охоронюваної інформації з боку зловмисника, незалежно від форми представлення даних.

Інформація вважається захищеною, якщо дотримуються три головних властивості[1].

Перша – цілісність – передбачає забезпечення достовірності і коректного відображення даних, що охороняються, незалежно від того, які системи безпеки і прийоми захисту використовуються в компанії. Обробка даних не повинна порушуватися, а користувачі системи, які працюють з захищеними файлами, не повинні стикатися з несанкціонованою модифікацією або знищенням ресурсів, збоями в роботі програмного забезпечення.

Друга – конфіденційність – означає, що доступ до перегляду та редагування даних надається виключно авторизованим користувачам системи захисту.

Третя – доступність – має на увазі, що всі авторизовані користувачі повинні мати доступ до конфіденційної інформації.

Досить порушити одне з властивостей захищеної інформації, щоб використання система стало безглуздим.

Виділяють наступні комплексні методи захисту інформації:

* фізичні засоби;
* апаратні засоби;
* програмні засоби;
* криптографічний та організаційні методи.

Більшість методів спрямована на взаємодію з комп’ютерними засобами управління інформацією тому, що на сьогоднішній день, майже, всі організації мають мережу комп’ютерів, за допомогою якої здійснюють більшість операцій з інформацією[1].

Фізичні засоби захисту – це засоби, необхідні для зовнішнього захисту засобів обчислювальної техніки, території та об'єктів. Вони реалізуються на базі ЕОМ, які спеціально призначені для створення фізичних перешкод на можливих шляхах проникнення і несанкціонованого доступу до компонентів інформаційних систем, що захищаються.

Апаратні засоби захисту – це різні електронні, електронно-механічні та інші пристрої, які вмонтовуються в серійні блоки електронних систем обробки і передачі даних для внутрішнього захисту засобів обчислювальної техніки: терміналів, пристроїв введення та виведення даних, процесорів, ліній зв'язку тощо.

Програмні засоби захисту, які вмонтовані до складу програмного забезпечення системи, необхідні для виконання логічних та інтелектуальних функцій захисту.

Апаратно-програмні засоби захисту – це засоби, які основані на синтезі програмних та апаратних засобів.

Криптографічні засоби захисту – це засоби, які за допомогою програмного забезпечення або приладів що здійснюють шифрування або передають інформацію у зашифрованому вигляді, щоб навіть при втраті зловмисник не зміг використати інформацію у своїх цілях.

Організаційні заходи захисту інформації складають сукупність заходів щодо підбору, перевірки та навчання персоналу, який бере участь у всіх стадіях інформаційного процесу.

Якщо у роботі підприємства хоча б одна із цих складових порушена, то підприємство матиме збитки, зниження продуктивності праці або нестиме репутаційні втрати[2].

Положення про забезпечення захисту інформації у Україні визначені на законодавчому рівні низкою Законів України, постанов Кабінету міністрів України, нормативних документів у галузі технічного захисту інформації:

* Закон України «Про інформацію» від 02.10.1992 №2657-XII;
* Закон України «Про захист інформації в інформаційно-телекомунікаційних системах» від 05.07.1994 № 80/94-ВР;
* Закон України «Про державну таємницю» від 21.01.1994 № 3855-ХІІ;
* Закон України «Про захист персональних даних» від 01.06.2010 № 2297-VI;
* Постанова Кабінету міністрів України «Про затвердження Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах» від 29.03.2006 №373;
* Постанова Кабінету міністрів України «Про затвердження Інструкції про порядок обліку, зберігання і використання документів, справ, видань та інших матеріальних носіїв інформації, які містять службову інформацію» від 27 листопада 1998 р. №1893;
* НД ТЗІ 3.7-003-05;
* ДСТУ 3396.1-96;
* НД ТЗІ 1.4-001-2000;
* НД ТЗІ 2.5-004-99;
* НД ТЗІ 2.5-005-99;
* НД ТЗІ 2.5-008-02;
* НД ТЗІ 2.5-010-03;
* НД ТЗІ 3.7-001-99;
* НД ТЗІ 3.6-001-2000;
* НД ТЗІ 1.1-002-99[5].

Належний захист інформації одна з головних задач сучасних компаній, які вирішують свою потребу у захисті кваліфікованим ІТ відділом. Між тим важливим є не тільки запобігання втрати або витоку інформації, зниження і протистояння кількості атак на їх ресурси та атак комп’ютерних вірусів.

Найти загальне рішення одночасно і назавжди неможливо, у цьому питанні потрібно постійно оновлювати методи захисту інформації та використовувати нові розробки та рішення у захисті інформації. Тому що з розвитком методів захисту інформації, постійне зростає кількість зловмисників, вдосконалюються методи що вони використовують, збільшуються обчислювальних можливості їх апаратних засобів.

Крім інформації що зберігається на електронних носіях. Для комерційних організацій, що мають різні відділи, велику територію слід використовувати фізичні методи захисту, рівні доступи та розмежування доступу для різних посад, відділів тощо[4].

1.1.2 Фактори, що обумовлюють впровадження методів захисту інформації. Основними факторами, які впливають на впровадження захисту інформації, є потреби підприємств та користувачів, а також наявність відповідних засобів для їх формування.

Потреби підприємства поділяються на:

Розробку правового забезпечення захисту інформації. Фактично це система нормативно-правових документів, актуальних для діяльності підприємства. З її допомогою, з одного боку, визначаються правила забезпечення інформаційної безпеки на підприємстві (наприклад, обов'язки співробітників), а з іншого – встановлюється відповідальність за їх порушення. До складу правового забезпечення включаються державні закони та акти (наприклад, закон про державну таємницю), внутрішні нормативні та організаційні документи підприємства[5].

Визначення потенційних загроз безпеки інформації. Їх можна розділити на три групи – це загрози, що виникають:

* внаслідок дій людини – це можуть бути як випадкові помилки фахівців підприємства при роботі з інформаційною системою (неправильне введення даних або їх видалення), так і навмисні дії (крадіжка документів або носіїв інформації);
* внаслідок некоректної роботи або відмови технічних або програмних засобів (наприклад, збій в роботі операційної системи, викликаний вірусом);
* через стихійних лих, природних катаклізмів, форс-мажорних обставин (повені, пожежі, смерчі, військові дії тощо)[3].

Список потенційних загроз для інформаційної безпеки підприємства може бути дуже великий. Рекомендується оцінити кожну з них з позиції здорового глузду або даних статистики, а потім виконати ранжування за ступенем ймовірності виникнення та обсягу потенційного збитку.

Складання переліку даних, що підлягають захисту. Інформація, яка використовується на підприємстві, може бути відкритою (доступна для всіх) або закритою (доступна для обмеженого кола осіб). До першого типу відносяться відомості, що не становлять державної або комерційної таємниці, не належать до категорії конфіденційної інформації (згідно із законодавством або внутрішніми документами підприємства). Збиток від втрати подібного роду відомостей не є значним, тому їх захист не пріоритетна.

До другого типу відносяться:

* дані, що становлять державну таємницю – їх перелік визначається законодавством;
* комерційні або службові відомості – будь-яка інформація, пов'язана з виробництвом, фінансами, що використовуються технологіями, витік або втрата якої може завдати шкоди інтересам підприємства;
* персональні дані співробітників[2].

Ця інформація повинна бути захищена в першу чергу. Для кожного типу такого роду даних вказується, як і де вони виникають, за допомогою яких програмних або технічних засобів ведеться їх обробка, які підрозділи (співробітники) з ними працюють і т.д.

Створення підрозділу, відповідального за питання захисту інформації. Як правило, на російських підприємствах існує поділ функцій, пов'язаних із забезпеченням інформаційної безпеки. Це має на увазі, що за розробку політики захисту даних, виконання організаційних заходів відповідає служба безпеки компанії, а питання, пов'язані із застосуванням будь-яких програмних і апаратних засобів, включаються до компетенції ІТ-підрозділу. Нерідко виникають ситуації, коли прагнення якомога надійніше захистити дані вступає в протиріччя з потребами бізнесу підприємства. У тому числі це відбувається, якщо заходи безпеки розробляються без урахування можливостей сучасних ІТ-засобів[3].

Правильним підходом є створення єдиної точки прийняття рішень, а саме створення підрозділу, завданням якого буде вирішення всього спектру питань щодо захисту інформації на підприємстві. До його складу необхідно включити як співробітників служби безпеки, так і ІТ-фахівців.

Визначення основних напрямків забезпечення інформаційної безпеки. В рамках вирішення цього завдання, зокрема, позначаються компоненти АСУ, які потребують захисту, визначаються необхідні програмні та технічні засоби, формуються організаційні заходи, спрямовані на захист інформації.

Таким чином, можна сказати, що в разі нехтуванням методами інформації компанії будуть нести великих економічних та репутаційних втрат. В іншому випадку цим можуть скористатися конкуренти і зловмисники[4].

1.1.3 Сучасний стан розвитку систем захисту інформації. Будь-яка локальна мережа є окремим випадком розподіленої системи. Головними відмінними рисами є кількість об'єднаних мережею автоматизованих робочих місць і їх територіальне розташування. Найчастіше, розподілені системи включають в себе значну кількість комп'ютерів, а також, окремі її сегменти можуть перебувати на великій відстані один від одного. Основним вразливим місцем даної технології є канал передачі інформації між сегментами. Оскільки, у міру переміщення по загальним комунікаційним лініям, дані можуть бути перехоплені, що веде до порушення мінімум одного з трьох основних властивостей інформації – конфіденційності (інформація стає відомою третім особам), а також, можливо, і інших: цілісності (дані можуть бути модифіковані) і доступності (дані можуть не дійти до адресата або дійти з затримкою). Крім того, існує можливість інформаційних атак на систему котра знаходиться під захистом ззовні, через глобальну мережу. Вони можуть здійснюватися за різними сценаріями і різними методами. Варто виділити наступні: вірусні атаки різних видів, мережеві атаки спрямовані на проникнення і мережеві атаки спрямовані на відмову роботи всієї системи, її структурних елементів або конкретного обладнання. Для створення системи захисту від вищеназваних загроз необхідно застосування комплексу програмних, апаратних або програмно-апаратних засобів захисту. Так, одним з ефективних інструментів для захисту каналу передачі інформації є створення і використання віртуальної приватної мережі (VPN – virtual private network), через яку проходить мережа загального доступу і шифрування даних. Для захисту від мережевих і вірусних атак застосовуються системи антивірусного захисту, системи виявлення та запобігання вторгнень і міжмережеві екрани[5].

Також для підвищення захищеності, інформація може бути попередньо зашифрована окремими засобами. Обидві ці функції об'єднують в собі програмні і апаратні комплекси, звані криптопровайдерами.

Системами виявлення та запобігання вторгнень є апаратні і програмні засоби, що служать для виявлення атак на мережеві вузли розподіленої системи і протидії їм. Принцип роботи даних систем базується на аналізі всього трафіку, що проходить як в одну, так і в іншу сторону. Найчастіше застосовуються технології сигнатурного аналізу та метод заснований на виявленні аномалій. Найбільш поширеними засобами захисту в даному сегменті є: «Cisco» «IPS Sensor» серії 4500 і «Network Security Platform» від «McAfee».

Програми антивірусного захисту застосовуються для запобігання і ліквідації наслідків зараження комп'ютерів шкідливим ПЗ. Головною негативною рисою шкідливого ПО є велика динаміка змін форми та синтаксису тіла вірусу, що дозволяє їм в деяких випадках обходити детектори антивірусних програм. Блок-аналізатор може використовувати метод статистичного, евристичного аналізу або їх комбінацію для підвищення ефективності. Також одним з важливих чинників застосування антивірусів полягає в підтримці вірусних баз в актуальному стані і їх регулярне оновлення.

Найбільш популярними являються: Norton 360, NOD 32, Avast, Dr. Web Security, 360 Security, Avira, MS Essential[2].

З вищесказаного можна зробити висновок, що сучасні методи захисту інформації базуються на наступних методах та принципах:

* Системний підхід до побудови систем захисту, який обумовлює оптимальне поєднання програмних, організаційних, фізичних і апаратних засобів, які застосовують на всіх етапах обробки інформації.
* Принцип постійного вдосконалення системи, сучасний захист інформації передбачає постійне вдосконалення системи у відповідності зі збільшенням ризиків витоку інформації. Даний процес безперервний і полягає в реалізації сучасних методів і шляхів вдосконалення систем інформаційної безпеки, постійному контролюванні, виявленні її вразливих місць і потенційних каналів витоку інформації. Безперервне вдосконалення системи обумовлено появою нових способів доступу до інформації ззовні.
* Забезпечення надійності систем інформаційного захисту, тобто контролю рівня надійності при відмові системи, появи збоїв, зломі та помилках.
* Контроль функціонування системи захисту.
* Безперервне вдосконалення засобів і методів контролю над працездатністю механізмів захисту.
* Удосконалення методів боротьби з шкідливими програмами і вірусами.
* Оптимізація витрат на створення і експлуатацію систем контролю, що виражені в економічній доцільності застосування систем інформаційної безпеки[5].

1.1.4 Аналіз причин неефективної експлуатації систем захисту інформації. Головним принципом захисту інформації є комплексний підхід, тобто використання всіх доступних методів або засобів систем захисту інформації на підприємстві. Придбання і встановлення найсучасніших та найдорожчих технічних засобів захисту інформації не захистить інформацію, якщо не було проведено комплексний аналіз, організаційні заходи: не було інструктажу персоналу, не проведено навчання персоналу, не був створений контролюючий відділ спрямований на забезпечення захисту інформації, кожен з працівників не ознайомлений з відповідальністю за порушення режиму секретності і т. ін[3].

За відсутності законного користувача, контролю та розмежування доступу до термінала кваліфікований порушник легко використовує його функціональні можливості для несанкціонованого доступу до інформації, що підлягає захисту, шляхом уведення відповідних запитів або команд.

За наявності вільного доступу до приміщення можна візуально спостерігати інформацію на засобах відбиття і документування, викрасти паперовий носій, зняти зайву копію, а також викрасти інші носії з інформацією: лістинги, магнітні носії та ін.

Особливу загрозу становить безконтрольне завантаження програмного забезпечення, в якому можуть бути змінені установки, властивості, дані, алгоритми, введено "троянську" програму або вкорінено комп'ютерний вірус, що виконують деструктивні несанкціоновані дії. Наприклад, записування інформації на сторонній носій, незаконне передавання у канали зв'язку, несанкціоноване друкування документів, порушення їх цілісності, несанкціоноване копіювання важливої інформації, вагомість якої визначається та обмежується на дуже короткий або, навпаки, тривалий час[3].

Загрозливою є ситуація, коли порушник – санкціонований користувач інформаційної системи, який у зв'язку зі своїми функціональними обов'язками має доступ до однієї частини інформації, а користується іншою за межами своїх повноважень. З боку санкціонованого користувача є багато способів порушення роботи інформаційної системи й одержання, модифікування, поширювання або знищення інформації, що підлягає захисту. Для цього можна використовувати, насамперед, привілейовані команди введення-виведення, неконтрольованість санкціонованості або законності запиту і звернень до баз та банків даних, серверів тощо. Вільний доступ дає порушникові можливість звертатись до чужих файлів і баз даних та змінювати їх випадково або умисно.

Під час технічного обслуговування апаратури можуть бути виявлені залишки інформації на її носіях (поверхні твердих дисків, магнітні стрічки та інші носії). Стирання інформації звичайними методами (засобами операційних систем, спеціальних програмних утиліт) неефективне з погляду технічного захисту інформації. Порушник може поновити і прочитати її залишки, саме тому потрібні тільки спеціальні засоби стирання інформації, що підлягає захисту[6].

Під час транспортування носіїв територією, яка не охороняється, виникає загроза перехоплення інформації, що підлягає захисту, і подальшого ознайомлення з нею сторонніх осіб.

Зловмисник може стати санкціонованим користувачем інформаційної системи у режимі розподілу часу, якщо він попередньо якось визначив порядок роботи санкціонованого користувача або якщо він працює з ним на одних лініях зв'язку. Він може здійснити підключення до лінії зв'язку між терміналом та процесором ЕОМ. Крім того, без переривання роботи санкціонованого користувача порушник може продовжити її від його імені, анулювавши сигнали відключення санкціонованого користувача.

Обробка, передавання та зберігання інформації апаратними засобами інформаційної системи забезпечуються спрацюванням логічних елементів на базі напівпровідникових приладів. Спрацювання логічних елементів зумовлено високочастотним зміщенням рівнів напруг і струмів, що призводить до виникнення в ефірі, ланках живлення та заземлення, а також у паралельно розміщених ланках й індуктивностях сторонньої апаратури електромагнітних полів, які несуть в амплітуді, фазі й частоті своїх коливань ознаки оброблюваної інформації. Використання порушником різних приймачів може призвести до несанкціонованого витоку та перехоплення дуже важливої інформації, що зберігається в інформаційній системі. Зі зменшенням відстані між приймачем порушника й апаратними засобами інформаційної системи ймовірність приймання таких інформаційних сигналів збільшується[5].

Як демонструє світовий досвід абсолютна більшість випадків втрати інформації, у тому числі в результаті атак, пов’язано з помилковими діями і неналежним ставленням до своїх обов’язків персоналу самих підприємств. Цілком зрозуміло, що проблема с помилковими рішеннями (діями) персоналу по захисту інформації особливо актуальна для підприємств, в яких не має штатних спеціалістів з захисту інформації. В своїй більшості це невеликі підприємства, офіси або відокремлені підрозділи державних установ, з невеликим штатом працівників або низькою заробітною платнею, що не дозволяє належним чином забезпечувати свою інформаційну безпеку силами штатних спеціалістів. У відповідності зі загальноприйнятою практикою, у випадку відсутності штатного спеціаліста або підрозділу по захисту інформації, керівник назначає відповідального спеціаліста по захисту інформації. Як правило, це спеціаліст у галузі інформаційних технологій, а іноді і робітник яких ніколи раніше не мав справи з захистом інформації, на котрого додають окрім своїх основних функціональних обов’язків додаються також обов’язки з проведення заходів по захисту інформації. Слід зауважити, що не існує ніяких нормативних або інших документів які визначають вимоги до кваліфікації такого спеціаліста. Безумовно, що вузько спеціалізовані рішення з інформаційної безпеки, які буде приймати спеціаліст відповідальний за захист інформації будуть характерні для його досвіду з минулого місця роботи або місця навчання. З вищенаведеного, можливо зробити висновок, що однією з головних умов у комплексному підході є вибір кваліфікованих працівників з захисту інформації, та належний інструктаж працівників, які працюють з конфіденціальною інформацією[6].

1.1.5 Аналіз вірусних загроз у діяльності підприємства. Комп'ютерний вірус – різновид шкідливого ПО, яке має властивість розмножуватися шляхом створення копій самого себе, а також впроваджуватися в код інших програм, в розділи системної пам'яті, завантажувальні сектори. Це спеціально написана програма, найчастіше невелика за розмірами, яка поширює свої копії за допомогою різних каналів зв'язку. Зазвичай вірус запрограмований для порушення роботи програм, блокування роботи користувачів, знищення файлів, приведення в неробочий стан апаратних комплексів ПК[3].

Комп'ютерні віруси можна розділити на певні типи, в залежності від їх місця існування:

* Завантажувальні віруси – віруси, які проникають в завантажувальний сектор пристроїв зберігання даних, таких як жорсткі диски, флешки, дискети і т.д., і здатні порушити доступність файлів.
* Файлові віруси – тип вірусів, які впроваджуються у виконувані файли (файли з розширенням COM і EXE) і негативно впливають на їх функціональність.
* Файлово-завантажувальні віруси – віруси, які об'єднують в собі функції двох попередніх типів вірусів;
* Віруси документи – З огляду на вірусів, які заражають файли офісних систем. Такий вид ще називають «макровірусами», оскільки зараження системи відбувається за допомогою зараження макросів програми.
* Мережеві віруси – тип вірусів, які поширюються за рахунок використання комп'ютерної мережі, тобто мережевих служб і протоколів[3].

Також віруси діляться на типи за принципом свого функціонування – віруси-паразити, віруси-станції, трояни, віруси-невидимки, віруси-шифратори та «відпочиваючі» віруси – ті які запрограмовані на те, щоб активізуватися в певний час.

Для того, щоб виявити, видалити або захистити комп'ютер від вірусів, розробляються спеціальні програми. Ці програми називаються антивірусними і являють собою багатофункціональний продукт, який поєднує в собі такі засоби як: превентивні, профілактичні, кошти «лікування» або видалення, а також відновлення порушених чи втрачених даних.

Антивірусні програми діляться на певні типи:

* Програми-детектори – ті, які допомагають знайти віруси в оперативній пам'яті або ж на носіях інформації, при цьому програми-детектори знайдені віруси не лікують.
* Програми-доктора – програми, які на відміну від попереднього виду, не тільки знаходять вірус, але і лікують заражений файл, повертаючи його в початковий стан.
* Програми-ревізори – такі програми мають властивість запам'ятовувати файл або системну область диска в його початковому стані, і пізніше порівнювати поточний стан з вихідним. При порівнянні файлу враховуються багато параметрів файлу, тому сховатися вірусу такі програми не залишають шансу.
* Програми-фільтри – призначені для виявлення підозрілих дій в роботі комп'ютера. При спробі активізації вірусу програма може блокувати його роботу.
* Вакцини – такі програми, які відразу запобігають зараженню різних файлів. Варто застосовувати такі програми, якщо програми-доктора відсутні. Але варто врахувати, що «вакцинація» можлива тільки проти вже відомих вірусів[3].

Проте, наявності одного лише антивірусного ПО на робочій станції є недостатнім для захисту загальносистемного і прикладного ПО і має підкріплюватися раціональними діями і обережністю при роботі в мережі самих користувачів.

За даними Брендона Гейлі у власному виданні. Статистика комп'ютерних вірусів. Відомо, що комп'ютерні віруси ставлять під загрозу користувачам конфіденційну інформацію, знищують дані та завдають пошкоджень обладнання вашої системи. Деякі з цих вірусів можуть призвести до збитків більше мільярдів.

Перелік типів вірусів та відсоток, які найчастіше вражають користувачів:

1. Віруси – 57%

2. Замасковані трояни – 21%

3. Троянські завантажувачі – 7%

4. Небажане програмне забезпечення – 4%

5. Рекламне програмне забезпечення – 3%

6. Віруси-експлойти – 3%

7. Віруси хробаки – 2%

8. Викрадачі паролів та інструменти моніторингу – 2%

9. Віруси-бекдори(потаємний хід) – 1%

10. Шпигунське програмне забезпечення – 0,01%[6].

Віруси, які завдали найбільших грошових збитків. Найдорожчий комп'ютерний вірус всіх часів був віднесений до "MyDoom", що призвело до відшкодування 38 мільярдів доларів. Він швидко рухався, заражаючи відкриті мережі та кожен комп'ютер, що має доступ до нього. У 2004 році цей вірус, за оцінками, вплинув на 25% всіх листів.

Нижче наведено список 10 найдорожчих програм вірусів, а також загальна сума збитків, що сталися:

1. MyDoom – $ 38 млрд;

2. So Big – $ 37,1 млрд;

3. ILOVEYOUE – $ 15 млрд;

4. Conficker – $9,1 млрд;

5. Code Red – $ 2 млрд;

6. Melissa – $ 1,2 млрд;

7. WannaCry – $ 1 млрд;

8. SirCam – $1 млрд;

9. SQL Slammer – $ 750 млн;

10. Nimda – $ 635 млн.[7].

Враховуючи економічне становище в деяких компаній, задля економії використовують попередні версії програмного забезпечення. Враховуючи той факт, що наприклад деякі версії операційних систем перестають обслуговуватись з годом, це вдало використовують зловмисники. Так 12 травня 2020 року велика кількість комп’ютерів з операційною системою Windows знаходились під вірусною атакою вірусом-вимагачем WannaCry. Вірус шифрував файли користувачів, щоб їх не можна було використовувати; за розшифрування зловмисники вимагали гроші. Уражено було близько 300 тисяч комп’ютерів у щонайменше 150 країнах світу. Збитки оцінювались більше ніж 1 мільярд доларів. Від атаки постраждала велика кількість державних компаній у різних країнах та більше ніж 70% від постраждалих компаній, знаходились на території Росії та України. Після атаки Microsoft випустила оновлення пакетів безпеки для операційних систем, що вже не підтримувались компанією а саме: Windows XP, Windows Server 2003 та Windows 8. Тому спеціалісти радять, комп’ютери, що мають доступ до важливої інформації використовувати без доступу до мережі Інтернет або належно обслуговати їх[8].

## 

## 1.2 Постановка задачі

**Цілі**: проведення аналізу стану, розробка та надання рекомендацій щодо впровадження комплексної систем захисту інформації на підприємстві Підприємство “НЕТСО”.

**Об’єкт**: підприємство Підприємство “НЕТСО”.

**Предмет**: аналіз стану та комплексний захист інформації на підприємстві Підприємство “НЕТСО”, було поставлене завдання на проведення аналізу стану та комплексного захисту інформації на підприємстві.

Виходячи з аналізу проблеми сформовано завдання та критерії проведення комплексного аналізу.

Основною задачею при реалізації проекту є забезпечення функцій:

* 1. Оцінка стану систем захисту інформації.
  2. Проведення аналізу мережі.
  3. Проведення аналізу програмного забезпечення, що встановлене на робочі комп’ютери.
  4. Проведення аналізу фізичних засобів захисту інформації.
  5. Проведення аналізу роботи з конфіденційною інформацією.
  6. Створення системи інформаційного захисту, програми передачі даних та повідомлень з використанням алгоритму шифрування Blowfish на базі мови програмування C#.

# РОЗДІЛ 2

# ПРОВЕДЕННЯ АНАЛІЗУ СТАНУ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ НА ПІДПРИЄМСТВІ “НЕТСО”

## 2.1 Загальні відомості про підприємство

Підприємство “НЕТСО” є юридичною особою приватного права, створеною за рішенням установчих зборів засновників в межах та у порядку, визначеному положеннями чинного, на момент її створення, законодавства України.

Організація у своїй діяльності керується Конституцією України, Цивільним кодексом України, Законом України «Про громадські об’єднання», Законом України «Про державну реєстрацію юридичних осіб, фізичних осіб-підприємців та громадських формувань», Законом України «Про освіту», іншими законодавчими актами, міжнародними договорами, згода на обов’язковість яких надана Верховною Радою України, та цим Статутом.

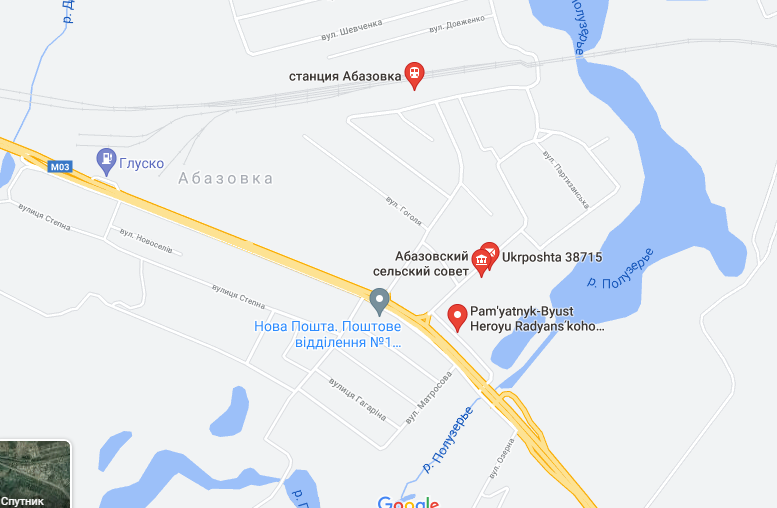


Рисунок 2.1 – Розташування підприємства



Рисунок 2.2 – Загальний вигляд офісного приміщення підприємства



Рисунок 2.3 – Загальний вигляд офісно-технічного приміщення підприємства



Рисунок 2.4– загальний вигляд виробничого приміщення підприємства



Рисунок 2.5– загальний вигляд станків виробничого приміщення підприємства



Рисунок 2.6– загальний вигляд станків виробничого приміщення підприємства

Підприємство “НЕТСО”, засноване в 1996 р має науково-виробничий профіль і спеціалізується на виконанні складних інженерних робіт.

Виробництво расташовано в села Абазівка Полтавського району Полтавської області по вул. Белоусько, будинок 52а.

Керівник: Казаров Размік Рафаелович

Основної діяльності підприємства є виготовлення та ремонт деталей по кресленням заказника, або по зразку будь якої складності, з особо твердого та крихкого метала, навіть, з самою складною геометрією з термообробкою твердості та точністю обробки до 0,002 мм за допомогою обладнання з числовим програмним управлінням (ЧПУ)

На підприємстві працюють більше 70 висококваліфікованих фахівців з вищою і спеціальною технічною освітою. Работу підприємства забезпечує інжинерно-технічний та виробничний підрозділи. Інженерно-технічний підрозділ користується сучасним програмним забезпеченням на сучасному обладнанні яки дают можливість моделювати навіть складні деталі та вузли.

Виробничі дільниці оснащені сучасним інноваційним високотехнологічним обладнанням для механічної обробки та зварювання, що дозволяє вирішувати різноманітні технічні задачі з виробництвом та відновлення деталей машин на високому професійному рівні.

Підприємство має можливість виконувати таки роботи як:

* копіювання форми перетину електрода
* вирізання деталей зі складним профілем
* розрізання заготовки високої твердості
* розточування деталей, шліфування (кругле, плоске)
* прошивання отвору, щілин та вікон
* складно контурна дротова вирізка
* виготовлення міцних різців, точних шаблонів, лекала, якісних штампів і матриць
* технічні ножі, гвинти, робочі колеса
* Прес-форми, штампи, висічний і пробивний інструмент для металообробні станки, шнеки, ролики, обичайки, вали, шліцьові вали, шестерні
* Термообробка, об'ємна гарт, ТВЧ, цементація і багато іншого

На плавки та механічної обробки дають можливість відновлювати деталі які мали різні механічні пошкодження: тріщини, відколи, надломи, деформації, знищують корозію та потребували, донедавна, однозначної заміни.

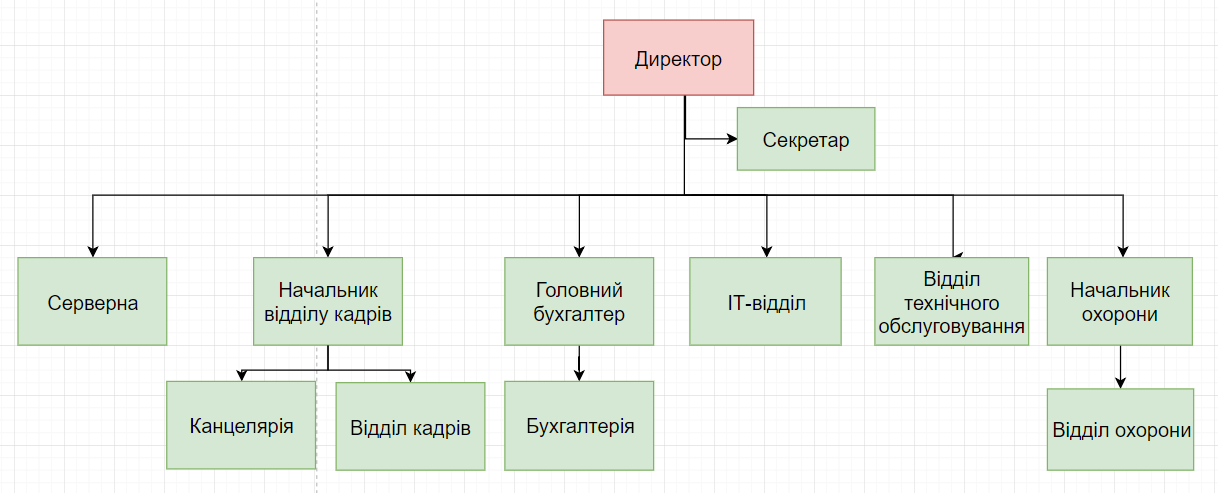


Рисунок 2.2 – Структура управління підприємства

## 2.2 Характеристка ком’ютерної мережі у відділах підприємства

Таблиця 2.1 − Кількість комп’ютерів та вид використаної топології

|  |  |  |
| --- | --- | --- |
| **Відділ** | **Техніка** | **Кількість** |
| Директор | Комп’ютер | 1 |
| Секретар | Комп’ютер | 1 |
| IT-відділ | Комп’ютер | 22 |
| Відділ охорони | Комп’ютер | 2 |
| Бухгалтерія | Комп’ютер  Принтер | 4  1 |
| Канцелярія | Комп’ютер | 2 |
| Відділ технічного обслуговування | Комп’ютер | 5 |
| Відділ кадрів | Комп’ютер | 3 |
| Серверна | Комп’ютер  Сервер | 1  1 |
| **Всього** | Комп’ютер  Принтер  Сервер | 41  1  1 |

## 

Таблиця 2.2 − Експлікація приміщень

|  |  |
| --- | --- |
| **Експлікація приміщень** | |
| **Відділ** | **Площа** |
| Директор | 14 м2 |
| Секретар | 5 м2 |
| IT-відділ | 132 м2 |
| Відділ охорони | 12 м2 |
| Бухгалтерія | 24 м2 |
| Канцелярія | 12 м2 |
| Відділ технічного обслуговування | 30 м2 |
| Відділ кадрів | 18 м2 |
| Серверна | 8 м2 |
| Коридор | 129,32 м2 |
| Підсобне приміщення | 11 м2 |
| Санвузол | 13,5 м2 |
| **Загальна площа** | 408,32 |

# План приміщення

План приміщення (Рис. 2.6) був розроблений за допомогою веб-сайту draw.io [9]

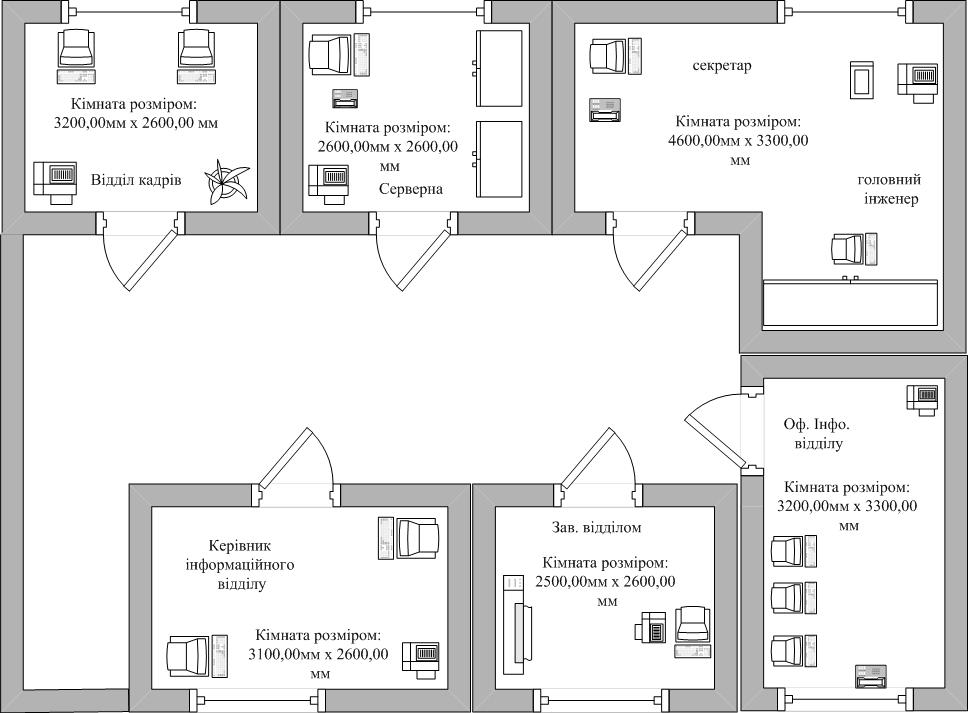


Рисунок 2.6 − План приміщення

# Апаратне забезпечення

Таблиця 2.3 − Апаратне забезпечення

|  |  |  |  |
| --- | --- | --- | --- |
| Назва | Модель | Кількість (шт.) | Ціна за шт. (грн.) |
| Монітор | Samsung Curved C24F396F (LC24F396FHIXCI) | 41 | 3 999 |
| Системний блок | Дон Кармани NO i7-8700 G2 (i7-8700/8GB-DDR4/HDD-500Gb) | 41 | 16 450 |
| Клавіатура | Logitech K100 PS/2 Black RUS (920-003200) | 41 | 229 |
| Миша | A4Tech N-708X USB Grey (4711421867009) | 41 | 249 |
| Принтер | Kyocera Ecosys M2040dn (1102S33NL0) | 1 | 9 000 |
| Сервер | HP Proliant DL 380p G8 (12x3.5) LFF | 1 | 18 053 |

## 2.3 Управління доступом на підприємстві

## Для управління доступом, потрібно розмежувати доступ до інформації на сервері для кожної посади. Для того щоб отримати доступ до різних видів інформації працівники використовують паролі, що надають певні права залежно від посади. Умовно за рівнем важливості інформація поділена на такі категорії:

* життєво важлива інформація – незамінна інформація, наявність якої необхідна для функціонування системи (інформація про банківські рахунки, особисті дані працівників і т. ін.);
* важлива інформація – інформація, що може бути замінена чи відновлена, але процес її відновлення важкий і зв’язаний з великими витратами (кількість продукції та сировини, бази даних постачальників і т. ін.);
* корисна інформація – інформація, яку важко відновити, однак система може досить ефективно функціонувати без неї;
* несуттєва інформація – інформація, без якої система продовжує існувати.

Для більш зручного та наочного представлення моделі керування та розмежування доступом на підприємстві використано матрицю доступу. Матриця доступу являє собою таблицю, у які зображено список робіт або процеси та користувачів, що беруть участь у робочому процесі. Елементами матриці є рівні доступу для кожного користувача враховуючи його повноваження відносно кожного списку робіт, процесу або об’єкту[9].

Керуючись постановою Кабінету міністрів України № 373. Потрібно виконувати наступні вимоги до забезпечення захисту інформації в системі:

1. Відкрита інформація під час обробки в системі повинна зберігати цілісність, що забезпечується шляхом захисту від несанкціонованих дій, які можуть призвести до її випадкової або умисної модифікації чи знищення.
2. Усім користувачам повинен бути забезпечений доступ до ознайомлення з відкритою інформацією. Модифікувати або знищувати відкриту інформацію можуть лише ідентифіковані та автентифіковані користувачі, яким надано відповідні повноваження.
3. Спроби модифікації чи знищення відкритої інформації користувачами, які не мають на це повноважень, неідентифікованими користувачами або користувачами з не підтвердженою під час автентифікації відповідністю пред'явленого ідентифікатора повинні блокуватися.
4. Під час обробки службової і таємної інформації повинен забезпечуватися її захист від несанкціонованого та неконтрольованого ознайомлення, модифікації, знищення, копіювання, поширення.
5. Доступ до службової інформації надається тільки ідентифікованим та автентифікованим користувачам. Спроби доступу до такої інформації неідентифікованих осіб чи користувачів з не підтвердженою під час автентифікації відповідністю пред'явленого ідентифікатора повинні блокуватися.
6. У системі забезпечується можливість надання користувачеві права на виконання однієї або кількох операцій з обробки службової інформації або позбавлення його такого права.
7. Вимоги до захисту в системі інформації від несанкціонованого блокування визначаються розпорядником інформації, якщо інше для цієї інформації або системи, в якій вона обробляється, не встановлено законодавством.
8. Порядок підключення систем, в яких обробляється службова і таємна інформація, до глобальних мереж передачі даних визначається законодавством.
9. Згідно вимоги до забезпечення захисту інформації в системі було створено матрицю доступу (Табл. 2.5) для наочного зображення доступу до інформації у кожного з користувачів[10].

Таблиця 2.4 – Матриця доступу на підприємстві “НЕТСО”

|  |  |  |  |
| --- | --- | --- | --- |
| Посада | Ресурс | | |
| База користувачів | Налаштування серверу | Повідомлення |
| Мережевий адміністратор | зчитування | Повний доступ | доступу немає |
| Керівник відділу | зчитування, запис (власні дані) | доступу немає | Зчитування, запис (власні дані) |
| Начальник групи (бригадир) | зчитування, запис (власні дані) | доступу немає | Зчитування, запис (власні дані) |
| Фахівець (спеціаліст по визначеному напряму) | запис (власні дані) | доступу немає | Зчитування, запис (власні дані) |
| Користувач | запис (власні дані) | доступу немає | Зчитування, запис (власні дані) |

# Розмежування доступу

**Розмежування доступу** — сукупність процедур, що реалізують перевірку запитів на доступ і оцінку на підставі Правил розмежування доступу можливості надання доступу.

Правила розмежування доступу — частина [політики безпеки](https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%BB%D0%B5%D0%BA%D1%81%D0%BD%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%B7%D0%B0%D1%85%D0%B8%D1%81%D1%82%D1%83_%D1%96%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D1%96%D1%97), що регламентує правила доступу користувачів і процесів до пасивних об’єктів.

При розгляді взаємодії двох об'єктів комп’ютерної системи, що виступають як приймальники або джерела інформації, слід виділити пасивний об'єкт, над яким виконується операція, і активний об'єкт, який виконує або ініціює цю операцію.

Коли користувачі або процеси намагаються одержати доступ до пасивних об'єктів, механізми, що реалізують [керування доступом](https://uk.wikipedia.org/wiki/%D0%9A%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC), на підставі політики безпеки і перевірки атрибутів доступу можуть «прийняти рішення» про легальність запиту. Використовуючи набір атрибутів доступу відповідно до прийнятої політики безпеки, можна реалізувати довірче керування доступом, адміністративне, контроль за цілісністю та інші види керування доступом.

### Таблиця 2.5 Облікові записи відділів



**Мандатне керування доступом** — розмежування доступу суб'єктів до об'єктів, засноване на призначенні мітки конфіденційності для інформації, що міститься в об'єктах, і видачу офіційних дозволів (допуску) суб'єктів на звернення до інформації такого рівня конфіденційності. Також іноді перекладається як **Примусовий контроль доступу**. Це спосіб, що поєднує захист і обмеження прав, що застосовується по відношенню до комп'ютерних процесів, даних і системних пристроїв, призначений для запобігання їх небажаного використання.

За допомогою мандатного керування доступу, ця політика безпеки централізовано контролюється адміністратором політики безпеки; користувачі не мають можливості перевизначити політику та, наприклад, надавати доступ до файлів.

### Таблиця 2.6 Мандатна модель



Згідно другої частини пункту сьомого постанови Кабінету міністрів України: У системі забезпечується можливість надання користувачеві права на виконання однієї або кількох операцій з обробки службової інформації або позбавлення його такого права. Для наочного зображення прав доступу до службової інформації створено таблицю рівнів секретності (Табл 2.7) та зображено рівні доступу за допомогою методів мандатного керування доступом.

Методи мандатного керування доступом застосовуються до тих баз даних, у яких збережена інформація має досить статичну і тверду структуру, що властиво, наприклад, деяким військовим або урядовим організаціям. Основна ідея полягає в тому, що кожному об'єктові даних привласнюється деякий класифікаційний рівень (classification level) (або необхідний гриф таємності, наприклад "Абсолютно секретно", "Секретно", "Для службового користування" і т.д.), а кожному користувачеві надається рівень допуску (clearance level) із градаціями, аналогічними існуючим класифікаційним рівням. Передбачається, що ці рівні утворять строгу ієрархічну систему (наприклад, "Абсолютно секретно" > "Секретно" > "Для службового користування" і т.д.)

2.4 Криптологічний захист інформації.

Криптологія – наука про захист інформації, шляхом її перетворення. Криптологія поєднує два напрямки – криптографію й криптоаналіз.

Криптографія займається пошуком і дослідженням методів перетворення інформації з метою приховання її змісту. Основні напрямки використання криптографічних методів – передача конфіденційної інформації з каналів зв'язку, установлення дійсності переданих повідомлень, зберігання інформації (документів, баз даних) на носіях у зашифрованому виді.

Для криптологічного захисту інформації на данному підприємстві пропонується шифрувати секретну інформацію, таку як паролі та логіни співробітників, секретну кореспонденцію що циркулює підприємством, та ін.

Розмежування доступу – сукупність процедур, що реалізують перевірку запитів на доступ і оцінку на підставі правил розмежування доступу. Правила розмежування доступу це частина політики безпеки, що регламентує правила доступу користувачів і процесів до пасивних об’єктів.

Коли користувачі або процеси намагаються одержати доступ до пасивних об'єктів, механізми, що реалізують керування доступом, на підставі політики безпеки і перевірки атрибутів доступу можуть «прийняти рішення» про легальність запиту. Використовуючи набір атрибутів доступу відповідно до прийнятої політики безпеки, можна реалізувати довірче керування доступом, адміністративне, контроль за цілісністю та інші види керування доступом.

Для відображення функціональностi комп’ютерної системи використовується концепція матриці доступу. Матриця доступу являє собою таблицю, уздовж кожного виміру якої відкладені iдентифікатори обʼєктів комп’ютерної системи, а елементами матриці виступають дозволені або заборонені режими доступу. Матриця доступу може бути двомірною (наприклад, користувачі/пасивні обʼєкти або процеси/пасивні обʼєкти) або тримірною (користувачі/процеси/пасивні обʼєкти).

## 2.5 Серверний захист

## На підприємстві “НЕТСО” сервер розташований у приміщенні ІТ-відділу, відповідальним за обслуговування серверу є системний адміністратор. На сервері існує рівень доступу до інформації згідно матриці доступу (Табл 2.6) дані для авторизації працівник отримає при проведенні організаційних робіт з забезпечення захисту інформації. Надано рекомендацію, щодо переміщення серверу або створення окремої шафи для серверу з додатковим фізичним захистом. Операційна система, що використовується на сервері – Ubuntu 14.04 LTS. Операційна система Ubuntu є надійною, безкоштовною та стабільною, не потребує частих перезавантажень, не потребує антивірусів, та має відносно легкий рівень у налаштуванні, реліз нової версії відбувається кожні півроку. Тому було надано рекомендацію, щодо оновлення операційної системи на сервері до версії Ubuntu 18.04 LTS, у цілях отримання останніх налаштувань та підтримки, оскільки обслуговування версії Ubuntu 14.04 закінчується у 2019 році в квітні місяці, а остання версія Ubuntu 18.04 LTS матиме, підтримки до квітня 2023 року. Головною перевагою Ubuntu є її безкоштовність та надійність відносно інших серверних ОС[10].

## 2.6 Антивірусний захист

Комп'ютерний вірус - це спеціально написана, як правило, невелика за розмірами програма, яка може записувати (впроваджувати) свої копії (можливо, змінені) в комп'ютерні програми, розташовані в здійснимих файлах, системних областях дисків, драйвери, документах, причому ці копії зберігають можливість до «розмноження»[3].

Процес впровадження вірусом своєї копії в іншу програму називається зараженням, а програма або об'єкт, що містить вірус - зараженим.

Комп'ютерні віруси є програмами, які можуть «розмножуватися» і таємно впроваджувати свої копії в файли, завантажувальні сектори дисків і документи[3].

Обов'язковою властивістю комп'ютерного вірусу є здатність до розмноження і непомітного для користувача впровадженню в файли, завантажувальні сектори дисків і документи. Назва «вірус» по відношенню до комп'ютерних програм прийшло з біології саме за ознакою здатності до само розмноження[3].

Стадії розвитку вірусу:

• прихований етап - дія вірусу не виявляється і залишається непоміченим

• лавиноподібне розмноження, але його дії поки не активізовані

• активні дії - виконуються шкідливі дії, закладені його автором.

Активізація вірусу може бути пов'язана з різними подіями:

- Настанням певної дати або дня тижня

- Запуском програми

- Відкриттям документа

**Avast Antivirus** являє користувачу нову функцію AutoSandbox, яка дозволяє автоматизувати процес приміщення підозрілих файлів в "пісочницю", де можна буде провести повний аналіз файлу і, при необхідності, вилікувати його. Ця функція дозволяє врятувати від миттєвого стирання досить великий відсоток файлів, уникнути системних помилок, пов'язаних з видаленням важливих системних файлів і тп. Додаток звертається з об'єктами акуратніше аналогів[4]!

**AVG Anti-Virus** може похвалитися відмінними показниками захисту системи і досить малим споживанням системних ресурсів. На відміну від платної версії, яка, звичайно, містить в собі більше інструментів і функцій, версія для безкоштовного використання працює набагато стабільніше, отримуючи при цьому ту ж саму технічну підтримку у вигляді оновлень. Швидкість роботи програми вражає, а сканер електронної пошти позбавить вас від необхідності встановлювати спеціалізовані додатки, так як справляється він на всі 100%[4].

**SystemCare Ultimate** стане в нагоді як власникам слабких машин, так і користувачам потужних комп'ютерів, тому що дозволить домогтися максимальної продуктивності. А завдяки вбудованій системі виявлення вірусів на основі движка BitDefender захистить від всіх актуальних загроз[4].

**Panda Antivirus Pro** відмінний вибір для користувачів, які хочуть отримати якісний захист від вірусів при мінімумі зусиль. Більшість функцій Панди автоматизовані, програма постійно сканує оперативну пам'ять і жорсткий диск на наявність загроз і підозрілих файлів. Новий движок програми дозволяє знизити споживання програмою системних ресурсів до мінімуму. Антивірусний завантажувальний диск Panda Cloud Cleaner дає можливість вилікувати заражену систему, яка не може сама завантажитися. Може трохи напружувати кількість помилкових спрацьовувань, але ж це не так і погано - програма піклується про Вас! Загалом, якщо Вам потрібен відмінний антивірус, який чудово справляється з підтримкою системи в хорошому стані навіть без участі користувача в цьому процесі, Panda Antivirus Pro - кращий вибір[4]!

**IObit Malware Fighter** не є класичним продуктом, як, наприклад, «Антивірус Касперського», але гарантують більшу ступінь захисту, ніж dr.web cureit і інші сканери, розраховані на звичайну перевірку ПК на віруси. Також він може бути встановлений в комплекті з програмним забезпеченням Advanced SystemCare - набором утиліт, які очищають систему, підвищують продуктивність комп'ютера, відновлюють випадково видалену інформацію і т.д. Знайдеться засіб для вирішення мало не будь-якої проблеми[4].

**360 Total Security** Антивірус є відмінним вибором як для початківців користувачів ПК, так і для просунутих. Перші отримують в своє розпорядження надійну систему з безліччю автоматизованих функцій, що дозволить захищати комп'ютер без прямої участі користувача. Другі ж по достоїнству оцінять гнучкі налаштування програми, можливість змінювати профілі, зберігаючи в них різні настройки, функції щодо оптимізації роботи системи і багато інших цікавих опції. Оформлення програми не викликає ніяких питань і дозволяє використовувати всі її аспекти без зайвих питань і допомоги довідки[4].

**ESET NOD32 Smart Security** має в своєму арсеналі все необхідне для захисту вашого ПК: кілька ступенів захисту від будь-якого типу небажаного ПО або вірусу, персональний фаєрвол для шифровки з'єднання, батьківський контроль, контроль і скан пристроїв, що підключаються, безкоштовна цілодобова технічна. Якщо необхідний антивірус для установки на ноутбук, НОД32 буде практично ідеальним рішенням, так як він має спеціальні профілі для роботи на портативних ПК, які дозволяють знизити витрату енергії. Але за все доводиться платити, і у ESET NOD 32 крім величезного набору якісних інструментів також досить високе споживання системних ресурсів, що частково компенсується профілями, де можна налаштувати кожен аспект. До слова, на офіційному сайті антивіруса є багато корисної інформації по оптимізації роботи додатка. Але в цілому, ESET NOD32 Smart Security по праву є однією з найпопулярніших і надійних антивірусних програм на сучасному ринку[4].

**Avira Antivirus** йде в комплекті з модулем Virus Guard, який автоматично сканує файли, які відкриває користувач, що підвищує рівень безпеки системи. Для забезпечення безпеки в мережах wi-fi також можна активувати компонент захисту Avira Phantom VPN, необхідний для захисту ПК від фішингових сайтів і вірусних загроз ззовні. В цілому, Авіра - одне з кращих рішень для боротьби з так званими "поліморфними" вірусами, які можуть наслідувати звичайними програмами. Також Ви можете встановити модуль сканування електронної пошти, захист від спаму та від програм з автодозвоном. Іншими словами, Авіра - хороший антивірус, який готовий довго і надійно захищати ваш комп'ютер, що б ви не робили[4].

**Bitdefender Antivirus Edition** ідеально підходить для домашнього використання, так як не навантажує систему зайвими процесами, забезпечуючи при цьому високий рівень захисту. На відміну від багатьох інших антивірусів, Бітдефендер не напружує постійно вискакують вікнами і не вимагає участі користувача в своїй роботі. Величезна база вірусів забезпечує найвищий рівень захисту, а інтуїтивний інтерфейс дозволяє використовувати його навіть користувачам, які ніколи раніше не стикалися з антивірусними програмами[4].

Рейтинг топ 10 антивірусів, ціна вказана 1 ПК 1 рік обслуговування найвищого степеню захисту тобто рівня Prime та Pro.



Рисунок 2.4 Рейтинг Антивірусів

На фірму вирішено було поставити антивірус Panda Antivirus Pro так як на підприємстві багато особливо важливої інформації, тому на антивірусі було вирішено не економити. Panda має одну з найбільш великих баз вірусів у світі і вже багато років входить в топ 5 кращих антивірусів світу!

## 2.7Організаційний та фізичний захист

## Відповідальним за проведення організаційних робіт з забезпечення захисту інформації, наказом підприємства було уповноважено виконавчого директора. Інструктаж на підприємстві проводиться згідно Законів України, та постанов Кабінету міністрів України, щодо захисту інформації. Кожен працівник освідомлений рівнем доступу та має дані для авторизації для доступу до інформації на сервері згідно (Табл 2.6) матриці доступу. Також кожен працівник освідомлений за відповідальність у разі порушення Закону України або нормативно правових-актів чи наказів підприємства, щодо збереження цілістності інформації[11].

## 

## 2.8 Підсумки аналізу та надані рекомендації

## Проведення аналізу заходів захисту інформації можна розділити на такі групи:

* Технічні (апаратні засоби). Це різні за типом пристрої (механічні, електромеханічні, електронні та ін), які апаратними засобами вирішують завдання захисту інформації. Вони або перешкоджають фізичному проникненню, або, якщо проникнення все ж таки відбулося, доступу до інформації, у тому числі за допомогою її маскування.
* Програмні засоби включають програми для ідентифікації користувачів, контролю доступу, шифрування інформації, видалення залишкової (робочої) інформації типу тимчасових файлів, тестового контролю системи захисту та ін. Переваги програмних засобів – універсальність, гнучкість, надійність, простота установки, здатність до модифікації та розвитку. Недоліки – обмежена функціональність мережі, використання частини ресурсів файл-сервера і робочих станцій, висока чутливість до випадкових або навмисним змін, можлива залежність від типів комп'ютерів (їх апаратних засобів). На даному підприємстві включають блокування портів проти розподілених атак спрямованих на відмову в обслуговуванні, а також доступ до інформації на сервері здійснювався завдяки паролю відповідно рівня доступу користувача, було надано рекомендацію по встановленню паролю для доступу до комп’ютерів.
* Криптографічні методи захисту інформації – це методи захисту даних із використанням шифрування. Криптографічні засоби відсутні.

З вищенаведеного, можна зробити висновок, що рівень захищеності підприємства є недостатнім у криптографічному та програмному методах. Для забезпечення програмного методу, було надано рекомендації, а для забезпечення криптографічного захисту було прийнято рішення розробити програмний засіб з криптографічним алгоритмом шифрування інформації та повідомлень.

# РОЗДІЛ 3

# РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ШИФРУВАННЯ ПОВІДОМЛЕНЬ ТА ФАЙЛІВ

## [3.1](https://mail.yandex.ua/neo2/#_Toc325035686) Вибір алгоритму шифрування

# Особливості роботи алгоритму шифрування Blowfish

Шифрування використовується людством з того самого моменту, як з'явилася перша секретна інформація, тобто така, доступ до якої повинен бути обмежений. Ідея шифрування полягає в запобіганні перегляду істинного змісту повідомлення (тексту, файлу і ін.) тими, у кого немає коштів його дешифрування.

До кінця 1993 року в світі криптографії виникла дуже незручна ситуація. Алгоритм симетричного шифрування DES, зі слабким 56-бітовим ключем, був близький до фіаско, а існуючі на той момент альтернативні варіанти, такі як Khufu, REDOC II, IDEA були захищені патентами і не доступні для вільного використання. Виникла потреба у алгоритмі шифрування, більш криптостійкому ніж DES, і в той же час без будь-яких обмежень на право свого використання. І він з'явився – у 1994 році Брюс Шнайер презентував свій алгоритм блочного шифру, який був названий Blowfish.

Відмінними рисами цього алгоритму став вищий ступінь криптостійкості, ніж алгоритму DES (в тому числі за рахунок використання змінної довжини ключа, до 448 біт), висока швидкість шифрування / дешифрування (за рахунок генерації таблиць заміни) і звичайно – можливість його вільного застосування для будь-яких цілей.

У загальному випадку алгоритм складається з двох етапів - розширення ключа і шифрування / дешифрування вихідних даних. На етапі розширення ключа, вихідний ключ перетворюється в матрицю раундових ключів (P) і матрицю підстановки (S, Substitution-box). Шифровка даних, а також створення матриці раундових ключів і підстановки, відбувається через використання мережі Фейстеля, що складається в свою чергу з 16 раундів.

Мережа Фейстеля – один з методів побудови блочних шифрів. Мережа складається з комірок. На вхід кожної комірки надходять дані і ключ. На виході кожної комірки отримують змінені дані і змінений ключ. Всі комірки однотипні, і кажуть, що мережа являє собою певну ітеровану структуру. При шифруванні та розшифруванні виконуються одні й ті ж операції; відрізняється тільки порядок ключів.

В алгоритмі Blowfish при шифруванні виконується 16 раундів (всередині мережі Фейстеля), а 17-й і 18-й ключі складаються з лівим і правим вихідним блоком останнього раунду.

Підготовчим етапом алгоритму Blowfish є етап розширення ключів. У процесі цього етапу будується матриця раундових ключів Pn і матриця підстановки - 4 блоки заміни S-Box (Substitution-box), кожен з яких складається з 256 32-х бітних елементів. Елементи цих матриць обчислюються за допомогою розглянутої вище мережі Фейстеля.

Функція раунду або ітерації (оскільки вона загальна для всієї використовуваної мережі Фейстеля) вельми проста і використовує лише кілька логічних операцій над матрицею підстановок:

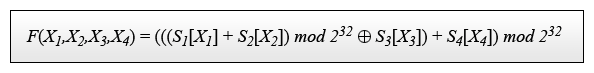


Рисунок 3.1 – Функція раунду



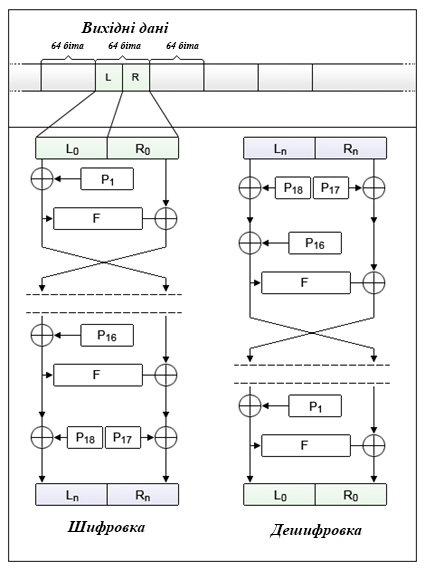


Рисунок 3.2 – Блок-схема алгоритму мережі Фейстеля

Якщо узагальнити алгоритм шифрування або дешифрування в алгоритмі Blowfish, то отримаємо такі кроки:

1. Виділяємо масив з 18 елементів для раундових ключів мережі Фейстеля і 4 матриць підстановки по 256 елементів в кожній.

2. Заповнюємо виділений масив значенням мантиси числа PI.

3.Робимо ітеративний XOR: Pi = Pi XOR Ki (де Pi - раундовий ключ, а Ki - вихідний ключ).

4. Шифруємо раундові ключі і матриці підстановки за допомогою мережі Фейстеля.

5. Шифруємо / дешифруємо блоки вихідних даних по 64 біта також за допомогою мережі Фейстеля.

Blowfish зарекомендував себе, як надійний алгоритм, тому реалізований у багатьох програмах, де не потрібна часта зміна ключа і необхідна висока швидкість шифрування / розшифрування.

# ПРАКТИЧНА ЧАСТИНА

# [3.2](https://mail.yandex.ua/neo2/#_Toc325035686) Вибір мови програмування та середовища розробки

# Для реалізації шифру Blowfish було обрано мову програмування С#. Тому, що вона має необхідний набір інструментів для роботи з шифрами, та на основі досвіду з інших навчальних дисциплін та набутих навичок.

C # (С Sharp) – об'єктно-орієнтована мова програмування. Розроблена в 1998-2001 роках групою інженерів під керівництвом Андерса Хейлсберг в компанії Microsoft як основна мова розробки додатків для платформи Microsoft .NET. Компілятор з C # входить в стандартну установку самої .NET, тому програми на ньому можна створювати і компілювати навіть без інструментальних засобів, на кшталт Visual Studio. C # відноситься до сім'ї мов з C-подібним синтаксисом, з них його синтаксис найбільш близький до C ++ і Java. Мова має статичну типізацію, підтримує поліморфізм, перевантаження операторів (в тому числі операторів явного і неявного приведення типу), делегати, атрибути, події, властивості, узагальнені типи і методи, ітератори, анонімні функції з підтримкою замикань, LINQ, виключення, коментарі у форматі XML. Перейнявши багато від своїх попередників – мов C ++, Java, Delphi, Модула і Smalltalk – С #, спираючись на практику їх використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем: так, C # не підтримує множинне успадкування класів (на відміну від C ++).

C # розроблявся як мова програмування прикладного рівня для CLR і, як такий, залежить, перш за все, від можливостей самої CLR. Це стосується, перш за все, системи типів C #, яка відображає BCL.

Присутність або відсутність тих чи інших виразних особливостей мови диктується тим, чи може конкретна мовна особливість бути трансльований в відповідні конструкції CLR. Так, з розвитком CLR від версії 1.1 до 2.0 значно збагатився і сам C #; подібної взаємодії слід очікувати і в подальшому. (Однак ця закономірність була порушена з виходом C # 3.0, що представляє собою розширення мови, що не спираються на розширення платформи .NET.) CLR надає C #, як і всім іншим .NET-орієнтованим мовам, багато можливостей, яких позбавлені «класичні» мови програмування.

Наприклад, прибирання сміття не реалізована в самому C #, а проводиться CLR для програм, написаних на C # точно так же, як це робиться для програм на VB.NET, J # і ін[11].

Завдяки достатньому та зручному набору інструментів і безкоштовності було, обрано середовище розробки Visual Studio Express.

# [3.3](https://mail.yandex.ua/neo2/#_Toc325035686) Програмна реалізація проекту

3.3.1 Опис алгоритму шифрування **Blowfish** та аналіз можливих атак.Симетричний алгоритм шифрування Blowfish. Розмір блоку якого становить 128 біт, ключ може бути 128, 192, 256 біт, прийнятий в якості стандарту шифрування владою Сполучених штатів Америки. Алгоритм є криптостійким, про це свідчить те що за його допомогою з 2003 року Агентство національної безпеки Сполучних штатів Америки постановили, що захисту шифру достатньо для захисту відомостей, які є державною таємницею.

Blowfish є симетричним блочним алгоритмом шифрування. Містить матрицю (двовимірний масив) байтів, розташованих наступним чином:

Таблиця 3.2 – Матриця байтів

|  |  |  |
| --- | --- | --- |
| 0 | 3 | …. |
| 1 | 4 | …. |
| 2 | 5 | …. |

Алгоритм здійснює шифрування завдяки системі раундів. Раунд – це ітерація циклу перетворення матриці байтів (Табл. 3.2). Кількість ітерацій залежить від довжини ключа, чим більша довжина ключа, тим більша кількість ітерацій.

Ключ раунду – унікальний ключ, який застосовується в кожному окремому раунді.

Таблиця підстановок – таблиця, котра задає відображення одного байту в інший в залежності від ключа.

Обернена таблиця підстановок – аналогічно до таблиці підстановок, задає обернену матрицю таблиці підстановок.

Шифрування відбувається за допомогою наступних функцій перетворення:

* Nr – кількість раундів.
* ExpandKey – функція для вираховування всіх ключів раундів.
* SubBytes – функція для підстановок байтів, які використовують таблицю підстановок.
* ShiftRows – функція, яка забезпечує циклічний зсув у формі на різні величини.
* MixColumns – функція, котра змішує дані всередині кожного стовбця форми;
* AddRoundKey – додавання ключа раунду з формою.

3.3.2 Тестування роботи програми**.** Скомпільований виконуючий файл, займає 348 кілобайт.

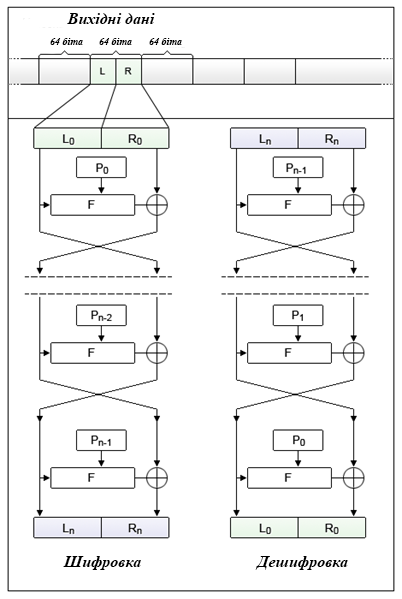


Рисунок 3.3 – Блок-схема алгоритму Blowfish

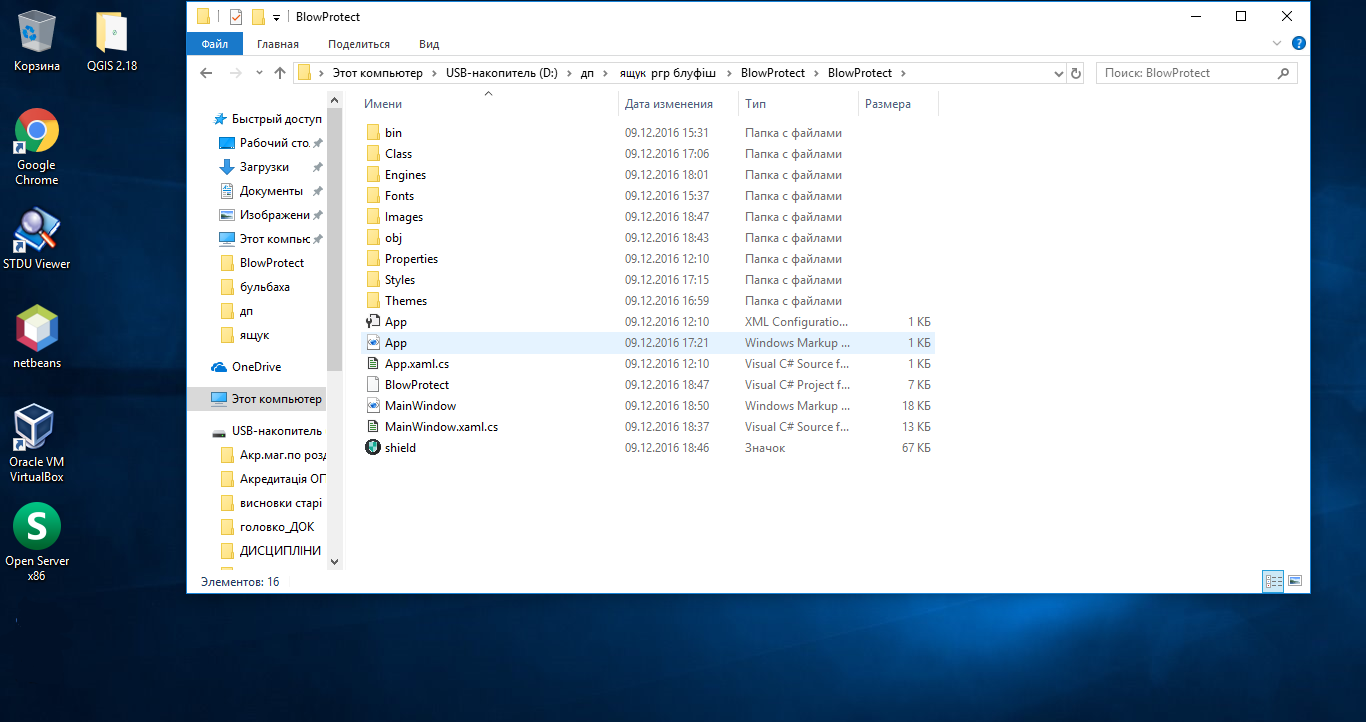


Рисунок 3.4 – Інтерфейс програми при запуску

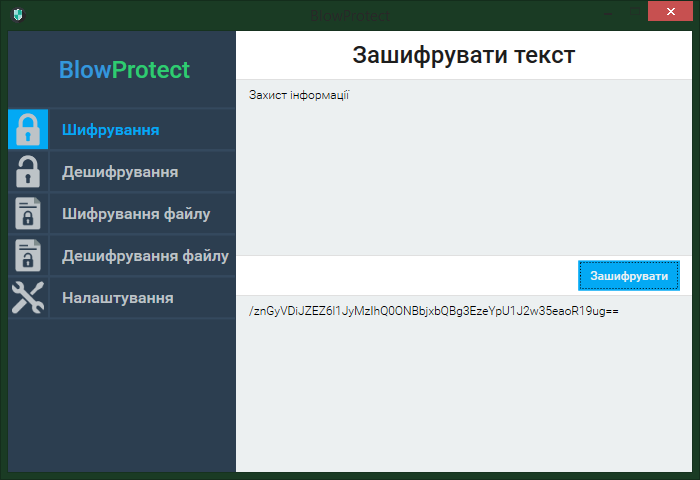


Рисунок 3.5 – Шифрування даних

Для зручності шифрування нового файлу здійснюється у ту саму папку що й знаходження початкового файлу. Зашифрований файл або розшифрований перейменовується з додаванням порядкового номеру у кінці назви файлу.

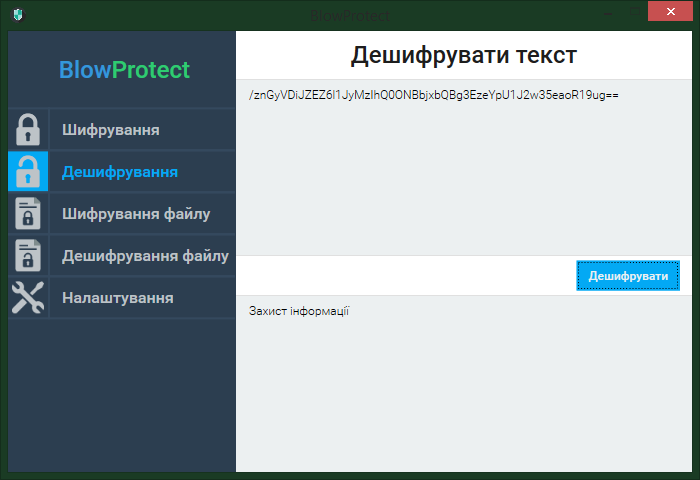


Рисунок 3.6 – Розшифрування даних

Для зручності розшифрований або зашифрований файл зберігається у таку ж саму директорію, що й обраний файл.

У разі вдалого або невдалого шифрування на екран виводиться повідомлення з результатом.

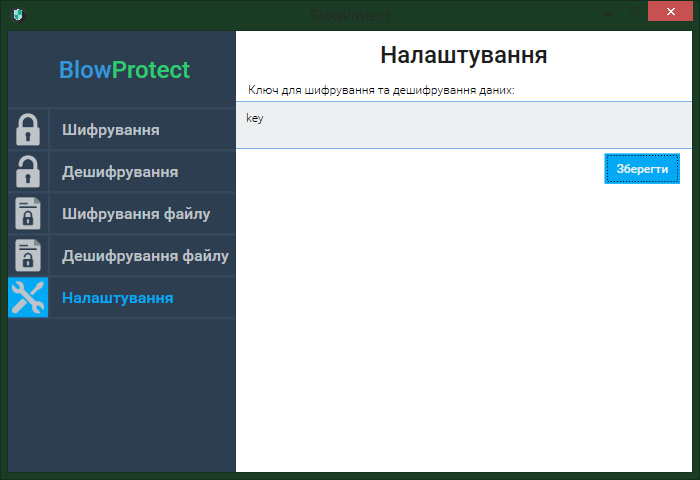


Рисунок 3.7 – Вікно налаштувань

**ВИСНОВКИ**

В дипломній роботі проведено аналіз стану та розроблено комплексні методи захисту інформації на підприємстві “НЕТСО”.

Виходячи з аналізу проблеми сформовано завдання та критерії проведення комплексного аналізу.

В ході роботи було надано рекомендації з захисту інформаційної системи підприємства, а саме захист комп’ютерної мережі, шляхом зміни та оптимізації топології комп’ютерної мережі, рекомендації щодо забезпечення захисту сервера від несанкціонованого доступу, було рекомендовано оновити операційну систему на сервері, встановити антивірус PANDA, задля забезпечення криптографічного захисту було розроблено програмне забезпечення на основі алгоритму шифрування Blowfish. По результатам роботи отримано довідку про впровадження на підприємстві “НЕТСО” (Додаток А).

Таким чином, проведення та впровадження систем комплексних заходів забезпечує безпеку інформації на підприємстві.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ситник Г. Інформаційна безпека [Електронний ресурс] / Галина Ситник // СерчІнформ – Режим доступу до ресурсу: https://searchinform.ru/informatsionnaya-bezopasnost/.
2. Дослідженння IBM і Ponemon Institute [Електронний ресурс] // IBM – Режим доступу до ресурсу: <https://www-03.ibm.com/press/ru/ru/pressrelease/50084.wss>. Шквір В. Д.
3. Наймасшатбніші комп'ютерні атаки у світі [Електронний ресурс] // ТАСС. – 2017. – Режим доступу до ресурсу: <http://tass.ru/info/4248876>.
4. Економічна енциклопедія: У трьох томах. Т. 1. / Редкол.: . В. Мочерний (відп. ред.) та ін. – К.: Видавничий центр “Академія”, 2000. – 864 с.
5. Закон України. Про захист інформації в інформаційно-телекомунікаційних системах // http://zakon.rada.gov.ua/laws/show/80/94-вр
6. Явтуховский Е. Ю., Кошелев С. О. Современные технологии защиты информации в распределённых системах // Молодой ученый. — 2016. — №28. — С. 43-45. — URL https://moluch.ru/archive/132/36986/ (дата обращения: 12.06.2018).
7. Комп'ютерні віруси. Визначення, класифікація і способи захисту [Електронний ресурс] – Режим доступу до ресурсу: <http://iteranet.ru/it-novosti/2013/08/10/kompyuternye-virusy-opredelenie-klassifikaciya-i-sposoby-zashhity/>.
8. Brandon G. 37 Shocking Computer Virus Statistics [Електронний ресурс] / Gaille Brandon. – 2017. – Режим доступу до ресурсу: <https://brandongaille.com/36-shocking-computer-virus-statistics/>.
9. Постанова Кабінету міністрів України [Електронний ресурс]. – 2006. – Режим доступу до ресурсу: <http://www.ukrbook.net/zakony/Sfera_inform/Pos_373.pdf>.
10. Сушко С. А. ПРАКТИЧЕСКАЯ КРИПТОЛОГИЯ ЛЕКЦИЯ 9 [Електронний ресурс] / С. А. Сушко – Режим доступу до ресурсу: <http://bit.nmu.org.ua/ua/student/metod/cryptology/%D0%BB%D0%B5%D0%BA%D1%86%D0%B8%D1%8F%209.pdf>.
11. Лекция 3 технологии программирования [Електронний ресурс] // Белорусско-Российский университет – Режим доступу до ресурсу: https://studfiles.net/preview/5618019/page:3/.

# ДОДАТОК А ДОВІДКА ПРО ВПРОВАДЖЕННЯ

ГО «Науково-практичний центр сприяння розвитку конкуренцій та захисту прав споживачів»

36014

м. Полтава, вулиця Зигіна, 1

АКТ

У ході виконання кваліфікаційної роботи було проаналізовано стан інформаційної безпеки на підприємствіта створено програмний додаток для захисту та передачі даних по мережі підприємства з використанням алгоритму шифрування Blowfish та асоціативного відбору.

Робота виконувалась по замовленню ТОВ «НЕСТО», з подальшим можливим працевлаштуванням.

Організація задоволена співпрацею зі студентом Саркісяном А.С. , було виявлено значну кількість не доопрацювань в сфері захисту інформації, які в подальшому могли привести до значних фінансових втрат.

Вважаємо, що студент Саркісян А.С. є кваліфікованим спеціалістом і заслуговує на оцінку «відмінно». Програмний додаток повністю відповідає всім вимогам, поставленим на етапі постановки задачі, та готовий до експлуатації.

17.06.2021р.

# ****ДОДАТОК В**** ****ВИХІДНИЙ КОД ПРОГРАМИ****

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

namespace BlowProtect

{

class BlowfishEngine

{

#region PreCalculated Tables

private readonly static uint[] PArray =

{

0x243F6A88, 0x85A308D3, 0x13198A2E, 0x03707344,

0xA4093822, 0x299F31D0, 0x082EFA98, 0xEC4E6C89,

0x452821E6, 0x38D01377, 0xBE5466CF, 0x34E90C6C,

0xC0AC29B7, 0xC97C50DD, 0x3F84D5B5, 0xB5470917,

0x9216D5D9, 0x8979FB1B

};

private readonly static uint[] SBox0 =

{

0xD1310BA6, 0x98DFB5AC, 0x2FFD72DB, 0xD01ADFB7,

0xB8E1AFED, 0x6A267E96, 0xBA7C9045, 0xF12C7F99,

0x24A19947, 0xB3916CF7, 0x0801F2E2, 0x858EFC16,

0x636920D8, 0x71574E69, 0xA458FEA3, 0xF4933D7E,

0x0D95748F, 0x728EB658, 0x718BCD58, 0x82154AEE,

0x7B54A41D, 0xC25A59B5, 0x9C30D539, 0x2AF26013,

0xC5D1B023, 0x286085F0, 0xCA417918, 0xB8DB38EF,

0x8E79DCB0, 0x603A180E, 0x6C9E0E8B, 0xB01E8A3E,

0xD71577C1, 0xBD314B27, 0x78AF2FDA, 0x55605C60,

0xE65525F3, 0xAA55AB94, 0x57489862, 0x63E81440,

0x55CA396A, 0x2AAB10B6, 0xB4CC5C34, 0x1141E8CE,

0xA15486AF, 0x7C72E993, 0xB3EE1411, 0x636FBC2A,

0x2BA9C55D, 0x741831F6, 0xCE5C3E16, 0x9B87931E,

0xAFD6BA33, 0x6C24CF5C, 0x7A325381, 0x28958677,

0x3B8F4898, 0x6B4BB9AF, 0xC4BFE81B, 0x66282193,

0x61D809CC, 0xFB21A991, 0x487CAC60, 0x5DEC8032,

0xEF845D5D, 0xE98575B1, 0xDC262302, 0xEB651B88,

0x23893E81, 0xD396ACC5, 0x0F6D6FF3, 0x83F44239,

0x2E0B4482, 0xA4842004, 0x69C8F04A, 0x9E1F9B5E,

0x21C66842, 0xF6E96C9A, 0x670C9C61, 0xABD388F0,

0x6A51A0D2, 0xD8542F68, 0x960FA728, 0xAB5133A3,

0x6EEF0B6C, 0x137A3BE4, 0xBA3BF050, 0x7EFB2A98,

0xA1F1651D, 0x39AF0176, 0x66CA593E, 0x82430E88,

0x8CEE8619, 0x456F9FB4, 0x7D84A5C3, 0x3B8B5EBE,

0xE06F75D8, 0x85C12073, 0x401A449F, 0x56C16AA6,

0x4ED3AA62, 0x363F7706, 0x1BFEDF72, 0x429B023D,

0x37D0D724, 0xD00A1248, 0xDB0FEAD3, 0x49F1C09B,

0x075372C9, 0x80991B7B, 0x25D479D8, 0xF6E8DEF7,

0xE3FE501A, 0xB6794C3B, 0x976CE0BD, 0x04C006BA,

0xC1A94FB6, 0x409F60C4, 0x5E5C9EC2, 0x196A2463,

0x68FB6FAF, 0x3E6C53B5, 0x1339B2EB, 0x3B52EC6F,

0x6DFC511F, 0x9B30952C, 0xCC814544, 0xAF5EBD09,

0xBEE3D004, 0xDE334AFD, 0x660F2807, 0x192E4BB3,

0xC0CBA857, 0x45C8740F, 0xD20B5F39, 0xB9D3FBDB,

0x5579C0BD, 0x1A60320A, 0xD6A100C6, 0x402C7279,

0x679F25FE, 0xFB1FA3CC, 0x8EA5E9F8, 0xDB3222F8,

0x3C7516DF, 0xFD616B15, 0x2F501EC8, 0xAD0552AB,

0x323DB5FA, 0xFD238760, 0x53317B48, 0x3E00DF82,

0x9E5C57BB, 0xCA6F8CA0, 0x1A87562E, 0xDF1769DB,

0xD542A8F6, 0x287EFFC3, 0xAC6732C6, 0x8C4F5573,

0x695B27B0, 0xBBCA58C8, 0xE1FFA35D, 0xB8F011A0,

0x10FA3D98, 0xFD2183B8, 0x4AFCB56C, 0x2DD1D35B,

0x9A53E479, 0xB6F84565, 0xD28E49BC, 0x4BFB9790,

0xE1DDF2DA, 0xA4CB7E33, 0x62FB1341, 0xCEE4C6E8,

0xEF20CADA, 0x36774C01, 0xD07E9EFE, 0x2BF11FB4,

0x95DBDA4D, 0xAE909198, 0xEAAD8E71, 0x6B93D5A0,

0xD08ED1D0, 0xAFC725E0, 0x8E3C5B2F, 0x8E7594B7,

0x8FF6E2FB, 0xF2122B64, 0x8888B812, 0x900DF01C,

0x4FAD5EA0, 0x688FC31C, 0xD1CFF191, 0xB3A8C1AD,

0x2F2F2218, 0xBE0E1777, 0xEA752DFE, 0x8B021FA1,

0xE5A0CC0F, 0xB56F74E8, 0x18ACF3D6, 0xCE89E299,

0xB4A84FE0, 0xFD13E0B7, 0x7CC43B81, 0xD2ADA8D9,

0x165FA266, 0x80957705, 0x93CC7314, 0x211A1477,

0xE6AD2065, 0x77B5FA86, 0xC75442F5, 0xFB9D35CF,

0xEBCDAF0C, 0x7B3E89A0, 0xD6411BD3, 0xAE1E7E49,

0x00250E2D, 0x2071B35E, 0x226800BB, 0x57B8E0AF,

0x2464369B, 0xF009B91E, 0x5563911D, 0x59DFA6AA,

0x78C14389, 0xD95A537F, 0x207D5BA2, 0x02E5B9C5,

0x83260376, 0x6295CFA9, 0x11C81968, 0x4E734A41,

0xB3472DCA, 0x7B14A94A, 0x1B510052, 0x9A532915,

0xD60F573F, 0xBC9BC6E4, 0x2B60A476, 0x81E67400,

0x08BA6FB5, 0x571BE91F, 0xF296EC6B, 0x2A0DD915,

0xB6636521, 0xE7B9F9B6, 0xFF34052E, 0xC5855664,

0x53B02D5D, 0xA99F8FA1, 0x08BA4799, 0x6E85076A

};

private readonly static uint[] SBox1 =

{

0x4B7A70E9, 0xB5B32944, 0xDB75092E, 0xC4192623,

0xAD6EA6B0, 0x49A7DF7D, 0x9CEE60B8, 0x8FEDB266,

0xECAA8C71, 0x699A17FF, 0x5664526C, 0xC2B19EE1,

0x193602A5, 0x75094C29, 0xA0591340, 0xE4183A3E,

0x3F54989A, 0x5B429D65, 0x6B8FE4D6, 0x99F73FD6,

0xA1D29C07, 0xEFE830F5, 0x4D2D38E6, 0xF0255DC1,

0x4CDD2086, 0x8470EB26, 0x6382E9C6, 0x021ECC5E,

0x09686B3F, 0x3EBAEFC9, 0x3C971814, 0x6B6A70A1,

0x687F3584, 0x52A0E286, 0xB79C5305, 0xAA500737,

0x3E07841C, 0x7FDEAE5C, 0x8E7D44EC, 0x5716F2B8,

0xB03ADA37, 0xF0500C0D, 0xF01C1F04, 0x0200B3FF,

0xAE0CF51A, 0x3CB574B2, 0x25837A58, 0xDC0921BD,

0xD19113F9, 0x7CA92FF6, 0x94324773, 0x22F54701,

0x3AE5E581, 0x37C2DADC, 0xC8B57634, 0x9AF3DDA7,

0xA9446146, 0x0FD0030E, 0xECC8C73E, 0xA4751E41,

0xE238CD99, 0x3BEA0E2F, 0x3280BBA1, 0x183EB331,

0x4E548B38, 0x4F6DB908, 0x6F420D03, 0xF60A04BF,

0x2CB81290, 0x24977C79, 0x5679B072, 0xBCAF89AF,

0xDE9A771F, 0xD9930810, 0xB38BAE12, 0xDCCF3F2E,

0x5512721F, 0x2E6B7124, 0x501ADDE6, 0x9F84CD87,

0x7A584718, 0x7408DA17, 0xBC9F9ABC, 0xE94B7D8C,

0xEC7AEC3A, 0xDB851DFA, 0x63094366, 0xC464C3D2,

0xEF1C1847, 0x3215D908, 0xDD433B37, 0x24C2BA16,

0x12A14D43, 0x2A65C451, 0x50940002, 0x133AE4DD,

0x71DFF89E, 0x10314E55, 0x81AC77D6, 0x5F11199B,

0x043556F1, 0xD7A3C76B, 0x3C11183B, 0x5924A509,

0xF28FE6ED, 0x97F1FBFA, 0x9EBABF2C, 0x1E153C6E,

0x86E34570, 0xEAE96FB1, 0x860E5E0A, 0x5A3E2AB3,

0x771FE71C, 0x4E3D06FA, 0x2965DCB9, 0x99E71D0F,

0x803E89D6, 0x5266C825, 0x2E4CC978, 0x9C10B36A,

0xC6150EBA, 0x94E2EA78, 0xA5FC3C53, 0x1E0A2DF4,

0xF2F74EA7, 0x361D2B3D, 0x1939260F, 0x19C27960,

0x5223A708, 0xF71312B6, 0xEBADFE6E, 0xEAC31F66,

0xE3BC4595, 0xA67BC883, 0xB17F37D1, 0x018CFF28,

0xC332DDEF, 0xBE6C5AA5, 0x65582185, 0x68AB9802,

0xEECEA50F, 0xDB2F953B, 0x2AEF7DAD, 0x5B6E2F84,

0x1521B628, 0x29076170, 0xECDD4775, 0x619F1510,

0x13CCA830, 0xEB61BD96, 0x0334FE1E, 0xAA0363CF,

0xB5735C90, 0x4C70A239, 0xD59E9E0B, 0xCBAADE14,

0xEECC86BC, 0x60622CA7, 0x9CAB5CAB, 0xB2F3846E,

0x648B1EAF, 0x19BDF0CA, 0xA02369B9, 0x655ABB50,

0x40685A32, 0x3C2AB4B3, 0x319EE9D5, 0xC021B8F7,

0x9B540B19, 0x875FA099, 0x95F7997E, 0x623D7DA8,

0xF837889A, 0x97E32D77, 0x11ED935F, 0x16681281,

0x0E358829, 0xC7E61FD6, 0x96DEDFA1, 0x7858BA99,

0x57F584A5, 0x1B227263, 0x9B83C3FF, 0x1AC24696,

0xCDB30AEB, 0x532E3054, 0x8FD948E4, 0x6DBC3128,

0x58EBF2EF, 0x34C6FFEA, 0xFE28ED61, 0xEE7C3C73,

0x5D4A14D9, 0xE864B7E3, 0x42105D14, 0x203E13E0,

0x45EEE2B6, 0xA3AAABEA, 0xDB6C4F15, 0xFACB4FD0,

0xC742F442, 0xEF6ABBB5, 0x654F3B1D, 0x41CD2105,

0xD81E799E, 0x86854DC7, 0xE44B476A, 0x3D816250,

0xCF62A1F2, 0x5B8D2646, 0xFC8883A0, 0xC1C7B6A3,

0x7F1524C3, 0x69CB7492, 0x47848A0B, 0x5692B285,

0x095BBF00, 0xAD19489D, 0x1462B174, 0x23820E00,

0x58428D2A, 0x0C55F5EA, 0x1DADF43E, 0x233F7061,

0x3372F092, 0x8D937E41, 0xD65FECF1, 0x6C223BDB,

0x7CDE3759, 0xCBEE7460, 0x4085F2A7, 0xCE77326E,

0xA6078084, 0x19F8509E, 0xE8EFD855, 0x61D99735,

0xA969A7AA, 0xC50C06C2, 0x5A04ABFC, 0x800BCADC,

0x9E447A2E, 0xC3453484, 0xFDD56705, 0x0E1E9EC9,

0xDB73DBD3, 0x105588CD, 0x675FDA79, 0xE3674340,

0xC5C43465, 0x713E38D8, 0x3D28F89E, 0xF16DFF20,

0x153E21E7, 0x8FB03D4A, 0xE6E39F2B, 0xDB83ADF7

};

private readonly static uint[] SBox2 =

{

0xE93D5A68, 0x948140F7, 0xF64C261C, 0x94692934,

0x411520F7, 0x7602D4F7, 0xBCF46B2E, 0xD4A20068,

0xD4082471, 0x3320F46A, 0x43B7D4B7, 0x500061AF,

0x1E39F62E, 0x97244546, 0x14214F74, 0xBF8B8840,

0x4D95FC1D, 0x96B591AF, 0x70F4DDD3, 0x66A02F45,

0xBFBC09EC, 0x03BD9785, 0x7FAC6DD0, 0x31CB8504,

0x96EB27B3, 0x55FD3941, 0xDA2547E6, 0xABCA0A9A,

0x28507825, 0x530429F4, 0x0A2C86DA, 0xE9B66DFB,

0x68DC1462, 0xD7486900, 0x680EC0A4, 0x27A18DEE,

0x4F3FFEA2, 0xE887AD8C, 0xB58CE006, 0x7AF4D6B6,

0xAACE1E7C, 0xD3375FEC, 0xCE78A399, 0x406B2A42,

0x20FE9E35, 0xD9F385B9, 0xEE39D7AB, 0x3B124E8B,

0x1DC9FAF7, 0x4B6D1856, 0x26A36631, 0xEAE397B2,

0x3A6EFA74, 0xDD5B4332, 0x6841E7F7, 0xCA7820FB,

0xFB0AF54E, 0xD8FEB397, 0x454056AC, 0xBA489527,

0x55533A3A, 0x20838D87, 0xFE6BA9B7, 0xD096954B,

0x55A867BC, 0xA1159A58, 0xCCA92963, 0x99E1DB33,

0xA62A4A56, 0x3F3125F9, 0x5EF47E1C, 0x9029317C,

0xFDF8E802, 0x04272F70, 0x80BB155C, 0x05282CE3,

0x95C11548, 0xE4C66D22, 0x48C1133F, 0xC70F86DC,

0x07F9C9EE, 0x41041F0F, 0x404779A4, 0x5D886E17,

0x325F51EB, 0xD59BC0D1, 0xF2BCC18F, 0x41113564,

0x257B7834, 0x602A9C60, 0xDFF8E8A3, 0x1F636C1B,

0x0E12B4C2, 0x02E1329E, 0xAF664FD1, 0xCAD18115,

0x6B2395E0, 0x333E92E1, 0x3B240B62, 0xEEBEB922,

0x85B2A20E, 0xE6BA0D99, 0xDE720C8C, 0x2DA2F728,

0xD0127845, 0x95B794FD, 0x647D0862, 0xE7CCF5F0,

0x5449A36F, 0x877D48FA, 0xC39DFD27, 0xF33E8D1E,

0x0A476341, 0x992EFF74, 0x3A6F6EAB, 0xF4F8FD37,

0xA812DC60, 0xA1EBDDF8, 0x991BE14C, 0xDB6E6B0D,

0xC67B5510, 0x6D672C37, 0x2765D43B, 0xDCD0E804,

0xF1290DC7, 0xCC00FFA3, 0xB5390F92, 0x690FED0B,

0x667B9FFB, 0xCEDB7D9C, 0xA091CF0B, 0xD9155EA3,

0xBB132F88, 0x515BAD24, 0x7B9479BF, 0x763BD6EB,

0x37392EB3, 0xCC115979, 0x8026E297, 0xF42E312D,

0x6842ADA7, 0xC66A2B3B, 0x12754CCC, 0x782EF11C,

0x6A124237, 0xB79251E7, 0x06A1BBE6, 0x4BFB6350,

0x1A6B1018, 0x11CAEDFA, 0x3D25BDD8, 0xE2E1C3C9,

0x44421659, 0x0A121386, 0xD90CEC6E, 0xD5ABEA2A,

0x64AF674E, 0xDA86A85F, 0xBEBFE988, 0x64E4C3FE,

0x9DBC8057, 0xF0F7C086, 0x60787BF8, 0x6003604D,

0xD1FD8346, 0xF6381FB0, 0x7745AE04, 0xD736FCCC,

0x83426B33, 0xF01EAB71, 0xB0804187, 0x3C005E5F,

0x77A057BE, 0xBDE8AE24, 0x55464299, 0xBF582E61,

0x4E58F48F, 0xF2DDFDA2, 0xF474EF38, 0x8789BDC2,

0x5366F9C3, 0xC8B38E74, 0xB475F255, 0x46FCD9B9,

0x7AEB2661, 0x8B1DDF84, 0x846A0E79, 0x915F95E2,

0x466E598E, 0x20B45770, 0x8CD55591, 0xC902DE4C,

0xB90BACE1, 0xBB8205D0, 0x11A86248, 0x7574A99E,

0xB77F19B6, 0xE0A9DC09, 0x662D09A1, 0xC4324633,

0xE85A1F02, 0x09F0BE8C, 0x4A99A025, 0x1D6EFE10,

0x1AB93D1D, 0x0BA5A4DF, 0xA186F20F, 0x2868F169,

0xDCB7DA83, 0x573906FE, 0xA1E2CE9B, 0x4FCD7F52,

0x50115E01, 0xA70683FA, 0xA002B5C4, 0x0DE6D027,

0x9AF88C27, 0x773F8641, 0xC3604C06, 0x61A806B5,

0xF0177A28, 0xC0F586E0, 0x006058AA, 0x30DC7D62,

0x11E69ED7, 0x2338EA63, 0x53C2DD94, 0xC2C21634,

0xBBCBEE56, 0x90BCB6DE, 0xEBFC7DA1, 0xCE591D76,

0x6F05E409, 0x4B7C0188, 0x39720A3D, 0x7C927C24,

0x86E3725F, 0x724D9DB9, 0x1AC15BB4, 0xD39EB8FC,

0xED545578, 0x08FCA5B5, 0xD83D7CD3, 0x4DAD0FC4,

0x1E50EF5E, 0xB161E6F8, 0xA28514D9, 0x6C51133C,

0x6FD5C7E7, 0x56E14EC4, 0x362ABFCE, 0xDDC6C837,

0xD79A3234, 0x92638212, 0x670EFA8E, 0x406000E0

};

private readonly static uint[] SBox3 =

{

0x3A39CE37, 0xD3FAF5CF, 0xABC27737, 0x5AC52D1B,

0x5CB0679E, 0x4FA33742, 0xD3822740, 0x99BC9BBE,

0xD5118E9D, 0xBF0F7315, 0xD62D1C7E, 0xC700C47B,

0xB78C1B6B, 0x21A19045, 0xB26EB1BE, 0x6A366EB4,

0x5748AB2F, 0xBC946E79, 0xC6A376D2, 0x6549C2C8,

0x530FF8EE, 0x468DDE7D, 0xD5730A1D, 0x4CD04DC6,

0x2939BBDB, 0xA9BA4650, 0xAC9526E8, 0xBE5EE304,

0xA1FAD5F0, 0x6A2D519A, 0x63EF8CE2, 0x9A86EE22,

0xC089C2B8, 0x43242EF6, 0xA51E03AA, 0x9CF2D0A4,

0x83C061BA, 0x9BE96A4D, 0x8FE51550, 0xBA645BD6,

0x2826A2F9, 0xA73A3AE1, 0x4BA99586, 0xEF5562E9,

0xC72FEFD3, 0xF752F7DA, 0x3F046F69, 0x77FA0A59,

0x80E4A915, 0x87B08601, 0x9B09E6AD, 0x3B3EE593,

0xE990FD5A, 0x9E34D797, 0x2CF0B7D9, 0x022B8B51,

0x96D5AC3A, 0x017DA67D, 0xD1CF3ED6, 0x7C7D2D28,

0x1F9F25CF, 0xADF2B89B, 0x5AD6B472, 0x5A88F54C,

0xE029AC71, 0xE019A5E6, 0x47B0ACFD, 0xED93FA9B,

0xE8D3C48D, 0x283B57CC, 0xF8D56629, 0x79132E28,

0x785F0191, 0xED756055, 0xF7960E44, 0xE3D35E8C,

0x15056DD4, 0x88F46DBA, 0x03A16125, 0x0564F0BD,

0xC3EB9E15, 0x3C9057A2, 0x97271AEC, 0xA93A072A,

0x1B3F6D9B, 0x1E6321F5, 0xF59C66FB, 0x26DCF319,

0x7533D928, 0xB155FDF5, 0x03563482, 0x8ABA3CBB,

0x28517711, 0xC20AD9F8, 0xABCC5167, 0xCCAD925F,

0x4DE81751, 0x3830DC8E, 0x379D5862, 0x9320F991,

0xEA7A90C2, 0xFB3E7BCE, 0x5121CE64, 0x774FBE32,

0xA8B6E37E, 0xC3293D46, 0x48DE5369, 0x6413E680,

0xA2AE0810, 0xDD6DB224, 0x69852DFD, 0x09072166,

0xB39A460A, 0x6445C0DD, 0x586CDECF, 0x1C20C8AE,

0x5BBEF7DD, 0x1B588D40, 0xCCD2017F, 0x6BB4E3BB,

0xDDA26A7E, 0x3A59FF45, 0x3E350A44, 0xBCB4CDD5,

0x72EACEA8, 0xFA6484BB, 0x8D6612AE, 0xBF3C6F47,

0xD29BE463, 0x542F5D9E, 0xAEC2771B, 0xF64E6370,

0x740E0D8D, 0xE75B1357, 0xF8721671, 0xAF537D5D,

0x4040CB08, 0x4EB4E2CC, 0x34D2466A, 0x0115AF84,

0xE1B00428, 0x95983A1D, 0x06B89FB4, 0xCE6EA048,

0x6F3F3B82, 0x3520AB82, 0x011A1D4B, 0x277227F8,

0x611560B1, 0xE7933FDC, 0xBB3A792B, 0x344525BD,

0xA08839E1, 0x51CE794B, 0x2F32C9B7, 0xA01FBAC9,

0xE01CC87E, 0xBCC7D1F6, 0xCF0111C3, 0xA1E8AAC7,

0x1A908749, 0xD44FBD9A, 0xD0DADECB, 0xD50ADA38,

0x0339C32A, 0xC6913667, 0x8DF9317C, 0xE0B12B4F,

0xF79E59B7, 0x43F5BB3A, 0xF2D519FF, 0x27D9459C,

0xBF97222C, 0x15E6FC2A, 0x0F91FC71, 0x9B941525,

0xFAE59361, 0xCEB69CEB, 0xC2A86459, 0x12BAA8D1,

0xB6C1075E, 0xE3056A0C, 0x10D25065, 0xCB03A442,

0xE0EC6E0E, 0x1698DB3B, 0x4C98A0BE, 0x3278E964,

0x9F1F9532, 0xE0D392DF, 0xD3A0342B, 0x8971F21E,

0x1B0A7441, 0x4BA3348C, 0xC5BE7120, 0xC37632D8,

0xDF359F8D, 0x9B992F2E, 0xE60B6F47, 0x0FE3F11D,

0xE54CDA54, 0x1EDAD891, 0xCE6279CF, 0xCD3E7E6F,

0x1618B166, 0xFD2C1D05, 0x848FD2C5, 0xF6FB2299,

0xF523F357, 0xA6327623, 0x93A83531, 0x56CCCD02,

0xACF08162, 0x5A75EBB5, 0x6E163697, 0x88D273CC,

0xDE966292, 0x81B949D0, 0x4C50901B, 0x71C65614,

0xE6C6C7BD, 0x327A140A, 0x45E1D006, 0xC3F27B9A,

0xC9AA53FD, 0x62A80F00, 0xBB25BFE2, 0x35BDD2F6,

0x71126905, 0xB2040222, 0xB6CBCF7C, 0xCD769C2B,

0x53113EC0, 0x1640E3D3, 0x38ABBD60, 0x2547ADF0,

0xBA38209C, 0xF746CE76, 0x77AFA1C5, 0x20756060,

0x85CBFE4E, 0x8AE88DD8, 0x7AAAF9B0, 0x4CF9AA7E,

0x1948C25C, 0x02FB8A8C, 0x01C36AE4, 0xD6EBE1F9,

0x90D4F869, 0xA65CDEA0, 0x3F09252D, 0xC208E69F,

0xB74E6132, 0xCE77E25B, 0x578FDFE3, 0x3AC372E6

};

#endregion

#region Constants

private static readonly int Rounds = 16;

private const int BlockSize = 8; // 64 bits block

private static readonly int SBoxSize = 256;

private static readonly int PArraySize = 18;

private readonly uint[] S0, S1, S2, S3; // the S-boxes

private readonly uint[] P; // the P-array

#endregion

/// <summary>

/// Cipher key

/// </summary>

private byte[] Key;

public BlowfishEngine()

{

S0 = new uint[SBoxSize];

S1 = new uint[SBoxSize];

S2 = new uint[SBoxSize];

S3 = new uint[SBoxSize];

P = new uint[PArraySize];

}

public void Init(string key)

{

if (key.Length >= 2 && key.Length <= 28)

{

this.Key = Encoding.Unicode.GetBytes(key);

SetKey(this.Key);

}

else

{

MessageBox.Show(

"Невірний ключ. Ключ повинен бути довжиною від 2 до 28 символів.",

"Помилка",

MessageBoxButton.OK,

MessageBoxImage.Error

);

}

}

public int GetBlockSize()

{

return BlockSize;

}

//==================================

// Private Implementation

//==================================

private uint F(uint x)

{

uint h = S0[x >> 24] + S1[(x >> 16) & 0xff];

return ((h ^ S2[(x >> 8) & 0xff]) + S3[x & 0xff]);

}

/\*\*

\* apply the encryption cycle to each value pair in the table.

\*/

private void ProcessTable(uint xl, uint xr, uint[] table)

{

int size = table.Length;

for (int s = 0; s < size; s += 2)

{

xl ^= P[0];

for (int i = 1; i < Rounds; i += 2)

{

xr ^= F(xl) ^ P[i];

xl ^= F(xr) ^ P[i + 1];

}

xr ^= P[Rounds + 1];

table[s] = xr;

table[s + 1] = xl;

xr = xl; // end of cycle swap

xl = table[s];

}

}

private void SetKey(byte[] key)

{

Array.Copy(SBox0, S0, SBoxSize);

Array.Copy(SBox1, S1, SBoxSize);

Array.Copy(SBox2, S2, SBoxSize);

Array.Copy(SBox3, S3, SBoxSize);

Array.Copy(PArray, P, PArraySize);

int keyIndex = 0;

for (int i = 0; i < PArraySize; i++)

{

// Get the 32 bits of the key, in 4 \* 8 bit chunks

uint data = 0x0000000;

for (int j = 0; j < 4; j++)

{

// create a 32 bit block

data = (data << 8) | (uint)key[keyIndex++];

// wrap when we get to the end of the key

if (keyIndex >= key.Length)

{

keyIndex = 0;

}

}

// XOR the newly created 32 bit chunk onto the P-array

P[i] ^= data;

}

ProcessTable(0, 0, P);

ProcessTable(P[PArraySize - 2], P[PArraySize - 1], S0);

ProcessTable(S0[SBoxSize - 2], S0[SBoxSize - 1], S1);

ProcessTable(S1[SBoxSize - 2], S1[SBoxSize - 1], S2);

ProcessTable(S2[SBoxSize - 2], S2[SBoxSize - 1], S3);

}

/\*\*

\* Encrypt the given input starting at the given offset and place

\* the result in the provided buffer starting at the given offset.

\* The input will be an exact multiple of our blocksize.

\*/

private byte[] EncryptBlock(byte[] inputBlock)

{

byte[] outputBlock = new byte[BlockSize];

uint xl = BitConverter.ToUInt32(inputBlock, 0);

uint xr = BitConverter.ToUInt32(inputBlock, 4);

for (int i = 0; i < Rounds; i += 2)

{

xl ^= P[i];

xr ^= F(xl);

xr ^= P[i + 1];

xl ^= F(xr);

}

xl ^= P[Rounds];

xr ^= P[Rounds + 1];

byte[] right = BitConverter.GetBytes(xr);

byte[] left = BitConverter.GetBytes(xl);

int halfSizeOfBlock = BlockSize / 2;

for (int i = 0; i < halfSizeOfBlock; i++)

{

outputBlock[i] = right[i];

}

for (int i = 0; i < halfSizeOfBlock; i++)

{

outputBlock[halfSizeOfBlock + i] = left[i];

}

return outputBlock;

}

public string Encrypt(string data)

{

try

{

int temp = 0;

*List*<byte[]> encryptedBlocks = new *List*<byte[]>();

byte[] input = Encoding.Unicode.GetBytes(data);

byte[] block = new byte[BlockSize];

byte[] outputBlock = new byte[BlockSize];

// Кодируем входную строку попутно разбивая её на блоки по 8 байт

for (int i = 0; i < input.Length; i++)

{

if (temp == BlockSize)

{

outputBlock = EncryptBlock(block);

encryptedBlocks.Add(outputBlock);

block = new byte[BlockSize];

outputBlock = new byte[BlockSize];

temp = 0;

}

block[temp] = input[i];

if (i + 1 == input.Length && temp < BlockSize)

{

outputBlock = EncryptBlock(block);

encryptedBlocks.Add(outputBlock);

}

temp++;

}

// Собираем закодированные блоки в одно целое

byte[] encryptedBytes = new byte[*encryptedBlocks*.*Count* \* BlockSize];

int counter = 0;

foreach (var encryptedBlock in encryptedBlocks)

{

for (int i = 0; i < BlockSize; i++)

{

encryptedBytes[counter] = encryptedBlock[i];

counter++;

}

}

// Кодируем в Base64 для нормального отображения и возвращаем полученную строку

return Convert.ToBase64String(encryptedBytes);

}

catch (*Exception*)

{

MessageBox.Show(

"Помилка при шифруванні даних.",

"Помилка",

MessageBoxButton.OK,

MessageBoxImage.Error

);

return data;

}

}

/\*\*

\* Decrypt the given input starting at the given offset and place

\* the result in the provided buffer starting at the given offset.

\* The input will be an exact multiple of our blocksize.

\*/

private void DecryptBlock(byte[] inputBlock, byte[] outputBlock)

{

uint xl = BitConverter.ToUInt32(inputBlock, 0);

uint xr = BitConverter.ToUInt32(inputBlock, 4);

for (int i = Rounds; i > 0; i -= 2)

{

xl ^= P[i + 1];

xr ^= F(xl);

xr ^= P[i];

xl ^= F(xr);

}

xl ^= P[1];

xr ^= P[0];

byte[] right = BitConverter.GetBytes(xr);

byte[] left = BitConverter.GetBytes(xl);

int halfSizeOfBlock = BlockSize / 2;

for (int i = 0; i < halfSizeOfBlock; i++)

{

outputBlock[i] = right[i];

}

for (int i = 0; i < halfSizeOfBlock; i++)

{

outputBlock[halfSizeOfBlock + i] = left[i];

}

}

public string Decrypt(string data)

{

try

{

int temp = 0;

string decryptedData = "";

byte[] input = Convert.FromBase64String(data); // Декодируем из Base64

byte[] block = new byte[BlockSize];

byte[] outputBlock = new byte[BlockSize];

// Декодируем входную строку попутно разбивая её на блоки по 8 байт

for (int i = 0; i < input.Length; i++)

{

if (temp == BlockSize)

{

DecryptBlock(block, outputBlock);

decryptedData += Encoding.Unicode.GetString(outputBlock);

block = new byte[BlockSize];

outputBlock = new byte[BlockSize];

temp = 0;

}

block[temp] = input[i];

if (i + 1 == input.Length && temp < BlockSize)

{

DecryptBlock(block, outputBlock);

decryptedData += Encoding.Unicode.GetString(outputBlock);

}

temp++;

}

// Возвращаем декодированныую строку, попутно убирая лишние символы в конце строки

return decryptedData.Replace("\0", "");

}

catch (*Exception*)

{

MessageBox.Show(

"Помилки при дешифруванні даних.",

"Error",

MessageBoxButton.OK,

MessageBoxImage.Error

);

return data;

}

}

}

}

Engine>BlowfishEngine.csusing System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace BlowProtect

{

public class MenuItem : Control

{

static MenuItem()

{

DefaultStyleKeyProperty.OverrideMetadata(

typeof(*MenuItem*),

new *FrameworkPropertyMetadata*(typeof(*MenuItem*))

);

}

public *ImageSource* ButtonImage

{

get { return (*ImageSource*)GetValue(ButtonImageProperty); }

set { SetValue(ButtonImageProperty, value); }

}

public static readonly *DependencyProperty* ButtonImageProperty =

DependencyProperty.Register(

"ButtonImage",

typeof(*ImageSource*),

typeof(*MenuItem*)

);

public string ButtonText

{

get { return (string)GetValue(ButtonTextProperty); }

set { SetValue(ButtonTextProperty, value); }

}

public static readonly *DependencyProperty* ButtonTextProperty =

DependencyProperty.Register(

"ButtonText",

typeof(string),

typeof(*MenuItem*),

new *PropertyMetadata*("Button Title")

);

public *SolidColorBrush* ImageBackground

{

get { return (*SolidColorBrush*)GetValue(ImageBackgroundProperty); }

set { SetValue(ImageBackgroundProperty, value); }

}

public static readonly *DependencyProperty* ImageBackgroundProperty =

DependencyProperty.Register(

"ImageBackground",

typeof(*SolidColorBrush*),

typeof(*MenuItem*),

new *PropertyMetadata*(

new *SolidColorBrush*(Colors.Transparent)

)

);

public *Brush* BasicForeground { get; set; }

private bool \_isActive;

public bool isActive

{

get

{

return \_isActive;

}

set

{

\_isActive = value;

if (value == true)

{

ImageBackground = new *SolidColorBrush*(Color.FromRgb(3, 169, 244));

Foreground = new *SolidColorBrush*(Color.FromRgb(3, 169, 244));

}

else

{

ImageBackground = new *SolidColorBrush*(Colors.Transparent);

Foreground = BasicForeground;

}

}

}

}

}

Class>MenuItem.cs