Міністерство освіти і науки, молоді та спорту України
Національний технічний університет
«Дніпровська політехніка»

Факультет *інформаційних технологій*
Кафедра *безпеки інформації та телекомунікацій*
(повна назва)

ЗАТВЕРДЖЕНО:
завідувач кафедри
Безпеки інформації та телекомунікацій
, проф. _____ В.І.Корнієнко
(підпис)
“___” _____2022 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА
дипломного проекту

магістра
(освітньо-кваліфікаційний рівень)

спеціальності: *125* Кібербезпека

на тему: *“ Методи захисту інформації в каналах зв'язку системи диспетчерської централізації "КАСКАД" ”*

Виконавець: _____ *М. С.Гаржа*
(підпис) (ініціали, прізвище)

Керівники	Прізвище, ініціали	Оцінка	Підпис
Керівник проекту	.т.н., проф. Корнієнко В.І.		
Спеціальної частини	ст. Кручінін О. В.		
Розділів:			
Економічний	к.е.н., доц. Пілова Д.П.		

Рецензент			

Нормоконтролер	ст. Тимофєєв Д.С.		

Дніпро
2022

ЗАТВЕРДЖЕНО:
завідувач кафедри
Безпеки інформації та телекомунікацій
(повна назва)
проф. _____ В.І. Корнієнко
(підпис)
“___” _____20__ р.

ЗАВДАННЯ
на дипломний проект

магістра
(освітньо-кваліфікаційний рівень)

спеціальності: *125 Кібербезпека*

студенту групи *125м-20-1 Гаржі Микиті Сергійовичу*

Тема дипломного проекту: *“Методи захисту інформації в каналах зв'язку системи диспетчерської централізації "КАСКАД" ”*

затверджена наказом ректора НТУ «Дніпровська політехніка» від 10.12.2021 р. № 1036-с

Розділ	**Зміст завдання**	**Термін виконання**
Спеціальна частина	*Розробити рекомендації щодо розробки і впровадження криптографічного протоколу*	10.01.2022 р.
Економічний	*Розрахувати економічну частину впровадження запропонованих заходів*	14.01.2022 р.

Завдання видали:

Керівник дипломного проекту к.т.н., проф. _____Корнієнко О.В.
(підпис)
Керівник спеціальної частини ст. викл. _____ Кручінін О.В.
(підпис)
Керівник економічного розділу к.е.н., доц. _____ Пілова Д.П.
(підпис)

Завдання прийняв на виконання: _____ Гаржа М. С.
(підпис)

Дата видачі завдання: *«__» _____ 2021 р.*
Термін подання дипломної роботи до ЕК: *« __ » _____ 2022 р.*

РЕФЕРАТ

Пояснювальна записка: 101 с., 15 рис., 5 табл., 3 додатків, 31 джерело.

Об’єкт розробки: методи захисту інформації в каналах зв’язку мікропроцесорної системи диспетчерської централізації «КАСКАД».

Предмет розробки: впровадження методів захисту інформації в каналах зв’язку мікропроцесорної системи диспетчерської централізації «КАСКАД».

Мета дипломного проекту: забезпечення заданого рівня безпеки інформації ,що передається по незахищеним каналам зв’язку в мікропроцесорній системи диспетчерської централізації «КАСКАД» за рахунок розробки криптографічного протоколу захисту інформації при передачі.

В технічному завданні визначені підстави та призначення розробки, вимоги до результатів виконання роботи, вихідні показники, стадії і етапи розробки.

В спеціальній частині розроблено рекомендації стосовно впровадження технічних заходів забезпечення заданого рівня захищеності каналів передачі інформації в частині системи, виконаний порівняльний аналіз та на базі нього обґрунтований вибір необхідних програмних засобів захисту каналів передачі інформації в мікропроцесорній системи диспетчерської централізації «КАСКАД».

В економічному розділі виконаний розрахунок витрат на запровадження даних засобів захисту до мікропроцесорної системи диспетчерської централізації «КАСКАД».

Практичним значенням даного дипломного проекту є забезпечення заданого рівня захищеності каналів передачі інформації з метою забезпечення стабільної роботи системи диспетчерської централізації об’єкту критичної інфраструктури.

ОБ’ЄКТ КРИТИЧНОЇ ІНФРАСТРУКТУРИ, АНАЛІЗ АВТОМАТИЗОВАНОЇ СИСТЕМИ КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ, КРИПТОГРАФІЧНИЙ ПРОТОКОЛ, АЛГОРИТМ ШИФРУВАННЯ, КОНТРОЛЬ ЦІЛІСНОСТІ ІНФОРМАЦІЇ.

THE ABSTRACT

Explanatory note: 101 pages, 15 pictures, 5 tables, 3 appendices, 31 sources.

The object of research: information protection methods in communication channels of dispatch centralization system "KASKAD".

The subject of research: introduction of methods of information protection in the communication channels of the microprocessor system of the dispatch centralization "KASKAD ".

Purpose of graduate project: the aim of this diploma project is to provide the necessary level of information security in microprocessor dispatch centralization system "KASKAD.

In the specification were determined: foundations and the purpose of development, requirements to the process executions results, initial indexes, development steps and stages.

In special part was worked out recommendations regarding introduction of information protection methods to provide required level of information security in a part of system, made a comparative analysis of necessary hardware and security software which are must be used with business.

In the economic part was calculated the costs for securities and economic effectiveness of putting them into operation.

In the part “Labor protection” was considered necessary labor safety events for workers and calculated server grounding.

The practical significance of this graduate work is to provide the necessary security of communication channels for stable functioning of dispatch centralization system of critical infrastructure object.

CRITICAL INFRASTRUCTURE OBJECT, ANALYSIS OF THE AUTOMATED CONTROL SYSTEM OF TECHNOLOGICAL PROCESSES, CRYPTOGRAPHIC PROTOCOL, ENCRYPTION ALGORITHM, CONTROL OF INTEGRITY OF INFORMATION.

РЕФЕРАТ

Пояснительная записка: 101 с., 15 рис., 5 табл., 3 приложений, 31 источник.

Объект разработки: методы защиты информации в каналах связи микропроцессорной системы диспетчерской централизации «КАСКАД».

Предмет разработки: внедрение методов защиты информации в каналах связи микропроцессорной системы диспетчерской централизации «КАСКАД».

Цель дипломного проекта: обеспечение заданного уровня безопасности информации, передаваемой по незащищенным каналам связи в микропроцессорной системе диспетчерской централизации «КАСКАД» за счет разработки криптографического протокола защиты информации при передаче.

В техническом задании определены основание и назначение разработки, требования к результатам выполнения работы, исходные показатели, стадии и этапы разработки.

В специальной части разработаны рекомендации по внедрению технических мер обеспечения заданного уровня защищенности каналов передачи информации в части системы, выполнен сравнительный анализ и на базе него обоснован выбор необходимых программных средств защиты каналов передачи информации в микропроцессорной системе диспетчерской централизации «КАСКАД».

В экономическом разделе выполнен расчет затрат на внедрение данных средств защиты в микропроцессорную систему диспетчерской централизации «КАСКАД».

Практическое значение данного дипломного проекта - обеспечение заданного уровня защищенности каналов передачи информации с целью обеспечения стабильной работы системы диспетчерской централизации объекта критической инфраструктуры.

ОБЪЕКТ КРИТИЧЕСКОЙ ИНФРАСТРУКТУРЫ, АНАЛИЗ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ, КРИПТОГРАФИЧЕСКИЙ ПРОТОКОЛ, АЛГОРИТМ ШИФРОВАНИЯ, КОНТРОЛЬ ЦЕЛОСТНОСТИ ИНФОРМАЦИИ.

СПИСОК УМОВНИХ СКОРОЧЕНЬ

АСК ТП – автоматизована система керування технологічними процесами;

ДБЖ – джерело безперервного живлення;

ЕОМ – електронно обчислювальна машина;

СЦБ – сигналізація, централізація та блокування;

МСДЦ – мікропроцесорна система диспетчерської централізації;

АРМ – автоматизоване робоче місце;

РСК – розподілені системи керування;

ЛП – лінійний пункт;

ЦП – центральний пункт;

ІБ – інформаційна безпека;

ІТ – інформаційні технології;

НСД – несанкціонований доступ;

ОС – операційна система;

ПЗ – програмне забезпечення;

ЕЦП – електронний цифровий підпис;

ЗМІСТ

С.

ВСТУП ... 9

1 СТАН ПИТАННЯ. ПОСТАНОВКА ЗАДАЧІ ... 12

1.1 Огляд автоматизованих систем керування технологічними процесами 12

1.2 Загальна характеристика мікропроцесорної системи диспетчерської централізації «КАСКАД» ... 20

1.2.1 Ключові фактори властивостей керування АСК ТП ... 23

1.2.2 Програмно-апаратний комплекс центрального пункту «ЦП КАСКАД» ... 29

1.2.3 Програмно-апаратний комплекс лінійного пункту «ЛП КАСКАД» .. 34

1.4 Висновок та постановка задачі ... 39

2 СПЕЦІАЛЬНА ЧАСТИНА ... 42

2.1 Вимоги до розробки ... 42

2.2 Базові криптографічні системи ... 43

2.2.1 Симетричне шифрування ... 43

2.2.2 Асиметричне шифрування ... 46

2.2.3 Криптографічні протоколи ... 51

2.2.4 Класифікація криптографічних протоколів ... 52

2.3 Розробка криптографічних протоколів ... 56

2.3.1 Обґрунтування вибору алгоритму шифрування даних ... 60

2.3.1.1 Алгоритм AES ... 60

2.3.1.2 Алгоритм ДСТУ ГОСТ 28147:2009 ... 64

2.3.1.3 Алгоритм ДСТУ 7624:2014 «Калина» ... 68

2.3.2 Обґрунтування вибору методу контролю цілісності ... 73

2.3.3 Хешування ... 73

2.3.4 Електронний цифровий підпис 78

2.3.5 Обґрунтування вибору методу захисту від дублювань команд..... 83

2.3.6 Обґрунтування вибору методів аутентифікації відправника та отримувача..... 84

2.4 Висновок 87

3 ЕКОНОМІЧНИЙ РОЗДІЛ 89

3.1 Визначення трудомісткості розробки криптографічного протоколу ... 91

3.2 Розрахунок витрат на впровадження криптографічного протоколу 92

3.3 Розрахунок збитків через відсутність систем захисту каналів передачі інформації. 94

3.4 Показники економічної ефективності 95

3.5 Висновок 96

ВИСНОВКИ 97

ПЕРЕЛІК ПОСИЛАНЬ 98

ДОДАТОК А. Відомість матеріалів дипломного проекту

ДОДАТОК Б. Перелік документів на оптичному носії

ДОДАТОК В. Відгуки керівників розділів

ВСТУП

Сучасний стан суспільства передбачає повсюдний перехід до цифрової обробки даних. Таким чином інформація стала більш цінною ніж самі матеріальні ресурси. Інформація стала засобом впливу на позиції на ринку, і саме тому потрібно ставитись до неї відповідним чином й з точки зору безпеки.

Зі збільшенням кількості автоматизованих систем керування технологічними процеси збільшується й кількість відповідних правопорушень. Тому, використання подобного роду систем на підприємстві, а особливо на об'єктах критичної інфраструктури, вимагає дотримання вимог інформаційної безпеки, а саме:

- гарантування безперервності та коректність функціонування автоматизованої системи;
- забезпечення захисту інформації, що передається;
- забезпечення цілісності інформації, що передається;
- забезпечення доступності інформації, що передається;
- забезпечення захисту від технічних збоїв обладнання.

Забезпечення захисту інформації, що є критичною для підприємства та держави, завжди було та буде одним з найважливіших аспектів в сфері безпеки інформації. Особлива увага загострюється в цьому напрямі у зв'язку зі змінами у законодавстві в останні роки. Та у зв'язку з тим, що автоматизація технологічних процесі вносить певну кількість недоліків у безпеки роботи подібного роду систем.

Згідно статті першої Закону України «Про критичну інфраструктуру»[8] захист критичної інфраструктури - всі види діяльності, що виконуються перед або під час створення, функціонування, відновлення і реорганізації об'єкта критичної інфраструктури, спрямовані на своєчасне виявлення, запобігання і нейтралізацію загроз безпеці об'єктів критичної інфраструктури, а також мінімізацію та ліквідацію наслідків у разі їх реалізації.

Згідно статті 5 Закону України «Про критичну інфраструктуру»[8] метою державної політики у сфері захисту критичної інфраструктури є забезпечення безпеки об'єктів критичної інфраструктури, запобігання проявам

несанкціонованого втручання в їх функціонування, прогнозування та запобігання кризовим ситуаціям на об'єктах критичної інфраструктури.

Закон регулює державну політику, що стосується функціонування, захисту та відновлення об'єктів критичної інфраструктури, визначає вимоги щодо визнання або не визнання об'єкту – об'єктом критичної інфраструктури.

Також слід зазначити, що хоча й закон регулює критерії за якими підприємства можуть відноситися до об'єктів критичної інфраструктури, він не встановлює чіткий перелік цих об'єктів. Це призводить до необхідності визначати подібний статус за додатковими критеріями та унеможливлює своєчасний захист по всім пунктам зазначеним у Законі України «Про критичну інфраструктуру».

Великі темпи автоматизації технологічних процесів стосуються також і об'єктів критичної інфраструктури. Адже керування процесами на підприємстві, що сильно розгалужене з географічної точки зору – достатньо складно та дорого. А в разі якщо ці процеси не є максимально автоматизованими це не тільки звеличує їх складність у декілька разів, але й звеличує ймовірність помилок та відказів у роботі таких систем через людські помилки.

Таким чином, технологічні процеси у багатьох галузях були автоматизовані повністю або частково. І саме від рівня автоматизації цих процесів залежить кількість вразливостей, які автоматизація процесів створює. Адже автоматичне керування процесами, що знаходяться на великій відстані один від одного унеможливлює повний контроль за виконанням усіх дій. А довгі лінії передачі даних автоматично становляться потенційними об'єктами втручання сторонніх осіб.

В разі коли зв'язок в системі автоматизованого керування процесами налагоджений по виділених лініях зв'язку – це робить неможливим втручання віддаленим чином, адже система є ізольованою від зовнішнього світу і керується виключно в локальній мережі, це все стосується не тільки керування, але й встановлення оновлень програмного забезпечення в систему. В найкращому випадку – програмне забезпечення буде встановлене разом із технічним обладнанням, що не дає можливості втрутитися у процес встановлення програмного забезпечення, або його оновлення.

І тому, найчастіше, саме канали зв'язку стають найслабкішим місцем системи та місцем для потенційних втручань зловмисників. А у випадку, коли незважаючи на те що інформація передається виділеними лініями зв'язку, не забезпечується захист переданої інформації – загроза втручання у такий процес стає досить ймовірною.

Важливо зауважити, що усі зусилля та витрати, що були зроблені для створення системи диспетчерської централізації – направлені на створення максимально стійкої системи, що забезпечить стабільну роботу незважаючи на більшість зовнішніх факторів. Таким чином, на чільне місце було поставлене забезпечення безпеки перевезень, а ніяк не безпеки інформації, що передається. І саме це створює вразливості в системі передачі інформації, через які можуть бути реалізовані загрози направлені на порушення цілісності та доступності інформації, що передається по цим каналам.

Забезпечення безпеки інформаційних ресурсів підприємства є комплексною задачею. Управління інформаційною безпекою вимагає проведення різнопланових заходів: починаючи з інструктажу персоналу, фізичної охорони приміщень і закінчуючи, системою резервування, проведенням планових перевірок.

Основна робота спеціаліста з захисту інформації – знизити кількість потенційних загроз в області інформаційної безпеки, а саме: розкрити суть проблеми, конкретизувати дестабілізуючі фактори і представити основні методи, здатні підвищити захищеність автоматизованої системи.

1 СТАН ПИТАННЯ. ПОСТАНОВКА ЗАДАЧІ

1.1 Огляд автоматизованих систем керування технологічними процесами

АСК ТП знаходяться в багатьох галузях промисловості, таких як енергетична, водопостачання, нафтогазова, хімічна, фармацевтична, целюлозно-паперова, харчова промисловість, транспорт, а також дискретне виробництво (наприклад, автомобілебудування, аерокосмічні та довготривалі товари).

Автоматизована система керування технологічними процесами (АСК ТП) – це загальний термін, що охоплює окремі типи систем контролю, включаючи системи супервізорного контролю та збору даних (SCADA-системи), розподілені системи керування (РСК) та інші конфігурації систем керування, такі як програмовані логічні контролери (ПЛК), які часто застосовуються в промислових секторах та критичних інфраструктурах[1]. Відповідно до державного стандарту України (ДСТУ) АСК ТП - це людино-машинна система, що забезпечує автоматизований збір інформації з первинних (ПП) або передавальних (ПрП) перетворювачів сигналів і її первинну обробку (фільтрування сигналів, лінеаризація характеристик ПП і ПрП, перетворення сигналів у фізичні одиниці виміру) для розрахунку, видачі та реалізації керуючих впливів на об'єкт керування відповідно до прийнятих критеріїв керування. АСК ТП здійснює реалізацію впливів на об'єкт керування в темпі з технологічним процесом, тобто в реальному часі, при цьому забезпечує керування об'єктом в цілому, а її технічні засоби беруть участь у виробленні рішень з керування. Зазначеними обставинами АСК ТП якісно відрізняється від традиційних систем автоматичного керування (САК) , які представляють технічні засоби для автоматизації дій людини на окремих ділянках технологічного процесу. АСК ТП складається з комбінації елементів контролю (наприклад, електричних, механічних, гідравлічних, пневматичних), що діють спільно для досягнення промислової мети (наприклад, виробництво, транспортування матерії чи енергії). Частина системи, в основному пов'язана з виробництвом продукції, називається процесом. Керуюча частина системи включає задання бажаного виходу або продуктивності. Керування може бути повністю автоматизованим або може включати людину в цикл. Системи можуть бути налаштовані для роботи у режимі відкритого циклу, замкнутого циклу та ручному режимі.

АСК ТП використовуються для управління географічно розподіленими активами, часто розподіленими на тисячі квадратних кілометрів, включаючи розподільчі системи, такі як розподіл води та системи збору стічних вод, сільськогосподарські системи зрошення, нафтогазові трубопроводи, електроенергетичні мережі та системи залізничного транспорту[1].

Хоча системи керування, що використовуються в галузях виробництва та розподілу, дуже схожі, вони відрізняються в деяких аспектах. Виробничі галузі, як правило, знаходяться в обмеженій фабриці або районі, поблизу електростанції у порівнянні з територіально розподіленими галузями розподілу. Зв'язки у виробничій промисловості зазвичай виконуються за допомогою технологій локальної мережі (LAN), які, як правило, більш надійні та швидкі у порівнянні з дальніми широкосмуговими мережами (WAN) та бездротовими/РЧ (радіочастотними) технологіями, що використовуються галузями розподілу. АСК ТП, що використовуються в галузях розподілу, призначені для вирішення проблем, пов'язаних з міжміським зв'язком, такими як затримки та втрата даних, спричинені використанням різних засобів масової інформації. Залежно від типу мереж, елементи керування безпекою можуть відрізнятися.

З розвитком технологій автоматизовані системи керування технологічними процесами поступово перетворилися із закритих керуючих пристроїв на багаторівневі промислові мережі на базі стандартних мережевих протоколів, які мають безліч подібностей з корпоративними мережами, що активно використовуються. На жаль, це стосується й уразливостей, які тісно пов'язані із загрозами кібербезпеці. Ці мережі схильні до зараження шкідливими програмами, злому, виведення з ладу ПЗ та інших видів зовнішнього впливу. Це істотно впливає на виробничі процеси, і з кожним роком кількість подібних інцидентів збільшується.

Фахівці з інформаційної безпеки виділяють кілька типів порушників для АСК:

1) ворожі держави та їхні силові структури,
2) терористичні організації,
3) промислові шпигуни та представники організованих злочинних груп,
4) хакери-активісти.

Дії кіберпідрозділів силових структур ворожих держав спрямовані на порушення функціонування об'єктів інфраструктури, що може призвести до руйнувань та людських жертв. Даний вид атак є одним з найнебезпечніших при кібервійнах. Фахівці вважають, що найближчими роками він буде найпоширенішим. Людство вже бачило його застосування на практиці: між 2009 та 2011 роками відбувалася так звана операція Stuxnet, під час якої американські спецслужби за допомогою комп'ютерної програми-хробака порушували роботу іранських заводів зі збагачення уранового палива.

Кібератака на енергетичні компанії України - перша зареєстрована успішна кібератака на енергетичну систему з виведенням її із ладу сталась 23 грудня 2015 року. Російським зловмисникам вдалось успішно атакувати комп'ютерні системи управління трьох енергопостачальних компаній України.

Найбільше від першої кібератаки постраждали споживачі «Прикарпаттяобленерго»: було вимкнено близько 30 підстанцій, близько 230 тисяч мешканців залишались без світла протягом однієї-шести годин. Атака відбувалась із використанням троянської програми BlackEnergy.

Нападники змогли отримати доступ до корпоративної мережі компанії завдяки вдалому зараженню комп'ютера одного із співробітників трояном BlackEnergy третьої версії. Протягом багатьох місяців вони проводили масштабну розвідку, досліджували та описували мережі і здобували доступ до служби Windows Domain Controllers, яка керує обліковими записами користувачів мереж. Тут вони зібрали реквізити співробітників, у тому числі паролі від захищеної мережі VPN, яку працівники енергокомпаній використовували для віддаленого підключення до мережі SCADA[4].

Здобувши доступ до внутрішньої мережі нападники переконфігурували пристрої безперебійного живлення (UPS), відповідальні за енергопостачання двох диспетчерських центрів.

Кожне обленерго використовувало власну систему управління розподілом енергії у своїх мережах, і під час фази розвідки нападники ретельно вивчили кожну з них. Тоді вони написали шкідливий мікрокод, яким замінили справжній вбудований мікрокод на конвертерах із серійного інтерфейсу на інтерфейс Ethernet

у понад десяти підстанціях (ці конвертери застосовуються для обробки команд, які надходять від мережі SCADA до систем управління підстанцією)[4].

Виведення з ладу конвертерів не давало операторам змогу посилати віддалені команди для повторного включення запобіжників після того, як енергію було вимкнено.

Основні методи атак на АСК ТП:

1) Найпоширеніший спосіб - безпосереднє відправлення команд обладнання. Більшість програмного забезпечення, що використовується в промисловості, не вимагає ідентифікації користувача. Зловмисникам достатньо проникнути в технологічну мережу підприємства та встановити зв'язок із потрібним об'єктом.
2) За допомогою спеціальних утиліт злочинці можуть перехопити керуючий екран. У цьому випадку всі рухи курсору, що здійснюються ними, будуть видно оператору, а самі зловмисники будуть обмежені правами активного користувача.
3) У деяких випадках внесення правок до бази даних дозволяє також змінити та пов'язані з нею об'єкти. Технологія Man-in-the-Middle використовується у випадках, коли зловмисникам відомі параметри функціонування мережі. Тоді вони можуть змінювати інформацію на головному екрані та отримати повний контроль над будь-якою системою.
4) Зламування датчиків виробництва та подача на них неправильних даних для перешкоджання коректній роботі обладнання.

Компанія Dragos, що спеціалізується на кібербезпеці промислової сфери, представила аналітичні дані за 2019 рік у регулярному звіті «ICS VULNERABILITIES – YEAR IN REVIEW» – досліджувалися вразливості в промислових системах (ICS-CERT), не враховуючи супутнє обладнання (мережеве, системне та системне)[3].

Основні висновки дослідження:

1) 77% оцінених уразливостей вимагають суттєвого доступу до мережі управління для експлуатації та вважаються «глибоко внутрішніми»,

2) 9% проблем відносяться до рішень, що межують і не пов'язані безпосередньо з АСК ТП, - використовуються для початкового доступу зловмисника до операцій з контролерами,
3) 26% відомих уразливостей не мали виправлень (patch) на момент аналізу,
4) 30% рекомендацій були опубліковані з некоректними даними, що заважають операторам точно визначати пріоритети в процесі планування виправлень (завдяки роботі компанії Dragos було доопрацьовано 212 із 438 некоректних описів та рекомендацій),
5) 40% уразливостей відносяться до робочих станцій та ПЗ взаємодії з користувачем, які для експлуатації потребують підключення до Інтернету, що в цій галузі часто неприпустимо[3].

За даними Kaspersky ICS CERT, у перші шість місяців 2020 року частка атакованих комп'ютерів зросла порівняно з попереднім півріччям із 38% до майже 40% у системах автоматизації будівель та з 36,3% до 37,8% в АСУ ТП нафтогазової галузі(рис. 1.1). До останніх відносять сервери управління та збору даних (SCADA), сервери зберігання даних, шлюзи даних, стаціонарні робочі станції інженерів та операторів, мобільні робочі станції інженерів та операторів, комп'ютери, що використовуються для адміністрування технологічних мереж, та комп'ютери, що використовуються для розробки ПЗ для систем промислової автоматизації[2].

Таким чином, можна побачити, що зараз стає дуже важливим питання захисту АСК ТП у різних сферах діяльності людини, тому що атак на подібні об’єкти стає все більше рік від року, а розвиток систем захисту, методичних вказівок щодо забезпечення саме кіберзахисту таких систем йде помітно повільніше. Тим самим створюючи загрози безпеці таких підприємств, та не тільки нанесення матеріальних збитків, але й загрози життям багатьох людей, що неприйнятне за будь-яких обставин.

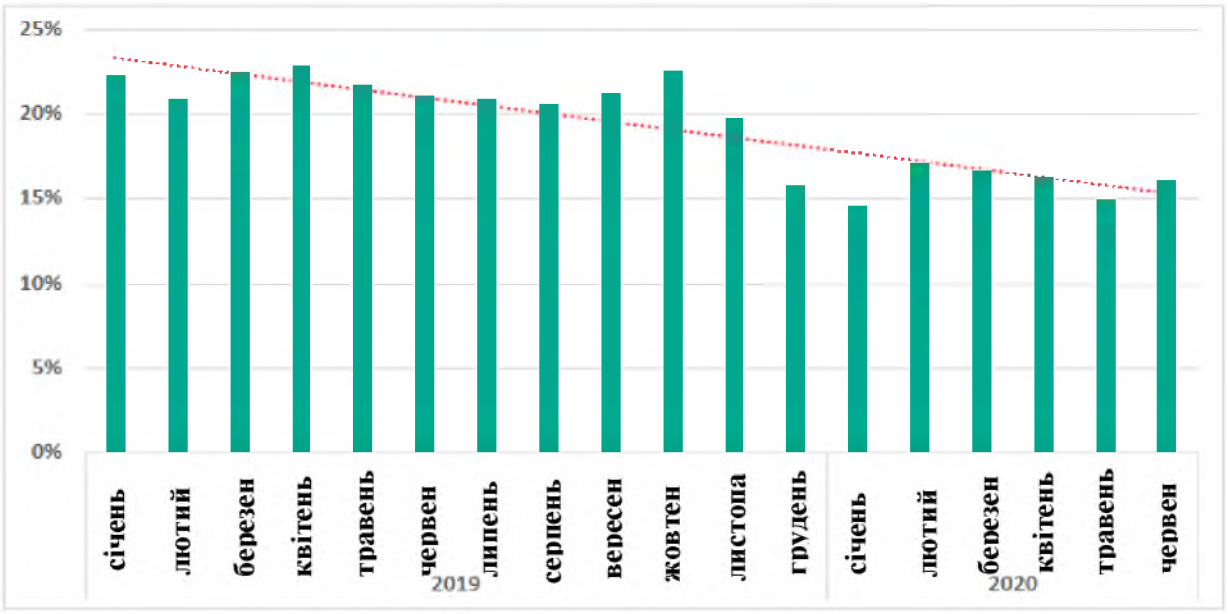


Рисунок 1.1 Частка атакованих комп'ютерів

В даній роботі розглядається АСК ТП Придніпровської залізної дороги. Дана АСК ТП є досить розгалуженою та покриває різні сфери життєдіяльності залізничного транспорту. Вона включає у себе збір інформації з різноманітних датчиків, розташованих по всій території залізничних шляхів, агрегація зібраної інформації на спеціальних серверах, що можуть забезпечити цілодобовий доступ до неї, передача забраних даних у різні точки залізниці з метою використання вже обробленої інформації на місцях(рис. 1.2).

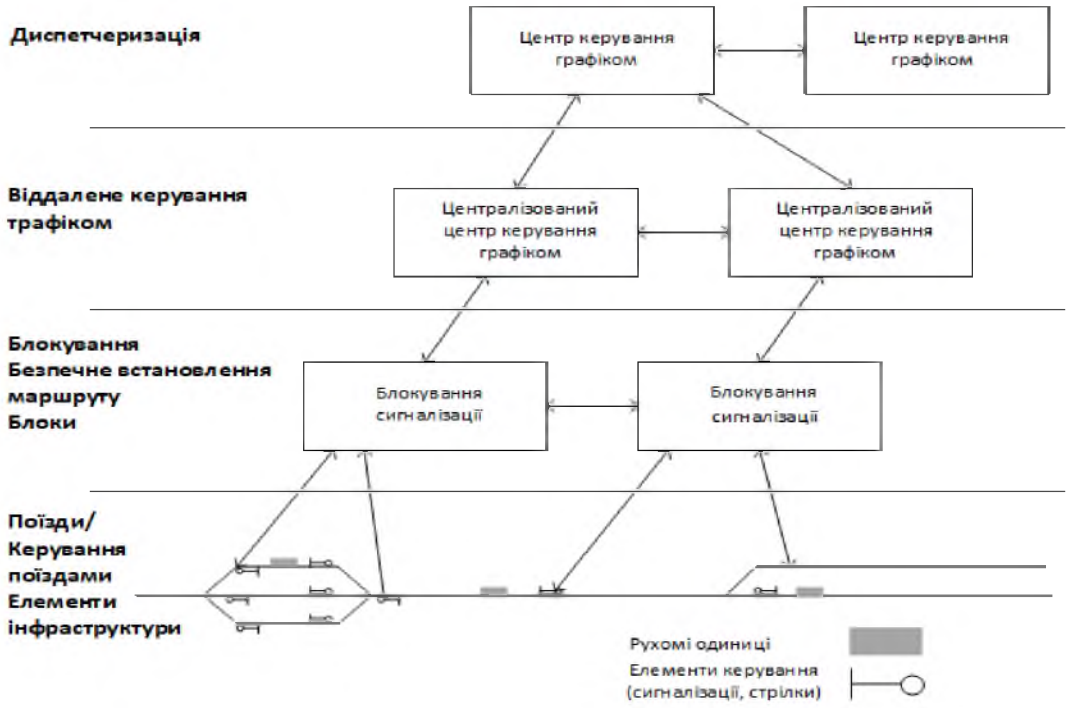


Рисунок 1.2 Ієрархічна структура системи диспетчерського керування рухом поїздів[6]

Інформація збирається по кожній зі станцій, що підпорядковуються Придніпровській залізниці, після збору інформації безпосередньо на самій станції, вона відправляється до інформаційно-обчислювального центру Придніпровської залізниці, де ця інформація безпосередньо збережується для подальшої агрегації, збереження в обробленому вигляді та можливості передачі цієї інформації до інших філій залізної дороги України. Інформація стосується не тільки функціонування Придніпровської залізниці, але й є важливою для інших філій України, тому що для влагодженої та безперебійної роботи всіх філій залізниці в Україні потрібна постійна комунікація їх один із одною.

Також потрібно відмітити, що не тільки дії, що стосуються пересування потягів, вагонів та вантажів, мають місце у роботі АСК ТП залізниці, є проміжною ланкою між управлінням залізниці в Києві та усіма підрозділами своєї філії. Територіальна розгалуженість залізниці зобов'язує її мати стабільний зв'язок з усіма окремими пунктами, що їй підпорядковуються, та забезпечувати безпеку при передачі та зберіганні інформації, що надається до розглядання.

Останнім часом в нашій державі були зроблені певні кроки в бік забезпечення безпеки об'єктів критичної інфраструктури. Був розроблений ряд документів, щодо

визначення, які саме об’єкти можуть підпадати під термін об’єкта критичної інфраструктури, та перелік вимог щодо процедури визначення підприємства, що є об’єктом критичної інфраструктури.

Нажаль, незважаючи на вже зроблені нашою державою кроки у цьому напрямі, все ще є моменти, які потребують доопрацювання. Одним з таких моментів є відсутність у нашій державі переліку об’єктів критичної інфраструктури. Тобто, незважаючи на те, що в нас є Закон України «Про критичну інфраструктуру», цей закон не визначає, які саме об’єкти підпадають під цей закон. Таким чином, неможливо сказати, без додаткового аналізу підприємства та даного закону, чи є те чи інше підприємство об’єктом критичної інфраструктури.

Згідно статті 8 Закону України «Про критичну інфраструктуру», до об'єктів критичної інфраструктури відносяться підприємства, установи, організації незалежно від форми власності, які:

1) провадять діяльність та надають послуги в галузях енергетики, хімічної промисловості, оборонно-промислового комплексу, транспорту, інформаційно-комунікаційних технологій, електронних комунікацій, у банківському та фінансовому секторах;
2) надають послуги у сферах життєзабезпечення населення, зокрема у сферах централізованого водопостачання, централізованого водовідведення, постачання теплової енергії, гарячої води, електричної енергії і газу, виробництва харчових продуктів, охорони здоров'я;
3) включені до переліку підприємств, що мають стратегічне значення для економіки і безпеки держави;
4) підлягають охороні та обороні в умовах надзвичайного стану і особливого періоду;
5) є об'єктами підвищеної небезпеки;
6) є об'єктами, які мають загальнодержавне значення, розгалужені зв'язки та значний вплив на іншу інфраструктуру;
7) є об'єктами, порушення функціонування яких призведе до кризової ситуації регіонального значення[8].

Отже, згідно визначення об'єктів критичної інфраструктури, можна зробити висновок, що АСК ТП Придніпровської залізниці відноситься до об'єктів критичної інфраструктури України, тому що:

1) Придніпровська залізна дорога провадить діяльність та надає послуги транспорту.
2) Згідно постанови Кабінету Міністрів України від 04.03.2015 р. №83[7] є об'єктом державної власності, що має стратегічне значення для економіки і безпеки держави.
3) Підлягає обороні в умовах надзвичайного стану і особливого періоду
4) Має розгалужені зв'язки , так як розміщена на великій території країни та має значний вплив на інші інфраструктури.
5) Порушення роботи залізниці може призвести до кризової ситуації не тільки регіонального значення, але й державного значення в цілому.

Тому усі подальші міркування та запропоновані рішення будуть відноситися до АСК ТП Придніпровської залізниці, як до об'єкта критичної інфраструктури, та до такого, який потребує підвищеної уваги до безпеки інформації.

1.2 Загальна характеристика мікропроцесорної системи диспетчерської централізації «КАСКАД»

Згідно з інформацією, наданою Головним управління автоматики, телемеханіки та зв'язку, при загальній довжині залізничних ліній Укрзалізниці 22301 км пристроями диспетчерської централізації (ДЦ) обладнано 13491,6 км, диспетчерського контролю (ДК) – 10258,4 км. Більшість з них - системи застарілих типів, але вже з 90-х років у зв'язку з подальшим розвитком засобів обчислювальної техніки, значним зменшенням їх вартості почались інтенсивні роботи з розроблення вітчизняних систем мікропроцесорної диспетчерської централізації (МПДЦ), мікропроцесорного диспетчерського контролю (МПДК), мікропроцесорних систем кодового управління (МСКУ).

Перший досвід експлуатації МСДЦ "КАСКАД" показав, що система в достатній мірі технологічна і повністю задовольняє потреби робітників господарства перевезень. Крім цього, за свідченнями персоналу дистанцій сигналізації та зв'язку, система надійна і практично не потребує обслуговування. У

наш час системі “КАСКАД” немає альтернативи по впровадженню на Укрзалізниці[5].

Програмно-апаратний комплекс мікропроцесорної системи диспетчерської централізації МСДЦ “КАСКАД” впроваджується на дільницях залізничного транспорту з метою підвищення ефективності управління вантажними та пасажирськими перевезеннями за рахунок:

1) автоматизації процесів збору та надання інформації про поїзне положення на регіоні управління;
2) телеуправління пристроями електричної централізації лінійних станцій в автоматичному та напівавтоматичному режимах;
3) підсилення контролю за станом об’єктів управління на підставі автоматично сформованої діагностичної інформації в реальному масштабі часу;
4) автоматизації та максимального спрощення операцій по управлінню рухом поїздів;
5) підвищення безпеки руху;
6) зменшення впливу суб’єктивного фактора при прийнятті рішень;
7) надання інформації користувачам різних рівнів та служб через локальну та глобальну мережі зв’язку;
8) використання сучасних графічних інтерфейсів, єдиного інформаційного простору, оперативного об’єднання або роз’єднання диспетчерських дільниць[5].

Система "КАСКАД" побудована за модульним принципом, максимально уніфікована, розроблена з використанням сучасних технологій та мікроелектронних виробів, що дозволяє досягти найвищих показників надійності. Програмне забезпечення має високий ступінь супроводження та максимальну незалежність від апаратної платформи.

Структура МСДЦ “КАСКАД” побудована згідно з діючою ієрархічною системою управління перевезеннями і включає 3 рівні(рис. 1.3):

1) Головне управління перевезень (ЦД);
2) диспетчерські центри управління залізниць (ДНЦ);
3) лінійні підприємства (ЛП);

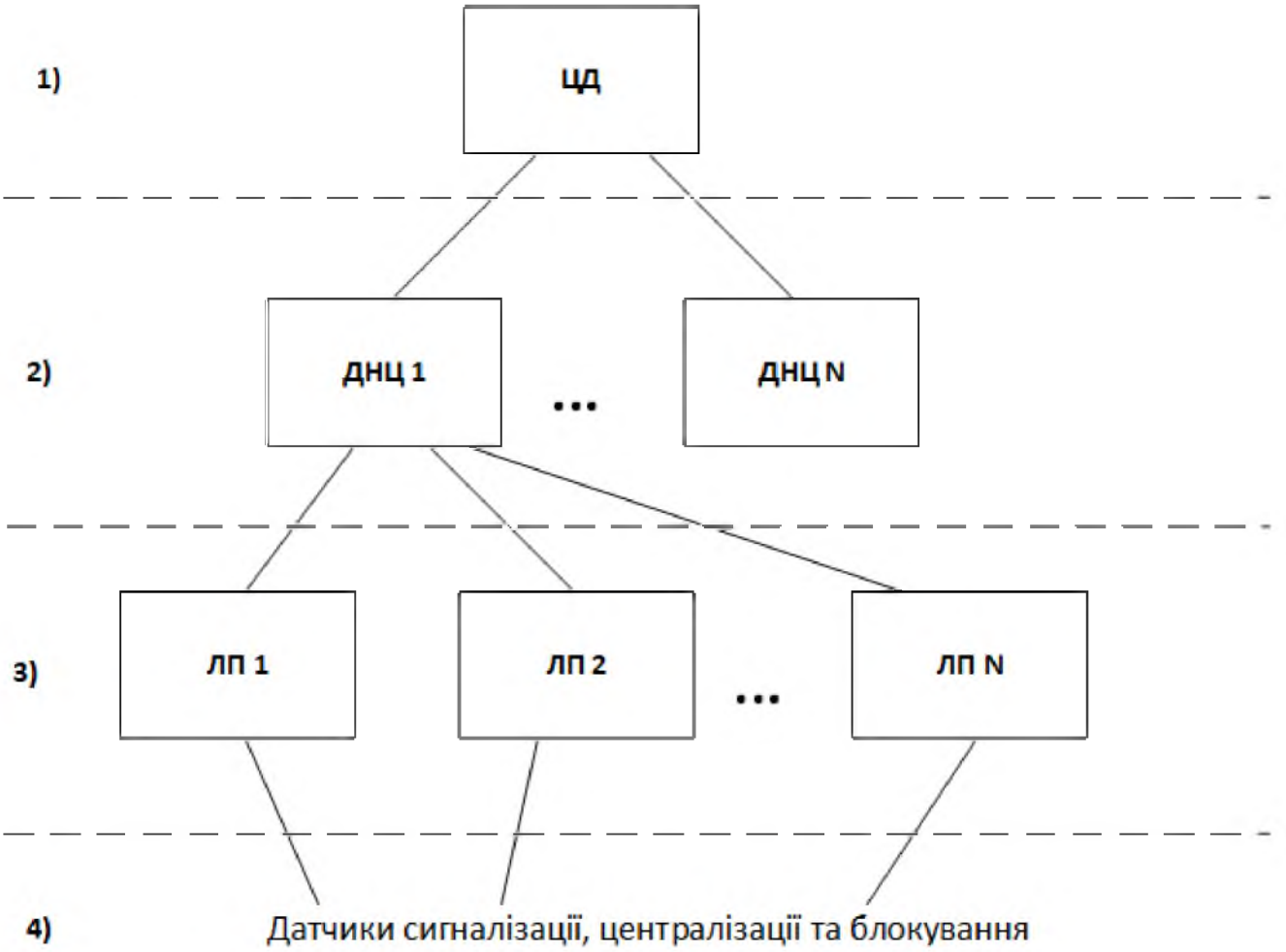

1)
ЦД
2)
ДНЦ 1
...
ДНЦ N
3)
ЛП 1
ЛП 2
...
ЛП N
4)
Датчики сигналізації, централізації та блокування

Рисунок 1.3 Ієрархічна структура системи управління перевезеннями

На першому рівні використовується стандартне програмне забезпечення, яке входить у пакет постачання операційної системи – Internet Explorer, Netscape Navigator, або інші. Другий рівень обладнаний комплексами центрального поста диспетчерської централізації "ЦП КАСКАД", а третій – комплексами лінійного пункту "ЛП КАСКАД". Лінійні пункти "ЛП КАСКАД" об'єднані з центром управління "ЦП КАСКАД" за допомогою кільцевих мереж зв'язку лінійних пунктів, які використовують виділені канали або фізичні лінії магістрального кабелю.

Комплекси "ЛП КАСКАД" географічно розташовуються на розподільних пунктах дільниці (станціях, блок-постах, роз'їздах) і складаються з мікропроцесорного контролера і комплекту уніфікованих модулів у відповідності зі складністю об'єкта автоматизації.

1.2.1 Ключові фактори властивостей керування АСК ТП

Ключові фактори, що керують проектними рішеннями щодо властивостей керування, зв'язку, надійності та резервування АСК ТП. Оскільки ці фактори сильно впливають на проектування АСК ТП, вони також допоможуть визначити потреби в системі безпеки.

Часові вимоги до керування. Процеси АСК ТП мають широкий спектр вимог до часу, включаючи дуже високу швидкість, узгодженість, регулярність та синхронізацію. Люди, можливо, не зможуть надійно та послідовно відповідати цим вимогам; Автоматичні контролери можуть бути необхідними[1].

Топологія локальної мережі кільцевого типу використовується в системі МСДЦ "КАСКАД". Комплекси "ЛП КАСКАД", які розташовані на постах ЕЦ залізничних станцій, є клієнтами локальної мережі кільцевого типу "LPnet".На фізичному рівні локальної мережі "LPnet" використовуються дві пари магістрального кабелю (1,05 мм) при відстані між сусідніми клієнтами мережі до 40-45 км. Максимальна чисельність станцій, об'єднаних одним кільцем локальної

мережі “LPnet”, розраховується на допустимий термін транспортування інформації, у найгіршому випадку до 6 секунд. Затримка транспортування інформації через “ЛП КАСКАД” складається з таких складових:

1) програмне забезпечення “ЛП КАСКАД” при формуванні пакета, Ts =0,02 с;
2) модем (Round Trip) “ЛП КАСКАД”, Trp=0,04-0,06 с;
3) передача інформаційного пакета по швидкості обміну, Tv с.

Tv – розрахунковий параметр залежить від швидкості обміну по мережі, розміру інформаційного пакета, ступеня захисту інформації по ймовірності трансформації сигналів ТУ, ТС.

Для реальних умов експлуатації локальних мереж зв’язку наведений у прикладі термін затримки транспортування інформації необхідно помножити на коефіцієнт надійності (Кн=1,5-2,0). Згідно до вимог транспортування інформації, що прописана у нормативних документах залізної дороги, термін транспортування не повинен перевищувати 6 с.

Справно діюча локальна мережа кільцевого типу дозволяє скоротити термін транспортування інформації вдвічі (3,75-5 с) та підвищити надійність мережі. Локальна мережа “LPnet” при застосуванні пристроїв доступу – аналогових модемів – відповідає вимогам по включенню у виділені канали зв’язку з аналоговим завершенням. Включення виділених каналів зв’язку в розрив кільця може відбуватися в будь-якому місці.

Географічний розподіл. Системи мають різний ступінь розподілу, починаючи від невеликої системи (наприклад, локального процесу керування ПЛК) до великих, розподілених систем (наприклад, нафтопроводів, електромереж). Більший розподіл зазвичай передбачає потребу в широкій області (наприклад, орендовані лінії, перемикання каналів та перемикання пакетів) та мобільного зв'язку[1].

Дільниця диспетчерського управління, у залежності від географічного розміщення станцій та перегонів, кількості об’єктів управління, системи організації зв’язку, може складатися з одного або декількох сегментів. Характеристики комплексу, максимальні значення щодо обладнання дільниці, а саме: кількість сегментів, станцій, блок-постів, блок-дільниць на перегоні, та загальну чисельність об’єктів автоматизації наведено в табл. 1.1.

Таблиця 1.1 Характеристики МСДЦ “КАСКАД”[5]

Найменування характеристики	Чисельне значення
Розмір сегмента диспетчерської дільниці	10 – 300 км
Кількість сегментів дільниці	1 – 10
Кількість станцій, роз’їздів, блок-постів сегмента дільниці	1 - 15
Загальна чисельність станцій, роз’їздів, блок-постів дільниці	до 150

Ієрархія. Супервізорне керування використовується для забезпечення центрального розташування, яке може об'єднувати дані з кількох місць, щоб підтримувати керуючі рішення на основі поточного стану системи. Часто ієрархічний/централізований контроль використовується для надання операторам повного уявлення про всю систему[1].

Комплекс “ЦП КАСКАД” розташовується безпосередньо в центрі управління перевезеннями залізниці і складається з робочих станцій, автоматизованих робочих місць диспетчерського персоналу, об’єднаних локальною мережею, сервера, комунікаційного обладнання.

Сервер дільниці виконує функції: збору, обробки та збереження інформації про стан об’єктів на дільниці, що контролюється, в режимі реального часу; передачі команд керування до об’єктів телеуправління; реєстрації та перевірки дій диспетчера; автоматичного керування схрещенням, обгоном і пропуском поїздів на заданих станціях; контролю доступу з керування дільницею; діагностики роботи комплексу.

Автоматизовані робочі місця взаємодіють через локальну мережу з сервером дільниці, який в свою чергу взаємо-діє з комп’ютерами лінійних станцій.

Керуюча складність. Часто функції керування можуть виконуватися простими контролерами та попередньо встановленими алгоритмами. Проте, більш складні

системи (наприклад, управління рухом поїздів) вимагають від операторів людини забезпечення того, щоб всі контрольні дії відповідали більшим цілям системи[1].

Система потребує роботи кваліфікованого персоналу при управлінні перевезеннями, при чому, направленість їх дій повинна бути різною за для забезпечення максимально ефективного управління перевезеннями та безпеки руху поїздів. До складу програмно-апаратного комплексу “ЦП КАСКАД”, розташованого в центрі управління перевезеннями, належать:

1) автоматизоване робоче (АРМ) місце поїзного диспетчера;
2) автоматизоване робоче місце енергодиспетчера;
3) автоматизоване робоче місце інженера СЦБ і зв’язку;

Таким чином потрібна злагоджена робота усіх робітників пункту, оперативна реакція на будь-які стандартні та нестандартні події, що можуть скластися у процесі контролю за перевезеннями. Вчасно прийняті міри по вирішенню екстрених ситуацій можуть не тільки значно зменшити збитки від самої події, але й врятувати людські життя, що беззаперечно є однією з найважливіших цілей.

Доступність. Потреби системи (тобто надійність) є важливим фак-тором проектування. Системи із сильними вимогами щодо доступності та часу роботи можуть вимагати додаткової резервної або альтернативної реалізації у всіх комунікаціях та керуванні[1].

Для забезпечення високої надійності та функціонування системи в різних режимах резервування в складі “ЛП КАСКАД” передбачено дві локальних міжмодульних мережі. Основний та резервний комплекти мають свою незалежну шину, джерело живлення, основну і резервну мережу. У свою чергу доступ до модулів (основного і резервного) може відбуватись з обох мереж. У разі пошкодження однієї з мереж або модуля, система продовжує функціонувати, при цьому діагностика стану пристроїв реєструє відповідну несправність.

Модулі живлення мають системи електронного захисту від перенапруг, перевищення допустимого струму та температур з двох напрямків: від первинного джерела та навантаження. У свою чергу кожен з модулів має свою систему захисту щодо струму та напруги. Для захисту по струму використовуються запобіжники “PolySwitch Ressettable Fuses”, які при перевищенні допустимої межі струму

відключають модуль від живлення та відновлюють свої властивості при зникненні перевантаження. Захист по напрузі забезпечується паралельною схемою силового напівпровідникового приладу “Transil diodes”[5].

У такій системі багаторівневого захисту пошкодження одного з модулів по мережі живлення не призводить до відключення живлення основного або резервного комплектів. Комплекс “ЛП КАСКАД” конструктивно побудований згідно з рекомендаціями міжнародного стандарту IEC 297 (DIN 41 494)[9].

Вплив збоїв**.** Невиконання керуючої функції може призвести до суттєво різного впливу для доменів. Системи, що мають більший вплив, часто потребують можливості продовжувати роботу за допомогою надмірного керування або здатності працювати в деградованому стані. Проектування має відповідати цим вимогам[1].

Для забезпечення роботи під час збоїв система має декілька різних варіантів розвитку подій. Перш за все слід відзначити, що система має резервні сервера для всіх своїх компонентів, а саме Web-сервер, сервер бази даних, резервні системи живлення та системи безперебійного живлення.

Усі вище перераховані компоненти встановлені задля забезпечення якомога більш стабільної роботи системи, недопущення збоїв, надання можливості реагувати оператору та мати час, як мінімум на мінімізацію можливих втрат, а як максимум - на усунення причин, які можуть вплинути на подальшу роботу як самої системи так і ділянки залізної дороги в цілому.

Також слід зазначити, що незважаючи на те, що керування системою проходить в автоматичному режимі, є ще один можливий варіант розвитку подій, коли усі автоматичні системи виходять з ладу та унеможливлюють автоматизоване керування транспортними процесами. На цей випадок система може переходити у режим ручного управління, усі дії щодо керування пересуванням залізничного транспорту можуть проводитись безпосередньо на самому залізничному покритті, через те, що лінії зв’язку диспетчерських станцій з інформаційно-обчислювальним центром, з яким проходить обмін інформацією щодо пересування потягів, також можуть постраждати, кожна із станцій має виділений канал зв’язку, який з’єднує її безпосередньо із інформаційно-обчислювальним центром.

Таким чином, система має надійні резервні сервери на випадок виникнення збоїв у роботі основного обладнання, та також не виключає сценаріїв, коли робота будь-якого із автоматизованих елементів управління системи(основних серверів, або резервних серверів) переривається повністю, або частково, тоді система може переходити у повністю ручне управління та продовжувати функціонувати таким чином упродовж усього періоду, який потрібен для усунення проблем.

Безпека. Область вимог безпеки системи також є важливим чинником проектування. Системи повинні мати можливість виявляти небезпечні умови та викликати дії, спрямовані на зменшення небезпечних умов до безпечних. У більшості критично важливих операцій надзвичайне керування та керування над потенційно небезпечним процесом є важливою частиною системи безпеки[1].

Вимоги до безпеки інформації в системі диспетчерської централізації та в системах такого типу в цілому, ставляться до властивостей інформації, які стосуються її доступності та частково цілісності. Усі вимоги, що ставляться до таких систем в Україні, стосуються забезпечення безпеки руху поїздів, створення та використання систем, які б могли забезпечувати ці вимоги є найголовнішим пріоритетом. Найважливішим завданням побудови оптимальної структури керування є вибір функціональної організації системи з подальшим поділом її на більш дрібні елементи з їхнім раціональним групуванням залежно від використовуваних методів і засобів для вирішення конкретного завдання. Другим завданням, що вимагає вирішення після вибору функціональної структури, є вибір набору функцій та алгоритмів їхньої реалізації у кожній з підсистем для отримання мінімуму витрат (коштів, часу) на досягнення цілей системи.

Оперативне диспетчерське керування призначене для реалізації руху запланованих у графіку руху поїздів(ГРП) з метою забезпечення їхньої безпеки, економічності і точності виконання ГРП. Безпека досягається виконанням встановлених норм, іноді навіть на шкоду іншим вимогам.

Таким чином, проаналізувавши усе згадане вище, можна зробити висновок, що системи такого типу в Україні створюються та використовуються з урахуванням норм безпеки пересування поїздів, але ніяким чином не йде мова про

забезпечення безпеки інформації, що передається в системах диспетчерської централізації.

Як вже було наведено, існують вимоги до доступності інформації, максимальних затримок при передачі, за яких інформація вважається актуальною, вимоги до відмовостойкості, вимоги до підтвердження самої доставки інформації, але ніде не йде мова про забезпечення конфіденційності та цілісності інформації. Немає вимог щодо цих властивостей інформації, немає реалізації забезпечення цих вимог як зі сторони програмного забезпечення так і технічного забезпечення.

Система диспетчерської централізації не передбачає шифрування інформації при передачі по каналах зв'язку, не передбачає перевірку цілісності інформації, щоб впевнитись в тому, що вхідна інформація не була ніяким чином модифікована при передачі по каналах зв'язку.

1.2.2 Програмно-апаратний комплекс центрального пункту «ЦП КАСКАД»

Зважаючи на вимоги до систем управління автоматизованими процесами, та аналіз виконання таких вимог в МСДЦ «КАСКАД», можна побачити, що в системі вже забезпечується виконання всіх вимог, окрім тих, що стосуються безпеки передачі інформації. Хоча й уся інформація передається по виділених каналах зв'язку, ця інформація не є ніяким чином захищеною. А довжина самих каналів передачі даних може сягати 20 км, що унеможливлює контроль за лінією упродовж усієї протяжності.

Комплекс "ЦП КАСКАД" розташовується безпосередньо в центрі управління перевезеннями залізниці(рис. 1.4) і складається з робочих станцій, автоматизованих робочих місць диспетчерського персоналу, об'єднаних локальною мережею, сервера, комунікаційного обладнання.

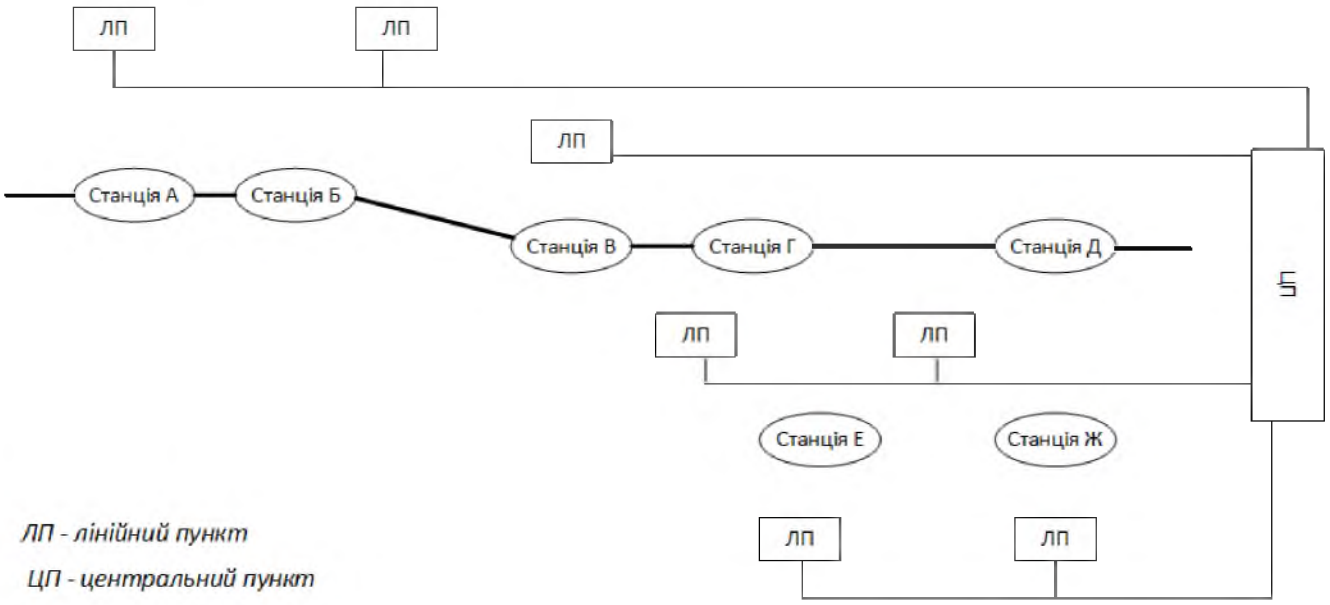


Рис. 1.4 Структурна схема локальної мережі зв'язку МСДЦ «КАСКАД»

Комплекс центрального поста "ЦП КАСКАД" об'єднується з комплексами лінійних пунктів "ЛП КАСКАД" через кільцеву локальну мережу зв'язку (рис.1.4) в єдину комп'ютеризовану систему централізованого управління.

Програмно-апаратний комплекс "ЦП КАСКАД" виконує функції обробки, збереження, формування, захисту інформації, людино-машинного інтерфейсу, підтримки глобальних та локальних мереж зв'язку.

До складу програмно-апаратного комплексу "ЦП КАСКАД", розташованого в центрі управління перевезеннями (рис. 1.5), належать:

1) автоматизоване робоче (АРМ) місце поїзного диспетчера;
2) автоматизоване робоче місце енергодиспетчера;
3) автоматизоване робоче місце інженера СЦБ і зв'язку;
4) локальна мережа АРМ;
5) резервований сервер бази даних з робочим місцем системного адміністратора;
6) каналоутворювальна апаратура зв'язку;
7) джерела безперебійного живлення;
8) системне та прикладне програмне забезпечення.

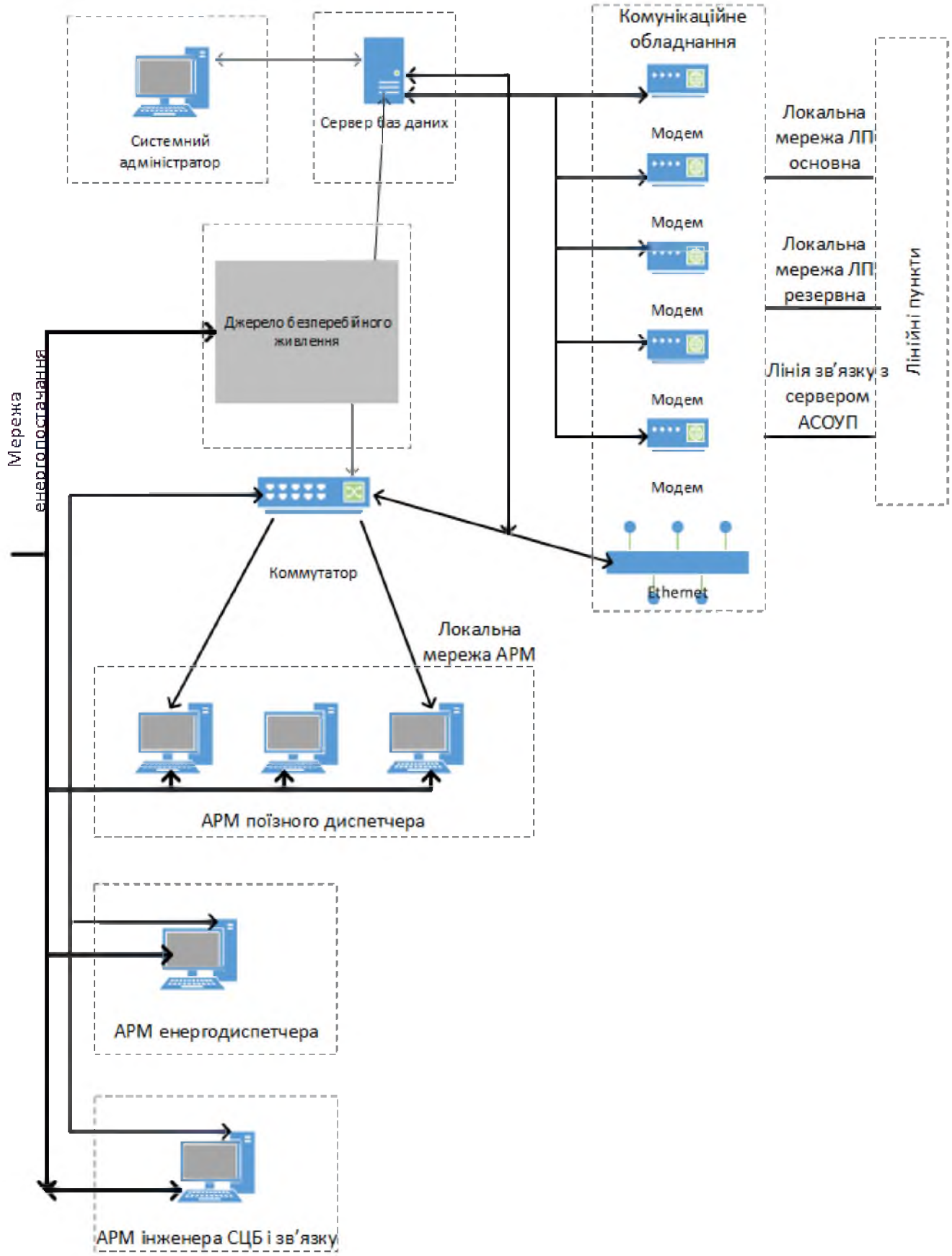


Рис. 1.5 Структурна схема програмно-апаратного комплексу «ЦП КАСКАД»

Програмно-апаратний комплекс АРМ поїзного диспетчера забезпечує контроль перевізного процесу, який здійснюється на основі інформації, отриманої від пристроїв СЦБ

Основні функції які забезпечує АРМ ДНЦ при управлінні перевізним процесом:

1) детальне відображення поїзної ситуації та стану об'єктів контролю на ділянці;
2) управління об'єктами СЦБ (пряме, з програмним стеженням, за заданими або накопиченими маршрутами, прогнозне управління);
3) автоматичну реєстрацію проходження поїзда дільницею;
4) ідентифікацію рухомої одиниці;
5) автоматичне управління схрещенням, обгоном та пропуском поїздів на заданих станціях;
6) інтерпретацію процесу проходження поїздів на ділянці у вигляді графіка виконаного руху в реальному режимі часу;
7) автоматичне формування графіка прогнозного руху;
8) автоматичне ведення системного журналу з реєстрацією сигналів телеуправління, телесигналізації, діагностики та дій поїзного диспетчера;
9) відображення за минулі періоди часу (до 30 діб) поїзної ситуації та стану об'єктів контролю на ділянці у вигляді “фільму”;
10) взаємодія із системою АСОУП.

Інформація відображається на екранах кольорових моніторів у режимі реального часу та за заданий період. Розподіл інформації на кольорових моніторах виконується за умовами оптимального управління поїзним диспетчером перевізного процесу. Кількість моніторів вирішується на етапі проектування в залежності від кількості та складності станцій на дільниці, мінімально – три.

Автоматизоване робоче місце енергодиспетчера забезпечує контроль стану та управління пристроями енергопостачання на дільниці, кольорове відображення стану пристроїв енергопостачання у вигляді мнемосхем з позначенням номера (назви) та стану об’єктів контролю, телеуправління пристроями енергопостачання, автоматичне ведення системного журналу з реєстрацією наказів телеуправління,

стану об'єктів телесигналізації, діагностики та дій енергодиспетчера за масштабом mчасу, відображення діагностичної інформації[5].

Автоматизоване робоче місце інженера СЦБ і зв'язку забезпечує контроль стану пристроїв СЦБ та інших шляхом відображення перевізного процесу в режимі реального часу, минулих подій і необхідної довідкової інформації.

Програмне забезпечення комплексу складається з операційної системи (ОС) та комплексу прикладного програмного забезпечення (ПЗ), орієнтованого на виконання завдань управління процесом перевезення. В якості операційної системи в МСДЦ «КАСКАД» використовується ОС «Linux».

Прикладне програмне забезпечення "ЦП КАСКАД" побудоване за модульним принципом, максимально уніфіковане, розроблене з використанням сучасних інструментальних засобів, забезпечує високий рівень супроводження і максимальну незалежність від апаратної платформи[5].

Програмне забезпечення складається з декількох серверів необхідних для забезпечення нормального функціонування комплексу:

1) сервер віддалених станцій;
2) сервер дільниці керування;
3) сервер бази даних;
4) WEB сервер користувачів глобальної мережі, АРМ поїзного диспетчера, енергодиспетчера, інженера СЦБ та інше.

У складі "ЦП КАСКАД" задіяні декілька локальних мереж, які відрізняються функціональним призначенням, топологією побудови, швидкостями обміну, віддаленістю абонентів локальних мереж:

1) локальна мережа «LPnet» призначена для підключення комплексів «ЛП КАСКАД» залізничних станцій полігону;
2) локальна мережа «Ethernet» призначена для підключення автоматизованих робочих місць диспетчерського персоналу ДНЦ, ДНЦЕ, та інших користувачів;
3) глобальна мережа «CPnet» призначена для об'єднання декількох комплексів «ЦП КАСКАД» в єдину інформаційно-обчислювальну систему вищого рівня.

Для підтримки перерахованих вище мереж використовуються модеми різного призначення та специфікації:

1) Аналогові модеми, що мають дводротове або чотиридротове завершення. Дані модеми підключаються до виділених дводротових фізичних ліній зв'язку протяжністю до 40 км, комутованих каналів зв'язку АТС та виділених ліній зв'язку для чотиридротових модемів. Вони забезпечують підтримку протоколів обміну інформацією в різних топологіях мережі «LPnet».
2) Цифрові модеми, що використовують технології SDSL для швидкісного обміну по виділеній дводротовій фізичній лінії зв'язку протяжністю до 4 км. Використовуються для мережних віддалених підключень АРМ диспетчерського персоналу, а також при включенні в глобальну мережу «CPnet».
3) Комутатор «Ethernet» є багатопортовим пристроєм канального рівня, відповідає вимогам специфікації IEEE 802.3 (10/100-Mbit Ethernet), забезпечує мережні підключення АРМ диспетчерського персоналу або інших користувачів на відстані до 100 м, має 24 порти, топологію зіркоподібного типу.

1.2.3 Програмно-апаратний комплекс лінійного пункту «ЛП КАСКАД»

Програмно-апаратні комплекси «ЦП КАСКАД» та «ЛП КАСКАД» є тісно взаємопов'язаними один із одним, від роботи кожного з них залежить функціонування МСДЦ «КАСКАД» в цілому, тому що без налагодженої роботи модулів «ЛП КАСКАД» неможливо правильно та своєчасно збирати дані з усіх пристроїв сигналізації та блокування на ділянках залізниці, а без «ЦП КАСКАД» неможливо ці дані збирати та надавати поїзним диспетчерам для подальшої роботи із ними.

Програмно-апаратний комплекс "ЛП КАСКАД" розроблений у відповідності до вимог, що пред'являються до систем промислового призначення високої надійності, забезпечує безперервний режим функціонування в умовах постів електричної централізації, резервування, діагностування обладнання. До складу комплексу "ЛП КАСКАД" входять уніфіковані модулі, кількість та їх типи

визначаються на етапі технічного проекту в залежності від складності об'єкта автоматизації (стрілки, сигнали, колії, переїзди та інше)[5].

«ЛП КАСКАД» складається з таких основних модулів, що є уніфікованими:

1) модуль телесигналізації («КАСКАД-ТС»);
2) модуль телеуправління («КАСКАД-ТУ»);
3) модуль телеуправління відповідальний («КАСКАД-ТВ»);
4) модуль модема («КАСКАД-ММ»);
5) модуль мікропроцесорного контролера («КАСКАД-МП»);
6) модуль вторинного живлення («КАСКАД-ВЖ»);
7) модуль електронного крейту («КАСКАД-КР»).

Подальше розширення функціональних можливостей комплексу потребує розроблення нових типів уніфікованих модулів.

Функціонально модулі поділяються на три категорії:

1) модулі взаємодії з пристроями СЦБ (введення/виведення): «КАСКАД-ТС», «КАСКАД-ТУ», «КАСКАД-ТВ»;
2) загальносистемні модулі: «КАСКАД-МП», «КАСКАД-ММ»;
3) модулі живлення та електронного крейту: «КАСКАД-ВЖ», «КАСКАД-КР».

Кожен з модулів першої категорії взаємодіє з мікропроцесорним контролером через міжмодульну послідовну локальну мережу. Локальна міжмодульна мережа забезпечує зв'язок між модулями взаємодії з пристроями СЦБ та модулем контролера міжмодульної мережі, який у свою чергу через системну шину взаємодіє з мікропроцесорним контролером.

Таким чином, можна побачити, що усі модулі є взаємопов'язаними в комплексі «ЛП КАСКАД», а основним елементом взаємодії виступає мікропроцесорний контролер, що займається збором та відправкою усіх даних, що він отримує від пристроїв СЦБ. Даний процесор, разом із модемами для передачі даних, створює ядро системи, яке буде відповідальним за усі дії .

Для забезпечення високої надійності та функціонування системи в різних режимах резервування в складі "ЛП КАСКАД" передбачено дві локальних міжмодульних мережі. Основний та резервний комплекти мають свою незалежну шину, джерело живлення, основну і резервну мережу. У свою чергу доступ до

модулів (основного і резервного) може відбуватись з обох мереж. У разі пошкодження однієї з мереж або модуля, система продовжує функціонувати, при цьому діагностика стану пристроїв реєструє відповідну несправність[5].

Функціональна схема програмно-апаратного комплексу «ЛП КАСКАД» підтримує такі основні режими:

1) двоканальної системи з незалежними каналами проходження інформації, при цьому інформація дійсна тільки у випадку, коли вона співпадає по двох каналах (два з двох). Підтвердженням проходження інформації є наявність сигналів зворотного зв'язку від кінцевих модулів та об'єктів управління;
2) двоканальної системи з незалежними каналами проходження інформації, при цьому інформація дійсна тільки у випадку, коли вона співпадає по двох каналах (два з двох). При наявності відповідного наказу система в місці виникнення пошкодження використовує обхідні шляхи для продовження дії пристроїв, але ступінь захисту в окремих випадках може знижуватись;
3) одноканальної системи з "гарячим" резервуванням всіх складових комплексу. У разі пошкодження в робочому каналі та при наявності відповідного наказу система переходить на резервний канал, а система діагностики дозволяє виявити та зафіксувати місце пошкодження в основному каналі (через резервний) до рівня модуля;
4) одноканальної системи з "гарячим" резервуванням частини складових комплексу: модулів телеуправління, телесигналізації або інших. У разі пошкодження модулів введення/виведення в схемах узгодження з пристроями СЦБ та при наявності відповідного наказу система переходить на резервні модулі, а система діагностики дозволяє виявити та зафіксувати місце пошкодження до рівня модуля;
5) одноканальної системи без резервування складових комплексу. У разі пошкодження система частково може зберігати дію, але потребує невідкладного ремонту. Система діагностики дозволяє виявити та зафіксувати місце пошкодження до рівня модуля тільки у випадках справної дії основних модулів (модема, мікропроцесорного контролера та лінії зв'язку).

У складі комплексу «ЛП КАСКАД» програмно-апаратними засобами забезпечується зовнішній інтерфейс RS485. Цей інтерфейс використовується для включення в систему МСДЦ «КАСКАД» системи диспетчерського контролю пристроїв на перегоні «ДК КАСКАД». Спрощена структурна схема комплексу «ЛП КАСКАД» наведена на рис. 1.6.

Для забезпечення обміну даними через кільцеву локальну мережу зв'язку лінійних пунктів і центрального поста використовується модуль модема «КАСКАД-ММ.2602». Фізичний рівень модуля модема забезпечено стандартними протоколами обміну, швидкість зв'язку може динамічно змінюватися в залежності від характеристик каналу.

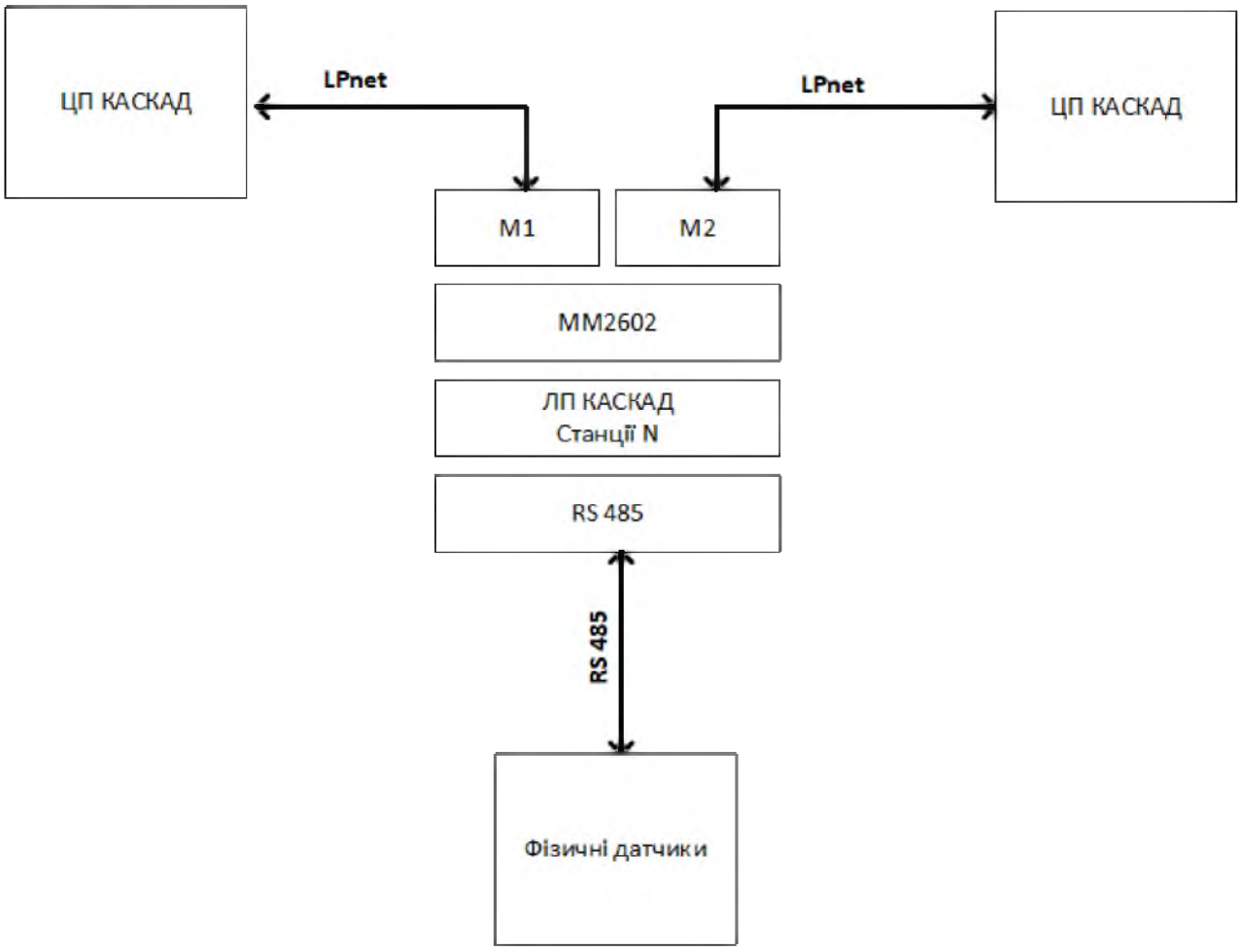


Рисунок 1.6 Спрощена структурна схема комплексу «ЛП КАСКАД»

Модуль забезпечує:

1) зв'язок по виділених фізичних парах у дводротовій схемі підключення;
2) підтримку міжстанційного телефонного зв'язку;

3) передачу голосових повідомлень з метою оповіщення пасажирів та інше.

Як провідний процесорний модуль у складі системи використовується модуль контролера «КАСКАД-МП.2616». Він забезпечує функції взаємодії з модулями комплексу, підтримує протоколи мереж зв'язку лінійних пунктів, забезпечує синхронізацію процесів з сусіднім каналом системи, перевіряє достовірність інформації в каналах обміну, підтримує протоколи локальної міжмодульної мережі, протоколи інформаційного обміну по послідовних портах та обміну з пристроями на перегоні (контроль перегріву букс, диспетчерський контроль та інше). Крім цього, модуль контролера забезпечує внутрішню діагностику та резервування.

Модуль контролера побудовано на процесорі ZFx86, що має тактову частоту 66 МГц. Пам'ять модуля складається з SDRAM на 16 Мбайт, Flash EPROM – 0,512Мбайт. Модуль комплектується твердотільним диском (DiskOnChip) об'ємом від 2 до 64Мбайт. Обмін інформацією між модулями забезпечується системною шиною ISA96 та за послідовним каналом RS232C. Напруга живлення модуля – 5V, споживання струму – до 0,7А.

По управлінню та контролю обладнання лінійних пунктів «ЛП КАСКАД» підключається до входів системи нижнього рівня – пристроїв станційної автоматики. У свою чергу, управління в системах електричної централізації здійснюється шляхом подачі команд керування на її входи, якими вважаються елементи управління на пульті. Такий підхід не порушує алгоритм роботи пристроїв СЦБ і забезпечує вимоги з безпечного функціонування. Фактично МСДЦ імітує дії чергового по станції у процесі встановлення маршруту. При цьому реалізація вимог, пов'язаних з особливостями завдання маршруту, управління стрілками та реалізацією команд управління, забезпечується на програмному рівні. За рахунок цього відпадає потреба у великій кількості допоміжних реле, які раніше встановлювалися для узгодження з системою диспетчерської централізації[5].

Реалізація деяких керівних функцій системи пов'язана з забезпеченням безпеки. Це виконання відповідних команд, до яких належать:

1) аварійна зміна напрямку руху на перегоні;
2) штучне розмикання секцій.

Реалізація наведених команд потребує видачі сигналу керування безпосередньо у схеми виконавчої групи електричної централізації. У зв'язку з цим до схем узгодження висунуті надзвичайно високі вимоги щодо забезпечення безпеки функціонування.

1.3 Висновок та постановка задачі

Таким чином, інформація, що циркулює у системі, відноситься безпосередньо до керування рухом поїздів. Потік інформації формується та передається з верхнього та нижнього рівнів структури комплексу, де верхнім рівнем виступає головне управління перевезень, що формує графіки пересувань поїздів, а нижнім рівнем виступають лінійні пункти, що здійснюють збір інформації з датчиків розташованих безпосередньо на самій залізниці, систем сигналізації, централізації та блокування. А у середині структури знаходиться центр диспетчерської централізації, який водночас виступає як концентратором даних, що отримуються від головного управління перевезеннями, для подальшого опрацювання, так і виконує функції аналізу даних, що надходять з нижніх структурних рівнів комплексу.

І, зважаючи на те, що основними вимогами до комплексу МСДЦ «КАСКАД» є безпека управління перевезеннями, на даний момент часу, не забезпечується безпека передачі інформації між рівнями комплексу, тобто передача інформації від лінійних пунктів, до центральних пунктів не захищається ніяким чином. Вся інформація передається у відкритому вигляді, ніяким чином не шифруючись, також немає систем перевірки цілісності інформації, що передається.

Проте, незважаючи на те, що вимоги системи, щодо цілісності інформації, що передається, не виконуються належним чином, система має достатній рівень захисту від технічних збоїв. Завдяки тому, що існують резервні лінії передачі даних на всіх рівнях комплексу, починаючи від датчиків, та закінчуючи головним управлінням перевезень. Також, окрім резервних каналів передачі даних, система обладнана резервними модулями живлення, на випадок збоїв у мережі енергопостачання, і може працювати у автономному режимі впродовж декількох годин.

Отже, проаналізувавши усе вище перераховане, стає зрозумілим, що частина системи «КАСКАД», а саме, канали передачі інформації між центральними та лінійними пунктами, є вразливою до атак типу man-in-the-middle, коли інформація може бути модифікованою у процесі її передачі незахищеними каналами зв'язку. Таким чином, загрозу представляють зловмисники, які можуть модифікувати будь-яку інформацію, що передається між лінійними пунктами «КАСКАД» та центральним пунктом «КАСКАД», завдяки не тільки відсутності перевірки цілісності в системі, але й тому, що велика протяжність фізичних ліній зв'язку, не дозволяє повністю контролювати їх на предмет втручання сторонніх осіб.

Інший вид загрози також пов'язаний з існуючими каналами зв'язку та засобами передачі інформації через них. Загрозу представляють зловмисники, що можуть втрутитись у роботу системи передачі інформації та надавати некоректні дані з систем сигналізації, централізації та зв'язку, таким чином змушуючи працівників диспетчеризації реагувати на стан датчиків певним чином, що може спричинити великих збитків. Ефект від такого роду атак буду кумулятивним, адже проблеми з контролем навіть однієї ділянки призведуть до ланцюгової реакції, що торкнеться усіх поїздів, прямуючих через неї, та тих що мали проїхати тим же напрямом.

Проаналізувавши структуру каналів передачі даних та методи їх передачі, стає видно, що в системі є вразливість пов'язана з відсутністю ідентифікації переданих даних. Таким чином є загроза дублювання команд/сповіщень, в разі їх перехоплення сторонніми особами. Відсутність чітких ідентифікаторів кожної з команд, що передаються незахищеними каналами дозволяє зловмисникам збирати передані та згодом імітувати відправку команд у великих обсягах, що призведе до некоректної роботи системи диспетчерської централізації.

Визначивши основні загрози, та вразливості системи передачі даних комплексу МСДЦ «КАСКАД», можна прийти до висновку, що система потребує впровадження засобів контролю цілісності та доступності інформації, що допоможуть усунути вище перераховані вразливості системи, та вберегти від реалізації загроз будь-якими зловмисниками.

2. СПЕЦІАЛЬНА ЧАСТИНА

2.1 Вимоги до розробки

Задля забезпечення виконання вимог безпеки інформації, що стосуються її цілісності та доступності, в частині системі потрібні механізми з їх реалізації. Такими механізмами можуть стати методи криптографічного захисту інформації, що забезпечують передачу інформації, без можливості як отримати несанкціонований доступ, так і контроль цілісності при передачі. Отже інформація буде передана у вигляді, в якому вона була відправлена та захищена від сторонніх осіб. Також потрібно взяти до уваги той факт, що для забезпечення цілісності інформації потрібна передача мітки, щоб бути впевненими в тому, що саме та інформація, що була відправлена, дійшла до кінцевого адресата та для усунення можливості дублювання команд, що надсилаються. Таким чином будуть

реалізовані механізми захисту каналів передачі інформації, що забезпечать функціонування системи без збоїв

Перш ніж визначатися з конкретними рішення криптозахисту каналів зв'язку, та методами передачі особливих міток, потрібно взяти до уваги характеристики системи, що розглядається.

Збереження основного функціоналу є однією з найголовніших задач, при впровадженні нових різновидів програмного забезпечення комплексу, або змін у діюче програмне забезпечення. Тим самим гарантуючи, що усі вимоги щодо забезпечення безпеки перевезення будуть виконані.

Лінійні пункти "КАСКАД" мають обмежені обчислювальні можливості, що робить необхідним звернути увагу не тільки на стійкість обраних рішень, але й на можливість застосування їх у даній конкретній системі.

Криптографічні методи захисту повинні забезпечувати достатній рівень стійкості та бути достатньо швидкими при шифруванні та дешифрування, щоб задовольнити встановлений рівень вимог по максимально допустимому часу передачі повідомлень від лінійних пунктів до центрального пункту, а саме, не перевищувати 6 секунд. Те ж саме стосується механізмів однозначної ідентифікації відправлених повідомлень.

Рішення, що потребують зміну технічного обладнання комплексу "КАСКАД", не можуть бути прийняті до уваги через велику матеріальну вартість, та достатньо великий проміжок часу на переобладнання системи, адже система має тисячі лінійних пунктів встановлених по всій залізниці.

Також слід зазначити, що ще однією з вимог до рішень з захисту інформації, що впроваджується, є відсутність необхідності підвищення рівня кваліфікації персоналу. Система вже є досить складною та розгалуженою, необхідність підвищення складності керування системою призведе до значних матеріальних витрат та, в перспективі, зробить складнішою пошук нового персоналу для керування та обслуговування.

2.2 Базові криптографічні системи

2.2.1 Симетричне шифрування

В загальному випадку криптосистема має наступну структуру(рис.2.1)

Рисунок 2.1 Узагальнена структура криптосистеми

Метод симетричного шифрування, як і випливає з назви, використовує один криптографічний ключ для шифрування і дешифрування даних. Використання одного ключа для обох операцій робить процес простим.

Робота криптосистеми може бути описана наступним чином:

1) З джерела ключів вибирається ключі (шифрування Ke і розшифрування Kd) і відправляються по надійним каналам передаючій і приймаючій стороні.
2) До вихідного (або відкритого) повідомлення, що призначене для передачі, застосовується алгоритм шифрування, внаслідок чого отримується зашифроване повідомлення.
3) Зашифроване повідомлення пересилається по каналу для обміну повідомленнями, який не вважається надійним (тобто зашифроване повідомлення може бути перехоплене порушником), приймаючій стороні.
4) На приймаючій стороні до зашифрованого повідомлення застосовується обернене перетворення для отримання вихідного повідомлення[10].

Алгоритми шифрування і розшифрування E і D відкриті, і секретність вихідного тексту p в даному шифротекста залежить від таємності ключа.

Найбільш видатною особливістю симетричного шифрування є простота процесу, так як використовується один ключ як для шифрування, так і для

дешифрування. Там, де необхідно зашифрувати великий обсяг даних, симетричне шифрування виявляється відмінним варіантом.

В результаті алгоритми симетричного шифрування:

1) Значно швидше, ніж їх аналоги асиметричного шифрування;
2) Потребують менше обчислювальної потужності;
3) Не знижується швидкість передачі даних.

Існують сотні алгоритмів симетричного типу. Найбільш поширені з них – AES, RC4, DES, 3DES, RC5, RC6 і т. д.

Найбільш популярними та, як наслідок найбільш ефективними є алгоритми: DES, 3DES та AES.

DES-алгоритм симетричного шифрування.

DES (data encryption standard), представлено в 1976 році, є найстарішим симетричним методом шифрування. Розроблений IBM для захисту конфіденційних урядових даних і офіційно прийнятий в 1977 році для використання федеральними агентствами в США. Алгоритм шифрування DES був одним з тих, який використовували в версії 1.0 і 1.1 TLS.

Криптоалгоритм DES реалізує архітектуру, що одержала назву збалансована мережа Файстеля. Відповідно до цієї архітектури весь процес шифрування складається з серії однотипних циклів(раундів)[13]. Даний принцип побудови займав домінуюче місце в криптографії блокових шифрів практично до наших днів. Він лежить в основі шифрів, що широко використовуються, включаючи криптоалгоритм визначений стандартом ДСТУ ГОСТ 28147:2009.

DES перетворює 64-бітні блоки даних відкритого тексту в зашифрований текст шляхом поділу на два окремих 32-бітних блока, застосовуючи процес шифрування до кожного окремо. Включає в себе 16 циклів різних процесів – таких як розширення, перестановка, заміна або інші операції – через які будуть проходити дані в зашифрованому вигляді. В кінцевому підсумку 64-бітові блоки зашифрованого тексту створюються в якості вихідних даних[11].

У 2005 році DES було офіційно оголошено застарілим і замінено алгоритмом шифрування AES. Найбільшим недоліком DES була занадто маленька довжина ключа шифрування, що полегшувало злом.

Алгоритм симетричного шифрування 3DES.

3DES також відомий як TDEA (triple data encryption algorithm), є оновленою версією алгоритму DES. 3DES розроблено для подолання недоліків алгоритму DES і введено в експлуатацію в кінці 1990 року. Оновлений алгоритм застосовував цикли DES тричі до кожного блоку даних. В результаті 3DES було набагато складніше зламати, ніж його попередника DES. Став частиною криптографічних протоколів, таких як TLS, SSH, IPsec і OpenVPN[11].

Уразливість Sweet32 алгоритму 3DES була виявлена Картікеяном Бхаварганом і Гаетаном Леурентом. Це відкриття змусило індустрію безпеки розглянути питання про старіння алгоритму, а Національний інститут стандартів і технологій США оголосив про це офіційно в проекті управління, опублікованому в 2019 році. Згідно з цим проектом, використання 3DES має бути скасовано в усіх нових додатках після 2023 року.

Алгоритм симетричного шифрування AES.

AES (advanced encryption system) також відомий як Rijndael, є одним з найбільш поширених алгоритмів шифрування. Був розроблений в якості альтернативи DES і після затвердження Національним інститутом стандартів і технологій США в 2001 році став новим стандартом шифрування. AES – це сімейство блокових шифрів з різною довжиною ключів і різними розмірами блоків.

AES працює методами підстановки і перестановки. Спочатку незашифровані дані перетворюються в блоки, а потім шифрування застосовується з використанням ключа. Процес шифрування складається з різних процесів, таких як зсуви рядків, змішування стовпців і додавання ключів. Залежно від довжини ключа виконується 10, 12 або 14 таких трансформацій (раундів). Варто відзначити, що останній раунд відрізняється від попередніх і не включає процес змішування[11].

На сьогоднішній день AES є найбільш популярним алгоритмом шифрування, він використовується для забезпечення безпеки в різних сферах, включаючи:

1) безпека процесорів і шифрування файлів;
2) протокол SSL / TLS;
3) безпека Wi-Fi;
4) шифрування мобільних додатків;

5) Virtual Private Network.

2.2.2 Асиметричне шифрування

Асиметричне шифрування, на відміну від симетричного, включає в себе кілька ключів для шифрування і дешифрування даних, які математично пов'язані один з одним. Один з цих ключів відомий як «відкритий ключ», а інший – як «закритий ключ». Асиметричний метод шифрування також відомий як «криптографія з відкритим ключем».

Відкритий ключ і приватні ключі генеруються парами випадковим чином, використовуючи алгоритм, і ключі мають математичну залежність один з одним. Ключ повинен бути більше довжини (128 біт, 256 біт), щоб зробити його міцнішим і унеможливити розрив ключа, навіть якщо відомий інший парний ключ. Дані шифруються за допомогою будь-якого з ключів та розшифровуються разом з іншим[12].

Хоча й асиметричне шифрування може виконувати ту й саму основну функцію, що й симетричне шифрування, а саме – забезпечення конфіденційності даних, ці види шифрувань достатньою відрізняються один від одного(табл.2.1).

Таблиця 2.1 - Різниця між симетричним та асиметричним шифруванням

Симетричне шифрування	Асиметричне шифрування
Використовує один ключ для шифрування та дешифрування.	Використовує два ключі, один для шифрування, а другий для дешифрування.
Зашифровані дані та ключі обмінюються.	Обмінюються лише зашифровані дані, а відкритий ключ доступний для всіх.
Більш швидкий	Повільно
Некерований, якщо ніхто з учасників не стає вищим.	Операції можна впорядкувати за допомогою пар відкритого та приватного ключів.
Ризик при обміні ключа в мережевому каналі.	Приватний ключ не обмінюється.

Існує багато алгоритмів асиметричного шифрування, кожний з яких гарантує достатній рівень надійності шифрування, та може використовуватися задля різних кінцевих цілей, в залежності від алгоритму, що використовується:

1) Ключова угода Діффі-Гелмана – ключ обміну інформацією
2) RSA – шифрування та цифровий підпис
3) ECC (еліптична крива криптографії) - функції схожі на RSA, використовується для обслуговування мобільних пристроїв.
4) Ель Гамель – використовується для цифрових підписів та обміну ключів.
5) DSA (алгоритм цифрового підпису) – використовується для створення цифрового підпису.

Перша перевага цього типу шифрування – безпека, яку він забезпечує. У цьому методі відкритий ключ – який є загальнодоступним – використовується для шифрування даних, в той час як розшифрування даних виконується з використанням закритого ключа, який необхідно надійно зберігати. Це гарантує, що дані залишаються захищеними від атак «людина посередині». Для веб-серверів і серверів електронної пошти, які постійно підключаються до сотень тисяч клієнтів потрібно управляти тільки одним ключем і захищати його. Інший ключовий момент полягає в тому, що криптографія з відкритим ключем дозволяє створювати зашифроване з'єднання без необхідності зустрічатися в автономному режимі, щоб спочатку обмінятися ключами[11].

Друга важлива особливість, яку пропонує асиметричне шифрування, – це аутентифікація. Як ми бачили, дані, зашифровані за допомогою відкритого ключа, можуть бути розшифровані тільки за допомогою закритого ключа, пов'язаного з ним. Отже, він гарантує, що дані бачить і дешифрує тільки той об'єкт, який повинен їх отримати. Простіше кажучи, це підтверджує, що ви розмовляєте або обмінюєтеся інформацією з реальною людиною або організацією.

Асиметричне шифрування може бути застосоване у різних цілях, таких як забезпечення:

1) Конфіденційність

Найбільш поширене застосування асиметричного шифрування - конфіденційність. Це досягається шляхом надсилання критичної інформації, зашифрувавши її відкритим ключем приймача, а одержувач розшифрує її власним приватним ключем.

2) Автентичність за допомогою цифрових підписів

Відправник приєднує свій приватний ключ до повідомлення як цифровий підпис та обмінюється з одержувачем. Одержувач використовує відкритий ключ відправника і перевіряє, чи належить приватний ключ належить відправнику, отже, з'ясовується справжність відправника.

3) Цілісність обміну інформацією

Один із способів хеш даних, що підлягають обміну, створюється та шифрується за допомогою відкритий ключа відправника. Зашифрований хеш і дані обмінюються з приймачем. Використовуючи приватний ключ відправника, одержувач розшифровує хеш, а також відтворює хеш. Будь-яка різниця між двома хешами вказує на зміст вмісту після втрати підпису та цілісності. Цей вид перевірки цілісності дотримується в цифрових операціях з готівкою та криптовалютами.

4) Відмова від авторства

При використанні інструменту шифрування цифрового підпису власник документа або інформації, який обмінявся ним з іншими, не може відмовитись від вмісту, а транзакція, здійснена в мережі, не може бути відхилена його автором.

Розглянемо два основних типи алгоритмів асиметричного шифрування.

1) Алгоритм асиметричного шифрування RSA

Як відомо, на сьогоднішній день RSA є найбільш використовуваним алгоритмом асиметричного шифрування. Його ефективність полягає в методі «первинної факторизації». По суті, обираються два різних випадкових простих числа заданого розміру (наприклад, 1024 біта кожне) і множаться, щоб створити ще одне гігантське число. Завдання полягає в тому, щоб визначити вихідні прості числа з помноженого гігантського. Виявляється, ця головоломка практично неможлива для сучасних суперкомп'ютерів, не кажучи вже про людей[11].

Великою перевагою RSA є його масштабованість, ключі можуть бути різної довжини шифрування: 768-бітний, 1024-бітний, 2048-бітний, 4096-бітний і т. д.

RSA засновано на простому математичному підході, тому його реалізація в інфраструктурі відкритих ключів стає легкою. Адаптивність і безпека зробили RSA найбільш використовуваним алгоритмом асиметричного шифрування для різних додатків, включаючи сертифікати SSL / TLS, криптовалюти та шифрування електронної пошти.

2) Алгоритм асиметричного шифрування ECC

У 1985 році два математика по імені Ніл Кобліц і Віктор Міллер запропонували використовувати еліптичні криві в криптографії. Майже через два десятиліття їх ідея втілилася в реальність, алгоритм ECC (Elliptic Curve Cryptography) почали використовувати в 2004-2005 роках.

У процесі шифрування ECC еліптична крива представляє набір точок, які задовільняють математичне рівняння ($y^2 = x^3 + ax + b$).

Як і RSA, ECC також працює за принципом незворотності. Простіше кажучи, в ECC число, яке символізує точку на кривій, множиться на інше число і дає іншу точку на кривій. Тепер, щоб зламати цю головоломку, ви повинні з'ясувати нову точку на кривій. Математика ECC побудована таким чином, що знайти нову точку практично неможливо, навіть якщо ви знаєте вихідну точку.

Не дивлячись на те, що в порівнянні з RSA, в ECC використовується коротша довжина ключа, він забезпечує більший рівень безпеки.

Ще однією перевагою використання більш коротких ключів в ECC є більш висока продуктивність. Короткі ключі вимагають меншого мережевого навантаження і обчислювальної потужності, і це чудово підходить для пристроїв з обмеженими можливостями зберігання і обробки. Використання алгоритму ECC в сертифікатах SSL / TLS значно скорочує час, необхідний для шифрування і дешифрування, що допомагає швидше завантажувати веб-сайт. Алгоритм ECC використовується для додатків шифрування, цифрових підписів, в псевдовипадкових генераторів і т. д[11].

Однак проблема масового використання ECC полягає в тому, що багато серверних програм і панелей управління ще не додали підтримку сертифікатів ECC SSL / TLS.

Гібридне шифрування(симетричне + асиметричне)

Зважаючи на те, що кожен із алгоритмів шифрування має свої сильні та слабкі сторони, кожен з них може бути використаний у різних цілях та для різних обсягів інформації, неможливо гарантувати, що один із вибраних алгоритмів зможе покрити усі потреби в безпечній передачі інформації.

Слід одразу зазначити, що гібридне шифрування не є «окремим методом», як симетричне або асиметричне, в ньому використовуються всі переваги обох методів і створюється надійна криптосистема, яка зможе гарантувати надійний захист інформації при передачі по незахищених каналах.

Кожен з алгоритмів шифрування має свої недоліки. Наприклад, метод симетричного шифрування відмінно підходить для швидкого шифрування великих обсягів даних. Але він не забезпечує перевірку особистості, що є необхідним, коли мова заходить про безпеку в локальних або глобальних мережах зв'язку. З іншого боку, асиметричне шифрування надає доступ до даних передбачуваного одержувача. Однак ця перевірка робить процес шифрування занадто повільним. Ідея гібридного шифрування народилася, коли стало критично важливо шифрувати дані з високою швидкістю надаючи при цьому перевірку особистості.

Метод гібридного шифрування використовується в SSL / TLS сертифікатах під час послідовного зв'язку між серверами і клієнтами (веб-браузерами) в процесі, відомому як «TLS handshake»[11]. Спочатку перевіряється особистість обох сторін з використанням закритого і відкритого ключа. Після того, як обидві сторони підтвердили свою особистість, шифрування даних відбувається за допомогою симетричного шифрування з використанням сеансового ключа. Це забезпечує швидку передачу великого обсягу даних, які ми відправляємо і отримуємо.

2.2.3 Криптографічні протоколи

Криптографічний протокол — це абстрактний чи конкретний протокол, що включає набір криптографічних алгоритмів. В основі протоколу лежить набір правил, що регламентують використання криптографічних перетворень та алгоритмів в інформаційних процесах.

Основними функціями криптографічних протоколів є:

1) Аутентифікація джерела даних;
2) Аутентифікація сторін відправника та отримувача;
3) Забезпечення конфіденційності даних;
4) Забезпечення неможливісті відмови;
5) Забезпечення неможливість відмови з доказом отримання;
6) Забезпечення неможливість відмови з доказом джерела;

7) Забезпечення цілісності даних;
8) Забезпечення цілісності з'єднання без відновлення;
9) Забезпечення цілісності з'єднання з відновленням;
10) Розмежування доступу.

Вимоги до безпеки протоколів:

1) Аутентифікація при розсилці за багатьма адресами або при підключенні до служби підписки/повідомлення
2) Авторизація (довіреної третьою стороною);
3) Властивості спільної генерації ключа:
4) Конфіденційність;
5) Анонімність:
6) Обмежена захищеність від атак типу "відмова в обслуговуванні";
7) Інваріантність відправника;
8) Неможливість відмови від раніше вчинених дій:
9) Безпечна тимчасова властивість.

2.2.4 Класифікація протоколів

1. Протоколи шифрування/розшифрування

В основі протоколу цього класу міститься деякий симетричний або асиметричний алгоритм шифрування/дешифрування. Алгоритм шифрування виконується на передачі відправником повідомлення, в результаті чого повідомлення перетвориться з відкритої форми в шифровану. Він виконується на прийомі одержувачем, де повідомлення перетвориться з шифрованого форми у відкриту. Так забезпечується властивість конфіденційності.

Для забезпечення властивості цілісності переданих повідомлень симетричні алгоритми шифрування / розшифрування, зазвичай, поєднуються з алгоритмами обчислення імітозахисної вставки (ІЗВ) на передачу та перевірки ІЗВ на прийомі, для чого використовується ключ шифрування. При використанні асиметричних алгоритмів шифрування / розшифрування властивість цілісності забезпечується окремо шляхом обчислення електронного цифрового підпису (ЕЦП) на передачу та перевірки ЕЦП на прийомі, ніж забезпечуються також властивості безвідмовності і автентичності прийнятого повідомлення[14].

2. Протоколи електронного цифрового підпису (ЕЦП)

В основі протоколу цього класу міститься певний алгоритм обчислення ЕЦП на передачі за допомогою секретного ключа відправника та перевірки ЕЦП на прийомі з допомогою відповідного відкритого ключа, що витягується з відкритого довідника, але захищеного від модифікацій. У разі позитивного результату перевірки протокол, зазвичай, завершується операцією архівування прийнятого повідомлення, його ЕЦП і відповідного відкритого ключа. Операція архівування може не виконуватися, якщо ЕЦП використовується тільки для забезпечення властивостей цілісності і автентичності отриманого повідомлення[14].

3. Протоколи ідентифікації/аутентифікації

В основі протоколу ідентифікації міститься певний алгоритм перевірки того факту, що ідентифікований об'єкт (користувач, пристрій, процес і т.д.), який пред'явив деякий ідентифікатор, знає секретну інформацію, відому тільки заявленому об'єкту, причому метод перевірки є, звичайно, непрямим, тобто без пред'явлення цієї секретної інформації[14].

Зазвичай з кожним ідентифікатором об'єкта пов'язується перелік його прав і повноважень у системі, записаний в захищеній базі даних. У цьому випадку протокол ідентифікації може бути розширений до протоколу аутентифікації, в якому ідентифікований об'єкт перевіряється на уповноваження до замовленої послуги[14].

Якщо в протоколі ідентифікації використовується ЕЦП, то роль секретної інформації відіграє секретний ключ ЕЦП, а перевірка ЕЦП здійснюється за допомогою відкритого ключа ЕЦП, знання якого не дозволяє визначити відповідний секретний ключ, але дозволяє переконатися в тому, що він відомий автору ЕЦП.

4.Протоколи аутентифікованого розподілу ключів

Протоколи цього класу поєднують аутентифікацію користувачів з протоколом генерації і розподілу ключів по каналу зв'язку. Протокол має двох або трьох учасників; третім учасником є центр генерації та розподілу ключів (ЦГРК).

Протокол складається з трьох етапів:

1) генерація;

2) реєстрація;

3) комунікація.

На етапі генерації сервер генерує числові значення параметрів системи, в тому числі, свій секретний і відкритий ключ.

На етапі реєстрації сервер ідентифікує користувачів за документами, для кожного об'єкта генерує ключову і/або ідентифікаційну інформацію і формує маркер безпеки, що містить необхідні системні константи і відкритий ключ сервера (при необхідності).

На етапі комунікації реалізується власне протокол аутентифікованого ключового обміну, який завершується формуванням спільного сеансового ключа.

У відповідності з наведеною класифікацією, розглянемо одні з основних типів криптопротоколів.

1. TLS протокол

TLS є абревіатурою для Transport Layer Security. Це тип цифрової безпеки, який дозволяє зашифрувати зв'язок між веб-сайтом та веб-браузером. Мета TLS – зробити з'єднання безпечним для передачі конфіденційної інформації, включаючи особисті дані, інформацію про платіж або реєстрацію.

Сертифікати SSL/TLS працюють через цифрову прив'язку криптографічного ключа до ідентифікуючої інформації компанії. Це дозволяє шифрувати передачу даних таким чином, що вони не можуть бути розшифровані третіми особами.

SSL/TLS працює як приватний, так і відкритий ключ, а також ключі сеансу для кожного унікального безпечного сеансу. Коли відвідувач вводить захищену SSL-адресу до свого веб-браузера або переходить на безпечну сторінку, браузер та веб-сервер встановлюють з'єднання.

Під час початкового підключення загальнодоступні та закриті ключі будуть використовуватися для створення ключа сеансу, який потім використовуватиметься для шифрування та дешифрування даних, що передаються. Цей ключ сеансу залишиться дійсним протягом обмеженого часу та використовуватиметься лише для цього сеансу.

Основне завдання TLS полягає в тому, щоб переконатися, що тільки одна людина — особа або організація, затверджена користувачем, може отримати

доступ до даних, що передаються. Це особливо важливо, коли мова йде про те, між якими пристроями та серверами передаються дані до того, як вони досягнуть свого пункту призначення.

TLS сертифікат можна використовувати практично на будь-якому пристрої, що робить його універсальним вибором безпеки в сучасному світі. Переваги використання SSL-сертифіката переважують час та кошти, необхідні для їх налаштування.

2. SSH протокол

SSH (Secure Shell) - це протокол віддаленого адміністрування, розроблений для здійснення віддаленого управління операційними системами і тунелювання TCP-з'єднання[15]. Використання цього протоколу допускає використання різних алгоритмів шифрування, що дозволяє безпечно працювати практично в будь-якому незахищеному середовищі: працювати з ПК через командну оболонку, передавати шифрованим каналом будь-який тип даних.

SSH це протокол, який використовує клієнт-серверну модель для аутентифікації віддалених систем та забезпечення шифрування даних, обмін якими відбувається у рамках віддаленого доступу.

За замовчуванням для роботи протоколу використовується TCP-22 порт: на ньому сервер очікує вхідне підключення і після отримання команди і проведення аутентифікації організує запуск клієнта, відкриваючи обрану користувачем оболонку. При необхідності користувач може змінювати порт, що використовується.

Для створення SSH підключення клієнт повинен ініціювати з'єднання з сервером, забезпечивши захищене з'єднання та підтвердивши свій ідентифікатор (перевіряються відповідність ідентифікатора з попередніми записами, що зберігаються в файлі RSA, та особисті дані користувача, необхідні для аутентифікації)[15].

Використання цього виду протоколу надає можливість надійно захищати з’єднання з сервером та чітко ідентифікувати користувачів, що намагаються зв’язатися за допомогою цього типу протоколу. За допомогою SSH протоколу додатково з’являється можливість шифрувати будь-який вид інформації, що

передається. Шифрування може бути здійсненим, використовуючи будь-який з видів алгоритмів: асиметричний, симетричний або хешування. Також слід зазначити, що даний вид протоколу забезпечує можливість безпечного використання будь-якого мережевого протоколу, це дозволяє передавати по каналу файли без обмеження по розміру.

2.3 Розробка криптографічного протоколу

Задля усунення вразливостей, через які можуть бути реалізовані загрози, знайдені в процесі аналізу необхідне використання криптографічних протоколів, так як вибір на користь використання лише одного з алгоритмів шифрування, не надає захисту від усіх видів загроз, що є актуальними для даної системи.

Таким чином, необхідне використання декількох видів шифрування, для реалізації різних функцій безпеки інформації при її передачі.

Використання криптографічних протоколів можливе у двох варіантах: використання вже існуючого протоколу для усунення вразливостей безпеки системи, та розробка нового криптографічного протоколу, як з нуля так і взяття вже існуючого алгоритму за основу, та розширення його функціями, необхідними системі, що розглядається.

Враховуючи той факт, що хоча й використання вже розроблених дозволить значно знизити матеріальні трати та час на впровадження, це значним образом впливає на гнучкість використання такого роду рішень. Адже вже створений продукт підтримується виключно підприємством, яким воно було створене, і в разі необхідності внесення змін в роботу криптопротоколу – буде неможливо зробити це вчасно, чи зробити взагалі.

Тому, проаналізувавши можливі наслідки використання готового рішення, можна прийти до висновку, що впровадження такого роду рішення не є доцільним з точки зору підтримання безпеки системи, та необхідності будь-якого роду змін в ній. Таким чином залишається варіант створення нового криптопротоколу, який буде відповідати вимогам безпеки системи, та зможе усунути вразливості, які існують на даний момент.

Протокол, що буде розроблений в результаті виконання роботи повинен виконувати наступні послуги:

1) аутентифікацію сторін при передачі;
2) шифрування переданих даних;
3) забезпечення цілісності переданих даних;
4) забезпечення захисту від дублювання переданих команд;

Криптографічний алгоритм повинен мати структуру, описану на рис.2.2, для забезпечення усіх вимог безпеки. Розробка криптопротоколу згідно структурних вимог потребує вибору рішень для кожного пункту вимог безпеки. Кожне з функціональних рішень повинне бути виконане одне за одним задля отримання бажаного результату.

Взаємодію на усіх етапах можна описати наступним чином(рис.2.2).

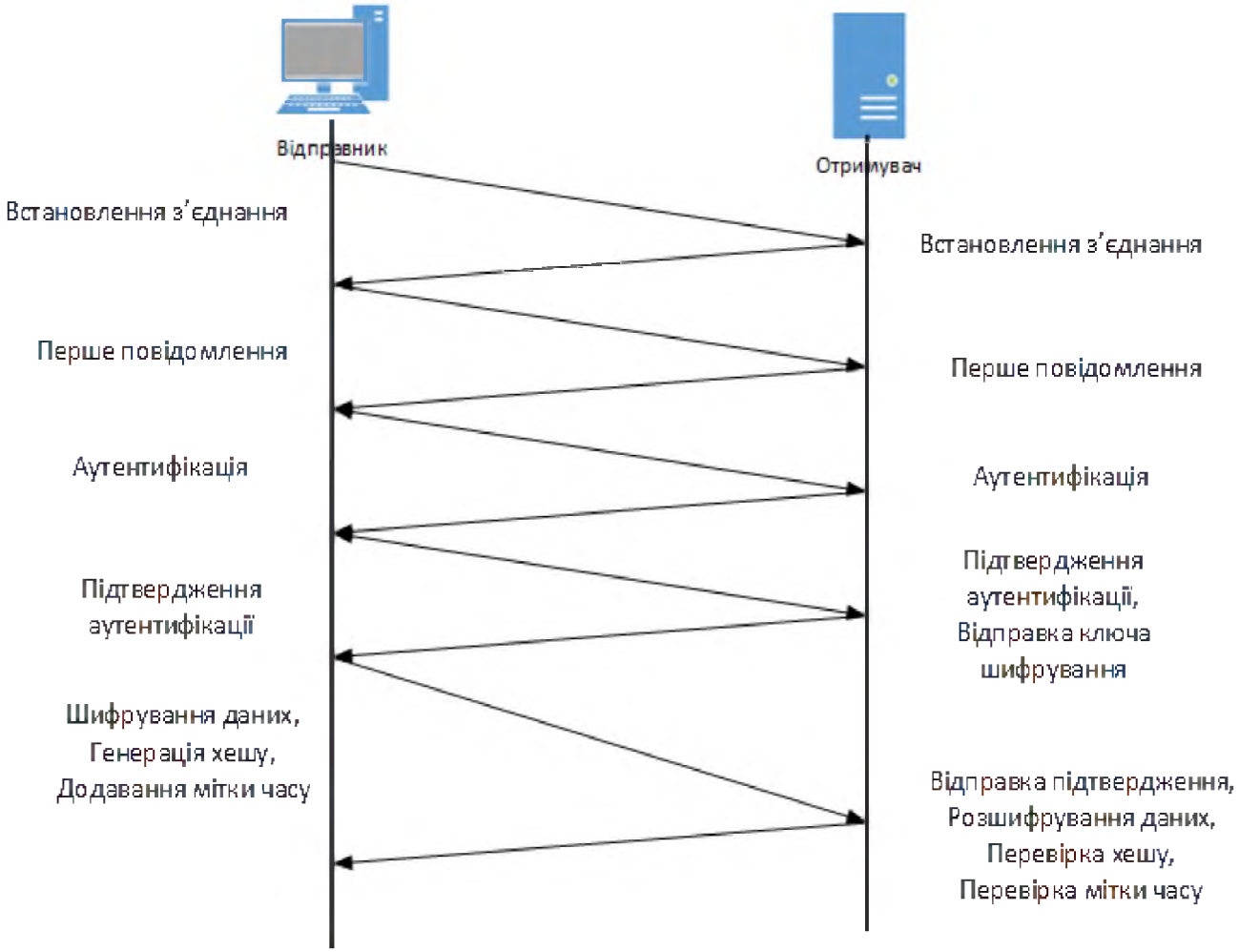


Рисунок 2.2 Схема обміну інформацією в криптопротоколі

Крок 1. Для аутентифікації повинен бути обраний асиметричний алгоритм, завдяки якому відправник та отримувач надсилають один одному пароль з міткою часу, які зашифровані публічним ключем. Надалі потрібно організувати шифрування даних, які будуть передаватися, і зважаючи на обмежені

обчислювальні можливості обладнання лінійних пунктів “КАСКАД”, для шифрування даних потрібно використовувати тільки симетричні алгоритми шифрування, через їх істотно вищу швидкість роботи.

Задля зменшення кількості запитів, що відправляються під час кожного сеансу передачі інформації, ключі публічний та приватний повинні бути завантажені на пристрої пунктів заздалегідь, при встановленні ПЗ. Таким чином їх можна буде використовувати неодноразово впродовж встановленого проміжку часу.

Крок 2. Після встановлення безпечного з’єднання потрібно згенерувати ключ шифрування, який буде використано для симетричного алгоритму шифрування даних, які передаватимуться по каналу. Даний ключ буде згенерований на боці ЦП “КАСКАД”, через те, що технічне обладнання центрального пункту занадто перевершує обладнання лінійного пункту в області обчислювальних можливостей. І після генерації буде переданий до лінійного пункту по вже встановленому захищеному з’єднанню, використовуючи сеансовий ключ, отриманий на попередньому кроку.

Крок 3. Після отримання ключа, який буде використаний для алгоритму симетричного шифрування, повідомлення, що передається, шифрується за допомогою обраного алгоритму.

Крок 4. Оскільки криптопротокол повинен не тільки забезпечувати конфіденційність інформації, але й її цілісність та доступність, при шифруванні потрібні бути виконані додаткові кроки, що забезпечать збереженні цих властивостей переданої інформації. А саме, потрібно забезпечити контроль цілісності зашифрованого повідомлення за допомогою передачі шифрованого хешу повідомлення, або електронного цифрового підпису(ЕЦП). Це наддасть змогу точно знати, що повідомлення не було модифіковане в процесі передачі.

Крок 5. І останній крок, який повинен бути зроблений на стороні відправника, перед безпосередньо відправкою повідомлення - забезпечення захисту від дублювань. Це може бути забезпечено за допомогою додавання до повідомлення зашифрованої мітки часу або унікального ідентифікатора.

Крок 6. Після виконання вище перерахованих повідомлення з інформацією, що потребує захисту - відправляється отримувачу.

На боці отримувача повинні бути зроблені кроки, що є протилежними зробленим на боці відправника. А саме, зашифроване повідомлення повинне бути розшифроване за допомогою згенерованого раніше ключа.

Після розшифровки повідомлення повинна бути перевірена його цілісність за допомогою обчислювання хеш функції прийнятого повідомлення та порівняння його з хешем, що прийшов. Або за допомогою перевірки ЕЦП.

І останнім етапом є порівняння мітки часу або унікального ідентифікатора відправленого повідомлення із тими, що були відправлені раніше.

В разі якщо хоча б один з вище перерахованих етапів на боці отримувача провалюється, повідомлення що надійшло буде вважатися не валідним. І система повинна сигналізувати про наявність помилки при передачі даних, яка в подальшому повинна бути оброблена належним чином.

Управління ключами при обміні повинно бути реалізоване як окремий механізм в рамках криптопротоколу. А саме, ключ шифрування, який отриманий в результаті обміну частинами ключа на першому кроці, буде використаний для шифрування секретів необхідних для роботи симетричного алгоритму. Задля зменшення кількості запитів під час кожного з'єднання, і як наслідок, зменшення часу обміну інформацією між лінійними та центральними пунктами «КАСКАД», отриманий ключ повинен мати термін закінчення дії. Даний термін доцільно буде встановити на відмітці 15 хвилин. Таким чином, після закінчення дії ключа, черговий сеанс відправки даних повинен передбачати процес обміну ключами з новим терміном дії.

2.3.1 Обґрунтування вибору алгоритму шифрування даних

Вибір алгоритму шифрування має бути зроблений на користь алгоритму, який буде підходити для системи лінійного пункту «КАСКАД» не тільки за критерієм криптостійкості, але й за іншим важливим показником, а саме, швидкістю шифрування та, як наслідок, вимоги до технічного обладнання, на якому шифрування буде виконуватися.

Провівши попередній огляд криптографічних шифрів, можна побачити, що коли мова йде про швидкість шифрування, симетричні шифри мають беззаперечну перевагу як з боку швидкості так і необхідних обчислювальних можливостей обладнання, що буде виконувати шифрування.

Таким чином буде доцільним розглядати симетричні алгоритми шифрування даних, як такі, що будуть використані для шифрування самих даних, що передаються. Беручи до уваги обмежені обчислювальні можливості обладнання лінійних пунктів, серед алгоритмів варто звернути увагу на найбільш актуальні та ефективні, такі як: AES, ДСТУ ГОСТ 28147:2009, ДСТУ 7624:2014 «Калина».

2.3.1.1 Алгоритм AES

Advanced Encryption Standard (AES) — це специфікація для шифрування електронних даних, заснована Національним інститутом стандартів і технологій США (NIST) у 2001 році. AES широко використовується сьогодні, оскільки він набагато сильніший, ніж DES і потрійний DES, незважаючи на складність. впровадження.

AES є ітеративним, а не шифром Фейстеля. Він заснований на «мережі підстановки-перестановки». Він складається з серії пов'язаних операцій, деякі з яких передбачають заміну вхідних даних конкретними виходами (підстановки), а інші передбачають перемішування бітів навколо (перестановки)[17].

AES виконує всі свої обчислення в байтах, а не бітах. Отже, AES розглядає 128 біт блоку відкритого тексту як 16 байтів. Ці 16 байтів розташовані в чотирьох стовпцях і чотирьох рядках для обробки як матриця.

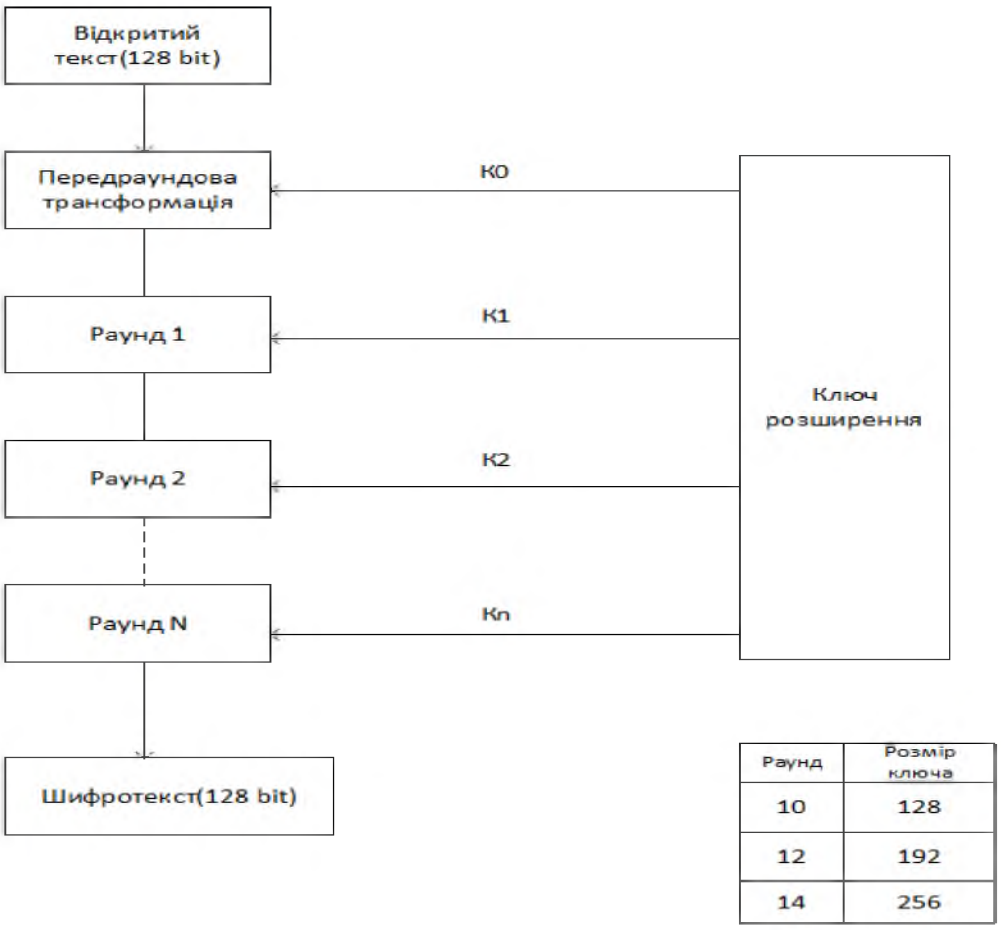
Кількість раундів в AES змінна і залежить від довжини ключа. AES використовує 10 раундів для 128-бітних ключів, 12 раундів для 192-бітних ключів і 14 раундів для 256-бітних ключів. Кожен з цих раундів використовує інший 128-бітовий ключ раунду, який обчислюється на основі вихідного ключа AES(рис. 2.3).

Алгоритм розкладу ключів використовується для обчислення всіх раундових ключів від початкового ключа. Таким чином, початковий ключ використовується для створення багатьох різних раундових ключів, які будуть використовуватися у відповідному раунді шифрування.

Шифрування.

AES розглядає кожен блок як сітку розміром 16 байт (4 байти x 4 байти = 128) у головному порядку стовпців.

```
[ b0 | b4 | b8 | b12 |
| b1 | b5 | b9 | b13 |
| b2 | b6 | b10| b14 |
| b3 | b7 | b11| b15 ]
```



Раунд	Розмір ключа
10	128
12	192
14	256

Рисунок 2.3 Схема шифрування AES

Кожен раунд складається з 4 кроків:

1. заміна байтів;
2. зсув рядків;
3. перестановка стовпців;
4. додається раундовий ключ.

В останньому раунді немає раунду перестановки стовпців. На першому крок раунду виконує заміну, а зсув рядків і перестановка стовпців виконують перестановку в алгоритмі.

Перший етап алгоритму реалізує заміну байтів.

На цьому кроці кожен байт замінюється іншим байтом. Це виконується за допомогою таблиці пошуку, яка також називається S-box. Ця заміна виконується таким чином, що байт ніколи не замінюється сам по собі, а також не замінюється іншим байтом, який є доповненням поточного байту. Результатом цього кроку є матриця розміром 16 байт (4 x 4), як і раніше.

На другому етапі здійснюється зсув рядків.

Цей крок виконується так само, як називається. Кожен рядок зсувається певну кількість разів.

1. Перший ряд не зміщується
2. Другий ряд зсувається один раз вліво.
3. Третій ряд двічі зміщується вліво.
4. Четвертий ряд тричі зміщується вліво.

```
[ b0  | b1  | b2  | b3  ]          [ b0  | b1  | b2  | b3  ]
| b4  | b5  | b6  | b7  |    ->    | b5  | b6  | b7  | b4  |
| b8  | b9  | b10 | b11 |          | b10 | b11 | b8  | b9  |
[ b12 | b13 | b14 | b15 ]          [ b15 | b12 | b13 | b14 ]
```

На третьому етапі здійснюється перестановка стовпців.

Кожен стовпець із чотирьох байтів тепер перетворюється за допомогою спеціальної математичної функції. Ця функція приймає на вхід чотири байти одного стовпця і виводить чотири абсолютно нових байти, які замінюють вихідний стовпець. Результатом є ще одна нова матриця, що складається з 16 нових байтів. Слід зазначити, що цей крок не виконується в останньому раунді.

На четвертому етапі здійснюється додавання раундового ключа.

16 байтів матриці тепер розглядаються як 128 біт і приєднуються до 128 біт раундового ключа(рис. 2.4). Якщо це останній раунд, то результатом буде зашифрований текст. В іншому випадку отримані 128 біт інтерпретуються як 16 байтів, і ми починаємо ще один подібний раунд.

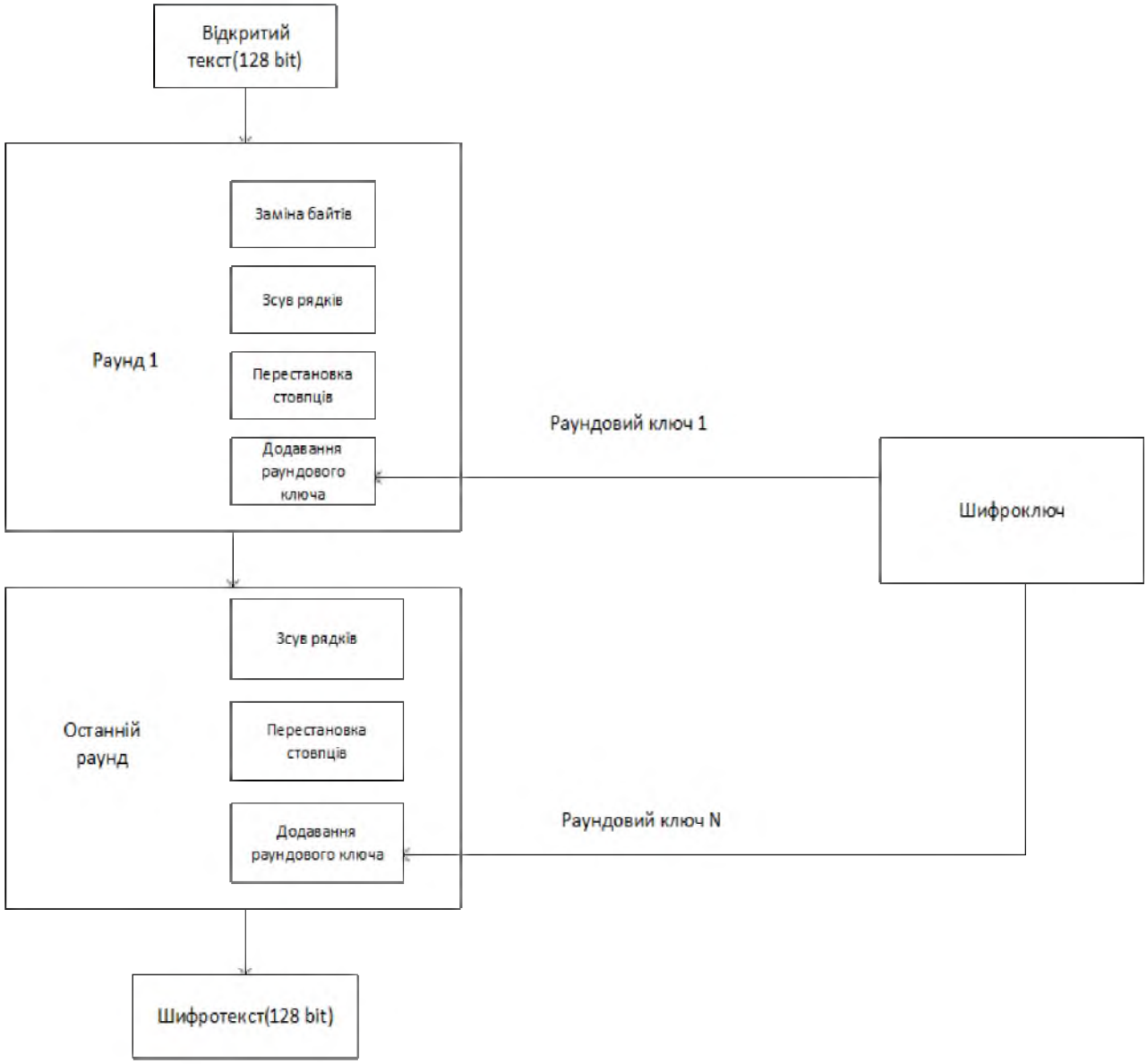


Рисунок 2.4 схема додавання раундового ключа

Після всіх цих раундів 128 біт зашифрованих даних повертаються як вихідні дані. Цей процес повторюється до тих пір, поки всі дані, що підлягають шифруванню, не пройдуть цей процес.

Дешифрування.

Етапи в раундах можна легко скасувати, оскільки ці етапи мають протилежність, яка при виконанні скасовує зміни. Кожні 128 блоків проходить 10, 12 або 14 раундів залежно від розміру ключа.

Етапи кожного раунду розшифровки такі:

1) додати раундовий ключ;
2) Інверсні стовпці змішування;
3) зсув рядків;
4) інверсна заміна байтів.

Набір інструкцій AES тепер інтегрований в центральний процесор (пропонує пропускну здатність кілька ГБ/с), щоб підвищити швидкість і безпеку програм, які використовують AES для шифрування та дешифрування. Незважаючи на те, що минуло 20 років з моменту його впровадження, нам не вдалося зламати AES алгоритм, оскільки він нездійсненний навіть за сучасних технологій. Поки що єдина вразливість залишається в реалізації алгоритму[17].

Світові тенденції свідчать про початок поступової відмови від цього шифру, як на рівні вибору перспективних рішень в міжнародних криптографічних конкурсах, так і в прикладних системах [7]. Зокрема, деякі компанії, лідери IT-індустрії, такі як Google, вже застосовують нові алгоритми замість AES.

Експерти з безпеки стверджують, що AES безпечний, якщо його правильно запровадити. Однак ключі шифрування AES необхідно захищати. Навіть найбільш стійкі криптографічні системи можуть бути вразливими, якщо хакер отримає доступ до ключа шифрування.

2.3.1.2 ДСТУ ГОСТ 28147:2009

В Україні з 1990 р. в якості БСШ використовувався алгоритм ГОСТ 28147-89 (офіційна назва ДСТУ ГОСТ 28147:2009) з розміром блоку і ключа 64 та 256 біт відповідно. Хоча він все ще забезпечує практичну стійкість, для нього вже відомі теоретичні методи криптоаналізу, зі складністю істотно меншою, ніж повний перебір ключів.

Шифрування.

Алгоритм криптографічного перетворення призначений для апаратної або програмної реалізації, задовольняє криптографічним вимогам та за своїми можливостям не накладає обмежень на ступінь секретності інформації. Схема роботи криптографічного алгоритму наведена на рисунку 2.5.

Криптосхема містить такі елементи:

1) ключовий запам'ятовуючий пристрій (КЗП) на 256 біт, що складається з 8-ми 32-розрядних накопичувачів (X0, X1, X2, X3, X4, X5, X6, X7);
2) чотири 32-розрядні накопичувачі (N1, N2, N3, N4);
3) два 32-розрядні накопичувачі (N5, N6) із записаними в них постійними заповненнями C2, C1;
4) два 32-розрядних суматора за модулем 232 (СМ1, СМ3);
5) 32-розрядний суматор порозрядного підсумовування за модулем 2 (СМ2);
6) суматор за модулем 2 без обмежень на розрядність (СМ5);
7) блок підстановки (К);
8) регістр циклічного зсуву на одинадцять кроків у бік старшого розряду (R) [18].

Рисунок 2.5 Схема шифрування

ДСТУ 28147:2009

Блок підстановки складається з восьми вузлів заміни К1, К2, К3, К4, К5, К6, К7, К8 з пам'яттю по 64 біти кожен. 32-розрядний вектор, що надходить на блок підстановки, розбивається на вісім послідовно йдуть 4-розрядних векторів, кожен з яких перетворюється в 4-розрядний вектор відповідним вузлом заміни, що представляє собою таблицю з шістнадцяти рядків, що містять по чотири біти заповнення в рядку. Вхідний вектор визначає адресу рядка у таблиці, заповнення цього рядка є вихідним вектором. Потім 4-розрядні вихідні вектори послідовно поєднуються в 32-розрядний вектор.

При додаванні та циклічному зрушенні двійкових векторів старшими розрядами вважаються розряди накопичувачів з великими номерами.

При записі ключа (W1, W2, … W256), Wq [0,1], q = 1 … 256, у КЗУ значення W1 вводитиметься в 1-й розряд накопичувача X0, значення W32 вводитиметься у 32-й розряд накопичувача X0, значення W33 вводитись у 1-й розряд накопичувача X1, значення W34 вводитиметься у 2-й розряд накопичувача X1, значення W256 вводитиметься у 32-й розряд накопичувача X7[18].

При перезапису інформації вміст р-го розряду одного накопичувача (суматора) переписується до р-го розряду іншого накопичувача (суматора).

Значення постійних заповнень С1, С2 (констант) накопичувачів N6, N5 наведено у додатку 2 ДСТУ 28147:2009.

Ключі, що визначають заповнення КЗП та таблиць блоку підстановки є секретними елементами і поставляються в установленому порядку.

Заповнення таблиць блоку підстановки є довгостроковим ключовим елементом, загальним для мережі електронно обчислювальних машин(ЕОМ).

Організація різних видів зв'язку досягається побудовою відповідної ключової системи. При цьому може бути використана можливість вироблення ключів (заповнень КЗП) у режимі простої заміни та зашифрування їх у режимі простої заміни із забезпеченням імітозахисту для передачі каналами зв'язку або зберігання в пам'яті ЕОМ[18].

У криптосхемі передбачено чотири види роботи:

1) зашифрування (розшифрування) даних у режимі простої заміни;
2) зашифрування (розшифрування) даних у режимі гамування;

3) зашифрування (розшифрування) даних у режимі гамування зі зворотним зв'язком;
4) режим вироблення імітовставки.

Розшифрування.

Криптосхема, що реалізує алгоритм розшифрування в режимі простої заміни, має той самий вид (рис. 2.5), що і зашифрування. У КЗУ вводяться 256 біт того ключа, у якому здійснювалося зашифрування відкритих даних. Зашифровані дані, що підлягають розшифруванню, розбиті на блоки по 64 біти в кожному.

Розшифрування здійснюється за тим же алгоритмом, що і зашифрування відкритих даних, з тією зміною, що заповнення накопичувачів X0, X1, ..., X7 зчитуються з КЗП у циклах розшифрування в іншому порядку.

2.3.1.3 ДСТУ 7624:2014 «Калина»

Національний стандарт шифрування ДСТУ 7624:2014 (Калина) [20] належить до SPN, байт-орієнтованих шифрів. Основні параметри шифру, такі як довжина ключа k і блоку даних l, кількість раундів t та кількість стовпців матриці стану c пов’язані залежностями представленими в табл. 2.2. Розмір блоку і довжина ключа використовуються у позначенні шифру у форматі Калина-l/k.

Таблиця 2.2 Параметри шифру «Калина»

Довжина ключа k, біт	Довжина блоку l, біт	Кількість раундів t	Кількість стовпців матриці стану c
128, 256	128	10	2
256, 512	256	14	4
512	512	18	8

При виконанні зашифрування або розшифрування операції виконуються над двовимірним масивом байт, названим поточним станом шифру. Поточний стан шифру можна представити у вигляді матриці розмірністю 8 x c байтів (вісім рядків по c байт).

В алгоритмі використовуються операції арифметичного додавання та віднімання за модулем 2^{64}, додавання за модулем 2, табличної заміни циклічного зсуву рядків та лінійного перетворення. Структура алгоритму «Калина» представлена на рисунках 2.6 та 2.7 [21].

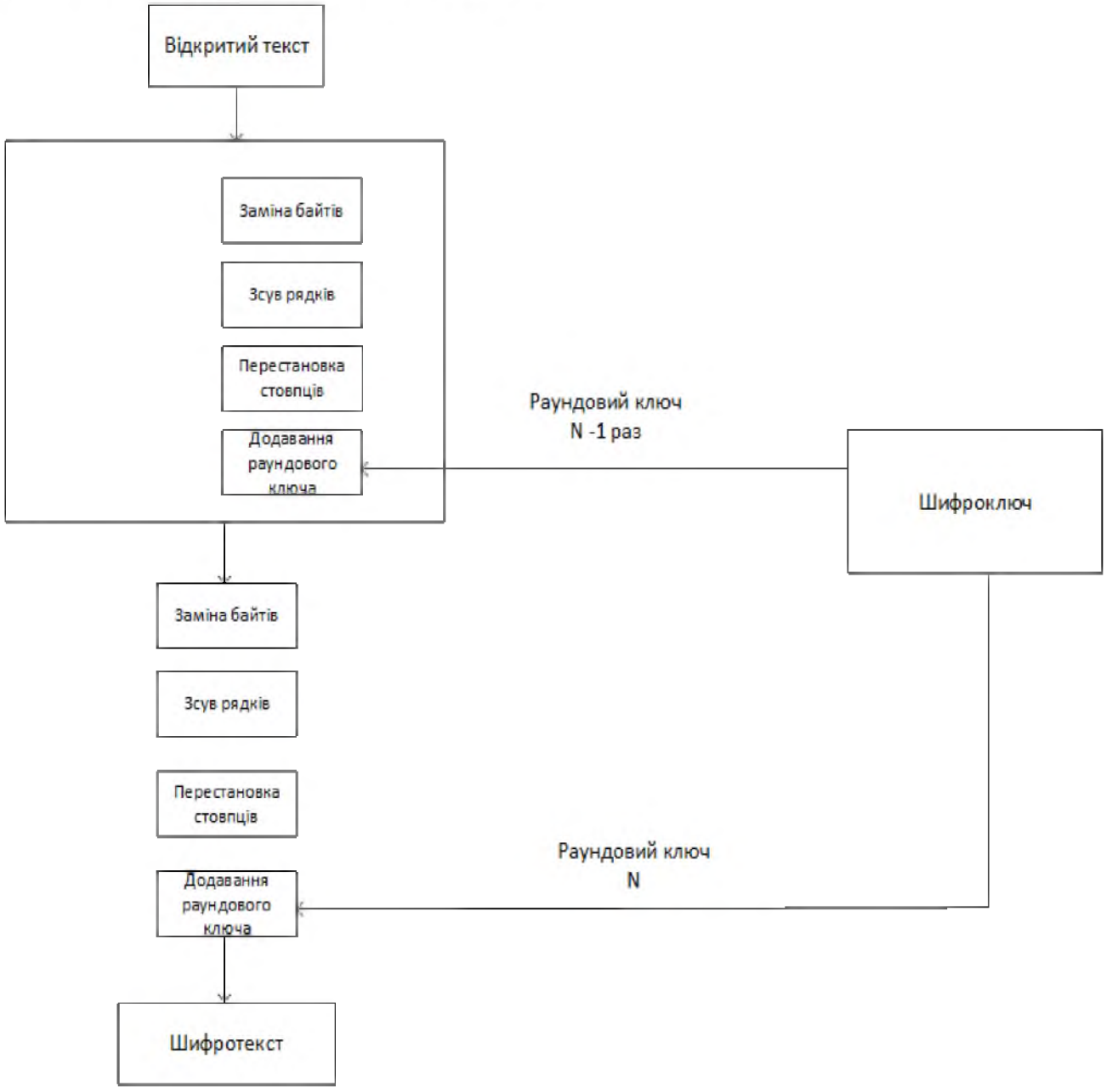


Рисунок 2.6 Схема шифрування алгоритму «Калина»

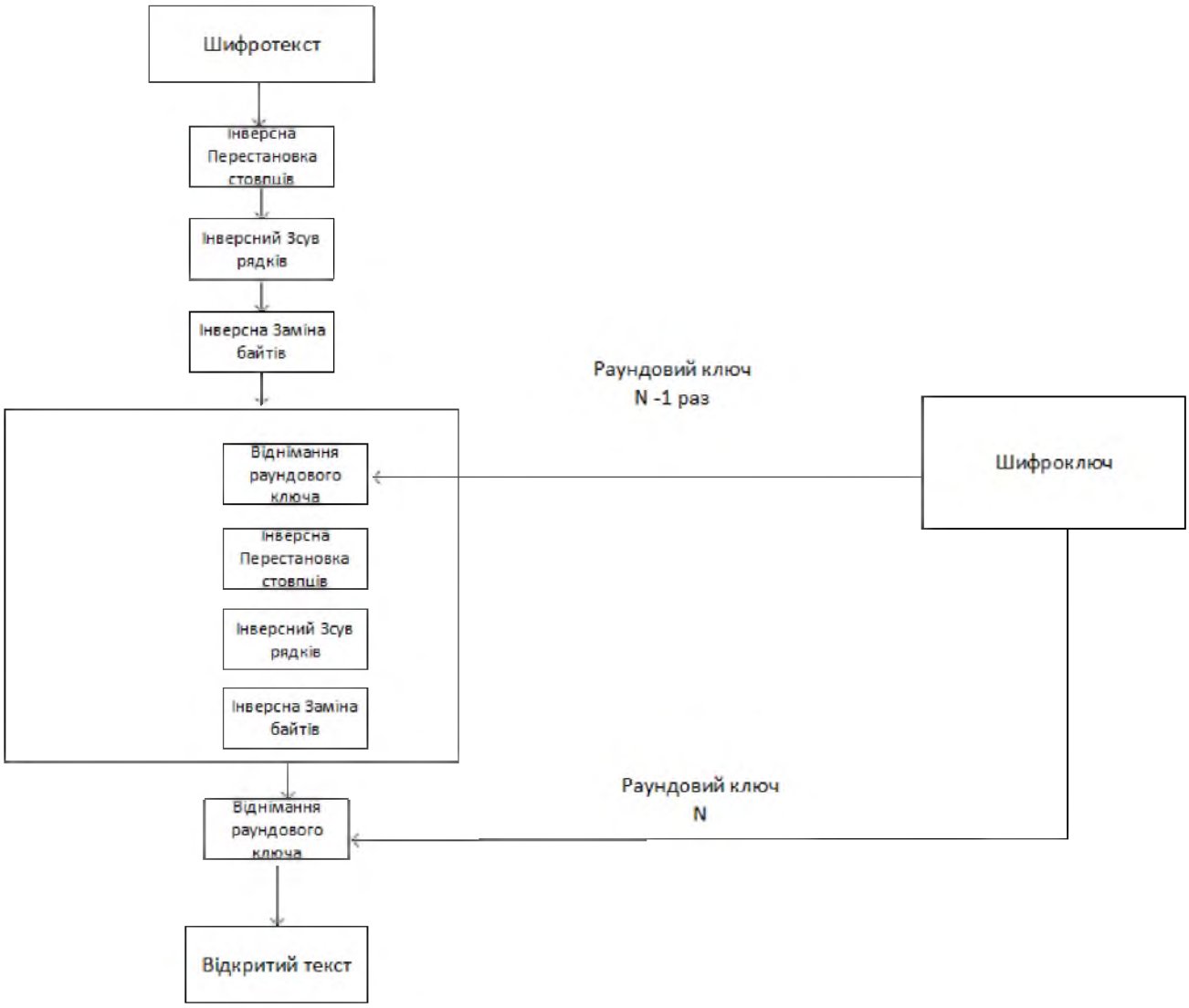


Рисунок 2.7 Схема розшифрування алгоритму «Калина»

Розглянемо дещо детальніше кожну з цих операцій.

Операції додавання та віднімання – реалізують арифметичне додавання або віднімання стовпців матриці стану і стовпців циклового підключа за модулем 2^{64}. Числа в стовпцях вважаються представленими у форматі little-endian, тобто менш значущі байти мають менший індекс.

Операція додавання за модулем 2 – виконує додавання за модулем 2 матриці стану і циклового підключа.

Операції заміни байтів та інверсної заміни – виконують підстановку кожного байту матриці стану на відповідний йому байт з однієї з чотирьох таблиць заміни S0-S3 та inv_S0-inv_S3 для операцій зашифрування і розшифрування відповідно.

Кожна таблиця має розмір 256 байт. Номер таблиці заміни визначається як індекс рядку байту за модулем 4 (i mod 4)[22].

Операції зсуву та інверсного зсуву – здійснюють циклічний зсув байтів рядків вправо чи вліво відповідно. Кількість позицій μ_i , на яку зсувається рядок, залежить від номера рядку і та довжини блоку l і обчислюється за формулою:

$$\mu_i = [i * l / 512] \qquad (2.1)$$

Операції перемішування та інверсного перемішування стовпців – здійснюють перетворення стовпців матриці стану шляхом виконання операцій множення і додавання в скінченному полі $GF(2^8)$ за модулем незвідного многочлена

$$\psi = x^8 + x^4 + x^3 + x^2 + 1. \qquad (2.2)$$

Для одержання раундових підключів з вихідного майстер-ключа використовується процедура розгортання ключів, в якій задіяні стандартні перетворення, розглянуті вище.

Таблиця 2.3 Розрахунок кількості процесорних інструкцій для обробки одного байту даних[16]

	Симетричний блоковий шифр		
	Калина	ДСТУ 28147	AES
Кількість операцій на байт	40,375	72	45,375

У роботах [16, 23] наведені результати порівняння продуктивності, що проводилося для шифру «Калина» (всі комбінації розміру блоку і довжини ключа), AES-128, AES-256, ДСТУ 28147:2009, в однакових умовах роботи (розмір блоку даних 1 Гбайт, режим простої заміни ECB, багатократне шифрування одного блоку).

Автори статті[24] прийшли до наступних висновків щодо 64-бітової платформи:

1) для 128-бітової довжини ключа швидкодія «Калини» вища за AES на 3% (86 Мбіт/с);
2) для 256-бітової довжини ключа швидкодія «Калини» повільніша за AES на 10% (для 128-бітового блоку) та швидше на 1% (для 256-бітового блоку);

3) швидкодія «Калини» при відповідній довжині ключа вища за ДСТУ 28147:2009 у 2,8 рази (для 128-бітового блоку) і 3,16 рази (для 256-бітового блоку.

Як свідчать публікації [16, 23, 24] шифр «Калина» орієнтований на досягнення високої продуктивності на 64-бітових сучасних мікропроцесорах загального призначення (Intel, AMD). Разом з тим, як декларують самі автори шифру, при розробці національного стандарту, що забезпечує високу стійкість і продуктивність, враховуючи відсутність в Україні власного мікроелектронного виробництва та неможливість надійного контролю іноземного, вимоги до ефективної реалізації нового шифру в системах з обмеженими ресурсами розглядалися як другорядні [16].

Таким чином можна побачити, що швидкодія усіх трьох шифрів хоча й відносно не відрізняється один від одного, шифр «Калина» має вищі показники ніж ДСТУ 28147 при будь-якій довжині ключа, та майже рівні показники відносно шифру AES.

Потрібно взяти до уваги той факт, що шифр ДСТУ 28147:2009, незважаючи на практичну стійкість, програє в цьому параметрі як шифру «Калина» так і AES. Це робить його менш привабливим з точки зору використання як основного в частині системи МСДЦ «КАСКАД».

Отже, основний вибір алгоритму криптографічного шифрування доцільно робити між шифрами «Калина» та AES. Обидва криптографічні алгоритми показали високий рівень стійкості та швидкодії, що є дуже важливим, зважаючи на те, що обраний алгоритм повинен виконуватися на обладнанні, яке має досить обмежені обчислювальні можливості.

Хоча й алгоритм AES має трохи кращі показники ніж алгоритм «Калина», слід звернути увагу на деякі ключові аспекти, що дозволять зробити вибір на користь одного з алгоритмів, що розглядаються. Незважаючи на високий рівень стійкості та високий рівень популярності алгоритму AES, даний вид алгоритму поступово стає неактуальним, зважаючи на той факт, що він потенційно може бути зламаний у найближчому майбутньому. Та слід зауважити, що структури безпеки США, які

на даний момент активно використовують даний шифр, мають план по відмові від нього до 2023 року, та переході на новий криптоалгоритм.

Проаналізувавши ключові аспекти, щодо використання шифру AES, можна прийти до висновку, що шифр «Калина» є найбільш доцільним рішенням. Цей шифр розроблено та введено в дію спеціалістами з України, тобто його буде доцільніше використовувати на державних підприємствах, на даний момент не має потенційних проблем з криптографічною стійкістю та повинен замінити діючий стандарт шифрування ДСТУ 28147:2009. І зважаючи на те, що в рамках обмежених обчислювальних можливостей, шифр «Калина» не поступається у швидкодії шифру AES, даний криптографічний алгоритм найбільш підходить та відповідає вимогам системи.

2.3.2 Обгрунтування вибору методу контролю цілісності

Вибір методу контролю цілісності інформації при передачі повинен бути зроблений на користь рішення, яке забезпечить не тільки можливість контролювати факт втручання у інформацію, що передається, але й за показником швидкодії обраного рішення в рамках обмежених обчислювальних можливостей технічного обладнання лінійних пунктів «КАСКАД».

Виходячи з того, що різноманітність методів контролю цілісності, які можуть бути інтегровані в діючу систему, достатньо невелика, вибір рішення повинен бути зроблений на базі криптографічних алгоритмів, що дозволять забезпечити достатній рівень контролю цілісності та стійкості інформації.

Таким чином буде доцільним розглянути рішення, що матимуть мінімальну обчислювальну складність при забезпеченні достатнього рівня криптостійкості, як такі, що будуть використані для контролю цілісності інформації. Варто звернути увагу на алгоритми, що реалізують хешування інформації та алгоритми електронного цифрового підпису.

2.3.3 Хешування

Хешування - перетворення вхідного масиву даних довільної довжини в вихідну бітову послідовність фіксованої довжини, яку можна використати для порівняння даних[25].

Хешування застосовується для порівняння даних: якщо у двох масивах хеш-коди різні, масиви гарантовано розрізняються; якщо однакові - масиви, швидше за все, однакові. У загальному випадку однозначної відповідності між вихідними даними і хеш-кодом немає в силу того, що кількість значень хеш-функцій менше, ніж варіантів вхідного масиву; існує безліч масивів, які дають однакові хеш-коди - так звані колізії. Ймовірність виникнення колізій відіграє важливу роль в оцінці якості хеш-функцій.

Основні питання що виникають при пошуку оптимального алгоритму хешування, є дослідження деяких властивостей цих алгоритмів. Такими властивостями є наприклад стійкість до пошуку першого та другого прообразів, обчислювальна складність та інші.

Для ідеальної хеш-функції виконуються такі умови:

1) хеш-функція є детермінованою, тобто те саме повідомлення призводить до одного і того ж хеш-значенню;
2) значення хеш-функції швидко обчислюється для будь-якого повідомлення;
3) неможливо знайти повідомлення, яке дає задане хеш-значення;
4) неможливо знайти два різні повідомлення з однаковим хеш-значенням;
5) невелика зміна в повідомленні змінює хеш настільки сильно, що нове та старе значення здаються некорелюючими.

Теоретично неможливо визначити хешфункцію так, щоб вона створювала випадкові дані з реальних невипадкових файлів. Однак на практиці реально створити досить хорошу імітацію за допомогою простих арифметичних дій. Більш того, часто можна використовувати особливості даних для створення хеш-функцій з мінімальним числом колізій (меншим, ніж при істинно випадкових даних)[26].

Алгоритм MD4. Алгоритм MD4 є більш ранньої розробкою автора Рона Ривеста. Спочатку даний алгоритм був опублікований в жовтні 1990 р, незначно змінена версія була опублікована в RFC 1320 в квітні 1992 р. [27].

Виокремлюють основні цілі алгоритму MD4:

1) Безпека: це звичайна вимога до хеш-коду, що полягає в тому, щоб було чисельно неможливо знайти два повідомлення, які мають один і той же дайджест.
2) Швидкість: програмна реалізація алгоритму повинна виконуватися досить швидко. Зокрема, алгоритм повинен бути досить швидким на 32-бітної архітектури. Тому алгоритм заснований на простому множині елементарних операцій над 32-бітними словами.
3) Простота та компактність: алгоритм повинен бути простим в описі і простим в програмуванні, без великих програм або підстановочних таблиць. Ці характеристики не тільки мають очевидні програмні переваги, але і бажані з точки зору безпеки, тому що для аналізу можливих слабких місць краще мати простий алгоритм.
4) Бажано little-endian архітектура: деякі архітектури процесорів зберігають ліві байти слова в позиції молодших адрес байта (little-endian). Інші зберігають праві байти слова в позиції молодших адрес байта (big-endian). Ця відмінність є важливою, коли повідомлення трактується як послідовність 32-бітових слів, тому що ці архітектури мають інверсне уявлення байтів в кожному слові. Ці цілі переслідувалися і при розробці алгоритму MD5 [27]. MD5 є більш складним і, отже, більш повільним при виконанні, ніж MD4. Вважається, що додавання складності виправдовується зростанням рівня безпеки.

MD4 (Message Digest 4) - хеш-функція, професором Рональдом Рівестом в 1990 році, і вперше описана в RFC 1186. Для довільного вхідного повідомлення функція генерує 128-розрядне хеш-значення, зване дайджестом повідомлення. Цей алгоритм використовується в протоколі аутентифікації MS-CHAP, розробленому корпорацією Майкрософт для виконання процедур перевірки достовірності віддалених робочих станцій Windows. Є попередником MD5.

Алгоритм MD4 складається з наступних кроків(рис. 2.8):

1) Додавання відсутніх бітів.
2) Додавання довжини повідомлення.

3) Ініціалізація MD-буфера.
4) Обробка повідомлення блоками 16 слів(32 біт).
5) Формування хешу.

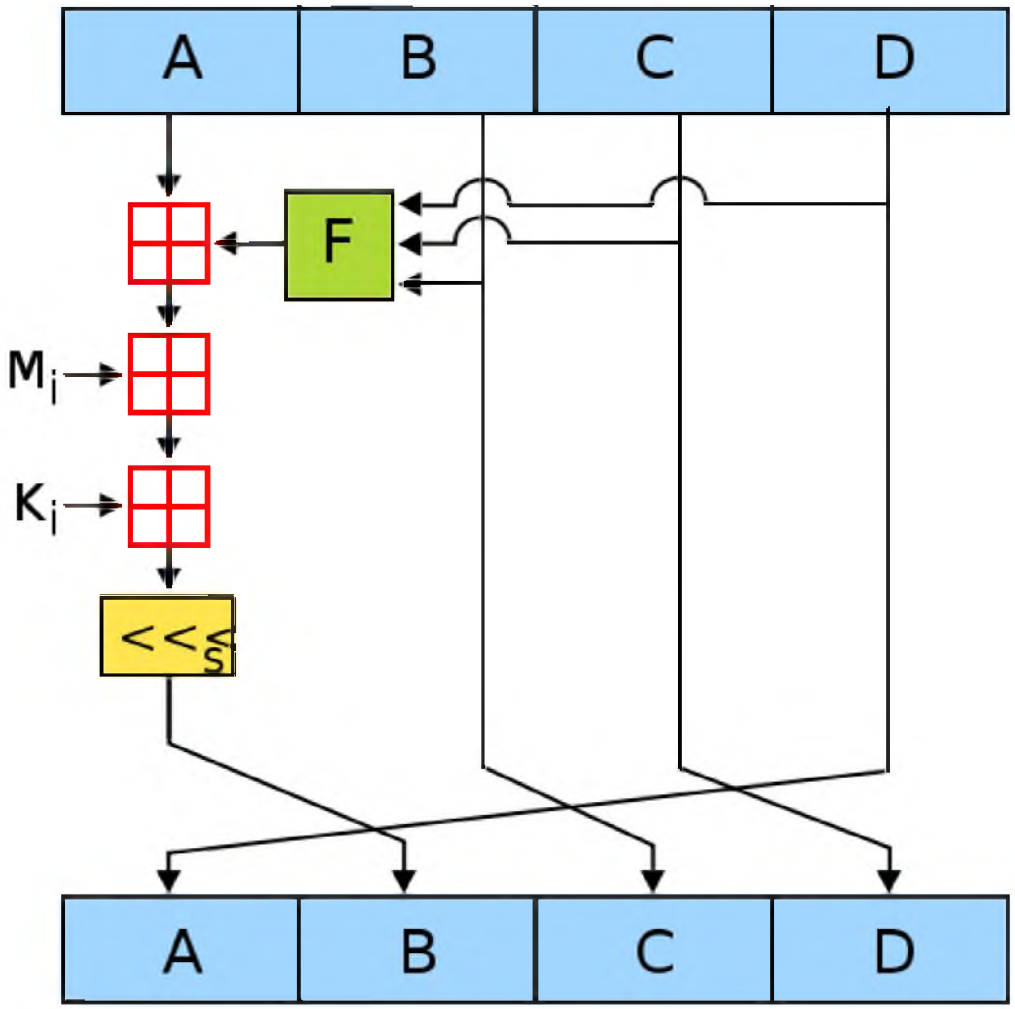


Рисунок 2.8 Одна операція алгоритму хешування MD4

Хешування з MD4 складається з 48 таких операцій, згрупованих у 3 раунди по 16 операцій. F - нелінійна функція; в кожному раунді функція змінюється. M_i означає 32-бітний блок вхідного повідомлення, а K_i - 32-бітова константа, різна для кожної операції. Блоки A, B, C та D – MD- буфер. <<<s – циклічний зсув 32 бітів вліво на s розрядів.

Алгоритмом передбачено, що на вхід подається повідомлення, що складається з b біт, хеш якого належить обчислити. b - Довільне невід'ємне ціле число; воно може бути нулем, не зобов'язана бути кратним восьми, і може бути як завгодно великим.

1) Додавання відсутніх бітів

Повідомлення розширюється так, щоб його довжина в бітах за модулем 512 дорівнювала 448. Таким чином, в результаті розширення, повідомленням бракує 64 біта до довжини, кратної 512 бітам. Розширення виробляється завжди, навіть якщо повідомлення спочатку має потрібну довжину.

Розширення проводиться таким чином: один біт, що дорівнює 1, додається до повідомлення, а потім додаються біти, рівні 0, до тих пір, поки довжина повідомлення не стане рівною 448 по модулю 512. У підсумку, до повідомлення додається як мінімум 1 біт, і як максимум 512[28].

2) Додавання довжини повідомлення

64-бітове представлення b (Довжини повідомлення перед додаванням набивальних бітів) додається до результату попереднього кроку. У малоймовірному випадку, коли b більше, ніж 2^{64} , Використовуються тільки 64 молодших біта. Ці біти додаються у вигляді двох 32-бітних слів, і першим додається слово, що містить молодші розряди.

На цьому етапі (після додавання бітів і довжини повідомлення) ми отримуємо повідомлення довжиною кратною 512 бітам. Це еквівалентно тому, що це повідомлення має довжину, кратну 16-ти 32-бітовим словам. Кожне 32-бітне слово містить чотири 8-бітних, але слідують вони не підряд, а навпаки (наприклад, з восьми 8-бітних слів (abcdefgh) ми отримуємо два 32-бітних слова (dcba hgfe)). Нехай M [0…N-1] означає масив слів jnhbvfyjuj повідомлення (тут N кратно 16)[28].

3) Ініціалізація MD-буфера

Для обчислення хешу повідомлення використовується буфер, що складається з 4 слів (32-бітових регістрів): (A, B, C, D).

Ці регістри ініціалізувалися наступними шістнадцятиричними числами (молодші байти спочатку):

слово A : 01 23 45 67,

слово B : 89 ab cd ef,

слово C : Fe dc ba 98,

слово D : 76 54 32 10

4) Обробка повідомлення блоками по 16 слів

Для початку визначимо три допоміжні функції, кожна з яких отримує на вхід три 32-бітових слова, і по ним обчислює одне 32-бітове слово.

$$F (X, Y, Z) = XY \text{ or } (not) XZ$$

$$G (X, Y, Z) = XY \text{ or } XZ \text{ or } YZ$$

$$H (X, Y, Z) = X \text{ xOr } Y \text{ xOr } Z$$

На кожну бітову позицію F діє як умовний вираз: якщо X , То Y ; Інакше Z . G діє на кожну бітову позицію як функція максимального значення: якщо принаймні в двох словах з X, Y, Z відповідні біти дорівнюють 1 , То G видасть 1 в цьому біті, а інакше G видасть біт, що дорівнює 0 . Цікаво відзначити, що якщо біти X , Y і Z статистично незалежні, то біти F (X, Y, Z) і G (X, Y, Z) будуть також статистично незалежні. Функція H реалізує побітовий xOr , Вона володіє такj. же властивістю, як F і G .

5) Формування хешу

Результат (хеш-функція) виходить як ABCD. Тобто, ми виписуємо 128 біт, починаючи з молодшого біта A, і закінчуючи старшим бітом D.

2.3.4 Електронний цифровий підпис

Згідно статті 1 Закону України «Про електронний цифровий підпис»[29] електронний цифровий підпис(ЕЦП) - вид електронного підпису, отриманого за результатом криптографічного перетворення набору електронних даних, який додається до цього набору або логічно з ним поєднується і дає змогу підтвердити його цілісність та ідентифікувати підписувача. Електронний цифровий підпис накладається за допомогою особистого ключа та перевіряється за допомогою відкритого ключа.

Електронний підпис призначений для ідентифікації особи, яка підписала електронний документ і є повноцінною заміною (аналогом) власноручного підпису у випадках, передбачених законом.

Використання електронного підпису дозволяє здійснити:

1) Контроль цілісності переданого документа: при будь-якому випадковому або навмисному зміні документа підпис стане недійсним, тому що обчислена вона на підставі вихідного стану документа і відповідає лише йому.

2) Захист від змін (підроблення) документа: гарантія виявлення підробки при контролі цілісності робить підроблення недоцільним у більшості випадків.
3) Неможливість відмови від авторства. Так як створити коректну підпис можна, лише знаючи закритий ключ, а він повинен бути відомим тільки власнику, то власник не може відмовитися від свого підпису під документом.
4) Доказове підтвердження авторства документа: Так як створити коректну підпис можна, лише знаючи закритий ключ, а він повинен бути відомим тільки власнику, то власник пари ключів може довести своє авторство підпису під документом. Залежно від деталей визначення документа можуть бути підписані такі поля, як "автор", "внесені зміни", "мітка часу" і т. д.

Існує кілька схем побудови цифрового підпису:

1) На основі алгоритмів симетричного шифрування. Дана схема передбачає наявність у системі третьої особи - арбітра, що користується довірою обох сторін. Авторизацією документа є сам факт зашифрування його секретним ключем і передача його арбітру[30].
2) На основі алгоритмів асиметричного шифрування. На даний момент такі схеми ЕП найбільш поширені і знаходять широке застосування[30].

Симетричні схеми ЕП менш поширені ніж асиметричні, так як після появи концепції цифрового підпису не вдалося реалізувати ефективні алгоритми підпису, засновані на відомих у той час симетричних шифрах. Асиметричні схеми цифрового підпису спираються на складні обчислювальні завдання, складність яких ще не доведена, тому неможливо визначити, чи будуть ці схеми зламані найближчим часом, як це сталося зі схемою, заснованої на задачі про укладання ранця. Також для збільшення криптостійкості потрібно збільшувати довжину ключів, що приводить до необхідності переписувати програми, що реалізують асиметричні схеми, і в деяких випадках перепроектувати апаратуру. Симетричні схеми засновані на добре вивчених блокових шифрах.

Однак у симетричних ЕП є і ряд недоліків:

1) потрібно підписувати окремо кожен біт інформації, що передається, що призводить до значного збільшення підпису. Підпис може перевершувати повідомлення за розміром на два порядки;

2) згенеровані для підпису ключі можуть бути використані тільки один раз, так як після підписування розкривається половина секретного ключа.

Таким чином використання симетричних алгоритмів у генерації ЕЦП є не доцільним, як рішення, що може бути прийнятим, через низку додаткових операцій, що будуть вимагати як додаткових обчислювальних операцій, так і надсилання додаткових запитів для обміну новими ключовими парами.

Загальновизнана схема цифрового підпису охоплює три процесу[30] :

1) Генерація ключової пари. За допомогою алгоритму генерації ключа рівноймовірним чином з набору можливих закритих ключів вибирається закритий ключ, обчислюється відповідний йому відкритий ключ.
2) Формування підпису. Для заданого електронного документа за допомогою закритого ключа обчислюється підпис.
3) Перевірка (верифікація) підписи. Для даних документа та підпису за допомогою відкритого ключа визначається дійсність підпису.

Для того, щоб використання цифрового підпису мало сенс, необхідно виконання двох умов:

1) Верифікація підпису повинна проводитися відкритим ключем, відповідним саме тому закритому ключу, який використовувався при підписанні.
2) Без володіння закритим ключем має бути обчислювально складно створити легітимний цифровий підпис.

Алгоритм створення та перевірки електронного цифрового підпису у загальному вигляді має наступний вид(рис. 2.9).

Рисунок 2.9 Узагальнений алгоритм створення та перевірки цифрового підпису

Схеми електронного підпису можуть бути одноразовими та багаторазовими. В одноразових схемах після перевірки справжності підпису необхідно провести заміну ключів, багаторазових схемах це робити не потрібно. Слід зауважити, що хоча й в разі використання багаторазових схем електронного цифрового підпису, немає необхідності змінювати ключові пари, закритий ключ є найбільш слабким місцем системи, адже в разі його викрадення буде скомпрометований увесь алгоритм створення ЕЦП. Таким чином створені ключі повинні мати термін дії і бути замінені в разі його закінчення, та з метою попередження можливості викрадення секретного ключа. Адже рівень захищеності секретного ключа прямим чином залежить від рівня захищеності ЕОМ, на якій він зберігається.

У деяких випадках, таких як потокова передача даних, алгоритми ЕЦП можуть бути занадто повільними. Таким чином, у системах де є необхідність у постійній та стабільній передачі даних у заданих рамках часу – ЕЦП не є доцільним та найбільш ефективним рішенням.

Також потрібно відзначити, що електронний цифровий підпис доцільно використовувати не на самому електронному документі чи обсязі даних, а на його хешу, через те що зміст самих даних може змінюватися і підписання цих даних

цифровим підписом забезпечить лише неможливість відмови від авторства, а не цілісність інформації при її передачі

Отже, кожний із варіантів забезпечення цілісності інформації при передачі має свої недоліки та переваги. Але хешування переданих даних в рамках криптопроколу, що розроблюється має переваги над використанням ЕЦП:

1) Алгоритм хешування виконується швидше за створення ЕЦП, адже не тільки вимагає меншої кількості обчислювальних операцій, а й тому що ЕЦП передбачає його використання на хешу повідомлення. Таким чином використовуючи ЕЦП у будь-якому разі доведеться обчислювати хеш повідомлення, що передається;
2) Процес створення ЕЦП заснований на асиметричних алгоритмах шифрування даних виконується значно довше ніж аналогічний процес шифрування хешу за допомогою симетричних алгоритмів шифрування;
3) Використання багаторазової схеми з ЕЦП вимагає періодичного оновлення ключів та завантаження їх на обладнання лінійних та центральних пунктів «КАСКАД»;
4) Використання ЕЦП забезпечує лише контроль цілісності повідомлення, що відправляється, і ніяким чином не захищає саме повідомлення, тому потрібне його використання наряду із симетричним алгоритмом шифрування, для захисту даних, що передаються.

Таким чином, в рамках як обмежених обчислювальних можливостей, так і у часових рамках відправки повідомлення (6 секунд), алгоритм електронного цифрового підпису значно програє використанню хешування у парі з симетричним алгоритмом шифрування.

І зважаючи саме на недоліки використання ЕЦП з обчислювальної точки зору, хешування даних разом з їх наступним шифруванням симетричним алгоритмом, що використовується в протоколі є найбільш доцільним рішення щодо контролю цілісності даних, що надсилаються.

2.3.5 Обґрунтування вибору методу захисту від дублювань команд

Захист від дублювань команд повинен забезпечувати однозначну унікальність повідомлення, що передається. Задля того щоб ніяким чином повідомлення, яке

вже було відправлене, не могло бути відправлене повторно та сприйняте як таке, що ще не надходило.

Даний механізм має бути реалізований за допомогою мінімальних обчислювальних операцій, щоб ідентифікація проходила однозначно та якомога швидше.

Для реалізації такого механізму ідентифікації повідомлень на діючому обладнанні, може бути використано декілька рішень: використання мітки часу у повідомленні, використання унікального ідентифікатора повідомлення.

Використання мітки часу означає додавання до тіла повідомлення мітки часу у мілісекундах, як унікального ідентифікатора. Мітка часу повинна випереджати поточну дату на 6 секунд, що вкладається у рамки валідності терміну передачі повідомлення. Цілісність та конфіденційність мітки часу буде забезпечена завдяки іншим функціям криптопротоколу, таким як хешування та шифрування, отже немає потреби хвилюватися, що мітка часу може бути змінена у процесі передачі.

Процес перевірки мітки часу повинен складатися з двох етапів порівнянь. Першим етапом повинне бути зроблене порівняння мітки часу з часом прийняття повідомлення у мілісекундах. Якщо мітка часу випереджає поточну дату більш ніж на 6 секунд - це означає, що мітка часу не є дійсною, а повідомлення було підроблене стороннюю особою. І в разі якщо мітка часу є меншою за поточну дату більше ніж на 6 секунд, повідомлення повинно бути визначеним підробленим.

Використання унікального ідентифікатора повідомлення означає генерування випадкової строки, яка додається до повідомлення, як ідентифікатор, і відправляється разом з ним. Перевірка на боці отримувача означає порівняння ідентифікатора, що надійшов разом із повідомленням, з ідентифікаторами, що зберігаються безпосередньо на ЕОМ. Такий метод гарантує, що повідомлення не може бути визнане валідним двічі, адже усі ідентифікатори записуються на пристрої одержувача.

Таким чином, обидва методи не дають можливості стороннім особам перехоплювати повідомлення, та накопичувати їх у себе з метою відправлення великої кількості вже згенерованих команд до лінійних або центральних пунктів “КАСКАД”. Але варто звернути увагу на те, що в разі використання унікальних

ідентифікаторів - потрібно буде їх зберігати безпосередньо на пристрої, який приймає повідомлення. Таким чином, хоча це й є прийнятним в разі використання цього методу захисту від дублювань на стороні центральних пунктів системи, такий метод не може бути використаним на боці лінійних пунктів, у зв'язку з обмеженими не тільки обчислювальними, але й обмеженими можливостями пристрою пам'яті.

Отже, використання мітки часу, поряд із алгоритмами хешування та шифрування, є найбільш ефективним та підходящим під вимоги системи рішенням, що забезпечить захист від дублювань команд.

2.3.6 Обґрунтування вибору методів аутентифікації відправника та отримувача

Обраний алгоритм аутентифікації відправника та отримувача повинен однозначно гарантувати, що зв'язок який буде встановлений, встановлюється саме з тим пунктом який був обраний відправником. Алгоритм повинен бути стійким та достатньо швидким, щоб використовуватися на обладнанні лінійних пунктів з обмеженими обчислювальними можливостями.

Ще одним важливим моментом аутентифікації повинно бути повідомлення підтвердження аутентифікації, яке з метою недопущення перехвату та підміни повинне теж бути зашифроване обраним алгоритмом.

Таким чином аутентифікація обох сторін пройде максимально безпечно з точки зору конфіденційності даних, необхідних для аутентифікації обох сторін.

З метою недопущення перехвату повідомлення аутентифікації та наступного його використання через певний проміжок часу, задля отримання з'єднання з однією із сторін, повідомлення аутентифікації повинно містити не тільки пароль аутентифікації кожної із сторін, але й мітку часу, яка буде зашифрована разом із паролем. Це зробить кожний запит аутентифікації унікальним з точки зору отриманого шифру, адже часова мітка буде кожен раз різною і сторонні особи не зможуть підібрати повідомлення аутентифікації на основі аналізу вже перехоплених ними, адже шифр кожного разу матиме різний вигляд.

В якості симетричного алгоритму шифрування пропонується обрати алгоритм RSA, як такий, що надає можливість шифрувати дані з великим рівнем стійкості та

швидкодії, при використанні ключів довжиною від 1024 до 2048 біт. Особливо, беручи до уваги той факт, що на даний момент, навіть використовуючи комп'ютерне обладнання з великими обчислювальними можливостями, злам зашифрованого повідомлення може займати тисячі років.

Алгоритм шифрування є математичною процедурою або набором кроків для кодування даних. RSA є найбільш широко використовуваним алгоритмом шифрування сьогодні. RSA відноситься до так званих асиметричних алгоритмів, у яких ключ шифрування не співпадає з ключем дешифрування. Один із ключів доступний усім (так робиться спеціально) і називається відкритим ключем, інший зберігається тільки у його господаря та невідомий нікому іншому. За допомогою одного ключа можна виконувати операції лише в один бік. Якщо повідомлення зашифровано за допомогою одного ключа, розшифрувати його можна тільки за допомогою іншого. Маючи один із ключів неможливо (дуже складно) знайти інший ключ, якщо розрядність ключа висока.

В основі RSA лежить завдання факторизації добутку двох простих великих чисел. Для шифрування використовується проста операція зведення в ступінь за модулем N. Для розшифрування необхідно обчислити функцію Ейлера від числа N, для цього необхідно знати розкладання числа n на прості множники (У цьому і полягає задача факторизації). У RSA відкритий і закритий ключ складається з кількох чисел. Закритий ключ зберігається в секреті, а відкритий ключ повідомляється іншому учаснику або десь публікується[31].

Ключ генерується за наступною схемою:

1) вибираються два простих числа p і q (такі, що p нерівно q);
2) обчислюється модуль N = p *q;
3) обчислюється значення функції Ейлер від модуля N: F(N)=(p−1)(q−1);
4) вибирається число e, зване відкритої експонентою, число e має лежати в інтервалі від 1 до F(N). Обчислюється число d, зване секретною експонентою, таке, що $d*e=1(\mathrm{mod}F(N))$, тобто є мультиплікативно зворотне до e за модулем F(N).

Задля шифрування відкрите повідомлення M розбивається на блоки, де кожен блок є меншим ніж модуль N. Після розбивання повідомлення, кожний блок шифрується за наступною формулою:

$$C = M^e \pmod N$$

Для розшифрування використовується секретний елемент шифру d, яким здійснюється розшифрування зашифрованого повідомлення C за формулою:

$$M = C^d \pmod N$$

Таким чином здійснюються операції шифрування та розшифрування за допомогою алгоримту RSA.

Слід зазначити, що факторизація N призведе до розкриття алгоритму RSA. Не було доведено, що немає поліноміального за витратами часу вирішення задачі знаходження простих факторів великої кількості (тобто можливо, що в майбутньому буде знайдено алгоритм, який факторизує N досить швидко для розтину RSA). Однак, незважаючи на постійне просування в алгоритмах факторизації, жоден з них не відповідає критерію поліноміальності за часом, що зробило б проблему RSA вирішуваною.

Важливо відзначити, що не було доведено, що безпека RSA цілком залежить від проблеми знаходження простих факторів великої кількості. Однак всі інші способи знаходження D по заданому E виявилися еквівалентними за витратами задачі факторизації N. Є ймовірність, що буде знайдений алгоритм для знаходження E-того кореня за модулем N простішим шляхом, ніж факторизація N. Проте, досі не був знайдено такий алгоритм і RSA витримав величезну кількість спроб розтину.

2.4 Висновок

В ході аналізу вимог, сформульованих необхідністю створення криптографічного протоколу з заданими функціями забезпечення безпеки, були розглянуті різноманітні рішення щодо забезпечення функцій протоколу. Таким чином були розглянуті вимоги до забезпечення аутентифікації, шифрування даних, забезпечення їх цілісності та захисту від дублювань.

Для реалізації функції аутентифікації був проаналізований та використаний асиметричний алгоритм шифрування RSA, який забезпечує високий рівень

захищеності даних, необхідних для аутентифікації, при передачі незахищеними каналами.

В якості рішення щодо шифрування даних, були проаналізовані різні симетричні алгоритми шифрування, такі як AES, ДСТУ 7624:2014 «Калина» та ДСТУ 28147:2009. В результаті аналізу було прийняте рішення використовувати шифр «Калина», як такий що зможе забезпечити високий рівень конфіденційності даних, та адаптований для виконання на обладнанні з обмеженими обчислювальними можливостями.

Задля забезпечення цілісності даних були проаналізовані два можливих методи, а саме, хешування даних та електронний цифровий підпис. Виходячи з обчислювальних можливостей обладнання пунктів «КАСКАД», та необхідності додатково обслуговувати їх в разі використання ЕЦП, вибір був зроблений на користь алгоритмів хешування даних, без додаткового їх підпису. В якості алгоритму хешування був обраний MD4, як такий що показує високий рівень хешування навіть на системах з 32-бітним процесором.

Захист від дублювань був розглянутий за допомогою використання унікальних ідентифікаторів у повідомленні, таких як мітка часу та випадковим чином згенерована строка. Обидва рішення не вимагають значних обчислювальних можливостей обладнання, але рішення все ж таки було прийняте на користь мітки часу. Через те, що при використанні випадкової строки, потрібно зберігати усі ідентифікатори у системі, задля подальшого їх порівняння на предмет співпадання. А в рамках невеликої кількості пам'яті лінійних пунктів «КАСКАД» це не є прийнятним рішенням.

В результаті аналізу та вибору методів, якими будуть реалізовані функції криптографічного протоколу, результат має вигляд наведений у табл. 2.4:

Таблиця 2.4 Розроблений криптографічний протокол

Криптографічний протокол	
Функція	Реалізація
Аутентифікація	Пароль із міткою часу, шифрований алгоритмом RSA
Шифрування	Алгоритм ДСТУ 7624:2014 «Калина»
Забезпечення цілісності	Хешування повідомлення за допомогою MD4
Захист від дублювань	Використання мітки часу у заданих рамках

Таким чином, розроблений криптографічний протокол захищає інформацію в незахищених каналах зв'язку від порушення її цілісності та доступності. І задовольняє усі вимоги, що стосуються технічних обмежень при реалізації криптографічного протоколу.

3 ЕКОНОМІЧНІЙ РОЗДІЛ

Економічно доцільним вважається, якщо витрати на забезпечення інформаційної безпеки не перевищують збиток при реалізації можливої загрози. Тому ,для обґрунтування економічної доцільності розробки та впровадження нового криптографічного протоколу, який забезпечить необхідний рівень захисту від загроз , необхідно розрахувати збитки від реалізації можливих загроз і порівняти їх з витратами на розробку та впровадження нового криптографічного протоколу відповідно.

3.1 Визначення трудомісткості розробки криптографічного протоколу

Нормування праці в процесі розробки нового криптографічного алгоритму істотно ускладнено через творчий характер праці спеціалістів з інформаційної безпеки. Проте трудомісткість розробки криптографічного алгоритму може бути розрахована на основі трудомісткості робіт, які виконуються.

Трудомісткість створення ПЗ визначається тривалістю кожної робочої операції, починаючи з складання технічного завдання і закінчуючи оформленням документації (за умови роботи одного програміста):

$$t = t_{тз} + t_в + t_а + t_{пр} + t_{опр} + t_д \text{ ,годин , (3.1)}$$

де $t_{тз}$ – тривалість складання технічного завдання на розробку ПЗ;

$t_в$ – тривалість визначення ТЗ літературних джерел за темою тощо;

$t_а$ – тривалість розробки блок-схеми алгоритму;

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

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

$t_д$ –тривалість підготовки технічної документації на ПЗ.

Складові трудомісткості визначаються на підставі умовної кількості операторів у програмному продукті Q (з урахуванням можливих уточнень у процесі роботи над алгоритмом і програмою).

Умовна кількість операторів у програмі:

$$Q = q * c (1 + p) \text{ , штук, (3.2)}$$

де q – очікувана кількість операторів;

c – коефіцієнт складності програми;

p – коефіцієнт корекції програми в процесі її опрацювання.

$$Q = 2500 * 1,5 * (1 + 0,1) = 4112$$

Коефіцієнт складності програми *c* визначає відносну складність програми щодо типового завдання, складність якого дорівнює одиниці.

Діапазон його зміни – 1,25…2,0.

Коефіцієнт корекції програми *p* визначає збільшення обсягу робіт за рахунок внесення змін в алгоритм або програму внаслідок уточнення технічного завдання. Його величина знаходиться в межах 0,05…0,1, що відповідає внесенню 3…5 корекції і переробці 5-10% готової програми.

Оцінка тривалості складання технічного завдання на розробку ПЗ $t_{тз}$ залежить від конкретних умов і визначається дипломником на підставі експертних оцінок за узгодженням із керівником проекту.

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

$$t_в = Q * B / (75…85) * k \text{ , годин, } (3.3)$$

Де B – коефіцієнт збільшення тривалості етапу внаслідок недостатнього опису завдання, B = 1,2…1,5 ;

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

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

$$t_в = 4112 * 1,3 / (80 * 1,1) = 60.75 \text{ годин}$$

Тривалість розробки блок-схеми :

$$ta = Q / (20…25) * k, \text{ годин. } (3.4)$$

$$t_a = 4112 / (22 * 1,1) = 104 \text{ годин}$$

Тривалість складання програми за готовою блок-схемою:

$$t_{пр} = Q / (20…25) * k, \text{ годин. } (3.5)$$

$$t_{пр} = 16500 / (22 * 1,1) = 104 \text{ годин}$$

Тривалість опрацювання програми ПК:

$$t_{опр} = 1,5Q / (4…5) * k \text{ , годин. } (3.6)$$

$$t_{пр} = 16500 * 1,5 / (4 * 1,1) = 1405 \text{ годин}$$

Тривалість підготовки технічної документації на ПЗ :

$$t_{д} = Q / (15…20) * k + Q / (15…20) * 0,75 \quad (3.7)$$

$$t_{пр} = 16500 / (17 * 1,1) + 16500 / 17 * 0,75 = 402 \text{ годин}$$

$$t = 60.75 + 104 + 104 + 1405 + 402 = 2076 \text{ годин}$$

3.2 Розрахунок витрат на впровадження криптографічного протоколу

Витрати на розробку нового криптографічного протоколу $К_{рп}$ складаються з витрат на заробітну плату спеціалістів з інформаційної безпеки $З_{зп}$ і вартості витрат машинного часу, що необхідний для розробки криптографічного протоколу $З_{мч}$[19]:

$$К_{рп} = З_{зп} + З_{мч} \quad (3.2)$$

Заробітна плата одного виконавця враховує основну і додаткову заробітну плату, а також відрахування на соціальні потреби(пенсійні відрахування , страхування на випадок безробіття, соціальне страхування тощо) та визначається за формулою[19]:

$$З_{зп} = t * З_{іб} \text{ , грн, } (3.3)$$

де t – загальна тривалість розробки криптографічного протоколу, годин;

$З_{іб}$ – средньогодинна заробітна плата спеціаліста з інформаційної безпеки з нарахування, грн/годину.

$$З_{зп} = 2076 * 300 = 622\,800 \text{ грн.}$$

Вартість машинного часу для розробки криптографічного протоколу на ПК визначається за формулою[19]:

$$З_{мч} = t * С_{мч}, \text{ грн, } (3.8)$$

де – t трудомісткість розробки криптографічного протоколу на ПК, годин;

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

Вартість 1 години машинного часу ПК визначається за формулою[19]:

$$С_{мч} = Р * t_{нал} * С_{е} * (Ф_{зал} * Н_{а}) / F_{р} + (К_{ппз} * Н_{апз}) / F_{р}, \text{ грн, } (3.9)$$

де Р – встановлена потужність ПК, кВт;

$С_е$ – тариф на електричну енергію, грн/кВт*година;

$Ф_{зал}$ – залишкова вартість ПК на поточний рік;

$Н_а$ – річна норма амортизації на ПК, частки одиниці;

$Н_{апз}$ – річна норма амортизації на ліцензійне програмне забезпечення, частки одиниці;

$К_{лпз}$ – вартість ліцензійного програмного забезпечення;

$F_р$ – річний фонд робочого часу (за 40-годинного робочого тижня $F_р = 1920$).

Залишкова вартість ПК визначається виходячи з терміну його експлуатації як різниця між первісною вартістю та зносом за час використання.

$С_{мч} = 1 * 1,68 + (65000 * 0,125) / 1920 + (7000 * 0,125) / 1920 = 6,36$ грн

Таким чином вартість машинного часу для розробки криптографічного протоколу складає:

$$З_{мч} = 2076 * 6,36 = 13203 \text{ грн.}$$

Витрати на розробку нового криптографічного протоколу, враховуючи той факт, що розробкою займатимуться 3 спеціалісти складатимуть:

$$К_{рп} = 622800 + 13203 = 636\,003 \text{ грн.}$$

Визначена таким чином вартість розробки криптографічного протоколу є частиною одноразових капітальних витрат разом з витратами на придбання та та налагодження апаратури необхідної для розробки нового криптографічного протоколу.

Таким чином, капітальні(фіксовані) витрати на проектування та впровадження проектного варіанта нового криптографічного протоколу складають[19]:

$$К = К_{рп} + К_{зпз} + К_{аз} + К_н + К_{пр}, \quad (3.10)$$

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

$К_{рп}$ – вартість розробки криптографічного протоколу;

$К_{зпз}$ – вартість закупівель ліцензійного основного та додаткового програмного забезпечення, тис.грн;

$К_{аз}$ – вартість закупівлі апаратного забезпечення та допоміжних матеріалів, тис.грн;

$К_н$ – витрати на налагодження системи інформаційної безпеки, тис.грн.

Вартість закупівлі апаратного забезпечення та допоміжних матеріалів для розробки рішення з інформаційної безпеки:

$$К_{аз} = 75000 * 3 + 500 * 3 = 226,5 \text{ тис.грн.}$$

Вартість закупівель ліцензійного основного та додаткового програмного забезпечення:

$$К_{зпз} = 7000 * 3 + 500 * 3 = 22,5 \text{ тис.грн.}$$

Витрати на налагодження системи інформаційної безпеки:

$$К_{н} = 25000 * 3 = 75 \text{ тис.грн.}$$

Таким чином, витрати на проектування та впровадження проектного варіанта нового криптографічного протоколу:

$$К = 636003 + 226500 + 22500 + 75000 + 60000 = 1020003 \text{ грн.}$$

3.3 Розрахунок збитків через відсутність систем захисту каналів передачі інформації

Для розрахунку збитку від реалізації загроз спричинених відсутністю систем захисту каналів передачі інформації в частині МСДЦ «КАСКАД» необхідно визначити наслідки реалізації цих загроз. До наслідків відносяться:

- зменшення кількості клієнтів;
- необхідність відшкодування збитків нанесених клієнту;
- затримки в роботі залізниці.

Вхідні дані:

- середня вартість вантажу одного вагону, $Q_{ваг}$;
- кількість вагонів в одному складі, $N_{ваг}$;
- відсоток втрати клієнтів, $p_{клієнт}$;
- обсяг прибутку від одного клієнта, $Q_{клієнт}$;
- кількість клієнтів за день, $N_{клієнт}$;
- час на відновлення втрачених клієнтів, $t_{віднов}$;

Вхідні дані представлені у таблиці 3.1

Таблиця 3.1

Пояснення	Умовне позначення	Значення

час відновлення після атаки	$t_{віднов}$	14 днів
середня вартість вантажу одного вагону	$Q_{ваг}$	100000 грн
кількість вагонів в одному складі	$N_{ваг}$	60
відсоток втрати клієнтів	$p_{клієнт}$	5%
обсяг прибутку від одного клієнта за день	$Q_{клієнт}$	300000 грн
кількість клієнтів за день	$N_{клієнт}$	300

Вартість втрат від розголошення інформації, що становить комерційну таємницю розраховується по формулі:

$$S_{втрат} = U + N_{ваг} * Q_{ваг} * 0,05 \qquad (3.11)$$

Враховуючи кількість клієнтів, що обслуговуються ПАТ «Укрзалізниця» за день, $N_{клієнт} = 300$, розголошення даних може спричинити відтік клієнтів у розмірі 5%, таким чином, враховуючи обсяг прибутку від одного клієнта, можлива втрата грошового прибутку складе:

$$U = Q_{клієнт} * N_{клієнт} * p_{клієнт} * t_{відновл} \quad (3.12)$$

U = 300000 * 300 * 0,05 * 14 = 63000000 грн, де U – можлива втрата прибутку за тиждень внаслідок відтоку клієнтів.

Підставивши вхідні дані до формули розрахунку вартості втрат отримаємо можливі збитки в разі витоку інформації, що становить комерційну таємницю:

$$S_{втрат} = 300000 * 300 * 0,05 * 14 + 60* 100000 * 0,1= 63600000 \text{ грн.}$$

3.4 Показники економічної ефективності

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

Щодо до інформаційної безпеки говорять не про прибуток, а про запобігання можливих втрат від порушення цілісності та доступності інформації, що потребує захисту,а отже:

$$ROSI = E/K, \text{ частки одиниці,} \quad (3.13)$$

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

E – загальний ефект від впровадження системи інформаційної безпеки, тис. грн. :

$$E = B*R - C, \quad (3.14)$$

де B – загальний збиток від порушення цілісності та доступності інформації, що потребує захисту, тис.грн;

R – очікувана ймовірність порушення цілісності та доступності інформації, що потребує захисту, частки одиниці;

C – щорічні витрати на експлуатацію системи інформаційної безпеки, тис.грн.

Щорічні витрати на експлуатацію системи інформаційної безпеки дорівнюються нулю C = 0. Це обумовлено, перш за все тим, що рішення, яке впроваджується, стосується суто програмного забезпечення і не потребує ніяких додаткових вкладень впродовж періоду експлуатації. Також, слід зазначити, що відсутність додаткових витрат на обладнання, курсів підвищення кваліфікації персоналу та будь-яких інших поточних витрат – є однією з основних вимог до рішення, що впроваджується. Таким чином, для рішення яке пропонується для впровадження, не потрібні додаткові витрати на експлуатацію.

Оцінивши можливості порушника втрутитися в передачу інформації виділеними лініями зв'язку, та ймовірність зміни інформації при її передачі в цілях нанесення збитку, за коефіцієнт ймовірності порушення доступності та цілісності інформації, що потребує захисту, прийнято R = 3%.

Таким чином:

$$E = 63600000 * 0{,}03 = 1908000 \text{ грн}$$

$$ROSI = 1908000 / 1020003 = 1{,}870$$

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

$$T_o = K / E = 1 / ROSI, \text{ років} \quad (3.15)$$

Отже, термін окупності капітальних інвестицій дорівнює:

$$T_o = 1 / 1,870 = 0,534 \text{ року}$$

3.5 Висновок

Розрахувавши збитки від реалізації можливих атак, які склали 63600000 грн., і порівнявши їх з витратами на розробку та впровадження системи захисту інформації комп'ютерної мережі 1023000 грн, та прийнявши до уваги загальний ефект від впровадження системи інформаційної безпеки 1908000 грн, можна зробити висновок, що система окупиться приблизно за півроку, таким чином впровадження засобів захисту є економічно доцільним. Термін окупності – приблизно півроку (6 місяців).

ВИСНОВКИ

У ході виконання дипломної роботи були вирішені наступні питання: проведений автоматизований системи керування технологічними процесами об'єкта критичної інфраструктури, його мережевої та фізичної структури, на основі аналізу каналів передачі інформації між частинами системи, а саме, лінійними та центральними пунктами, виявлені вразливості системи, через які можуть бути реалізовані загрози цілісності та доступності інформації. Були розроблені вимоги до розробки методів захисту каналів зв'язку частини системи. На основі вимог, що висунуті до методів захисту каналів передачі інформації, був розроблений криптографічний протокол. Враховуючи обмеження системи та характеристики різних алгоритмів, проведена порівняльна характеристика технічних рішень та обрані найбільш ефективні з пропонованих програмних засобів захисту каналів передачі інформації для частини системи мікропроцесорної диспетчерської централізації

В економічному розділі виконаний розрахунок витрат на впровадження даних засобів захисту та показники економічної ефективності від впровадження запропонованого рішення.

Практичне значення даного дипломного проекту полягає у забезпечені необхідного рівня захищеності від загроз цілісності та доступності у каналах передачі інформації, з детальною розробкою частин криптографічного протоколу, що був запропонований для впровадження в системі диспетчерської централізації «КАСКАД».

ПЕРЕЛІК ПОСИЛАНЬ

1. NIST Special Publication 800-82: Guide to Industrial Control Systems (ICS) Security – Mineola, 200 – 171 с.
2. Відомості про атаки на АСК ТП у світі (Електрон.ресурс) Спосіб доступу URL: https://smb.ixbt.com/articles/tehnologii-i-produkty/2019/05/28/ataki-na-asu-tp-i-iiot-prodolzhajut-nabirat-silu.html
3. Відомості про атаки на АСК ТП у світі (Електрон.ресурс) Спосіб доступу URL: https://www.securitylab.ru/blog/company/AngaraTech/347875.php
4. Кібератаки на енергетичні компанії України Спосіб доступу URL: https://texty.org.ua/articles/66125/Hakerska_ataka_Rosiji_na_ukrajinsku_jenergosystemu_jak-66125/
5. Мікропроцесорна диспетчерська централізація “КАСКАД” /М.І. Данько, В.І.Мойсеєнко, В.З. Рахматов, В.І. Троценко, М.М. Чепцов: Навч. посібник. – Харків, 2005. – 176 с.
6. Диспетчерське керування рухом поїздів на швидкісних та високошвидкісних магістралях: Навч. посібник /С. В. Панченко, Т. В. Бутько, А. В. Прохорченко та ін. - Харків: УкрДУЗТ, 2019. – 153 с.,
7. Постанова Кабінету Міністрів України від 04.03.2015 р. №83 Спосіб доступу URL: https://ips.ligazakon.net/document/view/KP150083?an=1
8. Закон України «Про критичну інфраструктуру» Спосіб доступу URL: https://zakon.rada.gov.ua/laws/show/1882-20#Text
9. Нормативні системи міжнародного електротехнічного стандарту IEC297 (Електрон. ресурс) Спосіб доступу URL: http://dspace.nbuv.gov.ua/bitstream/handle/123456789/122719/04-Krivenko.pdf?sequence=1

10. Технології захисту інформації / Ю. А. Тарнавський; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2018. – 162 с. (Електрон. ресурс) Спосіб доступу URL: https://ela.kpi.ua/bitstream/123456789/23896/1/TZI_book.pdf

11. Шифрування: типи і алгоритми(Електрон. ресурс) Спосіб доступу URL: https://hostpro.ua/wiki/ua/security/encryption-types-algorithms

12. Асиметричне шифрування (Електрон. ресурс) Спосіб доступу URL: https://uk.education-wiki.com/5679799-asymmetric-encryption

13. Основи КЗІ / Г.М.Гулак,Г.М. Мучачев: Навч. посібник. – Київ, 2011. – 197 с.

14. Криптографічні системи та протоколи : навч. посіб. [для студ. баз. напряму 6.170101 "Безпека інформ. і комунікац. систем" усіх форм навчання] / А. Е. Лагун ; М-во освіти і науки України, Нац. ун-т "Львів. політехніка". – Л. : Вид-во Львів. політехніки, 2013. – 96 с. : іл. – Бібліогр.: с. 95

15. SSH протокол (Електрон. Ресурс) Спосіб доступу URL: https://freehost.com.ua/faq/wiki/chto-takoe-ssh/

16. Обґрунтування вимог, побудування та аналіз перспективних симетричних криптоперетворень на основі блочних шифрів / [О. О. Кузнецов, Р. В. Олійников, Горбенко Ю. І. та ін.] // Вісн. Нац. ун-ту "Львів. політехніка". – 2014. – № 806. – С. 124-140.

17. AES (Електрон. Ресурс) Спосіб доступу URL: https://www.geeksforgeeks.org/advanced-encryption-standard-aes/

18. ДСТУ ГОСТ 28147 – 2009. Система обробки інформації. Криптографічний захист інформації. Алгоритм криптографічного перетворення.

19. Методичні вказівки до виконання економічної частини дипломного проекту фахівця за спеціальності 125 Кібербезпека / Упорядн.: І.В. Шереметьєва, Д.П. Пілова, Н.М.Романюк – Дніпро: Національний технічний університет «Дніпровська політехніка», 2017 – 17с.

20. ДСТУ 7624:2014. Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення. [Текст]. – Введ. 01–07–2015. – К.: Мінекономрозвитку України, 2015.

21. A New Encryption Standard of Ukraine: The Kalyna Block Cipher / [R. Oliynykov, I. Gorbenko, O. Kazymyrov et al] // Norwegian Information Security Conference (NISK-2015). – 2015. – 113 p.

22. ЕФЕКТИВНА РЕАЛІЗАЦІЯ АЛГОРИТМУ БЛОКОВОГО СИМЕТРИЧНОГО ШИФРУВАННЯ ДСТУ 7624:2014 («КАЛИНА») ДЛЯ 8/16/32-БІТОВИХ ВБУДОВАНИХ СИСТЕМ: Стаття /Я.Р. Совин, В.І. Отенко, Є.Ф. Штефанюк – 16 с.

23. Принципи побудови і основні властивості нового національного стандарту блокового шифрування України / [Р. Олійников, І. Горбенко, О. Казимиров та ін.] // Захист інформації. – 2015. – № 2(17). – С. 142-157.

24. R. Oliynykov, O. Kazymyrov, O. Kachko et al. Source code for performance estimation of 64-bit optimized implementation of the block ciphers Kalyna, AES, GOST, BelT, Kuznyechik. 2015 [Electronic]

25. Хешування і захист інформації (Електрон ресурс) Спосіб доступу URL: http://isearch.kiev.ua/ru/searchpracticeru/-internetsecurity-ru/837-hashing-message-digest

26. МЕТОДОЛОГІЧНИЙ ВИБІР ХЕШ-ФУНКЦІЇ ДЛЯ ОПТИМАЛЬНОЇ СИНХРОНІЗАЦІЇ БАЗИ ДАНИХ НАЦІОНАЛЬНИХ ПАРКІВ ХОРВАТІЇ: Стаття / Борд Р.В., Лозовська Л.І. – 5 с.

27. Кормен Т. Алгоритмы: построение и анализ. / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – М.: МЦНМО, 2001

28. Алгоритм MD4 (Електрон. ресурс) Спосіб доступу URL: https://znaimo.com.ua/MD4

29. Закон України «Про критичну інфраструктуру» Спосіб доступу URL: https://zakon.rada.gov.ua/laws/show/1882-20#Text

30. Електронний цифровий підпис (Електрон. ресурс) Спосіб доступу URL: https://znaimo.com.ua/%D0%95%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B8%D0%B9_%D1%86%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B8%D0%B9_%D0%BF%D1%96%D0%B4%D0%BF%D0%B8%D1%81#link63

31. Алгоритм RSA (Електрон. ресурс) Спосіб доступу URL: https://proverkassl.com/book_algoritm_rsa.html

ДОДАТОК А. Відомість матеріалів кваліфікаційного проекту

Найменування	Кількість аркушів	Примітка
Реферат	3	
Список умовних скорочень	1	
Зміст	3	
Вступ	3	
Стан питання. Постановка задачі	30	
Спеціальна частина	47	
Економічний розділ	7	
Висновки	1	
Перелік посилань	3	
Додаток А. Відомість матеріалів дипломного проекту	1	
Додаток Б. Перелік матеріалів на оптичному носії	1	
Додаток В. Відгуки керівників розділів	3	

ДОДАТОК Б. Перелік матеріалів на оптичному носії

1) Пояснювальна_записка_ГаржаМ.С.docx
2) Пояснювальна_записка_ГаржаМ.С.pdf
3) Презентація_ГаржаМ.С.ppt

Відгук

на кваліфікаційну роботу магістра на тему:
«Методи захисту інформації в каналах зв'язку системи диспетчерської централізації "КАСКАД"»
студента групи 125м-20-1
Гаржі Микити Сергійовича

Мета роботи – забезпечення заданого рівня безпеки інформації, що передається по незахищеним каналам зв'язку в мікропроцесорній системи диспетчерської централізації «КАСКАД».

Тема роботи безпосередньо пов'язана з об'єктом діяльності фахівця за спеціальністю 125 Кібербезпека – впровадження та використання методів та засобів криптографічного захисту інформації в інформаційних системах об'єктів критичної інфраструктури.

Задачі роботи (аналіз автоматизованої системи керування технологічними процесами (АСК ТП), аналіз актуальних загроз для інформації в АСК ТП, формування вимог до розробки, обґрунтування вибору криптографічних алгоритмів та проколів, адаптація криптографічних алгоритмів до обчислювальних можливостей апаратного забезпечення АСК ТП) віднесені в освітньо-кваліфікаційній характеристиці магістра до класу евристичних, вирішення яких основане на знаково-розумових уміннях фахівця.

Оригінальність технічних рішень полягає у розробці протоколу для каналів зв'язку між лінійними та центральним пунктами системи диспетчерської централізації.

Практичне значення результатів проектування полягає в можливості реалізації запропонованих рішень без необхідності значних змін у існуючій системі.

До недоліків дипломної роботи відносяться:

– недостатньо структуровано опис актуальних загроз;

– недостатньо обґрунтовано формат даних, які передаються каналом зв'язку;

– надмірно докладний опис деяких стандартних алгоритмів;

– відсутність практичної перевірки ефективності запропонованих рішень.

Оформлення пояснювальної записки до дипломного проекту виконано з деякими відхиленнями від стандартів.

Рівень запозичень у кваліфікаційній роботі не перевищує вимог положення про систему виявлення та запобіганню плагіату.

Ступінь самостійності виконання дипломної роботи висока.

За час дипломування Гаржа М.С. виявив себе фахівцем, здатним самостійно, на достатньо високому рівні вирішувати поставлені задачі.

В цілому кваліфікаційна робота виконана у відповідності до вимог, що ставляться до кваліфікаційної роботи магістра, заслуговує оцінки “відмінно”, а Гаржа М.С. присвоєння йому кваліфікації магістр з кібербезпеки, освітньо-професійна програма «Кібербезпека».

Керівник спеціальної частини
дипломної роботи магістра,
старший викладач _____ О.В. Кручінін

Керівник дипломної
роботи магістра,
д.т.н., професор _____ В.І. Корнієнко

В І Д Г У К
КЕРІВНИКА ЕКОНОМІЧНОГО РОЗДІЛУ
на кваліфікаційну роботу студента групи 125м-20-1
Гаржі Микити Сергійовича

на тему: «Методи захисту каналів зв'язку системи диспетчерської централізації «КАСКАД»»

РЕЦЕНЗІЯ

на кваліфікаційну роботу магістра на тему:
“Методи захисту інформації в каналах зв’язку системи диспетчерської централізації «КАСКАД»”

студента групи 125м-20-1

Гаржі Микити Сергійовича

Кваліфікаційна робота за спеціальністю 125 «Кібербезпека» студента Гаржі М.С. представлена пояснювальною запискою на 101 стор., містить 15 рис., 5 табл., 3 додатки, 31 джерело.

Обрана тема є актуальною, оскільки існує необхідність забезпечення захисту інформації в інформаційно-телекомунікаційних системах об’єктів критичної інфраструктури.

Зміст та завдання кваліфікаційної роботи повністю відповідають темі.

Обрані рішення є ефективним з точки зору забезпечення заданого рівня безпеки інформації, що обробляється та з точки зору швидкодії в рамках обмежених обчислювальних можливостей.

Основними перевагами у роботі є:

- мінімізація затрат на забезпечення необхідного рівня безпеки інформації, що обробляється;
- відсутність необхідності внесення змін у технічне обладнання;
- відсутність необхідності проходження курсів підвищення кваліфікації для співробітників.

До недоліків роботи слід віднести:

- недостатня обґрунтованість вибору механізму аутентифікації;
- відсутність методик оцінки стійкості запропонованих алгоритмів шифрування.

Проте вказані недоліки не знижують загальної цінності роботи.

В цілому кваліфікаційна робота виконана у відповідності до вимог, які пред'являються до кваліфікаційної роботи магістра і заслуговує оцінки "_____", а Гаржа Микита Сергійович присвоєння йому кваліфікації магістра з кібербезпеки.

Рецензент:

_____ _____

(підпис)