**Міністерство освіти і науки України**
**Київський національний університет імені Тараса Шевченка**

**Факультет інформаційних технологій**
**Кафедра кібербезпеки та захисту інформації**

ДОПУСТИТИ ДО ЗАХИСТУ:
завідувачка кафедри кібербезпеки
та захисту інформації
_____Наталія ЛУКОВА-ЧУЙКО
«14» червня 2022р.

# ПОЯСНЮВАЛЬНА ЗАПИСКА
## випускної кваліфікаційної роботи
***бакалавра***
(назва освітнього ступеня)

**галузь знань** 12 Інформаційні технології
(шифр і назва галузі знань)
**спеціальність** 125 Кібербезпека
(код і назва спеціальності)
**освітня програма** «Кібербезпека»
(назва освітньої програми)

**на тему:** «Вразливості, шляхи та рекомендації щодо захисту серверів та ЦОД»

**Виконавець:** студентка IV курсу, групи КБ-41

_____ **Марина СПИРИДОНОВА**
(підпис) (ім'я прізвище)

| | **Прізвище, ініціали** | **Підпис** |
|---|---|---|
| **Керівник** | Яніна ШЕСТАК | |

| | | |
|---|---|---|
| **Нормоконтроль** | Юрій ЩЕБЛАНІН | |

**Київ 2022**

**Міністерство освіти і науки України**

**Київський національний університет імені Тараса Шевченка**

**Факультет інформаційних технологій**
**Кафедра кібербезпеки та захисту інформації**

**ЗАТВЕРДЖЕНО:**
завідувач кафедри кібербезпеки
та захисту інформації
_____Наталія ЛУКОВА-ЧУЙКО
«01» листопада 2021 р.

# ЗАВДАННЯ
## на виконання дипломної роботи (проєкту)

**спеціальності** 125 Кібербезпека
(код і назва спеціальності)
**освітньої програми** «Кібербезпека»
(назва освітньої програми)

**студентці** **КБ-41** **Спиридоновій Марині Максимівні**
(група) (прізвище ім'я по-батькові)
**Тема дипломної роботи** Вразливості, шляхи, та рекомендації щодо захисту серверів та ЦОД

**1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ**

Тема дипломної роботи затверджена на засіданні кафедри кібербезпеки та захисту інформації протокол № 5 від 29.10.2021р.

**2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ**

Microsoft Threat Modeling Tool, Adobe Photoshop, моделі вразливостей, методи та способи впровадження систем цілодобового моніторінгу та звітності, основні етапи для запобігання вразливостей.

**3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ**

Необхідно ознайомитись з нормативно-правовою базою у сфері захисту інформації, стандартами ISO та NIST. Створити рекомендаційний алгоритм для дій щодо запобігання вразливостей та впровадження звітності по моніторингу.

## 4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

**Практична цінність** Результати бакалаврської роботи можуть бути використані при розробці систем моніторингу фізичних параметрів приміщень не тільки для ЦОД, але і для систем інтелектуального приміщення.

## 5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 01 листопада 2021 року

Завдання видав _____ Яніна ШЕСТАК
(підпис) (ініціали, прізвище)

Завдання прийняла до виконання _____ Марина СПИРИДОНОВА
(підпис) (ініціали, прізвище)

**КАЛЕНДАРНИЙ ПЛАН**

| № п/п | Найменування етапів робіт | Строки виконання робіт (початок-кінець) | Відмітка про виконання |
|---|---|---|---|
| 1 | Уточнення постановки задачі | 01.11.2021 - 27.01.2022 | *виконано* |
| 2 | Аналіз літератури | 28.01.2022 – 11.02.2022 | *виконано* |
| 3 | Аналіз основних відомостей про структуру серверу та ЦОД | 12.02.2022 – 24.02.2022 | *виконано* |
| 4 | Збір даних про основні вразливості серверів та ЦОД | 25.02.2022 – 24.03.2022 | *виконано* |
| 5 | Дослілдення вразливостей веб-серверів | 25.03.2022 – 07.04.2022 | *виконано* |
| 6 | Аналіз способів захисту веб-серверів та ЦОД | 08.04.2022 – 20.04.2022 | *виконано* |
| 7 | Дослідження методів тестування ІБ | 21.04.2022 – 05.05.2022 | *виконано* |
| 8 | Створення рекомендаційного алгоритму для захисту | 06.05.2022 – 01.06.2022 | *виконано* |
| 9 | Оформлення пояснювальної записки | 02.06.2022 – 06.06.2022 | *виконано* |
| 10 | Підготовка до захисту дипломної роботиВ | 07.06.2022 – 13.06.2022 | *виконано* |

Студент-дипломник _____ Марина СПИРИДОНОВА
(підпис) (ініціали, прізвище)

Керівник випускної кваліфікаційної роботи _____ Яніна ШЕСТАК
(підпис) (ініціали, прізвище)

Термін подання дипломної роботи до ЕК 06 червня 2022 року

# РЕФЕРАТ

Пояснювальна записка до дипломної роботи «Вразливості, шляхи та рекомендації щодо захисту серверів та ЦОД» складається зі вступу, основної частини, що містить 3 розділи, висновків і списку літератури та джерел. Загальний обсяг роботи – 62 сторінок. Робота містить 1 рисунок. Список використаних джерел включає 34 джерела.

**Метою дослідження** є розробка алгоритму для впровадження найкращих засобів цілодобового моніторингу поточного стану фізичного стану серверної кімнати ЦОД та засобів оповіщення про критичні ситуації.

Для досягнення зазначеної мети поставлено наступні завдання:

- дослідити структуру серверів та ЦОД
- провести аналіз найбільш поширених вразливостей, характерних для визначеної структури
- побудувати алгоритм з рекомендаціями щодо запобігання або швидкого вирішення загроз

**Об’єктом дослідження** є процесс захисту серверів та ЦОД від вразливостей.

**Предметом дослідже***ння* є методи дослідження вразливостей, шляхів та рекомендацій щодо захисту серверів і ЦОД

**Практичне значення отриманих результатів.** Результати дипломної роботи можуть бути використані при розробці систем моніторингу фізичних параметрів приміщень не тільки для ЦОД, але і для систем інтелектуального приміщення загалом.

**Ключові слова:** сервер, центр обробки даних, вразливості системи, модель загроз, безпека серверів, безпека центру обробки даних, методи тестування, системи реагування.

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

| | | |
|---|---|---|
| CERT | – | спеціалізований структурний підрозділ Державного центру кіберзахисту |
| CERT –UA | – | команда реагування на комп’ютерні надзвичайні події в Україні |
| CSS | – | каскадні таблиці стилів |
| CVSS | – | загальна система оцінки вразливостей |
| FIRST | – | форум команд реагування на надзвичайні події |
| FTP | – | протокол передачі файлів по мережі |
| HTML | – | мова розмітки гіпертекстових документів |
| HTTP | – | протокол передачі даних |
| ID | – | унікальна ознака об'єкта |
| IP | – | інтернет протокол |
| ISSAF | – | структура інформаційних систем з оцінки безпеки |
| ISO | – | міжнародна організація, що займається випуском стандартів |
| ДСТУ | – | державний стандарт України |
| ДПД | – | Діаграма потоків даних |
| НД ТЗІ | – | нормативний документ системи технічного захисту інформації |
| ОС | – | операційна система |
| ПЗ | – | програмне забезпечення |
| СКБД | – | система керування базами даних |

# ЗМІСТ

# ВСТУП

У роботі здійснено дослідження нових загроз безпеці в публічних хмарах і центрах обробки даних. По-перше, ми представляємо нову вразливість, яка називається зменшеним резервуванням охолодження, яка може бути використана для серйозного погіршення теплових умов у центрі обробки даних. Ми систематично вивчаємо спричинені пошкодження та пропонуємо динамічне тепловирівнювання навантаження, щоб пом’якшити загрозу. Потім ми розкриваємо проблему витоку інформації в хмарних сервісах. Ми демонструємо, що наслідки таких витоків можуть призвести до порушення конфіденційності або навіть перебоїв у роботі служби, і впроваджуємо систему захисту для захисту хмарних контейнерів.

Актуальність теми. Оскільки хмарні обчислення стали основним напрямком надання IT-послуг, важливо досягти високої доступності та надійності для обчислювальних послуг, розміщених у хмарі. На жаль, системні збої та відключення електроенергії в центрі обробки даних не рідкість. Одна з причин полягає в тому, що центри обробки даних повинні значно розширювати свої масштаби, щоб задовольнити постійно зростаючі потреби в послугах. Перевищення підписки широко використовується для розгортання більшої кількості серверів, ніж може підтримувати інфраструктура. Така агресивна політика, хоча й заощаджує витрати на модернізацію об'єктів, вичерпує резерви та допуски на несподівані збої та ненормальні умови. Ще гірше те, що існування цих проблем відкриває двері для зловмисників і робить центри обробки даних уразливими до різних атак відмови в обслуговуванні (DoS). Щоб уникнути перерв у роботі служби, необхідно заздалегідь виявити потенційні вразливості та створити більш надійні механізми захисту.

# РОЗДІЛ 1
# ПОНЯТТЯ СЕРВЕР ТА ПОНЯТЯ ЦОД. ЗАГАЛЬНІ ВІДОМОСТІ ЩОДО ЇХ СТРУКТУРИ

## 1.1. Основні поняття серверів та їх класифікація

Сервер (з англ. server, обслуговуючий) - існує ряд визначень щодо поняття сервер, враховуючи особливості використання.

1. Сервер (мережа) – логічний або фізичний вузол мережі, що обслуговує запити до однієї адреси та/або доменного імені (суміжних доменних імен), що складається з одного або системи апаратних серверів, на яких запущена або система серверних програм [23].

2. Сервер (програмне забезпечення) – програмне забезпечення, яке отримує запити від клієнтів (в архітектурі клієнт-сервер).

3. Сервер (апаратне забезпечення) – комп’ютер (або спеціальне комп’ютерне обладнання), призначений та/або спеціалізований для виконання певних функцій Сервісу.

3. Сервер в інформаційних технологіях - програмна складова комп'ютерної системи, що виконує сервісні функції за запитом клієнта і надає йому доступ до певних ресурсів.

Актуальним є поняття серверна програма (сервер) на комп’ютері, також відома як «сервер», це поняття взаємопов’язано з топологіэю мережі, тобто такий вузол називають «сервером». Серверна програма може працювати на звичайній робочій станції, або як серверна програма, що виконується на серверному комп’ютері. Така програма може діяти як клієнт (тобто не сервер по відношенню до топології мережі).

Терміни сервер і клієнт та призначені їм ролі утворюють концепцію програмного забезпечення «клієнт-сервер».

Щоби взаємодіяти з клієнтом (або клієнтами, якщо одночасно підтримується декілька клієнтів), сервер виокремлює необхідні комунікаційні ресурси між процесами (спільна пам’ять, канал, сокет тощо) і чекає запитів на відкриття з’єднання (або фактично запитів на обслуговування). Залежно від характеру такого ресурсу сервер може розміщувати процеси в одній комп’ютерній системі або процеси на інших машинах через канали даних (наприклад, COM-порт) або мережеві з’єднання.

Формат клієнтських запитів та відповідей сервера визначається протоколом. Специфікації відкритих протоколів описуються відкритими стандартами, такими як B. Інтернет-протоколи, визначені в документах RFC [11].

Залежно від завдань деякі сервери можуть бути відсутніми в режимі очікування без запитів на обслуговування. Інші можуть виконувати конкретні завдання (наприклад, збір інформації), для таких серверів робота з клієнтами може бути другорядним завданням.

Наприклад, сервери можна класифікувати за класом завдань, які вони виконують, або кількістю клієнтів, які надають послуги. Великі сервери відрізняються в  рядом аспектів.

- робоча група (робоча група);

-кафедра (кафедра);

-середня організація (середня);

- компанія (компанії).

Існує багато класифікацій серверів за класом завдань. Як результат виробники часто сегментують виготовлені сервери за типом конструкції. Ультратонкий (блейд), класичний підлоговий (баштовий), оптимізований для монтажу в стійку (стійка), високої масштабованостьі(ультра-масштабований) [31].

Визначимо тип сервера,  відповідно до наукової класифікації.

Веб-сервер

Багато в чому веб-сервер схожий на робота-фуршета. Клієнт там щось просить. У цьому випадку це файл, і веб-сервер отримує файл і відправляє його клієнту. У більшості випадків початковий веб-сервер нічого не робить з цим файлом

і просто повертає його клієнту. Сучасні веб-сервери розвинули здатність одночасно обробляти велику кількість запитів і швидко відповідати, обробляючи запити більш складними способами, ніж просто надсилання документа. В результаті веб-сервери вийшли на нову територію і стали називатися «серверами додатків» або «інформаційними серверами».

Сервер додатків

Сервер додатків розширює обробку інформації, й взаємодія з клієнтом схожа на роботу з програмою. Вона подібна до роботи з користувачем за комп'ютером, коли читач має можливість гортати сторінки ... Функціональність серверів та їх додатків може бути досягнута шляхом інтелектуального поєднання існуючих технологій. Наприклад, розробник із відкритим кодом може підключити веб-сервер Apache до мови сценаріїв PHP, щоб ефективно створити сервер додатків.

Файловий сервер

Файлові сервери є важливою складовою електронної інфраструктури компанії. Це дуже швидкий комп'ютер, підключений до мережі. Як правило він зберігає програми та дані, якими поділилися користувачі. Використовують великий жорсткий диск, до якого можуть отримати доступ всі комп’ютери у мережі компанії. Файловий сервер забезпечує зв'язок між мережевими станціями і дозволяє користувачам отримувати доступ до файлів, необхідних для роботи. Крім того, файлові сервери зазвичай обмежують несанкціонований доступ до даних [18].

Переваги такої електронноії інфораструктури компнії повязна з и тим, що інформація зберігається централізовано. Вона не поширюється на комп’ютери різних співробітників. До серверу компанії о можна отримати доступ з будь-якого комп’ютера, під’єднаного до сервера (який може бути віддаленим комп’ютером, який зв’язується із сервером по телефону). Часто сервери захищені від випадкового доступу, оскільки для входу на сервер підключається пароль. Ще однією важливою перевагою сервера є надійність зберігання інформації. Як правило сервер набагато краще захищений від збоїв, ніж ПК. Якщо один із жорстких дисків сервера виходить з ладу, є спосіб повністю відновити інформацію.

Насправді є  різниця між файловим сервером і сервером додатків. Перший зберігає програми та дані, а другий запускає програми для обробки даних.

«Бездротові» сервери

Термін «бездротовий» сервер застосовують описуючи  дві різні технології. Комп'ютер може бути типовим веб-сервером або сервером додатків. Такий ПК знає, як надіслати документ, написаний мовним стандартом бездротового пристрою. Для цього використвують мову Wireless Markup Language (WML). Сьогодні актуалізується адаптація веб-сервера для роботи як бездротового сервера. Такий сервер може обробляти документи WML. Є програми що  «навчає» сервер розпізнавати ці документи. Веб-сервер має лише повідомити клієнта, що документ у форматі.

Більш складним типом «бездротового» сервера є «бездротовий» шлюз. Такі шлюзи діють як посередники. Отримуючи запити від бездротових пристроїв і пересилаючи їх на традиційні веб-сервери. Більшість бездротових шлюзів керуються постачальниками послуг, які спеціалізуються на наданні бездротового доступу. Є можливість у користувачів обмежити своїх абонентів лише послугами, які ці шлюзи підтримують. Як правило такі системи орієнтовані на певний сегмент ринку [32].

Проксі-сервер

Він діє як посередник, який мережу компанії чи клієнта, та дозволяє отримувати інформацію з Інтернету, коли вона потрібна. Проксі-сервер може кешувати інформацію  за запитом на локальному жорсткому диску. З метою швидкої доставки інформації користувачам без необхідності повторного відвідування Інтернету.

Однак проксі-сервери зберігають не тільки часто запитувані дані. Коли користувачу не потрібен прямий доступ до Інтернету, так само є можливість задовольнити потреби клієнта. Проксі-сервери є методом підключення внутрішньої мережі компанії до Інтернету. Зі зростанням популярності та доступності широкосмугового доступу потреба в проксі-серверах зросла. Це з'єднання забезпечує пропускну здатність для підтримки кількох машин одночасно. За

допомогою проксі-рішень користувачам потрібно використовувати лише один із цих комп’ютерів. Такий ПК підтримує широкосмуговий зв’язок, а підприємства можуть економно використовувати простір IP-адрес. Це дозволяє зменшити оплату послуг провайдерам.

Брандмауер.

Проксі-сервери можна налаштувати на прийом або відхилення певних типів мережевих запитів як з локальної мережі, так і з Інтернету. У цій конфігурації проксі-сервер є брандмауером. Брандмауер є інструментом безпеки. Залежно від висоти брандмауера ви можете налаштувати його, щоб розрізняти багато типів вхідних і вихідних даних. Отримання дозволу на передачу даних в тому чи іншому напрямку залежить від певних умов, таких як: В. IP-адреса джерела даних. Хороший брандмауер також пропонує широкі можливості ведення журналів, оскільки докази мережевої активності є ключовими для розслідування випадкових або навмисних інцидентів [4].

Існують брандмауери у вигляді функціонально завершеної системи та проксі-сервери з можливостями брандмауера. Ви можете подумати, що брандмауер не є сервером у традиційному розумінні, тому що він схожий на «канаву», що розділяє сервери.

Поштовий сервер

Як і проксі-сервер, поштовий сервер (іноді його називають сервером обміну повідомленнями). Такий ПК має обробляти як вхідні, так і вихідні запити. Одним із завдань поштового сервера є прочитання адреси вхідного повідомлення та доставка повідомлення у відповідну поштову скриньку в Інтернеті. Залежно від складності поштового сервера адміністратор може мати більший чи менший контроль над цими локальними поштовими скриньками. Також враховуються особливості типу та розміру повідомлень, які можна отримувати, автоматичними відповідями.

DHCP-сервер

Основна перевага використання сервера DHCP полягає в тому, що ви можете змінювати конфігурацію вашої локальної мережі під час додавання або видалення комп’ютерів (наприклад, ноутбуків) у міру зростання локальної мережі.

У деяких випадках програмне забезпечення сервера DHCP вбудовується у відповідне обладнання.

FTP-сервер

Сервери FTP, засновані на протоколах передачі файлів, були де-факто стандартом для переміщення файлів через Інтернет протягом десятиліть. FTP-сервер підтримує прості завдання менеджера файлів (клієнтів). FTP-сервери вимагають особливої безпеки, але вони є найпоширенішим і зручним способом переміщення файлів з одного комп'ютера на інший між сусідніми локаціями та континентами однієї компанії [23].

Складний FTP-сервер надає адміністраторам детальний контроль над підключенням і дозволами на спільний доступ до файлів. Також враховуються типи файлів та особливості розміщенням. Існують ресурси для багатьох підключень до серверів, обмеження даних, мінімальні швидкості передачі тощо стають все більш популярними як інструменти для підвищення безпеки FTP-сервера.

Сервери друку

Ці сервери дозволяють будь-якому комп'ютеру, підключеному до вашої мережі, друкувати документ на одному або кількох спільних принтерах. У цьому випадку вам не доведеться комплектувати окремий пристрій друку для кожного комп'ютера. Крім того, сервер друку вирішує всі проблеми друку документів і звільняє комп'ютер для виконання інших завдань. Так, сервер друку зберігає надісланий для друку документ на диску, ставить його в чергу та друкує на принтері в порядку пріоритету.

Сервер віддаленого доступу

Ці системи дозволяють спілкуватися з офісною мережею засобами телефоної мережі. Наприклад, коли є ноутбук далеко від офісу, є можливість отримати доступ до потрібних файлів і перевірити, особисту електронну скриньку. Таким чином сервери віддаленого доступу використовують під час роботияк в офісі так віддаленій роботі.

Факс-сервер

Такі сервери заміняють традиційні факсимільні апарати. Його мета — контролювати процес надсилання та отримання факсів засобами комп'ютерної мережі. Його використовують для обміну документами. Це зручніше ніж телефонні лінії або стопки термопаперу або звичайного паперу. Існує ряд функцій факс-сервера які керують ресурсами телефону.

Серверні приставки

Насправді цей термін відноситься до будь-якого типу сервера на ринку, який уже налаштований і готовий до включення в мережу.

Сервер для інфраструктури електронного бізнесу

Середовища електронного бізнесу зазвичай вимагають різних серверів, кожен з яких має певні вимоги до продуктивності, масштабованості та доступності. Як правило, ці сервери можна розділити на кілька рівнів. Наприклад, інтерфейсний інтернет-сервер. Проміжний сервер додатків. Сервер бази даних.

## 1.2. Апаратне забезпечення

Сервер — це комп'ютер, обраний із групи персональних комп'ютерів (або робочих станцій). Його використовують виконання певного завдання без безпосередньої участі людини. Сервер і робоча станція можуть мати однакову апаратну конфігурацію. Вони також відрізняються лише залученням людини [7].

Ряд робочих завдань можна виконувати на робочій станції паралельно з роботою користувача. Таку робочу станцію умовно називають невиділеним сервером.

Серверам потрібна консоль (зазвичай монітор/клавіатура/миша) та участь людини лише на початкових етапах налаштування. Як правило під час обслуговування пристрою та керування в надзвичайних ситуаціях. Сьогодні більшість серверів керуються віддалено. Для надзвичайних ситуацій сервери зазвичай оснащені одним набором консолей на групу серверів (з перемикачем або без нього, наприклад, перемикачем KVM).

Завдяки спеціалізації, серверне рішення може виграти або втратити спрощену консоль (наприклад, порт зв’язку), і в цьому випадку початкову конфігурацію та керування в надзвичайних ситуаціях можна виконувати лише через мережу, а налаштування мережі можна скинути. за замовчуванням. ).

Спеціалізація серверного обладнання відбувається кількома способами. Більшість спеціалізацій збільшують витрати на обладнання.

Серверне обладнання зазвичай комплектується більш надійними елементами:

- Основна пам'ять із підвищеною відмовостійкістю, наприклад для ПК, пам’ять для серверів з технологією ECC (перевірка та виправлення помилок). На деяких інших платформах, наприклад B. Sparc (Sun Microsystems), вся пам'ять має виправлення помилок.

- Резервування, включаючи: джерела живлення (включаючи гарячі підключення), жорсткі диски (RAID; включаючи гарячі підключення та запасні). Не плутати з системами RAID на звичайних комп’ютерах; більш складне охолодження (функція) [10].

Розміри та інші деталі екстер'єру. Сервери (та інше обладнання), яке необхідно встановити на деякі стандартні шасі (наприклад, 19-дюймові стійки та шафи), стандартизовані та постачаються з необхідним монтажним обладнанням. Сервери, які не вимагають високої продуктивності та багатьох зовнішніх пристроїв, часто зменшуються. Часто це скорочення супроводжується скороченням ресурсів.

У так званому «промисловому варіанті», крім зменшеного розміру, корпус має більшу міцність, захист від пилу (оснащений змінними фільтрами) (а іноді і вологи), а також має конструкцію кнопки, що запобігає випадковому відключенню. Конструктивно апаратні сервери можуть бути виконані в настільному, підлоговому, рейковому та стельовому варіантах. Останній варіант пропонує найвищу щільність обчислювальної потужності на одиницю площі та максимальну масштабованість. З кінця 1990-х років блейд-сервери, компактні модульні пристрої, які знижують витрати на живлення, охолодження, технічне обслуговування тощо, стають все більш популярними у високонадійних, масштабованих системах.

Ресурс. За ресурсами (частота і кількість процесорів, обсяг пам’яті, кількість і потужність жорстких дисків, потужність мережевих адаптерів) сервери спеціалізуються у двох протилежних напрямках – збільшення ресурсів і зменшення ресурсів.

Зростання ресурсів спрямоване на збільшення потужності (наприклад, спеціалізації для файлового сервера) і продуктивності сервера. Коли продуктивність досягне певної межі, подальше зростання буде продовжуватися іншими методами, наприклад в паралельні завдання між кількома серверами. Зменшення ресурсів спрямоване на зменшення розміру та енергоспоживання серверів.

Апаратні рішення. Так звані апаратні рішення (апаратні маршрутизатори, мережеві дискові масиви, апаратні термінали тощо) утворюють найвищий ступінь спеціалізації серверів. Апаратне забезпечення таких рішень створено з нуля або перероблено з існуючої комп’ютерної платформи без сумісності, що унеможливлює використання пристрою із готовим програмним забезпеченням.

Програмне забезпечення в апаратних рішеннях завантажується виробником у постійну та/або енергонезалежну пам’ять. Апаратні рішення, як правило, більш надійні, ніж традиційні сервери, але менш гнучкі та універсальні. За ціною апаратні рішення можуть бути як дешевшими, так і дорожчими за сервери, залежно від класу обладнання [33].

Псевдоапаратні рішення. Останнім часом з'явилася велика кількість бездискових серверних рішень, заснованих на комп'ютерах (переважно x86) форм-фактора Mini-ITX і рідше зі спеціалізованою обробкою GNU/Linux на жорсткому диску SSD (флеш-пам'ять або флеш-карта ATA). Вони позиціонуються як «апаратні рішення». Ці рішення не належать до апаратного класу, а є звичайними спеціальними. Сервери розташовані в так званих серверних кімнатах. Серверами керують системні адміністратори [33].

## 1.3. Поняття центр обробки даних (ЦОД) та поняття системи управління центрами обробки даних

Центр обробки даних (ЦОД) — це вузькоспеціалізоване виробниче підприємство. Воно забезпечує прийом, аналіз, обробку, зберігання та поширення одного або кількох типів даних. Зокрема  узгоджує роботу взаємопов'язаних пристроїв і систем ІКТ з особливими вимогами для вирішення різноманітних завдань компаній і організацій. Центри обробки даних — це високотехнологічні об’єкти для розміщення комп’ютерів, призначених для обробки, зберігання та поширення інформації [24].

За оцінками експертів Frost & Sullivan і DirectINFO, у 2014 році світовий ринок послуг центрів обробки даних становив 5,6 мільярдів доларів.

Основне завдання дата-центру – забезпечити гарантований час роботи IT-інфраструктури підтримуваних ним компаній.

З технічної та організаційної точки зору ЦОД є поєднанням трьох інфраструктурних блоків: інфраструктури інформаційно-телекомунікаційних технологій (ІКТ), інфраструктури електропостачання та інфраструктури охолодження.

Інфраструктура ІКТ складається в основному з обладнання ІКТ з відповідним програмним забезпеченням. Найчастіше пристрої поділяють на три категорії: сервери, мережеві комутатори та прилади

Зберігання даних (пам'ять). Кожна група пристроїв виконує свою унікальну функцію. Крім того, інфраструктура ІКТ включає різноманітне програмне забезпечення, системи віртуалізації, бази даних, веб-хостинг, операційні системи та хмарні обчислення.

Живлення та охолодження ЦОД – це два наступні блоки інфраструктури, які необхідні для безперебійної роботи першого блоку. Електрика надходить переважно із зовнішньої мережі від двох незалежних джерел. Є деякі винятки, наприклад В. Електрика від паливних елементів і генераторів [35].

Консорціум Green Grid (TGG) запропонував набір індикаторів (метрик), які включають:

• Енергоефективність центру обробки даних (DCEP),

• Ефективність повторного використання енергії (ERE),

• Обчислювальна ефективність центру обробки даних (DCCE),

• Ефективність води (WUE),

• Карбонова ефективність (CUE) та інші.

Ці показники дозволяють операторам центрів обробки даних стратегічно оцінювати ефективність своєї роботи з точки зору витрат енергії.

Основні функції дата-центрів:

• зберігання та аналіз інформації різного розміру;

• Забезпечення безпеки технічних засобів, за допомогою яких інформація зберігається, передається та отримується;

• забезпечення максимальної доступності даних для споживачів;

• безперебійну та безперебійну роботу обладнання;

• Уніфікація розподілених систем;

• Надання конференц-дзвінків тощо [6].

Ці можливості значною мірою визначаються послугами, які запитують споживачі та які може надати центр обробки даних. Існує кілька класифікацій послуг центрів обробки даних. Таким чином, послуги ЦОД включають:

- Оренда приміщення (площа) в дата-центрі для розміщення обладнання (оренда місця під стійку замовника, оренда місця в стелажі для

хостинг серверів);

- Оренда обладнання ЦОД (оренда серверів, оренда виділених віртуальних серверів, оренда серверних частин (хостинг), оренда систем зберігання даних);

- Телекомунікаційні послуги (доступ до мережі компанії,

традиційні телекомунікаційні послуги, доступ до міжнародної точки пирингу);

- Додаткові послуги дата-центру.

В академічній літературі зазначається, що наведена вище класифікація достатньою мірою систематизує та раціоналізує основні послуги центрів обробки

даних. Водночас, однак, він не описує послуги на основі «хмарних» технологій. Крім того, через їх індивідуальний характер, вищезазначена класифікація не описує конкретні послуги з групи додаткових послуг центрів обробки даних.

Існує ще одна класифікація, за якою послуги ЦОД поділяються на стандартні та комплексні.

Типовими послугами є:

- Colocation - використання клієнтом технічних ресурсів дата-центру: розміщення власного телекомунікаційного обладнання, серверів, систем зберігання даних тощо.

- Виділений сервер - надання клієнтам серверів дата-центру в оренду

- Telehousing - розміщення та підключення до систем живлення та охолодження телекомунікаційного обладнання та різних елементів IT-інфраструктури клієнта в дата-центрі

- Спільний - оренда клієнтом дискового простору для обладнання ЦОД

Під комплексними послугами:

- Аутсорсинг інформаційних систем - послуга, що передбачає

придбання постачальником управління або права власності на частину або всю IT-інфраструктуру замовника

- Хостинг обслуговування та адміністрування ПЗ - послуга, яка передбачає централізоване управління тиражованим ПЗ, при цьому замовник має віддалений доступ до ПЗ, а самі програми знаходяться в інформаційному центрі постачальника [27].

- Хостинг інфраструктурних послуг - послуга, яка передбачає надання стандартних частин IT-інфраструктури для віддаленого використання на певний період.

Основними споживачам, які потребують послуг центрів обробки даних, є:

• Підприємства малого та середнього бізнесу, для яких дата-центри вирішують проблеми колокації та хостингу (Colocation, hosting).

• Банківські та різні фінансові установи, для яких центри обробки даних здійснюють фінансові операції.

• Телекомунікаційні компанії, для яких центри обробки даних забезпечують підключення цифрових послуг наземної телефонної лінії з мобільними операторами, операторами IP-телефонії тощо.

• IT-компанії, такі як Google, Amazon, eBay, Facebook, Yandex, Rambler та інші, а також телекомунікаційні компанії, що володіють і створюють нові центри обробки даних.

• Уряд та інші органи державної влади на федеральному та регіональному рівнях, у цьому випадку центри обробки даних необхідні для забезпечення функціонування електронного уряду тощо.

• Охорона здоров'я. Очікується, що в цьому сегменті попит на послуги центрів обробки даних буде зростати через тенденцію оцифровування звітів про лікування пацієнтів та інших медичних даних.

• Великий державний та недержавний бізнес, у тому числі численні групи підприємств із різних галузей економіки, для яких центри обробки даних забезпечують виконання всіх необхідних функцій [29].

Окремо слід розглянути так звані «хмарні обчислення», які не можна віднести до споживачів послуг дата-центрів, а є сервісом центрів обробки даних. Фактично «хмарні» технології є засобом поширення IT-додатків між фізичними серверами та фізичними центрами обробки даних.

Деякі автори виділяють три основні сервіси, засновані на «хмарних» технологіях:

• Інфраструктура як послуга (IaaS)

• Платформа як послуга (PaaS)

• Програмне забезпечення як послуга (Software as a Service, SaaS)

Найпоширеніші моделі послуг хмарних обчислень (SaaS, PaaS та IaaS) націлені на конкретні бізнес-моделі. Модель X (що завгодно) як послуга (XaaS) означає, що все, що можна доставити через Інтернет, можна продати. Це включає:

• Бекенд як послуга (BaaS)

•Зберігання як послуга (SaaS);

•Зв'язок як послуга (CaaS);

•Мережа як послуга (NaaS) і

•Моніторинг як послуга (MaaS) тощо.

Таким чином, послуги ЦОД є збірним поняттям для визначення таких видів діяльності, як: ефективна робота сховища даних, управління даними та розподіл даних. Для більш чіткого визначення та структурування комплексу послуг дата-центрів, на нашу думку, необхідно розглянути типові послуги центрів обробки даних та їх споживачів [14].

Структура дата-центрів

Існує багато різних типів центрів обробки даних, створених для різних додатків. Найпростіший спосіб класифікувати їх за розміром:

• міні-центри обробки даних невеликого розміру: ці системи використовують сотню стійків або менше, і зазвичай використовуються для невеликих підприємств, наприклад, для експериментальних установ (наприклад, невеликий модуль центру обробки даних з 10 стійками, розміщений Шведським інститутом комп'ютерних наук ICE (SICS)). в Лулео або модульних серверних контейнерах, наприклад, модульному центрі обробки даних Sun Microsystems;

• центри обробки даних середнього розміру: ці системи використовують від ста до тисячі стійок і зазвичай використовуються для підприємств середнього розміру;

• центри обробки даних великого розміру: у цих системах розміщено більше тисячі стійок і зазвичай використовуються такими великими корпораціями, як Microsoft, Google і Facebook. Ці приміщення можуть бути дуже широкими

Інфраструктуру в дата-центрі можна структурувати на три основні частини, а саме:

• Інформаційні технології (IT);

• Система Cooling Technologies (CT);

• Система розподілу електроенергії (PD) [16].

## Висновки за розділом 1

Збільшена швидкість передачі даних в мережах і продуктивність комп'ютерів, на сьогодні, дають можливість користувачам не лише спілкуватися в реальному часі за допомогою текстових повідомлень, але і здійснювати аудіо- і відеозв'язок.

Для функціонування повноцінного інтерактивного спілкування в фірмах та закладах необхідні наступні компоненти: •Сервер – програма, яка приймає повідомлення від клієнта, обробляє інформацію в ньому і відправляє їх за потрібною адресою. Працює на віддаленому комп’ютері в мережі. Клієнт – програма, яка відправляє повідомлення на сервер для інших абонентів та приймає повідомлення від них. Працює на локальному комп’ютері кожного співбесідника. Мережа – середовище, в якому передаються повідомлення. Це може бути, наприклад, Інтернет або локальна мережа на основі протоколу TCP/IP.

Також використвують ЦОД— це вузькоспеціалізоване виробниче підприємство. Воно забезпечує прийом, аналіз, обробку, зберігання та поширення одного або кількох типів даних. Центри обробки даних — це високотехнологічні об’єкти для розміщення комп’ютерів, призначених для обробки, зберігання та поширення інформації

# РОЗДІЛ 2
# АНАЛІЗ ВРАЗЛИВОСТЕЙ СЕРВЕРІВ І ЦОД

## 2.1. Атаки на центри обробки даних. Їх загроза та наслідки

Теплова атака на центри обробки даних.

Загроза безпеці, яку представляють теплові атаки на центр обробки даних, є реальною і її важко подолати, головним чином через п’ять наступних причин.

I. Основна причина загрози полягає у широкому застосуванні агресивних політик охолодження та керування живленням, таких як підвищення температури припливного повітря та перевищення передплати, що дозволяє розгортати більше серверів у центрі обробки даних із меншими витратами на охолодження. Хоча центри обробки даних спочатку були розроблені з достатнім резервуванням охолодження, ця агресивна політика значно зменшує резерви охолодження, роблячи самі центри обробки даних вразливими до аномальних теплових умов [11].

II. Хоча сучасні сервери оснащені різними датчиками рівня мікросхем, такими як датчик температури для кожного ядра, ці датчики на рівні чіпа не можуть надавати інформацію на рівні сервера (наприклад, температура на вході та виході сервера). Температура всередині не дорівнює температурі на вході та температурі на виході. Температура ядра змінюється набагато швидше. Для ефективної термічної атаки не потрібно постійно напружувати центральний процесор, а просто підтримувати температуру на виході на високому рівні. В результаті датчики Core не можуть забезпечити моніторинг температури на рівні сервера або постійного струму.

III. Хоча термодатчики, безумовно, допомагають контролювати температурні умови в комп’ютерній кімнаті, більшість сучасних центрів обробки даних мають лише кілька теплових датчиків або зондів для всього ЦОД. Розгортання датчиків на рівні сервера або стійки буде дуже дорогим. Навіть із датчиками рівня стійкості (лише 2-4 датчики на стійку) їм неможливо охопити сотні серверів у стійці. Таким чином, виникнення локальної гарячої точки неминуче. Без глобальних знань про

загальні теплові умови в комп’ютерній кімнаті захист на рівні мікросхем, як-от Intel RAPL (Running Average Power Limit), що забезпечує та обмежує споживання енергії для кожного пакета ЦП, не може запобігти виникненню локальних гарячих точок.

IV. Для захисту від теплових атак на центр обробки даних важливо враховувати термодинаміку ЦОД. Механізм регулювання температури на основі зворотного зв’язку допоможе обмежити температуру для локальних гарячих точок. Проте такого механізму керування зворотним зв’язком у нинішніх дата-центрах немає. На рівні мікросхеми існують деякі механізми керування зворотним зв'язком, що використовуються для захисту від перегріву; однак, на рівні ЦОД, без глобальних знань про теплові умови, дуже складно розгорнути механізми керування зворотним зв’язком для керування температурою всього ЦОД.

V. Оскільки робочі навантаження, що інтенсивні, самі по собі є безпечними і не використовують жодних системних уразливостей, важко відрізнити зловмисників від звичайних користувачів. Крім того, профіль потужності/теплового профілю на рівні процесу також не може захистити від теплових атак на рівні центру обробки даних. Зловмисники не обмежуються використовувати лише один процес, щоб генерувати набагато більше тепла. Вони можуть використовувати багато облікових записів для одночасного виконання різних робочих навантажень, щоб генерувати значну кількість тепла [28].

Теплова атака може значно підвищити температуру цільового сервера. Спочатку оцініть вплив теплової атаки на стійку сервера. У середовищі стійки рециркуляція повітря змушує сервери впливати на температуру один одного. В результаті теплових потоків різні розташування стійки призводять до різного температурного впливу. Наприклад, сервери в нижній частині стійки можуть мати менший вплив, оскільки холодне повітря подається з фальшпола і тече вгору до вентиляційних отворів на стелі. Як наслідок, нижні сервери зазвичай мають нижчу температуру. Щоб розглянути тепловий ефект рециркуляції повітря, ми порівнюємо три стратегії атаки для стійки: (1) атака гарячої точки; (2) атака без точки доступу; і (3) випадкова атака. Під час атаки на точки доступу ми в основному атакуємо сервери, які мають більший вплив на стійку (наприклад, у верхній частині стійки), і

тому створювати точки доступу легше. У атаці без точки доступу ми зосереджуємось на серверах, розташованих у позиції з невеликими тепловими ефектами (наприклад, у нижній частині стійки). У випадковій атаці ми випадковим чином вибираємо сервери для атаки. Ми атакуємо різну кількість серверів в одній стійці, а інші сервери залишаємо в неактивному стані.

Підвищення температури впливає на частоту збоїв обладнання. Таким чином, ми проводимо аналіз апаратних збоїв, використовуючи EM, TDDB та збій диска, щоб оцінити теплову атаку на стійку. Ми бачимо, що середній MTTF сервера знижується, коли більше серверів піддаються атаці. Наприклад, коли атакуються близько 10 серверів, теплові атаки можуть збільшити частоту збоїв на ~10% для всіх серверів у стійці, в результаті чого нормалізований MTTF становить ~90%. При атаці на 30 серверів нормований MTTF падає до 75%. Серед трьох типів атаки найбільшої шкоди завдає атака гарячої точки, оскільки ці позиції можуть максимізувати тепловий ефект. Це вказує на те, що рециркуляція повітря дійсно впливає на теплові умови сервера навіть в одній стійці [35].

IT-сфера розвивається, ускладнюється, тому й ускладнюються атаки. Технічно, DDos-атака як так і залишилося комплексом дій, де мета - "забити" канал зв'язку, при якому буде недоступний сайт/сервіс. Це можливо за різних сценаріїв. Тільки якщо раніше атакувалися сервери, що знаходяться у клієнта, тепер атакують потужності провайдерів. Але у провайдерів (ЦОД) набагато більше можливостей захисту. Через ускладнення IT-інфраструктури змінюються складові "архітектури атаки", через які йде атака. Як правило, мережа будь-якого підприємства складається з різноманітних рішень: свічі, роутери, всі вони різні за рівнем захисту, і тому вони можуть зазнати атаки.

Раніше для цього зламувалися робітники комп'ютери - але це затратно і не так ефективно. Останній тренд, про який повідомляють IT-інженери з кібер-безпеки - атака через елементи спостереження. Зламуються реєстратори та IP камери, оскільки найчастіше ці пристрої працюють на Open Source рішеннях і вразливі для зловмисників. Потім ця ботнет-мережа починає атакувати якусь адресу. Такою розподіленою мережею, яка атакує (шле запити) можуть бути будь-які «речі з

підключенням до інтернету» адже тій же IP камері, або холодильнику з підключенням до мережі, все одно, куди надсилати запит. Будь-які датчики, підключені до Інтернету, якщо вони інфіковані - можуть бути дуже небезпечною річчю. Таким чином, «інтернет речей», який обіцяє блага цивілізації, може обернутися вразливістю всієї величезної сучасної IT-інфраструктури.

Системи безпеки вже давно можуть збирати різні події у спільні лог-файли. При правильній архітектурі системи безпеки - система може зібрати всі розрізнені події в одну загальну картину, яка відразу прояснює клієнту, що і звідки в нього було атаковано. Наприклад, хтось сканував порти замовника, а потім розпочалася атака для доступу до даних тощо. - Такими можливостями «розуміння» подій захисту для своїх клієнтів мають провайдери та ЦОДи [22].

Основні види DDos:

UDP-флуд – мережна атака типу «відмова в обслуговуванні», яка використовує режим безсеансного протоколу UDP. Полягає у надсиланні безлічі UDP-пакетів (як правило, великого об'єму) на певні або випадкові номери портів віддаленого хоста, який для кожного отриманого пакета повинен визначити відповідний додаток, переконатися у відсутності його активності і відправити відповідне повідомлення ICMP «адресат недоступний». В результаті атакована система виявиться перевантаженою: у протоколі UDP механізм запобігання перевантаженням відсутній, тому після початку атаки паразитний трафік швидко захопить всю доступну смугу пропускання, і корисному трафіку залишиться лише мала її частина. Підмінивши IP-адреси джерел в UDP-пакетах, зловмисник може перенаправити потік ICMP-відповідей і тим самим зберегти працездатність атакуючих хостів, а також забезпечити їх анонімність.

HTTP-флуд - це найпоширеніша flood атака. В її основі – посилання HTTP-запитів GET на 80-й порт. Це призводить до такого стану завантаження сервера, що він виявляється нездатним для обробки інших запитів. Ця атака flood може бути націлена як на корінь сервера, так і на його скрипт, зайнятий виконанням ресурсомістких завдань. Розпізнання даної атаки можливе шляхом виявлення

швидкого зростання кількості запитів до одного або кількох скриптів на сервері та швидкого зростання логів сервера [21].

DoS (Denial of Service - відмова в обслуговуванні) - хакерська атака на обчислювальну систему з метою довести її до відмови, тобто створення таких умов, за яких сумлінні користувачі системи не можуть отримати доступ до системних ресурсів (серверів), що надаються, або цей доступ утруднений. Відмова «ворожої» системи може бути кроком до оволодіння системою (якщо в позаштатній ситуації ПЗ видає будь-яку критичну інформацію — наприклад, версію, частину програмного коду тощо). Але найчастіше це міра економічного тиску: втрата простої служби, що приносить дохід, рахунки від провайдера та заходи щодо уникнення атаки відчутно б'ють «мету» по кишені. В даний час DoS і DDoS-атаки найбільш популярні, тому що дозволяють довести до відмови практично будь-яку систему, не залишаючи юридично значущих доказів.

## 2.2 Аналіз вразливостей веб-серверів

Веб-сайт розміщено на веб-сервері. Веб-сервер – це комп’ютер під керуванням операційної системи, який зберігає файли сайту (документи HTML, стилі CSS, файли JavaScript, зображення) і передає їх на пристрої кінцевих користувачів (веб-браузери) [9].

Веб-сервер також може взаємодіяти з базою даних, якщо він реалізований для отримання інформації з бази даних і надання її в певному форматі HTML.

Коли справа доходить до програмного забезпечення, веб-сервер містить ряд компонентів, які відповідають за доступ користувачів до файлів на сервері, наприклад, сервери HTTP. HTTP-сервер — це програмне забезпечення, яке розуміє веб-адреси (URL-адреси) і HTTP. Сервер зберігає веб-сторінку, надає її клієнту за запитом та обробляє через HTTP. Він відіграє важливу роль у всій архітектурі як протокол для відправки інформації від клієнта до сервера і навпаки.

Не дивно, що мережі, як правило, піддаються атаці з боку хакерів, які атакують веб-сервери різними способами. В результаті вразливості в програмах,

базах даних, операційних системах або мережах можуть атакувати веб-сервери. Атака на веб-сервер означає порушення нормальної роботи [15].

Видаліть або змініть вузол, його вміст або отримайте привілейований доступ до машини.

Враховуючи різні технології, які використовуються компонентами веб-сервера, необхідно проаналізувати існуючі мережеві атаки на веб-сервер і способи їх захисту. Рисунок 2.1. показує стек уразливостей веб-сервера.

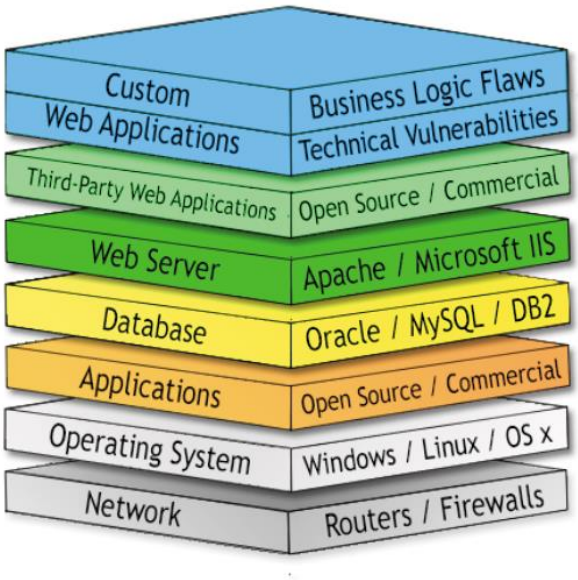


Рисунок 2.1 – стек уразливості веб-сервера

Основні типи мережевих атак і приклади сценаріїв атак.

SQL-ін'єкція

База даних є дуже зручною структурою для зберігання інформації. У більшості випадків доступ до бази даних здійснюється за допомогою спеціальної мови запитів (SQL). SQL реалізується за магідопомогою інтерпретатора. Інтерпретована мова — це мова, виконання якої містить компоненти часу виконання, які інтерпретують код мови та виконують інструкції, що містяться в ньому. Спосіб інтерпретації мови вносить набір уразливостей, які називаються реалізацією коду. Потенціал вставки виникає, коли недовірені дані надсилаються інтерпретатору як частина команди або запиту [26].

Процес, за допомогою якого програма отримує доступ до сховища даних, однаковий, незалежно від того, чи був цей доступ ініційований непривілейованим користувачем чи адміністратором програми. Веб-додаток діє як довільний контроль доступу до сховища даних,

Запит для отримання, додавання або зміни даних у базі даних на основі облікового запису та типу користувача. Успішна ін’єкційна атака, яка змінює запит (а не лише дані в запиті), може обійти будь-які засоби контролю доступу програми та отримати несанкціонований доступ. Крім того, якщо логіка програми контролюється результатом запиту, зловмисник може змінити запит, щоб змінити логіку програми.

XSS атака

Атака XSS — це атака на вразливість сервера, яка вводить довільний код HTML / JavaScript в результаті сценарію, якщо сценарій не фільтрує дані від користувача. Через те, що фільтрація не працює, сервер не перевіряє цю змінну на наявність заборонених символів-,, <,>,', «. Код може містити шкідливу інформацію, яка може загрожувати комп’ютеру жертви через вибухове зростання веб-браузерів. Міжсайтові сценарії також можуть містити шкідливий код JavaScript, який надсилає облікові дані сеансу на інший веб-сервер. Використання міжсайтових сценаріїв має на меті надати жертвам неправдиву інформацію, наприклад, неправдиву інформацію про "реальний світ", як-от новини, які, здається, надходять з іншого законного джерела. Можуть бути випадки. Вміст може включати форму входу, яка, коли її надсилається, подає облікові дані для входу на веб-сервер, що належить хакерам, а не на «справжній» сервер [28].

Цей тип атаки цікавий тим, що шкідливі скрипти з сервера виконуються на комп’ютері клієнта і запускаються самою жертвою.

CSRF атака

Підробка міжсайтових запитів (CSRF) — це атака, яка змушує кінцевого користувача виконувати непотрібні дії над веб-додатком, що перевіряється на даний момент. Уразливості CSRF можуть виникнути, якщо програма ідентифікує, використовуючи лише файли cookie HTTP.

Користувач, який надіслав конкретний запит. Браузери автоматично додають файли cookie до запитів незалежно від джерела запиту, що може дозволити зловмиснику створити зловмисний веб-сайт, який підробляє міждоменний запит до вразливої програми. Зловмисник створює підроблені HTTP-запити і змушує жертву надсилати їх за допомогою тегів зображень, XSS або іншим способом.

Зловмисник може змусити жертву виконати операцію зміни стану - вхід в систему, оновлення облікових даних, здійснення фінансових операцій [31].

Порушена аутентифікація

Розробники часто витрачають мало часу та уваги на цю вразливість, і цей тип атаки дозволяє зловмиснику перехопити або обійти метод аутентифікації, який використовується у веб-додатку, перехопивши ідентифікатор користувача.

Щоб відрізнити одного користувача від іншого, веб-додатки використовують так звані сесійні файли cookie. Коли користувач вводить логін і пароль і програма їх схвалює, спеціальний ідентифікатор зберігається в сховищі браузера, і браузер представляє його серверу щоразу, коли запитується сторінка веб-програми. Це спосіб для веб-додатків зрозуміти, що це потрібний користувач.

Якщо хакер отримує ідентифікатор сеансу неперевіреної системи, наприклад, перевіряє IP-адресу в сеансі, або перевіряє кілька з’єднань за один сеанс, зловмисник може отримати доступ з привілеями облікового запису користувача. Таким чином, мета шахрайської атаки аутентифікації — захопити один або кілька облікових записів і отримати ті самі привілеї, що й атакований користувач. Наступні типи вразливостей дозволяють зловмиснику обійти метод аутентифікації:

- Введення аутентифікації користувача не захищено під час збереження [35].

-Реєстраційні дані (пароль тощо), які легко отримати

-Ідентифікатор сеансу буде відображатися в URL-адресі (наприклад, якщо ви скопіюєте URL-адресу)

-Після виходу значення сеансу буде вичерпаним або недійсним.

-Після успішного входу ідентифікатор сесії не повертається.

-Ідентифікатор сеансу та інші облікові дані надсилаються через незашифроване з’єднання.

Неправильно налаштована атака

Атаки помилкової конфігурації програми використовують переваги недоліків конфігурації, які є у веб-додатках. Це одна з уразливостей, яку важко визначити. Уразливість залежить не від програмного коду, а від того, як обробляються налаштування програми.

Неправильні конфігурації безпеки можуть виникати в програмах на всіх рівнях, включаючи платформи, веб-сервери, сервери додатків, бази даних і фреймворки. Тому неправильні конфігурації безпеки можуть варіюватися від конфігурації платформи до конфігурації дозволів облікового запису бази даних. Багато програм мають непотрібні та небезпечні функції, які ввімкнено за замовчуванням, наприклад функції налагодження та забезпечення якості. Коротше кажучи, ці небажані функції дозволяють хакерам обходити методи аутентифікації та отримувати доступ до конфіденційної інформації. Крім того, завдання за замовчуванням можуть включати відомі імена користувачів та паролі, попередньо запрограмовані облікові записи, спеціальні механізми доступу та неправильні налаштування файлів, до яких можна отримати доступ через веб-сервер [19].

Якщо ваша програма використовує веб-сервер, платформу, платформу програми, базу даних, мережу або містить код, ви ризикуєте неправильно налаштувати безпеку системи. IT-спільноті добре відомо, що неправильна конфігурація є найсерйознішою вразливістю, з якою сьогодні стикаються підприємства. Найпоширеніші неправильні конфігурації в традиційних центрах обробки даних включають:

- Конфігурація за замовчуванням, яка не змінилася і залишається небезпечною

-Неповна конфігурація, яка мала бути тимчасовою

-Невірні припущення щодо очікуваної поведінки та вимог до підключення програми.

Без належного рівня розуміння неправильні налаштування безпеки створюють нові ризики для неоднорідних середовищ.

Приклади неправильних налаштувань безпеки:

-Непотрібний порт керування доступний кожному

додаток. Це робить вашу програму вразливою для віддалених атак.

-Вихідні підключення до різних інтернет-сервісів. Можна виявити

Небажана поведінка програми у критичних середовищах.

-Застарілі програми, які намагаються взаємодіяти з програмами, які більше не існують. Зловмисник може імітувати ці програми, щоб встановити з'єднання[17].

Розкриття конфіденційних даних

Багато вразливостей можна класифікувати як витоки конфіденційних даних і мають спільне те, що вони включають випадкове виявлення конфіденційної інформації, яку необхідно зашифрувати. По-перше, конфіденційні дані – це інформація, яку можна використовувати або маніпулювати в зловмисних цілях, наприклад, номери банківських карток, облікові дані та номери платників податків. Коли користувач вводить інформацію у веб-додаток, ми впевнені, що сервер захистить ці конфіденційні дані. Однак у багатьох випадках уразливі програми взагалі не шифрують конфіденційні дані і зберігають їх у базах даних, які можуть бути скомпрометовані ін'єкцією SQL та іншими типами атак. Багато веб-додатків використовують слабкі алгоритми шифрування або прості хеші для захисту конфіденційних даних, тому навіть якщо використовується шифрування, цього може бути недостатньо [19].

Вплив конфіденційних даних може бути спричинений як зовнішніми, так і внутрішніми атаками. Незадоволені працівники становлять більшу загрозу, ніж сторонні, оскільки вони вже мають доступ до інформації компанії і мають право зловживати нею. Хмарне сховище є зручним варіантом для зберігання даних, але воно забезпечує відкриту платформу для атак, якщо не захищено належним чином.

Ці вразливості зазвичай дуже важко використовувати для хакерів, але вплив настільки серйозний, що дуже важливо зрозуміти архітектуру програми та прийняти правильні рішення. Більшість, якщо не всі підприємства використовують веб-додатки для своєї діяльності, тому дані постійно піддаються внутрішнім і зовнішнім загрозам. Цей ризик може принести компанії величезні витрати. Включає витрати на відновлення безпеки, витрати на оповіщення та підтримку постраждалих, а також вартість нормативних штрафів.

Відсутність контролю доступу на функціональному рівні

Цю вразливість можна віднести до категорії «відсутність контролю доступу на функціональному рівні», якщо обробники конфіденційних запитів неадекватні або недостатньо чутливі.

Уразливості контролю доступу на функціональному рівні можуть бути результатом неналежного захисту конфіденційних обробників запитів у програмі. Програма просто приховує доступ до конфіденційних дій, не надає достатньої функціональності для захоплення певної дії або випадково відкриває дію за допомогою параметра запиту, керованого користувачем. Можна зробити. Ці вразливості є набагато складнішими і можуть бути результатом ледь помітних екстремальних наслідків основної логіки програми [11].

Прикладом цієї вразливості є неавторизований користувач, який може отримати доступ до URL-адрес, що містять конфіденційну інформацію, або надати функціональність лише авторизованим користувачам. Іншим прикладом поширеного типу цієї уразливості є просто приховування функції від користувача, але щоб дозволити запит, користувач може зрозуміти, як її використовувати.

**Висновки за розділом 2**

Загроза безпеці ПК або серверу можуть представляти теплові атаки на центр обробки даних. Також розповсюдженими є основні види DDos: UDP-флуд – мережна атака типу «відмова в обслуговуванні», яка використовує режим безсеансного протоколу UDP. Полягає у надсиланні безлічі UDP-пакетів (як правило, великого об'єму) на певні або випадкові номери портів віддаленого хоста. Підмінивши IP-адреси джерел в UDP-пакетах, зловмисник може перенаправити потік ICMP-відповідей і тим самим зберегти працездатність атакуючих хостів, а також забезпечити їх анонімність. HTTP-флуд - це найпоширеніша flood атака. В її основі – посилання HTTP-запитів GET на 80-й порт. Розпізнання даної атаки можливе шляхом виявлення швидкого зростання кількості запитів до одного або кількох скриптів на сервері та швидкого зростання логів сервера. DoS (Denial of

Service - відмова в обслуговуванні) - хакерська атака на обчислювальну систему з метою довести її до відмови, тобто створення таких умов, за яких сумлінні користувачі системи не можуть отримати доступ до системних ресурсів (серверів).

За видами кібератаки на підприємство чи організацію можна розподілити на нецільові атаки (фішинг), цільові атаки (фінансове шахрайство, викрадення баз даних, DDoS-атаки, вимагання) та внутрішні атаки (викрадення, знищення інформації, сприяння цільовим атакам).

# РОЗДІЛ 3
# СПОСОБИ ЗАХИСТУ СЕРВЕРІВ ТА ЦОД. ПОРІВНЯЛЬНИЙ АНАЛІЗ ІСНУЮЧИХ МЕТОДОЛОГІЙ ДЛЯ ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

## 3.1 Способи захисту веб серверу від атак

Захист від ін'єкції SQL

1. Оновіть усі компоненти програмного забезпечення веб-додатків, включаючи бібліотеки, плагіни, платформи, програмне забезпечення веб-сервера та програмне забезпечення сервера баз даних, використовуючи останні доступні оновлення системи.

2. Використовуйте принцип найменших привілеїв ззовні

Посилання для підготовки облікового запису, який використовується для підключення до бази даних SQL. Наприклад, якщо вашому веб-сайту потрібно використовувати лише оператор SELECT для вилучення веб-вмісту з вашої бази даних, вам не потрібно надавати цьому обліковому запису будь-які інші дозволи, такі як INSERT, UPDATE або DELETE. У багатьох випадках ви можете керувати цими привілеями, використовуючи відповідну роль бази даних для вашого облікового запису. Також не дозволяйте веб-програм підключатися до бази даних з правами адміністратора (наприклад, під обліковим записом "sa" в Microsoft SQL Server).

3. Налаштуйте відповідні звіти про помилки та обробку у вашому веб-сервері та коді, щоб запобігти надсиланню повідомлень про помилки бази даних у веб-браузер клієнта. Зловмисник може використовувати технічні деталі повідомлення про помилку, щоб налаштувати запит на належну роботу.

4. Створіть спеціальні символи за допомогою escape-символів

Символ escape — це лише спосіб повідомити MySQL, що він є частиною самого рядка, а не одинарними лапками, які завершують рядок. Додайте символ зворотної косої риски, щоб сказати MySQL, що це безпечно.

5. Використовуйте брандмауер веб-програм (WAF) для веб-програм, які мають доступ до бази даних. Це допомагає визначити спроби розгортання SQL і може запобігти спробам розгортання SQL[33].

Захист від атак XSS

1. Використовуйте атрибут HttpOnly. Коли веб-сервер встановлює файл cookie, він може надавати додаткові атрибути, які заважають вам отримати доступ до файлу cookie за допомогою шкідливого JavaScript.

Set-Cookie: [ім'я] = [значення]; HttpOnly

HttpOnly гарантує, що файли cookie будуть надіслані лише до домену, з якого вони були створені.

2. Закодуйте вихідну змінну. Програми для запобігання атак XSS

Вам потрібно переконатися, що всі вихідні дані на сторінці закодовані, перш ніж їх повернути кінцевому користувачеві. Кодування вихідної змінної замінює HTML-розмітку на альтернативне представлення, яке називається «сутність». При використанні об'єкта браузер відображає об'єкт, але не запускає його. Наприклад, <script> стає & lt ;. Сценарій & gt ;.

Однак, коли веб-браузер знаходить об'єкт, він перетворюється на HTML і відображається, але не запускається. Наприклад, якщо зловмисник вставляє <script> alert </ script> у змінне поле на веб-сторінці сервера, сервер використовує цю стратегію для повернення <script> alert </ script>.

Коли веб-браузер завантажує зашифрований сценарій, зашифрований сценарій перетворюється на <script> попередження </ script> і відображається як частина веб-сторінки, але браузер не запускає сценарій.

3. Використовуйте транскордонні політики. Якщо ви використовуєте політику перетину кордону, усі автентифіковані користувачі на вашому сайті повинні повторно ввести свої облікові дані, перш ніж їм буде надано доступ до певних сторінок або служб на вашому сайті. Навіть якщо автентифікований користувач має

файл cookie, який дозволяє йому автоматично входити в систему, його налаштування змусить користувача повторно ввести своє ім’я користувача та пароль перед входом на певну сторінку [15].

Причина, чому ця стратегія ефективна для припинення атак XSS, полягає в тому, що вона суттєво обмежує здатність хакерів XSS захоплювати сеанси.

Захист від атак CSRF

Щоб захистити себе від CSRF, потрібно зробити дві речі. Переконайтеся, що запит GET не має побічних ефектів і що запит без GET надсилається лише з коду клієнта.

Щоб це забезпечити, вам потрібно:

1. REST — це набір принципів проектування, які призначають певні типи дій (перегляд, створення, видалення, оновлення) різним дієсловам HTTP. Наступний повний дизайн REST зберігає ваш код чистим і допомагає розвивати ваш сайт. Крім того, REST стверджує, що запити GET використовуються лише для відображення ресурсів.

Запит GET не має побічних ефектів, що обмежує шкоду, яку можуть завдати шкідливо створені URL-адреси. Зловмисникам потрібно докласти додаткових зусиль для створення шкідливих запитів POST.

2. Використовуйте жетони проти підробок. Навіть якщо ваші дії редагування обмежені запитами, які не є GET, вони не повністю захищені. Запити POST можна надсилати на ваш сайт зі скриптів або сторінок, розміщених в інших доменах. Щоб переконатися, що обробляються лише дійсні запити HTTP, кожна відповідь HTTP має містити секретний та унікальний маркер. Сервер також перевіряє цей маркер, коли він неодноразово повертається у запиті за допомогою методу POST (або будь-якого іншого методу). Крім GET, насправді.

Кожного разу, коли сервер відображає сторінку, яка виконує конфіденційну дію, сервер повинен записати підроблений маркер у приховане поле форми HTML. Цей маркер має бути включений у форму AJAX або подання виклику. Сервер повинен перевірити маркер, коли повертаються повторні запити, і відхилити відсутній або недійсний виклик. Токени для захисту від підробок зазвичай (строго) є

випадковими числами і зберігаються в файлі cookie або на сервері, коли записуються в приховане поле [23].

Сервер порівнює маркер, доданий до вхідного запиту, зі значенням, збереженим у файлі cookie. Якщо значення однакові, сервер приймає дійсний запит HTTP.

3. Переконайтеся, що файл cookie надіслано з атрибутом cookie

Команда SameSite Google Chrome додала новий атрибут до заголовка Set-Cookie, щоб запобігти CSRF. Атрибут cookie для одного сайту дозволяє розробникам вказувати браузеру контролювати, чи надсилаються файли cookie із запитами, ініційованими сторонніми доменами. Налаштувати атрибут "Same-Site" для файлу cookie дуже легко.

Set-Cookie: CookieName = CookieValue; SameSite = Lax;

Set-Cookie: CookieName = CookieValue; SameSite = Строгий;

Значення "Strict" означає, що браузер видаляє файл cookie за кожним запитом, ініційованим стороннім доменом у домені. Це найбезпечніший параметр, який запобігає спробам шкідливих сайтів виконувати шкідливі дії під час сеансу користувача [26].

Ви можете використовувати значення Lax, щоб прикріпити файл cookie до домену в запиті GET від стороннього домену, але дозволені лише запити GET. У цьому випадку, якщо посилання надходить з іншого сайту (наприклад, результати пошуку Google), користувачеві не потрібно повторно входити на сайт.

Захист від атак порушеної аутентифікації

1. Використовуйте безпечний вбудований менеджер сеансів

Сторона сервера. Генерує новий випадковий ідентифікатор сеансу з високою ентропією після входу. Ідентифікатор сеансу не повинен бути включений в URL-адресу, він повинен бути надійно збережений і він має бути недійсним після входу, бездіяльності та абсолютного часу очікування.

Для цього встановіть заборону надсилання сеансів через URL-адреси.

php.ini

session.use_trans_sid = 0;

session.use_only_cookies = 1;

php_flagsession.use_trans_sid вимкнено

php_flagsession.use_only_cookies увімкнено

2. Застосуйте відповідний інструмент керування безпекою паролів.

Важливим питанням є безпека паролів. Постійна політика паролів ускладнює або навіть унеможливлює вгадування паролів за допомогою ручних або автоматичних засобів.

Пароль має містити не менше 10 символів і не обмежувати максимальну довжину. Зазвичай 128 символів.

Складність пароля. Механізм паролів повинен мати можливість вводити майже будь-який символ, включаючи пробіл. Для додаткової складності паролі мають бути чутливими до регістру. Механізм зміни пароля вимагає мінімальної складності, яка має значення для програми та користувача. приклад:

-Пароль повинен відповідати (принаймні) 3 з 4 правил складності

- Принаймні одна заголовна літера (AZ).

- Принаймні один символ рядка (a-z);

- Принаймні одна цифра (0-9).

- Принаймні один спеціальний символ (не розділовий знак).

- Два або менше однакових символи поспіль

3. Вимкніть часто використовувані топології паролів. Вимагає мінімальних змін топології між старими та новими паролями.

4. Передавайте паролі лише через TLS або інший надійний канал.

Сторінка входу та всі наступні сторінки автентифікації повинні бути доступні лише через TLS або інший надійний канал. Домашня сторінка входу повинна обслуговуватися через TLS або інший надійний канал.

5. Запит на повторну аутентифікацію важливих функцій. Щоб зменшити ризик CSRF та перехоплення сеансів, важливо запитувати облікові дані перед оновленням конфіденційної інформації облікового запису (паролі, електронні листи, критичні транзакції).

Захист від неправильно налаштованих атак

1) Зменште площу поверхні вразливостей за допомогою ітераційного процесу

2) Підтримуйте програмне забезпечення в актуальному стані

3) Вимкніть усі облікові записи за замовчуванням та регулярно змінюйте паролі

4) Розробити потужну архітектуру програми для шифрування даних, що містять конфіденційну інформацію.

5) Переконайтеся, що параметри безпеки фрейма та бібліотеки встановлені на захищені значення.

6) Виконуйте періодичні перевірки та запускайте інструменти для виявлення отворів у системі

7) Використовуйте ту саму конфігурацію для виробництва, розробки та виробництва, оскільки невідповідність відкриває двері для багатьох неправильних конфігурацій.

8) Автоматизуйте систему, коли це можливо, щоб уникнути людської помилки.

9) Переконайтеся, що програмне забезпечення оновлено, включаючи операційну систему, програми, систему керування базами даних, веб-сервер або сервер додатків.

10) Переконайтеся, що немає облікових записів за замовчуванням і їх паролі змінено. Обліковий запис за замовчуванням завжди викликає проблеми.

11) Чи встановлено непотрібні чи потенційно небезпечні функції?

Наприклад, ваша програма може включати функції налагодження, які дозволяють зловмиснику обійти автентифікацію та отримати доступ до конфіденційної інформації.

12) Виконуйте періодичне сканування вразливостей та перевірки безпеки, щоб вчасно виявляти неправильні налаштування [31].

Важливий захист від витоку даних

1. Забезпечує шифрування даних і реалізує перевірену технологію шифрування. Шифруйте дані для визначення доступності: дуже важливо шифрувати дані, інформацію в збереженому форматі або в дорозі. Завжди шифруйте

конфіденційні дані. Це пояснюється тим, що такі дані не слід зберігати або надсилати у форматі простого тексту. Усі відкриті текстові дані є прямим запрошенням до зловмисника.

Обмежте доступ до законних груп користувачів лише шляхом ідентифікації даних, які потребують додаткового захисту, та використання шифрування на основі ключів.

Приклад №1: Шифрування кредитної картки Програма використовує автоматичне шифрування бази даних для шифрування номерів кредитних карток у базі даних. Однак це також означає, що ці дані будуть автоматично розшифровані після видалення, що є недоліком ін’єкції SQL і дозволить отримати номер кредитної картки відкритим текстом.

Таким чином, система повинна мати номери кредитних карток, зашифровані відкритим ключем, і матиме можливість розшифрувати їх за допомогою приватного ключа.

2. Використовуйте захищений шлюз аутентифікації. Захистіть свій сайт за допомогою безпечного протоколу HTTPS (SSL / TLS) і переконайтеся, що всі дані, які надсилаються між вашим браузером і веб-сервером, зашифровані та зберігаються конфіденційно. За допомогою SSL дані передаються за допомогою пари відкритих і закритих ключів.

Приклад №2: SSL використовується не для всіх аутентифікованих сторінок

Зловмисники просто відстежують мережевий трафік (наприклад, відкриті бездротові мережі) і крадуть сеанс cookie користувача. Потім зловмисник повторює цей файл cookie, щоб зафіксувати сеанс користувача та отримати доступ до особистих даних користувача.

3. Реалізуйте надійний алгоритм хешування паролів. Хакери можуть скористатися слабкими сторонами алгоритмів хешування паролів, щоб викрасти конфіденційну інформацію, що зберігається на веб-серверах або серверах додатків.

Щоб реалізувати хешування паролів, вам потрібно використовувати лише криптографічну хеш-функцію.

Приклад №3: База паролів зберігає паролі кожного користувача, використовуючи незбалансований хеш

Пошкодження файлу дозволяє зловмиснику отримати файл паролів.

Усі хеші без цукру можна опублікувати за допомогою Rainbow, використовуючи попередньо обчислені хеші [22].

Важливо відзначити, що шифрування також має свої недоліки. Тому не можна використовувати старі або слабкі алгоритми шифрування. Це дає лише помилкове відчуття безпеки. Те ж саме стосується простих хешів, які можна повернути. Дуже важливо надати новітні потужні стандартні алгоритми, протоколи та ключі, а також належне керування керуванням ключами.

Функціональний рівень

Для програми потрібен послідовний, простий для аналізу модуль затвердження, який викликається всіма бізнес-функціями. Часто такий захист забезпечується одним або кількома компонентами за межами програмного коду.

-Ми рекомендуємо завжди застосовувати правила заборони за замовчуванням. За замовчуванням заборонено доступ до всіх функцій програми та дозволено доступ лише користувачам та іншим частинам програми, яким це потрібно. Кожен запит має бути перевірений під час доступу, навіть якщо доступ до функціональних можливостей веб-додатка надано. Переконайтеся, що запит надходить від дійсного авторизованого користувача.

-Використовуйте списки контролю доступу та автентифікацію на основі ролей, щоб перевірити відповідність вищезазначеному. Використовуйте принцип найменших привілеїв. Тобто він надає доступ до функцій лише тоді, коли це необхідно. Не намагайтеся надати спільний доступ, а потім позбавити доступу користувачам, яким не слід ділитися.

-Не намагайтеся покладатися на безпеку щодо неоднозначності. Він просто приховує посилання на функції кнопок та інтерфейсів. Хтось зустріне спосіб отримати доступ до прихованих функцій. Крім того, будь-хто, хто намагається атакувати веб-додаток, ігнорує інтерфейс користувача та надсилає запити

безпосередньо, щоб отримати відповіді від внутрішніх програм і механізмів баз даних.

-Перевірте всі URL-адреси, кнопки та інші способи доступу до функцій вашої веб-програми за допомогою облікового запису з низькими привілеями.

Перевірте, чи можете ви отримати доступ до функцій, до яких не мають доступу ці облікові записи. Існують інструменти, які допоможуть вам з цим завданням. Порівняйте вигляд веб-програми під час автентифікації як користувача з правами адміністратора зі скануванням під час автентифікації як звичайний користувач. Потім виділіть частини програми, які не повинні бути доступні звичайним користувачам.

Більшість веб-програм не показують посилань або кнопок для несанкціонованих функцій, але цей «контроль доступу на рівні презентації» насправді не забезпечує жодного захисту.

### 3.2 Шляхи забезпечення захисту ЦОД

Підхід до забезпечення безпеки інформації, що зберігається (обробляється) в ЦОД, повинен базуватися на вимогах конфіденційності, доступності та цілісності. Слід зазначити, що метою забезпечення безпеки та захисту функціональних додатків, сервісів та даних (інформації) є зниження до прийнятного мінімуму (аналіз ризику, схильність до ризику) збитків від можливих зовнішніх і внутрішніх впливів [31].

Основні етапи безпеки ЦОД:

- побудова моделі загрози;
- виділення об'єктів, на які можуть бути спрямовані загрози;
- побудова моделі дій правопорушника;
- оцінка та аналіз ризиків;
- розробка та впровадження методів і засобів захисту в системах ЦОД. Аналізуючи вищевикладене, стає зрозумілим, що без побудови Системи підтримки

ІБ (системи інформаційної безпеки) як вся організація, так і елемент (ЦОД) не можуть забезпечити необхідний захист. Основні об'єкти захисту в дата-центрі:

- інформація, що циркулює в системі;
- обладнання (елементи);
- програмне забезпечення.

Кібератаки – це всі види діяльності, які потребують ретельної підготовки, наприклад, соціальна інженерія, сканування мережі та виявлення слабких місць. Як правило, DDoS-атака — це коли це цілеспрямована команда одного із зловмисників, і, можливо, це частина загального плану дестабілізації. Наприклад, виключити сайт з індексу пошуку [22].

Все залежить від мети атаки. Для комерційних сайтів-агрегаторів метою атаки може бути призупинення доступу до сайту на тиждень (!). Це змушує сайт «виходити» з індексу пошуку. Наприклад, нещодавно атакували клієнтський сайт Avtobazar.ua. Адміністратори сайту витрачають багато часу (до місяця), щоб відновити своє попереднє місцезнаходження в пошукових системах перед атакою. Це катастрофа для сайту-агрегатора. Адже основний контент (оголошення про продаж авто) генерує сам користувач. Після вищезгаданої атаки придбане у провайдера управління сайтом «Автобазар» є найвищим ступенем захисту від DDOS в дата-центрі Cosmonov і не шкодує ні про що.

Існує сценарій DDos-атаки. Відправник ділить пакет «даних» на 10 частин, а одержувач отримує один пакет і чекає ще 9 частин, щоб відобразити отриманий пакет, щоб зарезервувати місце... Через це на сервері пам’яті не вистачає. Насправді такі пакети «даних» складаються з кількох кілобайт і призначені для блокування сервера.

Існує багато атак, пов’язаних із шифруванням, які вимагають багато часу та потужності ЦП від жертви. Зловмисник надсилає запит на встановлення зашифрованого з'єднання, більше нічого зробити не може, а одержувач починає підраховувати хеші і приймає ключ, що забезпечує його живлення, проти фальшивих запитів [11].

Важливі правильні налаштування безпеки IT

Крім того, ми не можемо усунути людський фактор, який зводить нанівець усі зусилля щодо захисту IT-межів. Часто мова йде не про якість рішень безпеки, а про їх правильне використання та налаштування. Це означає, що навіть після придбання дорогого рішення безпеки, сам інструмент необхідно використовувати з відповідною версією IT-інфраструктури.

Є прості атаки, складні атаки, дешеві атаки та дорогі атаки. Все це залежить від конкретного сайту чи служби, яку ви хочете атакувати. У найпростішому випадку канал клієнта забитий. У разі складної атаки з клієнтом щось відбувається програмно. Крім того, хакери, які продають DDos-атаки, часто атакують щось, щоб показати «свої можливості» потенційним клієнтам [22].

Як уже згадувалося, DDos-атаки можуть бути частиною загального плану зловмисника - як відволікання. Наприклад, після злому облікового запису компанії та виведення грошей мережа вимикається, компанія, яка стала жертвою злому, нічого не може зробити, і не може «наздогнати» вкрадені гроші.

Соціальна інженерія як засіб злому

У багатьох випадках зловмиснику не потрібно нічого зламати. Засобом «злому» є соціальна інженерія, прорахунок поведінки людей. Наприклад, якщо біля офісу «загубилася» флешка, хтось із співробітників її знайде, запрацює і вставить флешку. Не потрібно зламати. Мережа компанії заражена зсередини. , Обійти всі системи захисту. А великі корпоративні мережі не працюють. Характерною рисою цих «інфекційних хвороб» є те, що, як правило, всі вони інфіковані і їх дуже важко вилікувати. Мережні пристрої завжди повторно заражають один одного. Адміністратор відремонтував деякі хости, оскільки вони швидко повторно заразилися. Наприклад, таким чином була заражена серія електронних супермаркетів і МВТ/КВТ. IT-відділу довелося терміново розділити всю мережу на карантинні сектори і «ремонтувати» всі пристрої по черзі. Робота всієї великої компанії була паралізована на два дні. Втрати можна порахувати, оскільки адміністратор повинен був фізично видалити пристрій з вірусу, щоб очистити саму віддалену касу.

Так звані вразливості нульового дня також можуть бути небезпечними. Іншими словами, це «дірка», яка ще не відома виробнику як уразливість. Крім того, якщо ви відкриваєте файл PDF з Інтернету, ви не можете бути впевнені, що ваша IT-система в порядку. Та ж Microsoft завжди надсилає виправлення, щоб виправити вразливість, але просто думає, що адміністратору не потрібно встановлювати оновлення на парк машин, які залишаються вразливими до зовнішніх і внутрішніх загроз.

Основні види захисту від DDos

Перший рівень захисту: блокує атакувану IP-адресу. Використовуйте цей параметр, щоб повністю заблокувати весь трафік на певну IP-адресу клієнта. Сайт клієнта буде недоступний, доки його не буде захищено або не завершиться атака. Однак інші IP-адреси в клієнтській інфраструктурі переймуть і повністю відновлять доступ до даних сайту / служби / клієнта. Дані будуть відправлені колектору. Збірник може вирішити, чи повністю блокувати весь трафік на певний IP-адрес клієнта на основі ключових параметрів (пакетів, отриманих за секунду, або обсягу трафіку в Мбіт/с). Замки можна швидко встановити або зняти вручну.

Другий варіант захисту: ця послуга, яка блокує весь трафік на певний IP-адрес клієнта на певному порту, збільшує вартість каналу зв'язку клієнта, і жертви недоступні з усього Інтернету на забороненому порту. , Решта адрес клієнта буде продовжувати нормально працювати [24].

І, нарешті, третій і найбільш просунутий варіант: спеціальні програмно-апаратні рішення аналізують вхідні запити на розпізнавання атак. Першим кроком є фільтрація запитів, які не проходять алгоритм автентифікації браузера. Далі включається кілька алгоритмів для різних перевірок вхідного трафіку. Ці пристрої розміщуються в мережевих розривах і аналізують трафік без затримок. У той же час цей захист дозволяє прозоро проходити автоматизовані скрипти, оголошені як пошукові боти.

## 3.3 Аналіз структури системи реагування на комп'ютерні надзвичайні події

Використання вразливостей, які можуть включати веб-додатки, може призвести до втрати даних і пошкодження як звичайних користувачів, так і великих підприємств, тому проаналізуйте вітчизняні організації з кібербезпеки, щоб захистити веб-додатки. Важливо визначити найкращі методи, які слід зробити [32].

В Україні діє група реагування на надзвичайні ситуації, відома як CERT-UA. CERT-UA — структурний підрозділ національних служб спеціального зв'язку та захисту інформації в Україні. CERT-UA займається:

-Збирайте та аналізуйте дані про кіберзагрози.

-Оцінка безпеки державних інформаційних ресурсів.

-Участь у Форумі реагування на інциденти інформаційної безпеки.

- Підтримка боротьби з кіберзагрозами.

Також зазначимо, що тривають роботи щодо створення центру (ЦЕРТ-НБУ) з реагування на інциденти у сфері кібербезпеки в українській банківській системі та платіжному секторі, повідомляється на сайті Національного банку України.

Крім того, як повідомляється на сайті Дніпровської місцевої асамблеї, у Дніпрі відкрито центр реагування на кібератаки з експертами з інформаційних технологій [33].

Додамо, що українська стратегія кібербезпеки була затверджена 15 березня 2016 року. Таким чином, Україна зараз має систему реагування на кіберінциденти, покращеннями якої є розробка нормативно-правових актів, спрямованих на кібербезпеку, та розвиток технологій для захисту від кібератак. Форум FIRST (Група реагування на інциденти та безпеки), який складається з груп реагування на надзвичайні ситуації, був створений для обміну інформацією про кіберінциденти. FIRST Forum розробив систему оцінки вразливості CVSS.

Спільна система оцінки вразливостей (CVSS) є відкритою основою для передачі характеристик і серйозності вразливостей програмного забезпечення. CVSS складається з трьох груп показників: базової, тимчасової та навколишнього. Базова

група представляє внутрішню якість уразливості, тимчасова група представляє характеристики вразливості, які змінюються з часом, а група «тимчасова» представляє характеристики вразливості, характерні для середовища користувача. Базові показники дають оцінки в діапазоні 0-10. Оцінки можна змінювати, оцінюючи тимчасові та екологічні показники [21].

Оцінка CVSS також відображається у вигляді векторного рядка, який є стислим текстовим представленням значень, які використовуються для відображення оцінки.

Основні показники пояснюють складність використання вразливості,

Потенційна шкода конфіденційності, цілісності та доступності інформації

індекс:

1. Вектор атаки – це відстань від потенційного нападника до вразливого об’єкта. Можливі значення показників: мережа (N), суміжна мережа (A), локальна (L), фізична (P)

2. Складність використання вразливості – це якісна оцінка складності атаки. Чим більше умов необхідно виконати для використання вразливості, тим складнішою вона стає. Можливі значення показників: низький (L), високий (H)

3. Аутентифікація / Необхідні рівні привілеїв. Визначте, чи вимагає атака аутентифікація, і якщо так, то яка. Можливі значення показників: високий (H), низький (L), немає (N)

4. Необхідно взаємодіяти з користувачем-Визначити, чи вимагає атакувана система дії користувача для успішної реалізації атаки. Можливі значення показників: Немає (N), Обов’язково (R)

5. Ліміти експлуатації. Визначте, чи відрізняються експлуатовані та атакувані компоненти, тобто чи порушує експлуатація вразливості конфіденційність, цілісність та доступність інших компонентів системи. Можливі значення показників: незмінно (U), змінено (C)

6. Показники впливу – Визначає ступінь впливу на конфіденційність, цілісність та доступність компонента, що атакується. Можливі значення метрики: середній (M), високий (H).

Рейтинг присвоюється відповідно до значення рейтингу вразливості.

Отже, представлена система дає змогу оцінити серйозність уразливості. У цьому документі використовуються оцінки CVSS 3.0 для аналізу існуючих методологій.

### 3.4 Порівняння існуючих методів для тестування інформаційної безпеки

На даний момент найпоширенішими методами тестування на проникнення є:

- Посібник з методології тестування безпеки з відкритим кодом (OSSTMM);

- Спеціальна публікація Національного інституту стандартів і технологій (NIST) 800-115;

- Посібник з тестування OWASP;

-Стандарт виконання тесту на проникнення (PTES);

- Структура оцінки безпеки інформаційної системи (ISSAF).

- BSI-Дослідження моделей для тестування на проникнення.

Існує  такоє загальний регламент про захист даних (GDPR) – це регламент у законодавстві ЄС про захист даних та конфіденційність у Європейському Союзі (ЄС) та Європейському економічному просторі (ЄЕЗ). Він також стосується передачі особистих даних за межі країн ЄС та ЄЕЗ. Основна мета GDPR – надати контролю особам над їхніми персональними даними та спростити регуляторне середовище для міжнародного бізнесу шляхом уніфікації регулювання в ЄС[30].

1. Методологічний аналіз Посібник з методології тестування безпеки з відкритим кодом (OSSTMM)

Це досить формалізований і структурований документ для мережевого тестування. Цей документ має так звану «карту безпеки» (візуальний індикатор безпеки). Ця карта показує ключові зони безпеки. Він містить набір елементів, які необхідно перевірити на методичну відповідність.

Цей документ містить підпункти «Методика» / «Тестування технології безпеки Інтернету» / «Огляд мережі» / «Тестування брандмауерів». Працює з охороною. Він також описує конкретну правильну реакцію мережі на атаку та її

існування, наприклад, вимірювання часу відповіді пакету та перевірку на втрату пакетів на маршруті до цілі.

Переваги цієї техніки:

-Детальний опис процедури підготовки до тесту

- Детальний метод тестування та підхід

-Детальне пояснення основних термінів і понять у сфері інформаційної безпеки.

Недоліками цього методу є:

-формат;

-додаткового пояснення вимоги немає.

-остання безкоштовна версія OSSTMM V3 була випущена в 2010 році і частково застаріла. Остання версія доступна лише платним учасникам.

-Не містить опису інструментів, які потрібно використовувати для цього[30]

2. NIST Special Publications Methodology 800-115 Технічний посібник з тестування та оцінки інформаційної безпеки. Створено та підтримується NIST принаймні три етапи оцінки інформаційної безпеки: планування, впровадження, постоперація (аналіз даних, визначення причини вразливості, надання рекомендацій щодо усунення вразливості та написання звіту). )) Ідентифікує. У розділі «Методики оцінки вразливості цілі» описано тестування на проникнення, або фазове та логістичне тестування, як один із методів. Згідно з цим документом, тестування на проникнення може бути використане для визначення наступного, крім стандартних функцій:

-Наскільки добре система витримує реальну модель атаки?

-Типова складність, яку повинен подолати нападник.

-Додаткові заходи, які можуть зменшити загрозу для системи.

-Здатність виявляти атаки, захищати вашу систему та забезпечувати належну реакцію на атаки

Переваги цього документа:

-Загальний опис методів безпеки комп'ютерної системи та їх короткий опис (перевірка мережі, перевірка файлів журналів, перевірка конфігурації системи,

перевірка цілісності файлів, сканування уразливостей, сканування бездротової мережі тощо).

-Посилання на програмні продукти, які необхідно використовувати для тестування-Посилання на інші нормативні документи та методики.

Недоліками цього методу є:

-Цей документ був створений у 2008 році

– На даний момент це не відповідає поточному стану розвитку

Як проникнути в інформаційні технології та комп'ютерні мережі [35].

3. Методологічний посібник із тестування OWASP (Open Web Application Security Security Project). OWASP (Open Web Application Security Project) — міжнародна відкрита спільнота, яка зосереджена на покращенні безпеки програмного забезпечення. Кожен має право брати участь в OWASP, і всі матеріали розповсюджуються вільно. Посібник з тестування OWASP є ширшою методологією, ніж будь-який інший метод, оскільки він надає рекомендації щодо аналізу веб-додатків загалом (наприклад, вихідний код), а не лише тестування на проникнення. Ця техніка зосереджена на виявленні вразливостей Інтернету. -застосування.

Переваги цього документа:

-Керування OWASP надає всю необхідну інформацію для кожного етапу життєвого циклу розробки безпечного програмного забезпечення.

Найпопулярніша і повна колекція інструментів тестування безпеки веб-додатків, доступних в Інтернеті.

Недоліками цього методу є:

-Якщо ваш корпоративний веб-сайт або веб-додаток не є важливими з точки зору бізнесу, тестування проникнення за допомогою цієї методології не підходить.

4. Методологія PTES-Стандарти виконання тесту на проникнення-Технічні вказівки. Стандарт, розроблений для поєднання як бізнес-вимог, так і функцій безпеки, а також масштабованості тестування на проникнення. На початковому етапі підготовки детально вивчаються встановлені канали зв'язку, правила взаємодії та контролю, конкретні реакції та моніторинг інцидентів. Висвітлено наступні кроки.

-Збір інформації;

-Моделювання загроз;

- Метод аналізу вразливостей;

-Зловживання - Забезпечте контрзаходи та уникнення виявлення

Найкращий спосіб атакувати.

-Аналіз інфраструктури після зловживання, подальше проникнення, видалення та життєздатність.

-За результатами визначити структуру звіту

Тест Переваги цього методу:

-Технічне керівництво з детальною технічною інформацією щодо інструментів та команд на кожному етапі тестування на проникнення

Недоліками цього методу є:

-Недостатньо уваги приділено використанню методів соціальної інженерії.

5. Структура оцінки безпеки методології-інформаційної системи ISSAF. Призначений для внутрішнього аудиту. Цей документ охоплює величезну кількість питань, пов’язаних з інформаційною безпекою [28].

Є розділи, які описують оцінки безпеки для брандмауерів, маршрутизаторів, антивірусних систем тощо. Методологія ISSAF моделює вимоги заходів внутрішньої безпеки та зосереджується на оцінці безпеки комп’ютерних мереж, систем і додатків. Ця методологія більше зосереджується на безпеці комп’ютерної системи, ніж PTES, і точно визначає, як використовувати інструмент. Система оцінки безпеки інформаційних систем (ISSAF) розділена на дві частини: технологію та управління. Технічна частина містить найважливіший набір правил і процедур для створення відповідного процесу оцінки безпеки. Інструкція містить загальні рекомендації щодо створення ефективного процесу тестування.

Переваги даної методики:

-Допомагає закрити розрив між техніками з тестування безпеки та адміністраторами та впроваджує засоби контролю, необхідні для ефективного керування обома сторонами.

Недоліками цього методу є:

-Ця методологія застаріла (2005 р.).

6. Методологія BSI-Study тестових моделей на проникнення. Розроблено німецьким відділом федерального уряду з інформаційної безпеки. У цьому документі описано правильний тест на міцність для вашої системи. Важливо, що сам метод тестування детально описано, а й описує вимоги, правові аспекти методології та процедури, які необхідно виконати для успішного виконання тесту. Показано класифікацію випробувань на міцність та визначені їх критерії [12].

Переваги даної методики:

-Методологія дуже детальна і намагається передбачити всі аспекти як технічних, організаційних, так і юридичних тестів на міцність.

-Додаток містить опис програмного забезпечення, яке можна використовувати для перевірки об’єктів, описаних у методиці.

**Висновки за розділом 3**

Суть безпеки інформації – забезпечити безперебійну роботу організації та звести до мінімуму збиток від подій, що таять загрозу безпеки, за допомогою їхнього запобігання і зведення наслідків до мінімуму.

В основі захисту від вірусів є високі знання і розуміння правил безпеки, належні засоби управління доступом до систем.

Варто застосовувати для перевірки комп’ютерів і носіїв інформації, Серверів на наявність відомих вірусів або як запобіжний захід, або як повсякденна процедура.

Програмні засоби виявлення змін, внесених у дані, повинні бути по необхідності інстальовані на комп’ютерах для виявлення змін у виконуваних програмах. Обслуговування систем. Заходи для обслуговування систем вимагаються для підтримки цілісності і доступності сервісів.

Для забезпечення можливості відновлення всіх критично важливих виробничих даних і програм після виходу з ладу комп’ютера або відмови носія інформації - сервера, необхідно мати належні засоби резервного копіювання.

Сьогодні найпоширенішими методами тестування на проникнення є: Посібник з методології тестування безпеки з відкритим кодом (OSSTMM); Спеціальна публікація Національного інституту стандартів і технологій (NIST) 800-115; Посібник з тестування OWASP; Стандарт виконання тесту на проникнення (PTES); Структура оцінки безпеки інформаційної системи (ISSAF). BSI-Дослідження моделей для тестування на проникнення.

# ВИСНОВКИ

В умовах цифрового суспльства та розвитку нових ІТ-технологій, поняття інформаційної безпеки значно розширилося. Підкреслюючи значущість інформаційної безпеку, відзначають комплекс заходів, покликаний зменшити число ймовірних шкідливих сценаріїв чи розмір збитків, яких може зазнати підприємство чи організація у разі розголошення чи викрадення конфіденційної інформації.

Інформаційна безпека – це економічний параметр, який повинен враховуватися у роботі підприємства чи організації, а інформацію та дані це цінність, що підлягає захисту. Вона має бути доступною лише для авторизованих користувачів чи програм. Інформаційна безпека – збереження конфіденційності, цілісності та доступності інформації; крім того, можуть враховуватися інші властивості, такі, як автентичність, відстежуваність, неспростовність та надійність.

Захист серверів і центрів обробки даних від зловмисників залежить від технологій і компонентів, які використовуються для створення веб-додатків, і потенційних вразливостей у цих компонентах. Існують різні категорії вразливостей, і кожна атака уразливістю має свої особливості, але оскільки причиною вразливості є помилка розробки, впровадження та застосування компонента веб-додатка, пошук уразливості. відповісти на інформацію про це місце. Як в Україні, так і в інших країнах світу організовано бригади реагування на надзвичайні ситуації з експертів та дослідників. Проте деякі міжнародні стандарти контролюють процес розкриття вразливостей. Наприклад Одним із таких обмежень, наприклад, є набір правил GDPR. Загальний регламент про захист даних (GDPR) – це регламент у законодавстві ЄС про захист даних та конфіденційність у Європейському Союзі (ЄС) та Європейському економічному просторі (ЄЕЗ). Він також стосується передачі особистих даних за межі країн ЄС та ЄЕЗ. Основна мета GDPR – надати контролю особам над їхніми персональними даними та спростити регуляторне середовище для міжнародного бізнесу шляхом уніфікації регулювання в ЄС. Існує сім ключових принципів, що лежать в основі GDPR: Законність, справедливість та

прозорість. Обмеження призначення. Мінімізація даних. Точність. Обмеження зберігання. Цілісність та конфіденційність (безпека). Підзвітність.

Для пошуку вразливостей можна використовувати різноманітні інструменти, але ефективність їх використання залежить від алгоритму дії, що виконується цим пошуком. Алгоритми дій можуть бути представлені у вигляді таких спеціальних методів і охоплюють широкий спектр питань кібербезпеки, включаючи тестування безпеки фізичних середовищ, операційних систем і бездротових мереж. Іншими словами, для аналізу існуючих методів і вибору таких методів потрібен додатковий час. компонент.

Тому необхідно враховувати міжнародні досягнення в тестуванні веб-додатків і розробляти методи тестування на проникнення, які включають перелік можливих інструментів тестування. Однак більшість методів охоплюють широкий спектр питань кібербезпеки і вимагають додаткового часу для аналізу вразливостей відповідно до існуючих методів і вибору компонентів, які особливо підходять для тестування веб-додатків.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Закон України «Про інформацію» № 2657-XII від 2 жовтня 1992 року.

2. Закон України «Про захист інформації в інформаційно-телекомунікаційних системах».

3. Аль-Хакамі Алі Мохаммед Омар. Моделі процесів функціонування корпоративних центрів обробки даних : дисертація ... кандидата технічних наук : 05.13.01 / Аль-хакамі Алі Мохаммед Омар.

4. Послуги ЦОД (дата-центрів). URL: http://marketing.rbc.ru/reviews/it-business/chapter_6_1.shtml.

5. Sankar R. Burpsuite - A Beginner’s Guide For Web Application Security or Penetration Testing [Electronic resource] / Ravi Sankar. - 2018. - Mode of access to the resource: https://kalilinuxtutorials.com/burpsuite/.

6. [Electronic resource] / Archana Choudhary // Security Zone. - 2019. - Resource access mode: https://dzone.com/articles/sql-injection-attacks-know-how-toprevent-them.

7. Brewer J. Web Server Vulnerabilities and a Defense in Depth Strategy Using the Squid Proxy [Electronic resource] / Jim Brewer // GSEC Practical version 1.4b. - 2004. - Resource access mode: https://www.giac.org/paper/gsec/3729/web-server-vulnerabilities-defense-in-depthstrategy-squid-proxy/105970.

8. CALYPTIX. - 2017. - Resource access mode: https://www.calyptix.com/topthreats/top-8-network-attacks-type-2017/.

9. Choudhary A. SQL Injection Attacks: Know How to Prevent Them.

10. Cobb M. Cross-site scripting explained: How to prevent XSS attacks [Electronic resource] / Michael Cobb // 2009 - Resource access mode: https://www.computerweekly.com/tip/Cross-site-scripting- explained-How-toprevent-XSS-attacks.

11. Common Website Security Vulnerabilities [Electronic resource] // Common places. - 2019. - Mode of access to the resource:https://www.commonplaces.com/blog/6-common-website-security-vulnerabilities/.

12. Cross Site Scripting (XSS) Attack Tutorial With Examples, Types &Prevention [Electronic resource]. - 2019. - Resource access mode: https://www.softwaretestinghelp.com/cross-site-scripting-xss-attack-test/.

13. Cross-site Scripting (XSS) [Electronic resource]. - 2018. - Resource access mode: https://www.owasp.org/index.php/Cross-site_Scripting_(XSS).

14. Evteev D. SQL Injection from A to Z [Electronic resource] / Dmitry Evteev. - 2008. - Resource access mode: https://www.ptsecurity.com/upload/corporate/ru-ru/analytics/PT-devteev-AdvancedSQL-Injection.pdf.

15. Excess XSS [Electronic resource]. - 2016. - Mode of access to.

16. Ganore P. What Is A Web Server And How Does It Function? [Electronic resource] / Pravin Ganore. - 2017. - Resource access mode: https://www.milesweb.com/blog/hosting/web-server-function/.

17. How a Web server functions? [Electronic resource]. - 2006. - Resource access mode: https://www.eukhost.com/blog/webhosting/how-a-web-serverfunctions/.

18. How to Prevent SQL Injection Attacks [Electronic resource] // eSecurityPlanet. - 2018. - Resource access mode: https://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks.html.

19. How to Protect Against SQL Injection Attacks [Electronic resource] // UC Berkeley. - 2019. - Mode of access to the resource:https://security.berkeley.edu/education-awareness/best-practices-how-articles/systemapplication-security/how-protect-against-sql.

20. Melnick J. Top 10 Most Common Types of Cyber Attacks [Electronic resource] / Jeff Melnick. - 2018. - Resource access mode: https://blog.netwrix.com/2018/05/15/top-10-most-common-types-of-cyber-attacks/.

21. Methods of protection against CSRF-attack [Electronic resource]. - 2016. - Resource access mode: https://habr.com/ru/post/318748/.

22. Mietek Glinkowski. Data center defined. URL: https://library.e.abb.com/public/37d42b7f0a7eb124c1257c5a003f8425/06-10%204m301_EN_72dpi.pdf.

23. Resource access mode: https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/SQL_Injection _Prevention_Cheat_Sheet.md.resource: https://excess-xss.com.

24. Ricca F. https://dl.acm.org/citation.cfm?id=381476 [Electronic resource] / F. Ricca, P. Tonella. - 2001. - Resource access mode: https://dl.acm.org/citation.cfm?id=381476.

25. Singh S. 5 Practical Scenarios for XSS Attacks [Electronic resource] / Satyam Singh // Pentest Tools. - 2018. - Mode of access to the resource: https://pentesttools.com/blog/xss-attacks-practical-scenarios/.

26. SQL injection. Check, hacking, protection [Electronic resource] // BVN2. - 2011. - Mode of access to the resource: https://habr.com/ru/post/130826/.

27. SQL_Injection_Prevention_Cheat_Sheet [Electronic resource] - The Green Grid Association URL: http://www.thegreengrid.org/.

28. Top 8 Network Attacks by Type in 2017 [Electronic resource] // The Proposed EU General Data Protection Regulation. A guide for in-house lawyers, Hunton & Williams LLP, June 2015, p. 14.

29. The Proposed EU General Data Protection Regulation. A guide for in-house lawyers, Hunton & Williams LLP, June 2015, p. 14.

30. Vulnerabilities of web applications [Electronic resource]. - 2019. - Mode of access to the resource:ttps: //www.ptsecurity.com/upload/corporate/ruru/analytics/Web-Vulnerabilities-2019-rus.pdf.

31. Web Server and its Types of Attacks [Electronic resource]. - 2012. - Resource access mode: https://www.greycampus.com/opencampus/ethicalhacking/web-server-and-its-types-of-attacks.

32. Web Server Vulnerabilities Attacks: How to Protect Your Organization [Electronic resource] // Tech Funnel. - 2018. - Resource access mode:

https://www.techfunnel.com/information-technology/web-server-vulnerabilitiesattacks-how-to-protect-your-organization/.

33. What is a web server [Electronic resource]. - 2019. - Resource access mode: https://developer.mozilla.org/ru/docs/Learn/%D0%A7%D1%82%D0%BE_%D1%82% D0% B0% D0% BA% D0% BE% D0% B5_% D0% B2% D0% B5% D0% B1_% D1% 81% D0% B5% D1% 80% D0% B2% D0% B5% D1% 80.

34. Zilberstein O., Lyashenko M., Shklyar T. Data centers: trends and development prospects // International Journal of Applied Engineering Research. 2015. т. 10. №24. 45350-45359.