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

дипломної 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: «Удосконалений метод захисту серверів багатокористувацьких ігор»

Виконавець: студент IV курсу, групи КБ-41

_____ Юрій СЕВАСТЬЯНОВ
(підпис) (ім'я прізвище)

	Прізвище, ініціали	**Підпис**
Керівник	Олександр ЛАПТЄВ	

Нормоконтроль	Олександр ТОРОШАНКО	

Київ 2022

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ЗАТВЕРДЖЕНО:
завідувачка кафедри кібербезпеки
та захисту інформації
_____ Наталія ЛУКОВА-ЧУЙКО
«01» листопада 2021 р.

ЗАВДАННЯ
на виконання дипломної роботи

спеціальності 125 Кібербезпека
(код і назва спеціальності)

освітньої програми Кібербезпека
(назва освітньої програми)

Студентові **КБ-41** **Севастьянову Юрію Володимировичу**
(група) (прізвище ім'я по-батькові)

Тема дипломної роботи Удосконалений метод захисту серверів багатокористувацьких ігор

1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

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

2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Характеристика видів атак на сервери багатокористувацьких ігор, Засоби захисту серверів для багатокористувацьких ігор

3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Класифікація багатокориствуцьких ігор та шахрайств в них, засоби захисту серверів
від кібератак, сучасні методи та засоби захисту ігор

4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Практична цінність Підготовка апаратного та програмного обладнання з блокуванням можливості для створення декількох вікон додатку водночас

5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 01 листопада 2021 року

Завдання видав _____ Олександр ЛАПТЄВ
(підпис) (ініціали, прізвище)

Завдання прийняв до виконання _____ Юрій СЕВАСТЬЯНОВ
(підпис) (ініціали, прізвище)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів робіт	Строки виконання робіт (початок-кінець)	Відмітка про виконання
1	Уточнення постановки задачі	29.10.2021 – 26.01.2022	*виконано*
2	Аналіз літератури	27.01.2022 – 20.02.2022	*виконано*
3	Аналіз видів багатокористувацьких ігор	21.02.2022 – 03.03.2022	*виконано*
4	Дослідження методів атак на сервери багатокористувацьких ігор	04.03.2022 – 30.03.2022	*виконано*
5	Розгляд методів захистів від атак на сервери	31.03.2022 – 17.04.2022	*виконано*
6	Вибір реалізації атак та механізмів захисту	18.04.2022 – 07.05.2022	*виконано*
7	Реалізація системи захисту серверу	08.05.2022 – 26.05.2022	*виконано*
8	Оформлення пояснювальної записки	27.05.2022 – 08.06.2022	*виконано*
9	Підготовка до захисту	09.06.2022 – 13.06.2022	*виконано*

Завдання видав _____ Олександр ЛАПТЄВ
(підпис) (ініціали, прізвище)

Завдання прийняв до виконання _____ Юрій СЕВАСТЬЯНОВ
(підпис) (ініціали, прізвище)

Термін подання дипломної роботи до ЕК 06 червня 2022 року

РЕФЕРАТ

Пояснювальна записка дипломної роботи складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел та додатків. Основний текст займає 52 сторінок, включає в себе зміст, вступ, три розділи дипломної роботи, висновки та список джерел. Використаних джерел включає 21 У пояснювальній записці дипломної роботи міститься 6 рисунків.

Метою роботи є аналіз методів захисту серверів багатокористувацьких ігор

Об'єктом дослідження є процес захисту серверів багатокористувацьких ігор

Предметом дослідження є засоби та механізми, які реалізують захист серверів багатокористувацьких ігор

Методи дослідження дипломної роботи:

- аналіз літератури;
- аналіз документів;
- системний підхід;
- методи порівняння;
- структурний аналіз

Для досягнення зазначеної мети поставлено наступні завдання:

- Провести характеристику поняття ігор та їх класифікацію;
- Вивчити види багатокористувацьких ігор та шахрайства в них;
- Розібрати способи захисту веб серверів;
- Охарактеризувати види нелегального доступу в іграх;
- Привести приклади технічних засобів захисту ігор;
- Вивчити сучасні методи захисту від несанкціонованого доступу і мети його застосування;
- Розібрати підготовку програмного

Практичною цінністю отриманих результатів є підготовка апаратного та програмного обладнання та блокування можливості атаки без реєстрації користувача із використанням авторського додатку.

Ключові слова: кібератака, захист інформації, багатокористувацькі ігри, захист серверів, переповнення буфера, класифікація ігор

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ПЗ	–	Програмне забезпечення
XSS	–	Cross Site Scripting
API	–	Application Programing Interface
DNS	–	Domain Name System
URL	–	Uniform Resource Locator
DOS	–	Denial of Service
SQL	–	Structured Query Language
DRM	–	Digital Rights Management
LAN	–	Local Area Network
MMO	–	Massively Multiplayer Online
HTTP	–	Hypertext Transfer Protocol
HTTPS	–	Hypertext Transfer Protocol Secured
RPG	–	Role Playing Game
IP	–	Internet Protocol
SSL	–	Secure Sokets Layer
CMS	–	Content Management System
DDoS	–	Distributed Denial of Service

ЗМІСТ

ВСТУП

Комп’ютерні ігри це вже давно частина нашого життя, що осіла в серцях багатьох. Велика кількість людей проводить свій вільний час або весь час в іграх захоплюючись віртуальних світом. Нерідко дехто перетворює свій відпочинок у роботу та заробляє на цьому великі кошти.

Гра — це заснована на правилах формальна система зі змінним і кількісно виміряним результатом, де різним результатам присвоюються різні значення, гравець докладає зусиль, щоб вплинути на результат, гравець відчуває прив’язаність до результату, а наслідки діяльності є необов'язковий і договірний.

В даний час існуючих загроз забагато, тому і рівень захисту ігор досить низький. Оскільки методи сканування вразливостей досить загальні та охоплюють широкий спектр тем, вони можуть бути надлишковими, якщо їх використовують на конкретній системі, тому під час пошуку деяких вразливостей можно втратити необгрунтовано багато часу. Тоді пропонується зробити аналіз існуючих методів для розробки нових.

Метою роботи є аналіз методів захисту серверів багатокористувацьких ігор

Об’єктом дослідження є процес захисту серверів багатокористувацьких ігор

Предметом дослідження є засоби та механізми, які реалізують захист серверів багатокористувацьких ігор

Методи дослідження дипломної роботи:

- аналіз літератури;
- аналіз документів;
- системний підхід;
- методи порівняння;
- структурний аналіз

Для досягнення зазначеної мети поставлено наступні завдання:

- Провести характеристику поняття ігор та їх класифікацію;
- Вивчити види багатокористувацьких ігор та шахрайства в них;

- Розібрати способи захисту веб серверів;
- Охарактеризувати види нелегального доступу в іграх;
- Привести приклади технічних засобів захисту ігор;
- Вивчити сучасні методи захисту від несанкціонованого доступу і мети його застосування;
- Розібрати підготовку програмного

Практичною цінністю отриманих результатів є підготовка апаратного та програмного обладнання та блокування можливості атаки без реєстрації користувача з відсутністю унікального ключа CD-KEY із використанням авторського додатку.

Ключові слова: кібератака, захист інформації, багатокористувацькі ігри, захист серверів, переповнення буфера, класифікація ігор

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ПОНЯТТЯ ЗАХИСТУ СЕРВЕРІВ ІГОР

1.1 Ігри та їх класифікація

Класифікація комп'ютерних ігор

Комп'ютерні та відеоігри можуть бути класифіковані за двома ознаками: жанр і кількість гравців.

Класифікація ігор з жанру

Чітка класифікація утруднена через те, що часом важко віднести гру до якомусь конкретному жанру. Гра може являти собою як змішання існуючих жанрів, так і не ставитися до жодного з них. Незважаючи на це, в ході розвитку комп'ютерних ігор склалася наступна класифікація:

- 3D Shooter (3D-шутери «дії»)

Назва походить від поняття 3D - 3 dimensions (три виміри) і shooter (англ. «стрілець»). Основний принцип полягає в зображенні віртуального простору і предметів за допомогою ігрової програми, виконуваної на комп'ютері. При цьому гравець може впливати на віртуальну ігрову середу. Застосовується для позначення всіх видів комп'ютерних ігор, що містять елементи бою у віртуальному тривимірному просторі. В основному використовується техніка «шутер від першої особи» - при цьому зображення на екрані монітора комп'ютера імітує вид з очей гравця. З точки зору організації гри розрізняються Singleplayer і Multiplayer - гра поодинці проти комп'ютера і гра з іншими гравцями.

Приклади: Doom, Quake, Counter-strike, Half-life, Unreal, Tomb Raider.

- Arcade (аркада)

Це ігри де зазвичай потрібно покладатися на свої рефлекси та реакцію, в якій є система бонусів, що нараховує гравцю очки та поступово відчиняє різні елементи гри. Кінець 1970-х - 1980-ті роки були золотим століттям аркадних ігор. Вони користувалися відносною популярністю навіть на початку 1990-х років. Однак популярність цієї платформи поступово знижувалася, оскільки набули популярності

консольні та комп’ютерні ігри

- Arcade Racing (аркадні гонки)

Аркадні гонки характеризуються легким, віддаленим від реальності управлінням

Приклади: серія Trackmania, Go for ride.

- Classic Arcade (класичні Аркади)

Головною ціллю є проходження рівнів за максимально короткий проміжок часу та збір бонусів на рівнях. До цього можна віднести різні Арканоїди і пінболли.

Приклади: Pacman, Digger, Battle City.

- Fighting (бійки)

Ігри де 2 персонажі б'ються на арені один з одним, використовуючи різні види ударів та комбінації, в таких видах ігор є велика кількість персонажів та різні види ударів зі своїми комбінаціями клавіш. Як на ПК так і ігрових приставках.

Приклади: Mortal Combat, Street Fighter, Tekken.

- Platformer (платформери)

Платформери починали свою історію з ігрових приставок, саме на них цей жанр найбільш популярний. Основна ціль гравця це подолання перешкод, ям, шипів, ворогів і тд, використовуючи стрибки.

Приклади: Mario, Aladdin.

- Scrollers (Скролери)

Це тип відеоігор, де для перегляду дій використовується кут камери бічного огляду. Як правило, 2-D з ігровими персонажами, які переміщуються з лівого боку екрана на правий. Деякі скролери вимагають від користувачів рухатися в одному безперервному напрямку (зазвичай вправо). Однак багато бічних скролерів дозволяють повертатися назад, а також рухатися вгору, вниз, вліво і вправо. Ігри з боковим скроллером були популярні в золоту еру аркадних відеоігор і консолей третього покоління. Жанр асоціюється зі стрімкими екшенами, оскільки в епоху прокрутки з боку було створено ряд класичних затискачів кнопок.

Приклади: Jets'n'guns, AirStrike, DemonStar, KaiJin.

- Virtual Shooting (Віртуальний тир)

Ігровий процес являє собою відстріл несподівано з'являються ворогів, але головною особливістю є неможливість керувати рухом персонажу та самою камерою, весь час потрібно лінійно йти вперед.

Прикладом є: Mad Dog McGee, серія House Of The Dead.

- Simulation (симулятори)

Гра-симуляція. В ній імітується управління складними технічними системи, а саме бойовими винищувачами, автомобілями, літаками, танками і тд.

Приклади: серія Need for Speed, Descent III, Aviator

- економічні симуляції

В економічних симуляціях представлені різні економічні процеси і взаємодії різних величин.

Приклади: Sims, Civilization.

- Strategy (стратегії)

Для гри потрібно формувати свою стратегію, для перемоги в військових операціях або посилення своєї держави.

Основним персонажем може бути одна людина, ціла країна, підрозділ або планета.

Розрізняти:

* покрокові стратегічні ігри Turn-Based Strategy. Гравці по черзі роблять ходи, і кожному гравцю відводиться необмежений або обмежений час на свій хід.

* стратегічні ігри в реальному часі Real Time Strategy. В ній всі гравці виконують свої дії одночасно, без обмеження у часі.

Приклади: WarCraft, StarCraft, Dune.

- Sport (спортивні)

Як і випливає з назви - імітація будь-якої спортивної гри, наприклад футболу.

Приклади: FIFA, NBA, Tennis

- Adventure (пригоди), або Quest

Гра-розповідь, в якій керований гравцем герой просувається по сюжету і

взаємодіє з ігровим світом за допомогою застосування предметів, спілкування з іншими персонажами і вирішення логічних завдань.

Приклади: Space Quest; Myst, Мор. Утопія

- Role-Playing Games (RPG) (рольові ігри)

Правильна назва цього жанру Computer RPG (CRPG), так як ці ігри є адаптованими для комп'ютера традиційними рольовими іграми.

Приклади: Сапер (Minesweeper); Sokoban.

- Traditional (традиційні) і board (настільні)

Комп'ютерна реалізація настільних ігор, наприклад шахів.

Приклади: CGoban

- текстові

Нове віяння в ігровій культурі. Найчастіше, жанр являє собою текстовий квест, кількість учасників в якому не обмежена. Іноді така гра може тривати роками.

Класифікація ігор за кількістю гравців

- single player для гри поодинці, проти комп'ютера.

- multiplayer для гри кількох людей в локальній мережі, модему або Інтернету.

- hot seat і splitscreen розраховані на багато користувачів на одному комп'ютері

На сучасних персональних комп'ютерах бувають рідко, але часто зустрічаються на старих ПК та приставках. Hot seat-гра по черзі на одному комп'ютері. У режимі splitscreen екран ділиться на дві частини, кожен з гравців грає на своїй частині.

- багатокористувацькі через електронну пошту (PBEM)

В основному зустрічається в покрокових стратегіях. Результати ходу записуються в спеціальний файл і відсилаються іншому гравцеві через електронну пошту.

- масові (MMO, Massively Multiplayer Onine)

Масові ігри по Інтернету. Найбільш часто зустрічаються жанри-настільні та рольові ігри (Т.зв. MMORPG, або Massively Multiplayer Online RPG). Серед них розрізняють також браузерні ігри (ігри, які не потребують установки будь-якого клієнта), а також текстові онлайнові ігри - жанр MUD.

1.2 Багатокористувацькі ігри та шахрайство в них

Ідея мережевих багатокористувацьких ігор не була популярна у основних виробників ігор до 1990-х, втім, немає правил без винятків . У цій главі ми спочатку познайомимося з короткою історією еволюції багатокористувацьких ігор, від перших мережевих ігор 1970-х до великої сучасної індустрії . А потім буде представлений огляд архітектури двох популярних мережевих ігор з 1990-х — «Starsiege: Tribes» і «Age of Empires» . Багато прийоми, що застосовувалися в цих іграх, використовуються і в наші дні, тому даний огляд допоможе краще зрозуміти складності створення мережевої багатокористувацької гри.

Історія багатокористувацьких ігор

Перші прабатьки сучасних мережевих багатокористувацьких ігор почали з'являтися ще в 1970-х і працювали на університетських мейнфреймах. Однак Ігри цього роду не набули широкого поширення, поки Інтернет не перетворився з інформаційного дива в щось буденне - це сталося в другій половині 1990-х.у цьому розділі ми поговоримо про те, як з'явилися перші мережеві ігри і в яких напрямках йшло їх розвиток майже півстоліття, починаючи з появи перших ігрових «динозаврів».

Локальні багатокористувацькі ігри

Деякі з ранніх відеоігор були локальними багатокористувацькими, в тому сенсі, що допускали можливість участі двох і більше гравців в грі на одному комп'ютері. До числа таких ігор відносяться найперші ігри, включаючи «Tennis for Two» (1958) і «Spacewar!» (1962). Локальні багатокористувацькі ігри програмуються практично так само, як однокористувацькі. Єдина відмінність полягала в наявності декількох зображень ігрової сцени на екрані і/або підтримки декількох пристроїв введення. Так як програмування локальних багатокористувацьких ігор дуже схоже з програмуванням одно - користувальницьких ігор, в цій книзі до нього ми більше не повернемося.

Ранні мережеві багатокористувацькі ігри

Перші мережеві багатокористувацькі ігри працювали в невеликих мережах, що

складалися з великих ЕОМ. Головна відмінність мережевої багатокористувацької гри від локальної багатокористувацької полягає в тому, що в активний сеанс гри включені два або більше комп'ютера, з'єднаних один з одним. Однією з таких мереж великих ЕОМ була система PLATO, розроблена в Іллінойському університеті. Саме в системі PLATO була створена одна з перших мережевих ігор — покрокова стратегія «Empire» (1973). Приблизно в той же час побачила світ і мережева гра від першої особи «Maze War», і досі немає єдиної думки про те, яка з двох може претендувати на первородство.

З початком поширення персональних комп'ютерів наприкінці 1970 - х розробники шукали способи організації взаємодії двох комп'ютерів через послідовні порти. Послідовний порт дозволяє передавати дані по одному біту, і основним його призначенням була взаємодія з зовнішніми пристроями, такими як принтери або модеми. Однак з його допомогою також можна було зв'язати два комп'ютери і організувати обмін даними між ними. Це дозволяло створити ігровий сеанс, що об'єднує кілька персональних комп'ютерів, завдяки чому з'явилися перші мережеві ігри для PC. У грудневому номері журналу «BYTE» за 1980 рік була опублікована стаття Вассермана і Страйкера про те, як програмувати так звані багатомашинні ігри на Бейсику.

Але великий недолік використання послідовних портів полягав у тому, що комп'ютери зазвичай мали не більше двох таких портів (якщо, звичайно, не використовувалась карта розширення). Це означало, що для з'єднання більш ніж двох ком - пьютеров через послідовні порти необхідно було використовувати гірляндную схему з'єднання, в якій безліч комп'ютерів об'єднувалося в кільце. Така організація комп'ютерів є одним з варіантів топології мережі, про які докладніше розповідається в розділі 6 «топології мереж і приклади ігор».

Так, незважаючи на наявність технології, доступної вже на початку 1980-х, більшість ігор, випущених протягом десятиліття, не використовувало наявні сеті - ШІ можливості. Тільки в 1990-і ідея об'єднання декількох комп'ютерів в локальну мережу для гри отримала подальший розвиток, про що розповідається нижче в цьому розділі.

Багатокористувацький світ

Багатокористувацький світ, або МПМ (Multi-User Dungeon, MUD), — це звичайна текстова багатокористувацька гра, в якій кілька гравців під - ключаються до загального віртуального світу. Ігри цього виду вперше з'явилися на великих ЕОМ у великих університетах, а сама абревіатура MUD походить від назви гри «MUD» (1978), створеної Робом Трушоу з Ессекського університету. Багатокористувацький світ можна вважати ранньою комп'ютерною версією рольової гри «Dungeons and Dragons», хоча не всі багатокористувацькі світи були рольовими іграми.

З ростом потужності персональних комп'ютерів виробники апаратного забезпечення почали випускати модеми, що дозволяють двом комп'ютерам обмінювати - ся даними по стандартних телефонних лініях. Незважаючи на те що швидкості обміну були надзвичайно низькими за сучасними мірками, модеми давали можливість грати в багатокористувацьких світах вже поза стінами університетів. Деякі запускали гри МПМ в електронних дошках оголошень (Bulletin Board System, BBS), що дозволяли декільком користувачам підключатися за допомогою модемів і виконувати безліч дій, в тому числі і грати в ігри.

Ігри в локальних мережах

Термін локальна мережа (Local Area Network, LAN) використовується для опису об'єднання декількох комп'ютерів у відносно невелику мережу. Для створення локальних з'єднань могли використовуватися найрізноманітніші механізми-одним із прикладів служить об'єднання в мережу через послідовні порти, як описано вище в цьому розділі. Однак зоряний час локальних мереж пробив з появою Ethernet (протоколу, який докладніше розглядається в главі 2 «Інтернет»).

Гру «Doom» (1993) багато в чому можна вважати прабатьком сучасних мережевих ігор, хоча вона і не була першою грою, що забезпечує багатокористувацький режим в локальній мережі. Перша версія шутера від першої особи, випущена компанією id Software, підтримувала до чотирьох гравців в одному сеансі гри з можливістю грати разом проти комп'ютера або кожен сам за себе. Так як «Doom» був активною грою з швидко змінюється ігровою ситуацією, він зажадав реалізації декількох найважливіших ідей, описуваних в цій книзі. Звичайно, всі ці

прийоми зазнали суттєвих змін після 1993 року, але першість «Doom» в даній області ніким не заперечується. Детальніше історія створення «Doom» описується в книзі «Masters of Doom» (див.посилання в кінці глави).

Багато ігор, які передбачали використання багатокористувацького режиму, підтримували взаємодії гравців не тільки в локальних мережах, але також через модемні або телекомунікаційні з'єднання. Через кілька років вже переважна більшість мережевих ігор забезпечило підтримку локальних мереж. Це призвело до появи клубів, оснащених з'єднаними в мережу комп'ютера - ми, в яких збиралися любителі мережевих ігор. Незважаючи на те що деякі се - теві багатокористувацькі ігри все ще випускаються з підтримкою локальних мереж, в останні роки на ринку домінують багатокористувацькі онлайн ігри.

Онлайн-ігри

В онлайн-грі гравці зв'язуються один з одним за допомогою деякої глобальної мережі. В даний час онлайн-ігри часто називають інтернет-іграми, але термін «онлайн» набагато ширший і включає також деякі ранні мережі, такі як CompuServe, які раніше не були з'єднані з Інтернетом.

З початком бурхливого розвитку Інтернету в кінці 1990 - х почався розквіт онлайн-ігор. У числі ігор, які завоювали популярність в цей період, можна назвати «Quake» (1996) компанії id Software і «Unreal» (1998) компанії Epic Game.

Хоча може здатися, що онлайн-ігри реалізуються так само, як ігри для локальних мереж, головною проблемою онлайн-ігор є затримки, неминуче виникають при передачі даних по мережі. Первісна версія «Quake», наприклад, взагалі не передбачала взаємодії через Інтернет-з'єднання, і тільки з виходом виправлень QuakeWorld з'явилася можливість гри через Інтернет. Методи компенсації мережевих затримок більш детально розглядаються в главі 7 «затримки, флуктуації і надійність» і в главі 8 «Покращена обробка затримок».

З появою в 2000 - х таких служб, як Xbox Live і PlayStation Network, що були прямими нащадками служб для персональних комп'ютерів типу GameSpy і DWANGO, стала можливою Підтримка онлайн-ігор на ігрових консолях. У години пік цими службами користуються мільйони активних користувачів, хоча, з

рас-простором потокового відео та інших служб для ігрових консолей, не всі ці активні гравці дійсно грають в ігри. У розділі 12 «ігрові служби» розповідається, як інтегрувати одну з таких служб — Steam — в гру для персонального комп'ютера.

Масові багатокористувацькі онлайн-ігри

Навіть в наші дні більшість багатокористувацьких онлайн-ігор обмежується підтримкою невеликого числа гравців в одному ігровому сеансі-зазвичай від 4 до 32. Однак в масовій багатокористувацької онлайн грі (Massively Multiplayer Online Game, MMO) в одному ігровому сеансі можуть брати участь сотні, якщо не тисячі гравців. Більшість масових онлайн-ігор - це рольові ігри (Role-Playing Games), і тому їх позначають абревіатурою MMORPG. Але суті-ють також і інші види масових онлайн-ігор, наприклад шутери від першої особи (First-Person Shooters, MMOFPS).

У багатьох випадках ігри MMORPG можна вважати графічним розвитком багато-користувацьких світів. Деякі з ранніх ігор MMORPG фактично з'явилися до широкого поширення Інтернету і функціонували в комутуючих мережах, таких як Quantum Link (пізніша назва - America Online) і CompuServe. Однією з перших таких ігор стала «Habitat» (1986), в якій були реалізовані деякі елементи новітньої технології, описані в книзі Мор - нігстара і Фармера (див.посилання в кінці глави). Однак тільки з поширенням Інтернету гри цього жанру стали набувати популярності. Першим справжнім хітом стала «Ultima Online» (1997).

Інші ігри MMORPG, такі як «EverQuest» (1999), також мали певний успіх, але справжній вибух інтересу в усьому світі стався з виходом «World of Warcraft» (2004). У якийсь момент гра від Blizzard зібрала більше 12 мільйонів активних передплатників по всьому світу і стала настільки значною частиною популярной культури, що в 2006 році лягла в основу одного з епізодів американського мультиплікаційного серіалу «Південний парк» (South Park).

Створення масових онлайн-ігор пов'язане з вирішенням складних технічних завдань, частина з яких обговорюється в главі 9 «масштабованість». Однак більшість прийомів, використовуваних при цьому процесі, далеко виходить за рамки нашої книги. Крім того, перш ніж навіть розглядати саму можливість створення масової онлайн-гри, слід ознайомитися з основами створення менш масштабних сітьових

ігор.

Мобільні мережеві ігри

З появою ігор для мобільних пристроїв не залишилися осторонь і багатокористувацькі ігри. Багато з них на мобільних платформах є асинхронними - зазвичай зі зміною ходу - і не вимагають передачі даних в реальному часі. У цій моделі гравці сповіщаються про те, що настає їх право зробити хід, і мають досить багато часу, щоб виконати його. Асинхронна модель існувала з самого моменту появи мережевих багатокористувацьких ігор. Деякі електронні дошки оголошень мали єдину телефонну лінію для вхідних з'єднань, а це означало, що в кожен момент часу до таких систем міг бути підключений тільки один користувач. Тобто гравець повинен був підключитися, зробити свій хід і відключитися. Потім в наступний момент другий гравець може підключитися, щоб зробити хід у відповідь.

Як приклад мобільної гри, що використовує асинхронну багатокористувацьку модель, можна привести «Words with Friends» (2009). З технічної точки зору асинхронна мережева гра простіше в реалізації, ніж гра, що діє в режимі реального часу, особливо для мобільних пристроїв, тому що прикладні програмні інтерфейси мобільних платформ (Application Program Interface, API) вже мають функції для асинхронних взаємодій. Використання асинхронної моделі для мобільних ігор на ранній стадії розвитку мобільних мереж було практично неможливим через низьку надійність цих мереж порівняно з дротовими з'єднаннями. Однак з швидким зростанням числа пристроїв, що підтримують Wi-Fi, і поліпшенням якості мобільних мереж стало з'являтися все більше мережевих ігор реального часу для цих пристроїв. Прикладом такої гри, що використовує переваги мережевих взаємодій в реальному масштабі часу, є «Hearthstone: Heroes of Warcraft» (2014).

1.3 Способи захисту веб серверу від атак

Більшість ігор вразливі для хакерів. При цьому не завжди застосовуються сучасні способи забезпечення інформаційної безпеки.

Захист ігор стає завданням і розробників, і користувачів. Якщо кілька років тому заходи захисту обмежувалися налаштуванням веб-сервера, ретельним очищенням жорсткого диска від зайвих і застарілих файлів і кодів, регулярним контролем за незмінністю файлів, то в міру посилення активності хакерів і почастішання DDoS-атак потрібні більш серйозні заходи безпеки.

Від типу гри залежить модель загроз, від яких його слід захищати.

Основні загрози

Зловмисники цікавляться способами злому веб-гри в різних цілях. Злом облікового запису допомагає отримати цінний ресурс без оплати, викрасти чужого гравця в популярній грі, використовувати аккаунт як учасника в бот-мережі. Для зломщика зазвичай не мають значення персональні дані особи, чий аккаунт зламується, йому цікавий лише факт доступу до ресурсу.

Особливість злому ресурсів - він не персоніфікований, а автоматизований, проводиться масово за допомогою спеціальних програм. Отримана інформація продається або використовується в цілях зломщика. Власник ресурсу, що працює з клієнтами за моделлю веб-гри, повинен вміти захистити сайт від найпоширеніших способів злому.

Найпопулярніші методи захисту

Звичайні методики захисту ігор від хакерів не вимагають серйозних та доступні більшості власників інтернет-магазинів і аналогічних ресурсів.

Серед них найпопулярніші:

- аудит (превентивний захід);
- використання захищених протоколів передачі даних;
- застосування ПЗ, що забезпечує безпеку.

Всі способи необхідно застосовувати в комплексі.

Перевірка сайту на уразливості

Перш ніж розробляти методику захисту веб-додатки від потенційних загроз, сайт слід перевірити на уразливості. Перевірка проводиться ручним або автоматизованим способом. Програми, доступні в платній і безкоштовній версіях, протестують додаток на основні ризики. Такі програмні продукти існують в двох

варіантах: Black hat, що моделюють дії зломщиків, і White hat, планомірно виявляють всі недоліки системи методом сканування.

Серед найефективніших безкоштовних інструментів-додатків слід назвати:

OpenVAS сканує локальні мережі на вразливості;

OWASP Xenotix XSS Exploit Framework перевіряє сайт на XSS-вразливість, - можливість впровадження в веб-сторінку шкідливого коду, що викрадає дані акаунтів користувачів та іншу інформацію. Код впроваджується через уразливості на сервері або пристрої Користувача;

Approof від Positive Technologies вивчає конфігурацію веб-програми і знаходить зайвий або шкідливий код.

Також з аудитом впораються безкоштовні онлайн-сервіси:

SecurityHeaders.io проаналізує відповіді сервера на запити і виявить вразливі місця;

Observatory by Mozilla-безкоштовний сервіс з відкритим кодом для виявлення проломів в безпеці. Може залучати ресурси інших сервісів перевірки безпеки і додавати їх дані до звіту. Оцінює ступінь безпеки за шкалою від A до F, де F – найнижчий рівень. У 2016 році 91% перевірених сайтів був на рівні F;

One button scan відповідає за сканування таких елементів сервера, як DNS, HTTP-заголовки, SSL, перевіряє на уразливості використовувані сервіси;

SSL Server Test перевіряє наявність SSL-вразливостей;

Snyk проконтролює наявність вразливостей в JavaScript, Ruby і Java-додатках, самостійно виправляє недоліки в безпеці. Успішно працює разом з GitHub-репозиторієм.

Платні ресурси дадуть більше можливостей для перевірки сервісу на уразливості, вони швидше оновлюються в міру зміни структури і характеру загроз.

За результатами аудиту IT-фахівець може випробувати шок - так багато загроз буде виявлено, але не всі вони однаково важливі або реалізовані, хакери використовують найпростіші методи злому.

Після виправлення серйозних вразливостей сканування слід провести повторно. Закінчивши автоматичну перевірку, можна організувати злом веб-додатки

вручну. Для цього потрібно змінити значення запитів POST (відправка даних на ресурс) і GET (запит даних у ресурсу) в HTTP. Краще використовувати проксі-сервер, що перехоплює HTTP-запити. Також необхідно обійти валідацію даних (перевірку відповідності запиту заданим вимогам) і впровадити на сайт SSL-інфекцію, що перехоплює дані користувачів. Якщо системи моніторингу показують суттєві уразливості, необхідно посилювати захист у виявлених областях.

Використовуючи платні або безкоштовні ресурси моніторингу систем безпеки необхідно перевірити гіпотетичну можливість хакера обійти вимоги обов'язкової аутентифікації, передбачені для деяких сторінок веб-додатки. Для цього потрібно використовувати такі традиційні способи злому як зміна параметрів URL (зокрема, ID користувача) або зміна Cookie.

HTTPS

Другим за популярністю способом захистити дані користувача після ідентифікації є застосування захищеного протоколу передачі даних HTTPS. Hyper Text Transfer Protocol Secure захищає інформацію про користувача веб-додатки за допомогою шифрування трафіку. Він забезпечує збереження конфіденційності та цілісності інформації, не допускаючи витік або підміну даних.

Більшість ресурсів використовує технологію давно, це стало хорошим тоном, що підтверджує готовність їх власників захищати інтереси клієнтів. Пошуковик Google піднімає у видачі сайти, що використовують цю технологію.

HTTPS необхідний, якщо користувачі передають сервісу такі відомості, як:

- номери кредитних карт;

- персональні дані;

- адреси сторінок, на які вони заходять.

При генерації запиту з форми авторизації застосовуються cookie-файли, вони підлягають відправці на сервер при кожному запиті. При слабкому захисті веб-додатки нічого не заважає зловмисникові перехопити файли і підробити запит, отримавши права користувача. Застосування HTTPS для кожної сторінки сайту знизить ступінь цього ризику.

Вирішити задачу нескладно, будуть потрібні наступні кроки:

- генерувати SSL-сертифікат, на деяких ресурсах

- це робиться безкоштовно;

- отримати та встановити сертифікат;

- підключити для веб-додатки підтримку HTTPS.

Додатковою можливістю після налаштування HTTPS стане застосування Hyper Strict Transport Security (HSTS). Це опція примусового використання протоколу HTTPS, навіть якщо сервер не підтримує його застосування. Однак і захищені протоколи не врятують веб-додаток, якщо саме програмне забезпечення застаріло.

Оновлення ПЗ

Власник програми повинен тримати руку на пульсі оновлень програмного забезпечення. Хакери тестують всі оновлення і знаходять в них уразливості іноді раніше, ніж розробники. Особливо активно зламуються операційні системи, технології управління HTTP і системи управління контентом (CMS).

У ситуації, коли сервіс встановлений на чужому хостингу, завдання своєчасної заміни операційної системи лягає на плечі провайдера. Якщо хостинг власний, ОС потрібно міняти відразу після виходу оновлень. Сайт може працювати на операційній системі, призначеної для цього типу веб-додатків (движка), стороннього виробника, особливо це характерно для інтернет-магазинів. Необхідно відстежувати всі оновлення ПЗ і встановлювати нову версію відразу після її виходу. Розробники сповістять про це власника ресурсу розсилкою, а найбільш популярні автори движків, WordPress і Umbraco, повідомляють про оновлення в момент входу в панель управління сайтом.

Web-сайти часто мають залежні компоненти (програмні модулі менеджменту контенту). Для управління ними використовуються менеджери пакетів, наприклад, Composer, NPM або RubyGems. За їх оновленнями, щоб уникнути проблем безпеки також необхідно стежити.

Захист від SQL-ін'єкцій

Безпека веб-застосунку залежить від того, наскільки ефективно власнику вдається уникнути SQL-ін'єкцій. Цей метод хакерської атаки виглядає як запит до сайту і його бази даних за допомогою поля форми або параметра URL. Якщо при

конструюванні ресурсу застосовувався мову Transact SQL, в запит вставляється шкідливий код, з легкістю змінює або знищує дані, що містяться в таблицях.

Уникнути ризику вийде, якщо застосовувати параметризовані запити, в яких задіяно кілька мов програмування.

Додаткові способи забезпечення безпеки

Робота з веб-сервісами вимагає використання широкого інструментарію для забезпечення безпеки. Крім основних перерахованих методів, часто застосовуються:

- шифрування паролів;

- уникнення міжсайтового скриптинга;

- контроль завантаження файлів на сервер.

Спільне застосування всіх доступних рішень забезпечить безпеку на максимально доступному рівні.

DRM в комп'ютерних іграх використовується для різних цілей, але в цілому всі схеми спрямовані на захист від копіювання і поширення піратських копій ігор. Найчастіше при запуску таких ігор необхідно вставити диск з грою в оптичний привід, при цьому перевіряються низькорівневі особливості ліцензійних CD і DVD-дисків, які неможливо відтворити при копіюванні в домашніх умовах. Також подібні системи DRM часто встановлюють в систему драйвер для захисту від емуляторів дисководів (наприклад, DAEMON Tools і Alcohol 120%), а іноді вимагають реєстрації через Інтернет.

Ігрові приставки, такі як Xbox 360, Xbox One, PlayStation 3 і Playstation 4, також містять систему перевірки диска на ліцензійність.

У деяких комп'ютерних іграх DRM захист використовується для обмеження числа систем, на яких можна встановлювати дане ПЗ. Для контролю використовується онлайн-аутентифікація на серверах видавця. Більшість таких DRM схем дозволяють зробити 3-5 установок, проте деякі дозволяють скасувати активацію за допомогою деінсталяції. Подібні схеми викликають багато критики, так як обмежують користувачів від законного використання придбаних продуктів, наприклад, якщо у користувача вдома більше 5 комп'ютерів, він не може встановити придбану продукцію на всі машини.

Приблизно з середини 2008 року випуск Mass Effect запустив цілу хвилю продуктів, що використовують DRM схему SecuROM, яка вимагає онлайн-аутентифікації на серверах видавця. В цьому ж році, використання подібного захисту в грі Spore від Electronic Arts призвело до того, що більшість користувачів воліла використання піратської версії гри. Однак, незалежні дослідники з TweakGuides прийшли до висновку, що подібне використання DRM не впливає на кількість піратських копій гри, відзначаючи той факт, що інші ігри (на кшталт Call of Duty 4: Modern Warfare, Assassin's Creed, Crysis), що використовують схему SafeDisc, що не вдається до онлайн-аутентифікації, також поширювалися в порівнянних зі Spore кількостях серед піратів. До того ж, ігри, що використовують онлайн аутентифікацію також, як і Spore, начебто BioShock, Crysis і той же Mass Effect, в списках найбільш скачуваних ігор на різних торрент-трекерах не значаться.

Постійна онлайн-аутентифікація

Багато видавців, серед яких, наприклад Electronic Arts, Ubisoft, Valve і Atari, використовували онлайн DRM схеми аж до початку 2009 року. Наприкінці 2008 року, компанія Ubisoft провела експеримент, випустивши гру Prince of Persia без DRM захисту, з метою перевірити «на скільки люди мають рацію» щодо того, що DRM тільки посилює піратство і провокує людей використовувати не ліцензійні копії. Хоч сама компанія так і не оголосила результати експерименту, незалежні експерти з Tweakguides помітили, що всього лише з двох торрентів на Mininova гру скачало більше 23 тисяч людей протягом 24 годин після релізу.

Ubisoft офіційно оголосили про повернення онлайн аутентифікації 9 лютого 2010 року. Вони представили свою нову онлайн ігрову платформу Uplay, яку почали використовувати в таких іграх, як Silent Hunter 5, The Settlers 7 і Assassin's Creed II. Silent Hunter 5 зламали протягом 24 годин з моменту релізу. Однак, користувачі піратської версії могли грати тільки в початкові рівні гри. Система Uplay працює таким чином, що на призначений для користувача ПК гра встановлюється в повному обсязі, а докачує вміст з ігрових серверів Ubisoft у міру проходження гри. Трохи більше, ніж через місяць після релізу на ПК, в перший тиждень квітня, було випущено ПЗ, за допомогою якого можна було обійти DRM захист в Assassin's Creed

II. ПО являло собою емулятор сервера Ubisoft для гри. Трохи пізніше, в цьому ж місяці, була випущена версія, яка прибирала необхідність в з'єднанні з серверами повністю.

На початку березня 2010 року сервера Ubisoft піддалися масштабній dos-атаці, що призвело до закриття доступу до ігор для ~5% гравців. В якості компенсації ЗА принесені незручності, компанія надала постраждалим користувачам по безкоштовній скачуваній грі. З березня 2010 року сервера Uboisoft більше не падали.

Приклад Ubisoft наслідували й інші розробники, такі як Blizzard Entertainment. Вони також перейшли на варіант захисту, коли більша частина ігрової логіки знаходиться «на стороні», або обробляється серверами творця гри. Blizzard використовує подібний підхід у своїй грі Diablo III. Electronic Arts використовували такий підхід у своїй перезавантаженні серіалу SimCity. Треба сказати, що подібний підхід негативно вплинув на обидві компанії, бо вони просто не змогли впоратися з кількістю гравців на серверах, що призвело до численних скарг і зростаючого невдоволення користувачів. Electronic Arts намагається прибрати необхідність постійного підключення до серверів, але поки це не представляється можливим, бо вся гра була створена з урахуванням цього.

Втручання в ПЗ

Деякі студії в якості захисту використовують не зовсім стандартні підходи. Bohemia Interactive використовує DRM схему (починаючи з 2001 року, з виходом Operation Flashpoint: Cold War Crisis), яка при запуску нелегальної копії гри, просто заважає грати. Гра починає створювати ситуації, в яких у гравців знижується точність зброї, або, наприклад, самі гравці перетворюються в птахів. Компанія Croteam в своїй грі Serious Sam 3: BFE використовувала схожий підхід, націковуючи на гравців, що використовують нелегальні копії гри, монстра, якого неможливо було вбити.

Критика DRM

Деякі критики DRM вважають, що DRM використовуються не щоб захистити виняткові права і обмежити масове незаконне копіювання («піратство»), а щоб змусити законослухняних клієнтів платити більше за звичні дії на кшталт

«сумлінного використання» або «вільного використання» творів. Наприклад, звичайну електронну книгу можна читати і на настільному комп'ютері, і на мобільному пристрої, Слухати за допомогою синтезатора мови, Копіювати в буфер обміну цитати (нікого при цьому не повідомляючи), а DRM дозволяє змусити користувача купувати окремі версії для кожного способу використання.

Контролююча особа (а іноді й інші особи) може збирати інформацію про поведінку покупця: його режим дня, способи використання твору тощо.

Продукція без DRM

Багато видавців, реагуючи на численну критику DRM, випускають свою продукцію зі спеціальною позначкою «DRM-Free», що на російську мову можна перекласти як «вільно від DRM», або «Без DRM». Багато великих компаній підтримують цю політику:

- Apple Inc. продають музику без DRM через свій iTunes Store з квітня 2007 року, відзначаючи всю музику знаком «DRM-Free» з січня 2009 року. Музика все ж містить цифрові водяні знаки для ідентифікації покупця. Інша продукція, що продається через iTunes (наприклад електронні книги, фільми, Додатки і т.д.) продовжує підтримувати DRM.
- Tor Books, великий видавець книг у жанрах наукова фантастика та фентезі, продає книги без DRM з липня 2012 року. Через рік вони заявили, що продовжать дотримуватися політики «DRM-Free», так як відсутність DRM захисту ніяк не шкодить їх бізнесу. Більш дрібні видавці почали позбавлятися від DRM ще раніше.
- GOG.com, цифровий постачальник відео ігор для ПК, також дотримується суворої політики щодо DRM. Весь їх каталог ігор продається без DRM, в той час як більшість цифрових магазинів продовжують використовувати DRM.
- DotEmu-ще один цифровий магазин класичних відео ігор, який пропонує в своєму каталозі ще й власні порти класичних ігор на мобільні пристрої. Все «DRM-Free».

- The Humble Indie Bundle-серія продуктів, створена Humble Bundle Inc., містить набори ігор, музики та електронних книг без DRM. Крім того, компанія дотримується цікавої цінової політики - користувач платить стільки, скільки вважає за потрібне.
- Crowdfunding-нова течія в області створення і просування проектів. Суть даної течії полягає в тому, що гроші на створення проекту збираються у користувачів, без безпосередніх видавців. Наприклад, на сайті kickstarter.com, зібрані кошти можуть сягати кількох мільйонів.

РОЗДІЛ 2 ДОСЛІДЖЕННЯ НЕЛЕГАЛЬНОГО ДОСТУПУ ДО РЕСУРСІВ В ІГРАХ

2.1 Загальна характеристика видів атак

Сервери мають обмеження на одночасну обробку запитів. Також для оптимізації навантаження передбачено обмеження пропускної здатності каналу, Що з'єднує мережу і сервер. Для обходу обмежень зловмисники організовують спеціальну мережу з шкідливим ПЗ («ботнет»)

Вхідні в інфраструктуру «ботнет» комп'ютери не пов'язані між собою. Вони використовуються для генерації надлишкового трафіку, здатного перевантажити атакується систему. Для цього на комп'ютери ставиться троян, який запускається віддалено. Атаці піддається DNS сервер, пропускний канал і інтернет-з'єднання.

Розпізнати атаку можна за такими ознаками:

Некоректна робота серверного ПЗ і ОС: зависання, довільні завершення сесій та ін.;

Пікове навантаження на сервер: навантаження на ЦП, оперативну пам'ять, диск та інші компоненти сервера, що перевищує середні значення;

Зростання числа запитів на порти;

Однакова модель поведінки: зловмисники намагаються маскувати шкідливий трафік, закладаючи в алгоритми симуляцію дій користувачів (скачування файлів, перегляди сторінок, використання пошуку та ін.). Виявлення масового вчинення однотипних дій може послужити сигналом;

Однотипні запити до портів і сервісів: виявити зрослу навантаження, однотипні запити до служб сервера можна з аналізу логів. Масові запити, якщо генерують їх користувачі не схожі на типову аудиторію, є хорошим маркером.

Атака спрямована на перевантаження брандмауера, центральної мережі або системи, що розподіляє навантаження. При атаках такого виду поширене використання мережевого флуду, при якому генерується маса однотипних

запитів-пустушок, що перевантажують канал. Основний упор тут робиться на методику обробки клієнтських запитів до сервера.

Як правило, Мережева служба працює за методом FIFO, згідно з яким в пріоритеті перше звернення. Однак, при флуді генерується такий обсяг запитів, що апаратних ресурсів сервера не вистачає для завершення обробки першого запиту.

Сервер отримує надлишковий обсяг HTTP-запитів клієнтів, в результаті чого всі вузли зв'язку стають недоступними.

Перевантажує сервер жертви службовими командами, на які машина повинна давати Ехо-відповіді. Класичний приклад-ping-флуд, коли на сервер безперервно відправляються ICMP-пакети для перевірки доступності вузла.

На сервер відправляється надлишковий обсяг SYN-запитів на TCP-підключення. Згідно з алгоритмом "потрійного рукостискання", сервер повинен відповісти на SYN-запит клієнта пакетом з прапором ACK (Acknowledge). Після цього буде встановлено з'єднання. У випадку з SYN-флудом, черга SYN-запитів на сервері переповнюється.

При цьому заголовки SYN-пакетів підробляються таким чином, щоб відповідні пакети з сервера йшли на неіснуючі адреси. Таким чином, зловмисник створює ланцюжок наполовину відкритих з'єднань, що забивають канал і роблять неможливим доступ рядових користувачів до сервера і його службам.

Атакується пристрій отримує множинні UDP-запити зі зміненими IP-адресами джерел. Так зловмисник зберігає анонімність паразитної мережі, забиваючи смугу пропускання сервера. Суть атаки в наступному: з шкідливої мережі на жертву направляється потік UDP-запитів. Сервер повинен обробити запит, розібравши приходить пакет і визначивши для нього відповідний додаток (сервіс, порт)

Потім потрібно перенаправити запит туди і в разі успіху повернути відповідь служби. У разі відсутності активності буде відправлено повідомлення «Адресат недоступний» по протоколу ICMP. Оскільки в пакетах була змінена адреса джерела ініціатора запиту, то ICMP-відмови йдуть на інші вузли. Тим часом, шкідливий алгоритм продовжує підтримувати чергу запитів переповненою.

Атаки рівня інфраструктури

Атаці піддаються Оперативна пам'ять, процесорний час, а також підсистема зберігання даних на сервері. При цьому пропускний канал не перевантажується.

Існують кілька видів таких атак.

Обчислення

Процесор отримує запити на «важкі» обчислення. Зважаючи на надлишок запитів сервер починає давати збої і користувачі не отримують доступ до сервера, його служб і ресурсів.

Переповнення диска

Дисковий простір сервера починає заповнюватися «сміттєвим» вмістом за допомогою шкідливого коду зловмисників. Переповнення диска порушує роботу веб-сервісів, функціонал яких побудований на активній роботі з файловою системою (зберігання, доступ і відтворення мультимедіа та іншого контенту). Для заповнення використовуються лог-файли (дані про запити і сесіях, що формуються на стороні сервера). Запобігти навмисне заповнення диска можна обмеживши розмір лог-файлів.

Обхід системи квотування

Зловмисник отримує доступ до CGI-інтерфейсу сервера і з його допомогою використовує апаратні ресурси машини в своїх інтересах.

Неповна перевірка користувача

Зловмисник може використовувати ресурси сервера нескінченно довго.

Атака другого роду

На сервері викликається помилковий сигнал про перевантаження, або її загрозу, в результаті мережевий вузол на час стає недоступним.

Атаки рівня додатків

При таких атаках використовуються закладені в серверне ПЗ упущення, що створюють уразливості. Класичний приклад-атака «пінг смерті», коли на атакується машину направляється надлишковий обсяг ICMP-пакетів, що переповнюють буфер пам'яті.

DNS-атаки

Атаки цього виду спрямовані на:

Використання вразливостей в ПЗ DNS-серверів «вразливість нульового дня», «швидкий потік», «DNS-спуфінг»;

Обвалення DNS-серверів: через відключення служби DNS Користувач не зможе зайти на сторінку сайту, оскільки його браузер не знайде IP-адресу потрібного вузла.

2.2 Технічні засоби захисту ігор

Основні види мережевих атак :

1. Використання спеціалізованих додатків.
2. Переповнення буферу (Buffer Overflow Attacks).
3. проведення збору відомостей з вільного доступу по мережі,
4. Spoofing,
5. DOS та DDOS атаки для перевантаження або відновлення працездатності сервера для обслуговування користувачів.
6. MITM (Man in the Middle) впровадження для перехвату пакетів всередині системи.
7. Fishing використання соціальної інженерії для обману цілі, метою проходження єю по нібито знайомої для неї адреси, але насправді підробки для своїх брудних цілей.
8. XSS - атака спрямована на уразливості в сервера.

Про перших трьох варіантах варто розповісти окремо, так як вони найскладніші і найпоширеніші.

Mailbombing

Mailbombing - це процесс спаму пошти користувача, а саме відправлення листів багатої кількості листів, для відмови роботи поштового сервера або лише самої скриньки.

Провести цю атаку досить легко, дізнавшись електронну адресу жертви та

адресу сервера, з якого можно відправити листи анонімно.

Для захисту від цієї атаки потрібно пам'ятати, що потрібно не давати адресу своєї пошти сумнівним джерелам. Фахівці задають певні налаштування на сайті провайдера. Включення обмеження на кількість листів с певної ір адреси. Якщо додаток бачить, що кількість листів перевищує норму, листи самі відправляються до кошику, але можно використовувати різні адресу для розсилання.

Спеціальні програми

Використання спеціальних додатків для зупинки роботи сервера.

Для цього використовуються віруси, руткіти, сниффери та трояни.

Вірус-шкідливий це невелика частина програмного забезпечення, яка підтримує реальні програми. Наприклад, вірус може приєднатися до такої програми, як програма електронних таблиць. Кожного разу, коли запускається програма електронних таблиць, запускається і вірус, і він має шанс розмножуватися (підключаючись до інших програм) або спричиняти хаос.

Троянський кінь - це просто комп'ютерна програма. Програма стверджує, що виконує одну річ (вона може вважатися грою), але натомість завдає шкоди, коли ви її запускаєте (вона може стерти ваш жорсткий диск). Троянські коні не мають можливості автоматично розмножуватися.

Сніффер це процес збору, збору та моніторингу фрагментів даних, які передаються через комп’ютерну мережу або Інтернет. Це означає, що кожен пакет, який проходить через Інтернет або локальну мережу, збирається для широкого спектру цілей, таких як моніторинг трафіку та пропускної здатності, підтримка мереж, аналіз даних, зібраних пристроєм, тощо.

Руткіт визначається як шкідливе комп’ютерне програмне забезпечення, приховане глибоко всередині ПК і залишається непомітним. Хоча це програмне забезпечення само по собі не може бути шкідливим, воно приховує черв’яків, ботів і шкідливе програмне забезпечення. Зловмисники можуть мати кореневий доступ до комп’ютера користувача за допомогою шкідливого програмного забезпечення. Таким чином, це вважається надзвичайно небезпечним для конфіденційності користувачів, і користувачам ПК потрібно антируткіте програмне забезпечення.

Переповнення буфера

Атаки переповнення буфера використовують відсутність обмежень для перевірки розміру вхідних даних, що зберігаються в буферному масиві. Записуючи дані за кінець виділеного масиву, зловмисник може вносити довільні зміни до стану, збереженого суміжною програмою з масивом. На сьогоднішній день найпоширенішою структурою даних, яка може бути пошкоджена таким чином, є стек, який називається атакою на розбиття стека.

Для захисту потрібно, виявити та усунути уразливості.

Способи захисту від мережевих атак:

1. Використання фаєрволу, що фільтрує вхідний та вихідний трафік, та антивірусів із вчасним їх оновленням.
2. Використання програм, що можуть блокувати дії сніферів та руткітів.

Комплексний захист від мережевих атак

Наша компанія випускає комплексний продукт Ікс, здатний забезпечити захист вашого комп'ютера «по всіх фронтах». У функціонал входить:

- міжмережевий екран;
- система виявлення вторгнень;
- лічильник трафіку;
- готовий VPN-сервер;
- фільтр трафіку;
- і багато іншого.

Інтернет контроль сервер - це універсальний і ефективний засіб контролю над внутрішніми мережами будь-якої організації незалежно від її сфери діяльності. Ікс дозволить захистити дані корпоративної мережі.

Налаштування програми проста у виконанні, за допомогою документації та відеоуроків з нею впорається навіть початківець користувач.

2.3 Сучасні методи захисту від атак і мета їх застосування

Запобігання та захист від DDoS-атак

Найбільш ефективний спосіб захисту від DDoS атак на сайт-це фільтрація підозрілої мережевої активності на рівні хостинг або інтернет-провайдера. Причому виконуватися це може як засобами мережевих маршрутизаторів, так і за допомогою спеціального обладнання.

Власник же сайту, веб-сервісу або іншого мережевого проекту, зі свого боку, для мінімізації ризиків і втрат від DDoS повинен:

Ретельно обстежити логіку свого продукту: ще на етапі розробки і тестування можна виключити помилки і уразливості;

Вести контроль версій ПЗ і мережевих служб: необхідно своєчасно оновлювати програмне забезпечення мережевих служб (СУБД, PHP та ін.). Також потрібно підтримувати код самого продукту в актуальному і стабільному стані. Рекомендується навіть розгортати проект на декількох серверах — продуктовому (бойовому), тестовому (для обкатки нового функціоналу) і бекап-сервері (для зберігання резервних копій і архівів вихідних кодів). Також рекомендується використовувати системи контролю версій (Git) для можливості відкату проекту до попередньої стабільної збірки;

Стежити за доступом до мережевих служб: делегування прав на операції вимагає опрацювання. Необхідно забезпечити кілька рівнів доступу (Майстер, гостьовий та ін.) до мережевих служб сервера і архіву версій проекту. Список осіб, які мають доступ до ресурсів сервера, необхідно підтримувати в актуальному стані — наприклад, своєчасно відключати доступ співробітникам, що звільнилися. Також потрібно скидати паролі та облікові записи при будь-якій підозрі на компрометацію;

Контролювати панель адміністратора: рекомендується обмежити доступ до панелі внутрішньої, або VPN-мережею;

Сканувати систему на наявність вразливостей: в цьому допоможуть публічні рейтинги (наприклад, OWASP Top 10), або інструменти розробника;

Використовувати брандмауер додатків: автоматизуйте перевірку мережевого

трафіку і валідації запитів до портів і служб сервера;

Розподіляти трафік за допомогою CDN: за рахунок розподіленого зберігання контенту навантаження на ресурси сервера оптимізується, що прискорює обробку трафіку і запитів;

Вести списки контролю доступу (ACL): для персонального обмеження доступу до мережевих вузлів;

Очищати кеш DNS: для захисту від спуфінга;

Використовувати захист від спаму: одне з джерел вразливостей-форми зворотного зв'язку. Зловмисники можуть направити своїх ботів масово заповнювати їх відправляти однотипні дані на сервер. Для фільтрації такого трафіку форми потрібно переводити на JS-компоненти або оснащувати їх капчами та іншими інструментами перевірки;

Використовувати контратаку: шкідливий трафік можна перенаправити на мережу атакуючого. В результаті це не тільки збереже доступність вашого сервера, але і тимчасово виведе зловмисника з гри;

Використовувати розподілене зберігання і бекапіроване: в разі відмови одного або декількох серверів вашої мережі ви зможете відновити роботу ресурсу на іншій машині. До цього часу там вже буде розгорнута функціональна копія вашого проекту;

Використовувати апаратні засоби захисту від DDoS: Impletec iCore, DefensePro та ін.;

Ретельно вибирати хостинг-провайдера: необхідно вибирати постачальника, що дає гарантії захисту від усіх сучасних загроз. Також важливо мати цілодобову лінію підтримки, панель адміністратора з необхідними інструментами аналітики за конкурентними Умовами.

Захист DNS

Брандмауери і системи запобігання вторгнень на сервери самі по собі уразливі і розраховувати тільки на їх надійність не варто.

Для TCP-трафіку рекомендується використовувати хмарні сервіси для фільтрації підозрілих запитів. Також рекомендується:

Проводити моніторинг DNS: підозрілу мережеву активність можна відстежити. Для цього рекомендується використовувати комерційні DNS-рішення, або Open-source продукти (наприклад, BIND). Ви зможете в режимі реального часу відстежувати мережевий трафік і запити до DNS. Для економії часу також рекомендується побудує базовий профіль мережевої інфраструктури і оновлювати його в міру масштабування бізнесу;

Розширювати апаратні ресурси DNS: компромісне рішення, що дозволяє захистити інфраструктуру від дрібномасштабних атак. Закупівля додаткових потужностей також пов'язана і з вкладеннями;

Використовувати DNS Response Rate Limiting(RRL): це знижує ймовірність використання вашого DNS-сервера в атаці DDoS Reflection. RRL знижує швидкість обробки повторних запитів. Цей параметр підтримується більшістю DNS;

Будувати конфігурації високої доступності: DNS служба розгортається на ha-сервері, що дозволяє відновити роботу Сервісу на резервної машині в разі якщо основна виявиться недоступною.

Географічно розподілена мережа також може послужити засобом захисту від DDoS. Існує два підходи до побудови такої мережі:

Anycast: різні DNS сервери використовують загальний IP-адресу, а при обробці трафіку запити направляються на найближчий сервер. Такий підхід, в порівнянні з описаним нижче, є більш оптимальним, оскільки трафік і навантаження розподіляються між декількома машинами. Це робить інфраструктуру більш стійкою до DDoS;

Unicast: за кожним DNS-сервером закріплюється унікальна IP-адреса. Служба DNS підтримує таблицю серверів і відповідних їм адрес ресурсу. При обробці запитів для балансування трафіку і навантажень IP-адреса вибирається у випадковому порядку. Такий підхід до організації DNS-мережі простіше в реалізації, однак при цьому страждає стійкість інфраструктури. Зловмисники можуть ініціювати ланцюжок спрямованих атак на DNS-сервери, послідовно виводячи їх з ладу.

РОЗДІЛ 3 РЕАЛІЗАЦІЯ АТАК ТА МЕХАНІЗМ ЗАХИСТУ ДЛЯ БАГАТОКОРИСТУВАЦЬКИХ ІГОР

3.1 Особливості реалізації атак багатокористувацьких ігор

Сучасні комп'ютерні ігри (Кі) - величезна індустрія з грошовим оборотом, порівнянним з нафтовим бізнесом. Особливою популярністю користуються мультиплеєрні (багатокористувацькі) ігри (МПІ, англ. Mass Multiplayer Online Game, MMOG). МПІ-Мережева комп'ютерна гра, в якій велика кількість гравців взаємодіють один з одним у віртуальному світі. Зазначена популярність багато в чому пов'язана з тим, що в основі взаємодії користувачів лежить модель free-to-play – гра доступна безкоштовно, а прибуток йде від продажу ігрових предметів, що прискорюють отримання досвіду, і різної декоративної екіпіровки.

В індустрії Кі з'явилися різноманітні способи монетизації. Розробники МПІ створюють віртуальні простори, що функціонують на основі власної економічної системи. Гроші цієї системи залучають не тільки інвесторів, але і зловмисників. Число шкідливих програм, «крадуть» ігрові предмети і «викрадають» акаунти користувачів, зростає швидкими темпами. Особливо вразливі мобільні додатки, так як багато з них вимагають введення гравцем даних банківської карти. Будь-яка галузь, яка оперує персональними даними, як правило, стає об'єктом атак з боку тих, хто хоче отримати ці дані. Ігрова індустрія - не виняток.

Типова онлайнова гра розділена на серверну частину і ігровий клієнт, що встановлюється на комп'ютерах або мобільних пристроях гравців (користувачів).

Програмна платформа Кі забезпечує технічну базу, на основі якої реалізуються такі функції, як рендеринг графіки, імітація фізичних процесів, штучний інтелект, керуючий поведінкою ігрових персонажів, мережа, управління пам'яттю і т.д. враховуючи дуже високу складність цих програмних платформ, неможливо очікувати відсутності в них багів. І вони дійсно є завжди. Ці недоліки позначаються на роботі самих ігор, а не апаратних платформ, на яких вони реалізовані і

функціонують, – локальних комп'ютерах, серверах або мобільних пристроях.

В коротко Проаналізовані деякі уразливості сервера гри «Project I. G. I. 2: Covert Strike».

Існує багато типів атак з використанням помилок в програмних кодах ігор. Одна з таких атак - «Format string attack» - полягає в неправильній передачі параметрів у функцію printf.

Атака на основі форматування послідовності знаків. Атакуючий зазвичай використовує директиву %n, яка записує кількість символів, збережених даною функцією під область пам'яті, зазначену в наступному аргументі, як у прикладі на рис. 3.1.

Рис. 3.1. Пакет, який використовується в атаці, що форматує рядки

Цю атаку дуже легко здійснити. Найпростіший спосіб - ввести комбінації символів

%n % N в ігровому чаті. Це закриває додаток на сервері. Не тільки чат схильний до такої помилки. Введення зазначеної комбінації символів в будь-якому місці призводить до тієї ж реакції додатки. Наведений рядок коду сприймається як команда сервера. Ця комбінація символів також може бути відправлена в пакеті, що містить ім'я гравця.

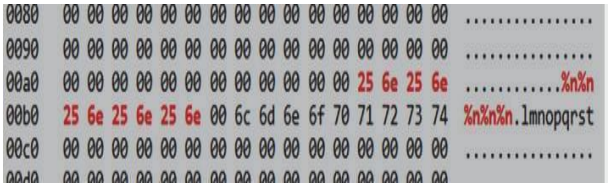

Рис. 3.2. Атакуючий рядок у пакеті з ім'ям гравця

Інше місце, де можна використовувати атаку, - це чат гри. В якості вмісту повідомлення відправляється вказаний рядок. На рис. 3.3 показаний фрагмент такого пакета.

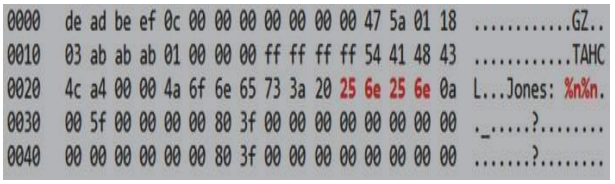

Рис. 3.3. Фрагмент пакета з повідомленням, що містить атакуючий рядок

Для цього типу атаки був створений захист у вигляді патчів. Однак такий захист призводить до можливості реалізації інших видів атак, не менш небезпечних. Наприклад, проблема зміни так званої ігрової карти. Якщо карта на сервері змінилася, гравці можуть зустріти труднощі з ідентифікацією.

Атака з переповненням буфера імені гравця. В області інформаційної безпеки корпоративних ресурсів дуже гостро стоїть проблема атак на мережу шляхом переповнення буфера. Основна особливість такої атаки полягає в наступному: якщо атакуючий зможе «підсунути» комп'ютера деякі інструкції у вигляді коду, комп'ютер виконає ці інструкції. Це є основою для нападу, пов'язаного з переповненням буфера. З формальної сторони переповнення буфера виникає, коли комп'ютерна програма записує дані («підсунуті» Інструкції) за межами простору, виділеного в пам'яті буфера.

Це призводить до перезапису інших даних в пам'яті, які можуть знадобитися для правильного функціонування програми. Одним із способів використання цієї атаки є переповнення буфера для імені гравця. Додаток має обмеження імені в 19 символів. Можна увійти в гру, ввівши відповідні параметри в командному рядку. Якщо вводиться рядок символів довжиною більше 64 в параметрі name, гравець увійде в систему з ім'ям лише з 64 символів. Щоб скористатися помилкою переповнення буфера, потрібно підготувати мережеві пакети, що відповідають за приєднання до гри. Для цього слід відправити пакет з ім'ям, наприклад, довжиною 66 символів (рис. 3.4).

Рис. 3.4. Фрагмент пакета переповнення буфера імені гравця

Досить добре з подібною проблемою можна впоратися, використовуючи міжмережеві екрани для аналізу мережевого трафіку.

Атака з переповненням буфера ключа. Це атака, аналогічна попередній, але тут використовується ключ до гри. При вступі в гру кожен гравець відправляє свій ключ (CD-KEY) в пакеті, який верифікується. Приклад пакета показаний на рис. 3.5.

Вміст пакету включає в себе зашифрований ключ гравця. Цей вміст можна розділити на дві частини. Перша, зазначена підкресленням, ніколи не змінюється. Її розмір становить 32 байта. Решта змінюється в кожному відправленому пакеті.

Якщо посилається пакет з довшим ключем на сервер, серверний додаток перестане працювати. Захист сервера від переповнення буфера імені гравця також є захистом від атаки з переповненням буфера ключа, оскільки цьому пакету передує підключення до гри.

Атака за запитом про ключ. Механізм перевірки CD-ключа має і інше слабке місце. Щоб перевірити, чи використовується іншими гравцями ключ, представлений даним гравцем, додаток використовує онлайн-валідацію. Ключ відправляється на сервери Gamespy, і потім відправник отримує відповідь про результати валідації. Всі повідомлення шифруються за допомогою функції XOR, використовуючи для шифрування рядок gamespy. У зазначеній відповіді зустрічаються запити, розділені символом backslash. При читанні запитів з'являється помилка. Якщо гравець відправить на сервер повідомлення з одним символом backslash, додаток закриється. Проблема тут криється в погано запрограмованому аналізаторі запитів. Це можна проілюструвати наступним фрагментом коду:

int size = strchr(buff + 1, '\\') - buff; if(size > 32) return; strncpy(querybuff, buff + 1, size);

Змінна buff містить запит. У ньому відшукується знак «\\». Потім перевіряється умова, і витягнутий текст поміщається в змінну querybuff. Помітна помилка в цьому програмному коді. Значення, що повертається функцією strchr, не перевіряється, тому, якщо функція не знаходить косу риску і повертає «0», функція strncpy видасть виняток, тому що значення змінної size буде негативним.

Рішенням описаної проблеми є патч, випущений Luigi Auriemm. Тип signed змінений на unsigned, тому значення не може бути негативним.

Атака на основі модифікації карти. Про Карту ми згадували в аналізі атаки на основі форматування послідовності знаків.

Редактор карт доступний кожному. Деякі модифікації можуть закрити програму. Файли карти на сервері повинні збігатися з файлами карти у гравця. Карта містить різні об'єкти: Будівля, Стіна, сходи та ін Кожен об'єкт має свій ідентифікаційний номер. Це важливо для інтерактивних об'єктів, таких як двері, кнопки, сходи. Коли гравець використовує один з об'єктів, він відправляє на сервер пакет з ідентифікаційним номером об'єкта. Потім сервер відсилає пакети з інформацією про діяльність гравця іншим учасникам гри. Завдяки цьому кожен може побачити ефект від використання об'єкта, наприклад відкриття дверей. Проблема виникає, коли гравець (або зловмисник) використовує об'єкт з ідентифікаційним номером, якого сервер «не знає». В цьому випадку програма закривається.

Карта може складатися з обмеженої кількості об'єктів. В аналізованій грі можна створити максимум 4096 об'єктів. Це полегшує захист від цієї атаки.

Якщо у всіх буде однаковий файл, який використовує максимальну кількість об'єктів, гра не закриється. Інший спосіб-створити базу даних всіх використовуваних ідентифікаційних номерів об'єктів і перевіряти, чи містить пакет ідентифікатор з цієї бази даних.

Крім розглянутих, існують і інші види атак на серверні і клієнтські додатки даної та інших комп'ютерних ігор. Часто вирішенням виникаючих проблем

займаються не тільки розробники ігор, але і самі гравці. В останньому випадку з'являються спеціалізовані програмні засоби.

Спеціалізовані програмні засоби для захисту сервера гри. У доступних джерелах міститься мало інформації про програмні продукти, призначені для вирішення зазначених завдань.

Ймовірно, одним з перших був багатовіконний додаток Project1. Воно має багато функцій, що полегшують роботу адміністратора (наприклад, відправка команд на сервер, написання загальних повідомлень гравцям, механізми попередження обману гравців). Project1 надає багато важливої інформації про сервер: кількість гравців, поточна карта, час гри, список гравців кожної команди, статистика гравців, IP-адреси, розмови в чаті.

Додаток Mautorun засноване на аналізі мережевих пакетів. Набуло великої популярності серед адміністраторів серверів.

3.2 Спосіб захисту багатокористувацької ігри

Однією з програм, що запобігають атакам на сервер, є Autobahn. Його основна функція полягає у виявленні атак з переповненням буфера. Це виявлення базується на аналізі різниці в часі, що відповідає відправленню пакетів приєднання до гри. Як правило, атакуюча сторона відправляє пакети протягом 1 с. приклад відповідних лінійок коду (насправді – двох) виглядає так:

[13:20:01] Server info sent to 192.168.1.1:26014
[13:20:01] NETWORKPACKET_TYPE_
CLIENTCONNECT [192.168.1.1:26015]

Можна помітити, що обидва рядки були створені протягом 1 С.це означає, що були відправлені шкідливі пакети. Програма AutoBan витягує IP-адресу і блокує його. Якщо сервер працює швидко, останній пакет, який повинен закрити сервер, буде нейтралізований.

Другий спосіб виявити атаку - це перевірити порти. Якщо хакер відправляє кожен пакет з іншого сокета, то це призводить до зміни порту. Якщо дві лінії вказують на різні порти, це-ймовірно, атака.

Авторське додаток для захисту сервера МПІ. Для нейтралізації описаних вище атак на сервер аналізованої гри нами розроблено спеціальний додаток.

Клієнтська програма використовується для взаємодії з користувачем, відповідає за реєстрацію і вхід в систему, налаштування облікового запису і приєднання до гри. Другий модуль - це PHP-сервер з базою даних на платформі MySQL. Його завдання-проаналізувати дані, отримані з клієнтської програми, перевірити їх коректність і повернути необхідну інформацію гравцеві. Інформація про користувачів зберігається в базі даних. Останній модуль являє собою серверну програму, яка була інтегрована в існуючу програму управління сервером. Модуль призначений для перевірки того, чи запитує користувач доступ до сервера, а також для контролю винятків в брандмауері. Додаючи виключення в брандмауер, користувач отримує доступ до сервера.

Клієнтська частина написана на C# з використанням технології. NET. основною перевагою цієї технології є доступ до багатьох бібліотек, що містять готові рішення аналізованої проблеми.

Серверний додаток створено на Java з використанням технології Maven і бібліотеки jnetpcap, призначеної для аналізу мережевого трафіку. Комунікаційний сервер створений за допомогою технології управління базами даних MySQL і мови PHP.

До особливостей розробленого додатка можна віднести наступне.

Після отримання списку очікують користувачів програма викликає команду для додавання інструкції в брандмауер сервера:

```
iptables -A INPUT -s 192.168.0.1 -p udp -
-dport 26001 -j ACCEPT
```

Таке правило приймає UDP-пакети, що надходять на порт 26001 з IP-адреси 192.168.0.1.

Унікальний ключ CD-KEY повинен бути збережений в системному реєстрі.

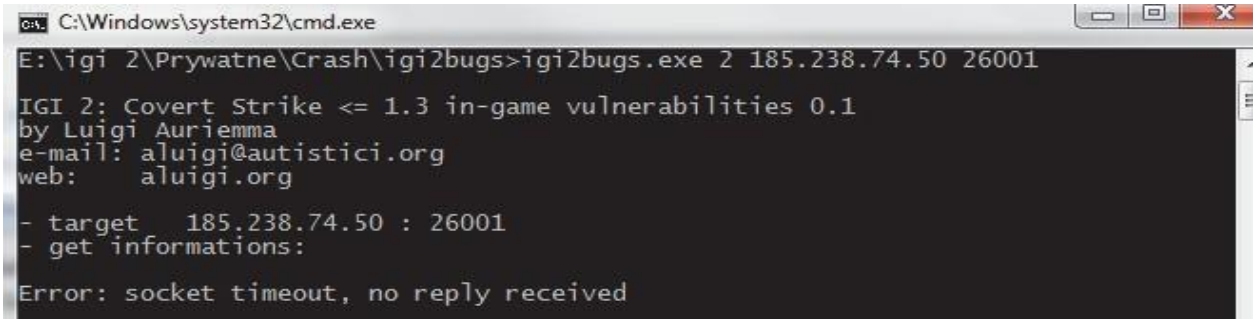

Рис. 3.7. Ілюстрація реалізації атаки без реєстрації користувача

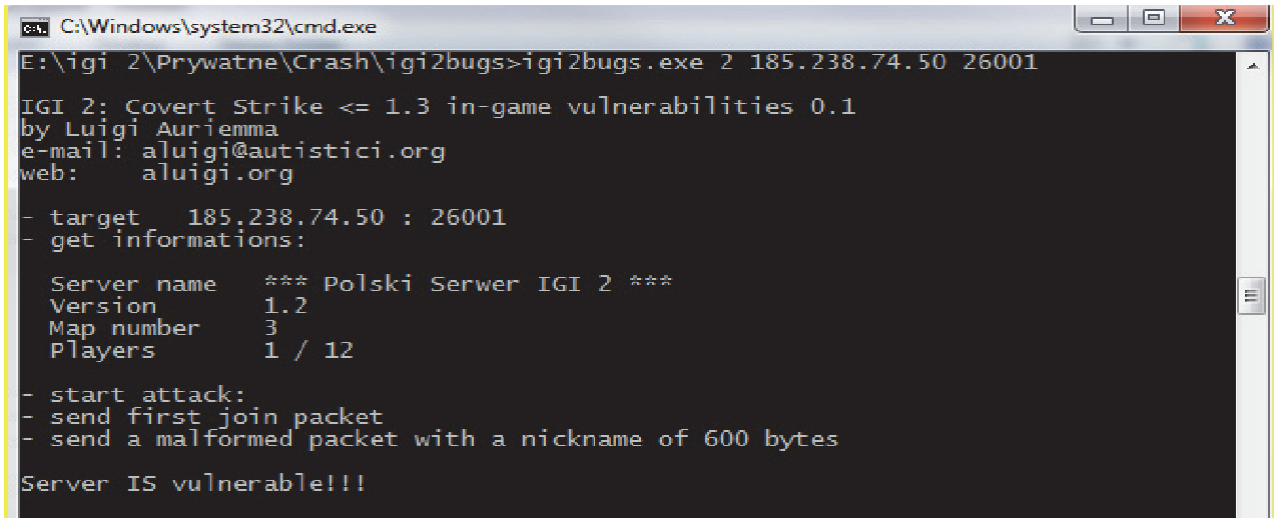

Рис. 3.8. Ілюстрація реалізації атаки зареєстрованого Користувача

Програма на сервері перевіряє статус гравців кожні 5 з, відправляючи запит в базу даних. Доступ до сервера виходить шляхом додавання необхідної інструкції в брандмауер. Якщо спробувати отримати доступ до сервера без запиту доступу, то відповідні пакети будуть проігноровані, тому що весь вхідний трафік буде заблокований. Спроба увійти на сервер без використання клієнтської програми не призведе до отримання відповіді від сервера.

Спроба атакувати сервер за допомогою добре відомої програми igi2bugs також зазнає невдачі, як це показано на скріншоті (мал. 3.8), оскільки сервер відхиляє весь трафік. Це робить систему несприйнятливою до атак незареєстрованих користувачів.

Якщо Користувач зареєструється, він все одно не зможе атакувати, оскільки він не відправив запит на доступ до сервера. Тільки після натискання кнопки «приєднатися до гри» («Join game») клієнтська програма відправляє такий запит. На жаль, система не в змозі заблокувати спробу атаки зареєстрованого гравця. Таку ситуацію ілюструє рис. 3.8.

У більшості комп'ютерних ігор останнього покоління практично немає вразливостей, властивих описаної грі. Але немає жодного додатка, абсолютно захищеного перед атаками. Неможливо також захистити багатокористувацьку комп'ютерну гру на 100%. Однак можна знизити її вразливості на основі аналізу попередніх подій.

Аналіз мережевого трафіку між сервером і клієнтським додатком (гравцем) показав, що більшість атак на сервер багатокористувацької комп'ютерної гри можна заблокувати.

При виборі системи, на якій буде запускатися ігровий сервер, необхідно враховувати час відгуку при додаванні необхідних інструкцій в брандмауер. Система Linux дозволяє фільтрувати мережевий трафік, використовуючи iptables. Брандмауер Windows повільніше реагує на додавання нових інструкцій. Також доступні поліпшені серверні додатки, які блокують деякі види атак.

Наш практичний досвід показав досить високу ефективність використання масштабування базової програмної платформи комп'ютерної гри «Project I. G. I. 2: Covert Strike» для захисту ігрового сервера від зловмисників.

ВИСНОВКИ

У роботі описана методика можливості підвищення ефективності та швидкості захисту ігрових серверів. Для досягнення мети роботи був проведений аналіз функціонування та стану веб-серверів. Поширення вразливостей веб-серверів і можливість атак через уразливості також виявляються і аналізуються, адже важливим етапом тесту на проникнення є Пошук вразливостей. Також були проаналізовані структури, що шукають слабкі місця. В ході дослідження ми знайшли стандарти обробки даних про вразливості, показники критичності вразливостей та інструменти для їх пошуку. І прийшов до висновку, що більшість методів охоплюють широкий спектр тим кібербезпеки, тому необхідно додатковий час на аналіз вразливостей за існуючими методами і зокрема на підбір тих компонентів, які підходять для тестування веб-додатків.

За допомогою результатів аналізу і досліджень адаптована методологія, яка проникає в мережу і враховує міжнародні досягнення в цьому напрямку. Саме тому він дозволяє перевіряти найпоширеніші уразливості. А вибір перевірки вразливостей тільки з критичним рівнем ризику підвищує ефективність. Практичний інтерес роботи полягає в скороченні тривалості та обґрунтуванні вибору засобів тестування на проникнення.

Вивчення атак і способів захисту від них показало, що від деяких атак захиститися неможливо, це атаки типу читання моделей або текстур з відеопам'яті, коли моделі завантажуються туди з ігрового додатка, більш того захист буде гальмувати вниз гру, ускладнити установку програмного забезпечення. Тому необхідно заздалегідь визначити, які ресурси найбільш важливі для правильного функціонування ігрового додатка, і захистити їх.

В даний час більшість розробників відмовляються захищати свої ігрові програми, тому що це не тільки дорогий процес, але і не приносить користі ігровим додаткам. Деякі інтернет-магазини погоджуються продавати ігрове додаток до тих пір, поки в ньому немає DRM (наприклад, GoG.com).

Великі компанії виграють від безпеки, але лише протягом обмеженого часу, поки її не зламають. А потім вони знімають захист з гри, щоб додаток працювало як задумано.

У незахищеному багатокористувацькому ігровому онлайн-додатку зловмисник завдає шкоди не тільки своєму клієнту і розробнику, а й іншим користувачам цієї відеоігри, тому захист онлайн-додатки є обов'язковим для всіх розробників.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Дацко М. А. Моделювання складних об'єктів. / М. А. Дацко. – М. : Максімас, 2015. – 111 с.

2. Крейтон, Р.Х. Основи розробки ігор у Unity / Р.Х. Крейтон. – Packt Publishing, 2010, – 83 с.

3. Любанова, Т.П. Бизнес-план: опыт, проблемы. Содержание бизнес-плана, пример разработки / Т.П. Любанова, Л.В. Мясоедова, Т.А. Грамотенко, и др.. – М.: Приор, 2012. – 204 с.

4. Хокінг Д. М. Unity в дії. Мультиплатформенна розробка на практиці. / Д. М. Хокінг, 2016. – 336 с.

5. Хорхе Паласиос Unity 5.x. Программирование искусственного интеллекта в играх. Руководство / Паласиос Хорхе. – М.: ДМК Пресс, 2017. – 427 с.

6. Apperley T. H. Genre and game studies: Toward a critical approach to video game genres / T. H. Apperley // Simulation & Gaming. – 2006. – Vol. 37. – No. 1. – P. 6 – 23.

7. Buckland Mat. Programming Game AI by Example – Texas, Wordware Publishing. – 2004. – s. 25 – 43.

8. Clearwater D. What Defines Videogame Genre? Thinking about Genre Study after the Great Divide / D. Clearwater // The Journal of the Canadian Game Studies Association. – 2011. – No. 5. – s. 29 – 49.

9. Goldstone W. Unity Game Development Essentials. / W Goldstone. Birmingham: Packt Publishing Ltd. – 2009. – 316 s.

10. Gregory Jason. Game Engine Architecture. – New York, CRC Press. – 2009. – No. 5. – s. 15 – 24.

11. Gregory Jason. Game Engine Architecture: Second Edition. – New York, CRC Press. – 2014. – No. 32. – s. 23 – 30.

12. Lengyel Eric. Mathematics for 3D Game Programming and Computer Graphics: Third Edition. – Boston, Course Technology. – 2012. – 215s. 82

13. McShaffry, Mike, Graham David. Game coding complete: Fourth Edition. – Boston, Cource Technology. – 2013. – 184s.

14. Microsoft documentation Справочник по языку C# [Електронний ресурс] / режим доступу: https://docs.microsoft.com/ru-ru/dotnet/csharp/languagereference/language-specification/introduction.

15. Unity Asset Store [Електронний ресурс] / режим доступу: https://assetstore.unity.com/

16. Unity Manual [Електронний ресурс] / режим доступу: https://docs.unity3d.com/Manual/index.html.

17. Tobold What is an MMORPG actually? – [електронний ресурс]. – Режим доступа: http://tobolds.blogspot.com/2003/07/what-is-mmorpg-actually.html

18. Luo L— Study on Security Protection of User's Personal Information in Social Networks. Journal of Library and Information Sciences — 2012. — № 14. — С. 36-40.

19. Зима В.М., Молдовян А.А., Молдовян Н.А. Защита компьютерных ресурсов от несанкционированных действий пользователей. – Учеб пособие. – СПб: Издательство ВИКА им. А.Ф. Можайского, 1997.

20. Афанасьева К.О. Авторське право: Практичний посібник. – К.: Атака, 2006. – 224 с.

21. Unity Objuscator [електронний ресурс]. – Режим доступа: http://devxdevelopment.com/