GU J Sci, Part A, 7(2): 43-57 (2020)

Gazi University

Journal of Science

PART A: ENGINEERING AND INNOVATION

http://dergipark.gov.tr/gujsa

ONLINE Game Security: A Case Study of an MMO Strategy Game

Ahmet EFE[1,*], Emre ÖNAL[2]

[1]*Ankara Development Agency, Ankara, Turkey*
[2]*Department of Computer Science, Yıldırım Beyazıt University, Ankara, Turkey*

Article Info

Received: 14/02/2019
Accepted: 20/07/2020

Keywords

Online Games,
Game Security,
Risks and threats,
Cyber game

Abstract

Just as the Internet and the digital age are transforming life practices in every direction, the habits of playing games are taking its dominance with pervasive addiction while security concerns are alarming. Different terms are being used for the security issues of games in the literature like cheat, exploit, hack, attack. Massively multiplayer online role-playing games are joining the ranks of software popular enough to be bombarded by attacks. In this study, we explore the security issues of an online game and its protection measures on a sample game. We will use a Massively Multiplayer Online (MMO) strategy game that we have been developing as our case study, which utilizes different aspects of MMO concept that contains many security issues. We tried to point out the vulnerable points of MMO games and provide sufficient solutions to these problems by using an MMO strategy game as in our case study.

1. INTRODUCTION

Game producers have online games that are suitable for almost every area of interest such as action, strategy, leap and running, shooting, simulation, role games or sports. In online games, children and young people can play live with other groups, make exchanges and work on campaigns or strategies together. However, in detail, the obligation to engage their talents at the top level and learn to accept their playmates is the issue. Most online games are linked to social networks like Facebook in order to increase social collaboration and interaction. Apart from this, players can post their game scores and they are usually rewarded with a game bonus as a result.

Online games differ not only with what is played, but also with where they are played. Browser-games are online games that can be played through the normal web-browser. It is not necessary to install software to play this kind of games. The browser is used as a point of intersection between the player and the game world. Browser-games are usually fairly simple games with very low entry disabilities that only take a little time to start with.

Massively Multiplayer Online Games (MMOGs) are very popular. MMOGs are often role games: Players choose a virtual game figure, an avatar, and constantly improve their skills with their gaming experience. In a group of playmates called a clan or guild they work together to solve missions. Virtual worlds like Second Life[1] are not online games in the narrow sense, but they are often discussed in this context. Here, users are usually living a life in a kind of parallel world, which depicts a realistic environment in a very detailed way. The player here is in constant connection with other players or their game figures.

*Corresponding author, e-mail: icsiacag@gmail.com

[1] Second Life is an online multiplayer open world game. In the game, which has its own currency, economy and social community, you choose a character and avatar and log in. It offers a chance to interact with other people and shop in the same way as in the real world. For details see: https://secondlife.com/

Many online games require the player's record and personal data to be disclosed. Here is a special note and simple rules: “The less is better”. If an e-mail address is requested, it is reasonable to use anonymous or just an address set up for such purposes. See general business terms (AGB) and / or data protection provisions to understand what data is being used for (for example, for advertising purposes). It is much better to choose servers that do not need personal data such as e-mail addresses, social-media-profiles or phone numbers.

Within the last decade, there has been an explosive growth in online gaming. Accordingly, the number of players has increased enormously. As an example in September 2016 Riot games reported that their game League of Legends, which is one of the most popular online games, has over 100 million active players monthly [1].

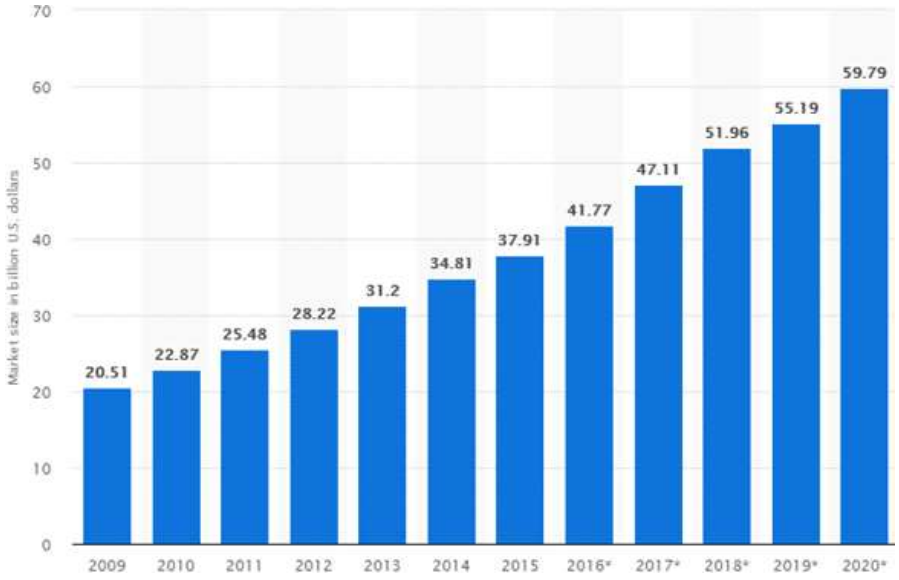

Figure 1. *PC online game market value worldwide from 2011 to 2019 (in billion U.S. dollars)[2]*

Online gaming includes a variety of different types of games such as from Mobile Gaming, to Social gaming, free to play gaming sites, massive multiplayer online games (otherwise known as MMOs), online casinos, bingo, online lotteries, sports betting, eSports betting platforms and services. One thing is for sure according to the trends, this industry it will keep on growing. From the popular co-op sandbox survival game Fornite[2], to the old school Candy Crush[3], or casino games such as Blackjack[4], everyone seems to be involved in some kind of online gaming activity nowadays. According to Statista, the online gambling gaming industry alone rose from 20.51 Billion US Dollars in 2009 to 47.11 US Billion Dollars in 2017 and it is expected to rise to 59.79 US Billion Dollars by 2020.

When computer games used to be single-player, the main security concern was copyright protection. In-game cheatings were not considered as an important problem for single-player systems. However, online game systems brought many different security concerns together. First massively distributed client-server architecture has much vulnerability. Number of the users (players) of these systems may reach millions, and many of these users are emotionally attached to this system. Lifetime of online games is much more than the lifetime of single player (offline) games. So it can continue to make money for long period of time but some players may try to abuse existing exploits for gaining advantages over other players or for other clandestine purposes. If the game company could not handle these problems successfully and provide a safe gaming environment to its players, they tend to leave the game much more sooner, which in turn may result a massive loss for the gaming company.

[2] For details see: https://www.epicgames.com/fortnite/en-US/buy-now/battle-royale
[3] For details see: https://king.com/es/game/candycrush
[4] For details see: https://www.casino.com/uk/blackjack/

Also for some of the games, it is possible to make real-life profit via selling in-game items, characters, in-game currency etc. This makes online games target for people who have other intentions than enjoyment. As a result, security of online games includes classic security concerns as confidentiality, integrity, availability and privacy.

Considering that online gaming will be included in the artificial intelligence platforms, deep learning and noral netwoks, the vulnerabilities and threat vectors that should be taken into consideration might severely vary. AlphaGo, an artificial intelligence used in Google-signed games, is the first in history, and more importantly, it means that an important threshold in terms of artificial intelligence has been overcome. Because the most important feature that distinguishes AlphaGo[5] from software that can achieve a similar victory is that it has been developed with a general purpose algorithm. For the comparison, IBM's Deep Blue[6] computer, which defeated Gerry Kasparov in 1997, achieved this victory only by taking “training” for chess. AlphaGo's software, on the other hand, allows the game to interpret patterns related to the game, to think and make decisions in the long term. Similarly, DeepMind researchers have announced that it has developed artificial intelligence technology that has learned to master video games on its own and has mastered it as it plays. The program started by learning Atari games and left the records of human opponents behind in 29 of 49 different Atari games. Another player on the artificial intelligence board in games in artificial intelligence studies is Facebook. Mark Zuckerberg provided information on Facebook's work in this area that the researchers developed an artificial intelligence capable of playing online games.

VR games are undoubtedly one of the most vocal projects of recent years. One of the most influential points in VR games is that the headsets and glasses that offer virtual reality become compatible with the new generation game consoles. These games can be played in the online platforms with interaction. This situation causes game makers to turn to games that increase experience. Consoles that can transfer physical movements to the game become stronger every year. Looking at this technology, which is at the baby-crawling level for now, is expected to become more realistic in the coming periods with noral networks. When game consoles are combined with AI online, the concept of artificial intelligence, it is thought that it will bring the game technology that we have difficulty in imagining with the possible risks and vulnerabilities that we can imagine.

In this study we will explore the security issues of an online game and its protection measures on a sample game. We will use a Massively Multiplayer Online (MMO) strategy game we have been developing as our case study, which utilizes different aspects of MMO concept that contains many security issues. First, we will explain the architecture and communication mechanism of the client and the server applications. Then we will introduce the vulnerable points of this architecture and provide possible solutions to these problems.

There are different studies in the literature about this topic. Yan examined the impact of new security requirements on the design of online games by using an online game called Bridge as their case study [2]. Hu and Zambetta provided a study specifically focused on security issues in massive online games [3]. They also provided a framework for the classification of cheating in MMO games. Woo and Kim prepared a survey on academic researches and industry practices related to online game security [4].

2. GENERAL RISKS AND PROBLEMS OF ONLINE GAMES

By 2020, with the number of online players reaching 50 million globally and online-digital games have a growing user base in Turkey. In fact, the ability of the "digital locals" born into technological

[5] For details see: https://deepmind.com/research/case-studies/alphago-the-story-so-far
[6] For details see: https://www.scientificamerican.com/article/20-years-after-deep-blue-how-ai-has-advanced-since-conquering-chess/

developments to keep up with the evolving developments has caused a serious increase in the playing of digital games.

Digital games attract people of all ages from children to adults. In addition to the benefits, digital games, where risks are also present, will come to a conscious, supervised and minimized risk -habits of usage if the balance is placed on a scale. Besides, the security risks in digital games and the precautions that can be taken against these risks are at the beginning of the topics to be emphasized. Online games involve a variety of risks in social and technological areas, which in general are as follows:

- Risks from communication with people who want to steal personal and financial information.
- Risks caused by malicious people who want to take advantage of computer security vulnerabilities.
- Risks that are caused by criminals seeking victims on the Internet and in the real world.
- Risks from malware such as Trojans, computer worms, spyware, bots and viruses.

The average US family has at least one dedicated game console, PC or smartphone, according to the study by the Entertainment Software Association. Mobile devices are becoming an important part of the gaming industry. NPD Group reports that 63 percent of children between the ages of two and 17 are using mobile devices to play games. Although online games can offer quality social interaction, there is also a dark side. From cyberbullying to online hunters and hidden fees, there are a variety of issues to be concerned about when playing games on the Internet. The most important thing a parent can do is teach safe internet use at an early age and continue as children grow. When they understand the risks and the importance of security, they are much more likely to come to you on issues they are concerned about.

2.1. Cyberbullying

For most children, fleeing to the internet world offers the opportunity to get away from real life. Nobody knows who they are, which school they went to and how they looked. However, this anonymity has two aspects. Cyberbullying has been recognized as a serious social concern. Considering the varied contexts of online engagement by children and youth is increasingly necessary to adequately understand their experiences and the impact of their participation. An online context which requires further attention is gaming platforms, which are especially popular among boys.[7]

2.2. Privacy Issues

Stay Safe Online recommends that children never use usernames derived from their real names, or their sex or age[8]. According to US-CERT, the social nature of online games allows cybercriminals to manipulate conversations. They can choose a child on a public chat channel and start sending personal messages asking for detailed personal information. By combining data from games and other sources, pirates can create accounts on a child's behalf or gain access to existing accounts.

2.3. Personal Information Left on Consoles and Computers

Another online gaming hazard is caused by consoles or PCs themselves[9]. After exceeding their usage period, most families send these devices to the local electronic recycling center or sell them on exchange sites. Users often forget to delete their files and personal information and put their financial and private information at risk. You need to delete all personal data from game consoles, tablets and smartphones and then do a factory reset. Required tools or procedures may vary depending on the type of device. It is therefore important to investigate this for each device.

[7] For more information see: https://link.springer.com/article/10.1007%2Fs10560-017-0498-0
[8] For more information see: https://www.trendmicro.com/vinfo/us/security/news/online-privacy/data-privacy-and-online-gaming-why-gamers-make-for-ideal-targets
[9] For more information see: https://www.priv.gc.ca/en/privacy-topics/technology/mobile-and-digital-devices/digital-devices/gd_gc_201905/

2.4. Webcam Risks

According to Business Insider, over 4,500 webcams in the US have been hacked last year and published on a Russian website. All connected devices such as webcams or audio devices can be controlled by attackers and used to exploit your children. To reduce this risk, regularly scan your system and make sure your webcam's default setting is "off"[10].

2.5. Online Hunters

Online hunters are often older people who play video games to fool smaller victims. As a result, inappropriate messages, webcam chats, and even face-to-face meetings that can lead to sexual abuse[11]. Online games offer hunters the opportunity to create a common online experience, thereby becoming a child's advocate or teammate. After defeating a tough enemy or discovering a new area in the game, hunters connect with young players and create common experiences that turn into personal questions. In most cases, hunters try to provoke children against their parents and become "the only one who understands them."

2.6. Hidden Fees

Some online games use the "free game" model. It gives you some content for free and requires payment to access other parts of the game. According to Mashable, Windows 10 users need to pay to be able to play certain classic game modes without interrupting ads. Players can use real money to buy a virtual sword or armor, or pay by credit card to get gold or experience for their own characters. Furthermore, many games offer players the opportunity to buy upgrades using microtransactions. The current research reveals the hidden costs that microtransactions have, as players respond negatively to other players using them.[12]

In most cases, these games require a credit card to register and start playing, and are automatically charged when users decide to purchase new items or services. Never give your card number for free games. Even if your child is playing more traditional membership-based games, it's a good idea to check your credit card statements regularly to make sure there are no charges for disapproved purchases. If you allow your child to use your smartphone or tablet, seriously consider turning off "in-app updates" to prevent your child from inflating your statement with in-app purchases without you noticing.

2.7. Malware

Trojans can replace a legitimate application and install the harmful version on Google Play or another legitimate marketplace. PC World reported a recent example of this: when this Trojan was downloaded, it was able to run, control the user's Android device and make it a part of a larger "botnet" to be used in DDoS attacks. Hacking groups, such as Lizard Squad, seek first and foremost to gain attention, and they realize that online gaming has an inherent “disruption amplification” effect—making them very attractive targets for those hungry for notoriety.[13] For this reason, the victims do not understand that the source is online games. The lesson here is that you should always pay attention to the applications you download. Applications may appear legitimate or disguised as legitimate applications.

[10] For more information see: https://www.avg.com/en/signal/how-hackers-can-hijack-your-webcam-to-spy-on-you
[11] For more information see: https://www.nytimes.com/interactive/2019/12/07/us/video-games-child-sex-abuse.html
[12] For further details see: https://www.ijis.net/ijis10_1/ijis10_1_evers_et_al.pdf
[13] For further details see: https://www.imperva.com/blog/ddos-attacks-on-online-gaming-servers/

3. ZERODAY EXPLOITS OF GAMES

There are many zero-day exploits that are published in the exploitdb[14] database. We have searched zeroday exploits which are published in the exploitdb. Detailed information of some declared vulnerabilities can be reached in the hyperlinks of the title given below according to their types and platforms.

Table 1. *Demonstration of some of the vulnerabilities confirmed in the expoloit database*

Date	Title	Type	Platform
2014-07-28	WordPress Plugin FB Gorilla - 'game_play.php' SQL Injection	WebApps	PHP
2013-05-07	MyBB Game Section Plugin - 'games.php' Multiple Cross-Site Scripting Vulnerabilities	WebApps	PHP
2011-04-03	RealNetworks GameHouse 'InstallerDlg.dll' 2.6.0.445 ActiveX Control - Multiple Vulnerabilities	Remote	Windows
2009-08-27	StandAloneArcade 1.1 - 'gamelist.php' Cross-Site Scripting	WebApps	PHP
2009-08-27	E-Gold Game Series: Pirates of The Caribbean - Multiple SQL Injections	WebApps	PHP
2009-08-30	e-Soft24 Flash Games Script 1.0 - Cross-Site Scripting	WebApps	PHP
2010-07-21	Monolith Lithtech Game Engine - Memory Corruption	DoS	Multiple
2010-07-05	Multiple Tripwire Interactive Games - 'STEAMCLIENTBLOB' Multiple Denial of Service Vulnerabilities	DoS	Windows
2009-12-31	Freewebscriptz Online Games Login - Multiple SQL Injections	Remote	Windows
2010-07-21	id Software id Tech 4 Engine - 'idGameLocal::GetGameStateObject()' Remote Code Execution	Remote	Windows
2010-05-13	GameCore 2.5 - 'GameID' Integer Overflow	Remote	Windows
2010-05-09	Torque Game Engine - Multiple Denial of Service Vulnerabilities	DoS	Windows
2008-09-11	Epic Games Unreal Engine 436 - Multiple Format String Vulnerabilities	Remote	Multiple
2008-06-23	PEGames - Multiple Cross-Site Scripting Vulnerabilities	WebApps	PHP
2008-05-20	Starsgames Control Panel 4.6.2 - 'index.php' Cross-Site Scripting	WebApps	PHP
2008-05-08	Ourgame 'GLIEDown2.dll' ActiveX Control - Remote Code Execution	Remote	Windows
2014-01-24	Daum Game 1.1.0.5 - ActiveX 'IconCreate Method' Remote Stack Buffer Overflow	Remote	Windows
2008-02-05	GlobalLink 2.6.1.2 - 'HanGamePlugincn18.dll' ActiveX Control Multiple Buffer Overflow Vulnerabilities	Remote	Windows
2007-12-22	MyBlog 1.x - 'Games.php?ID' Remote File Inclusion	WebApps	PHP
2007-08-20	Epic Games Unreal Engine Logging Function - Remote Denial of Service	DoS	Multiple
2007-08-18	gMotor2 Game Engine - Multiple Vulnerabilities	Remote	Multiple
2006-03-06	Game-Panel 2.6 - 'login.php' Cross-Site Scripting	WebApps	PHP
2013-06-24	Top Games Script 1.2 - 'play.php?gid' SQL Injection	WebApps	PHP
2005-10-26	IPBProArcade 2.5.2 - 'GameID' SQL Injection	WebApps	PHP
2005-05-26	Clever's Games Terminator 3: War of the Machines 1.16 Server - Remote Buffer Overflow	Remote	Multiple
2005-05-24	Gearbox Software Halo Game Server 1.06/1.07 - Infinite Loop Denial of Service	DoS	Windows
2005-05-17	War Times - Remote Game Server Denial of Service	DoS	Windows
2005-03-20	FUN labs Game Engine - Multiple Remote Denial of Service Vulnerabilities	DoS	Windows
2005-02-28	MercurySteam Scrapland Game Server 1.0 - Remote Denial of Service	DoS	Multiple
2005-02-21	Bontago Game Server 1.1 - Remote Nickname Buffer Overrun	Remote	Multiple
2005-02-02	People Can Fly Painkiller Gamespy 1.3 - CD-Key Hash Remote Buffer Overflow	Remote	Multiple
2004-12-17	Interactive Studio GamePort 3.0/3.1/4.0 - Arbitrary Application Execution	Remote	Windows
2004-12-10	Gamespy Software Development Kit - CD-Key Validation Buffer Overflow	DoS	Linux
2004-11-22	Gearbox Software Halo Game 1.x - Client Remote Denial of Service	DoS	Multiple
2004-11-05	Monolith Lithtech Game Engine - Multiple Remote Format String Vulnerabilities	Remote	Multiple
2004-09-20	Impressions Games Lords of the Realm III - Nickname Remote Denial of Service	DoS	Windows
2004-09-09	Gearbox Software Halo Combat Evolved 1.x - Game Server Remote Denial of Service	DoS	Windows
2013-02-17	Scripts Genie Games Site Script - 'index.php?id' SQL Injection	WebApps	PHP
2004-04-22	Epic Games Unreal Tournament Engine 3 - UMOD Manifest.INI Arbitrary File Overwrite	Remote	Multiple
2004-04-17	BSD-Games 2.x - Mille Local Save Game File Name Buffer Overrun	Local	BSD
2004-03-11	Targem Games Battle Mages 1.0 - Remote Denial of Service	DoS	Multiple
2004-03-10	Epic Games Unreal Tournament Server 436.0 - Engine Remote Format String	DoS	Multiple
2004-02-24	Gamespy Software Development Kit - Remote Denial of Service	DoS	Linux
2004-02-24	RedStorm Ghost Recon Game Engine - Remote Denial of Service	DoS	Multiple
2004-02-24	Digital Reality Game Engine 1.0.x - Remote Denial of Service	DoS	Windows

[14] The Exploit Database is maintained by Offensive Security, an information security training company that provides various Information Security Certifications as well as high end penetration testing services. The Exploit Database is a non-profit project that is provided as a public service by Offensive Security. For more information see: https://www.exploit-db.com/

Date	Title	Type	Platform
2004-02-21	LGames LBreakout2 2.2.2 - Multiple Environment Variable Buffer Overflow Vulnerabilities	Local	Linux
2004-02-16	Freeform Interactive Purge 1.4.7/Purge Jihad 2.0.1 Game Client - Remote Buffer Overflow	Remote	Multiple
2004-02-09	Nadeo Game Engine - Remote Denial of Service	DoS	Linux
2004-02-02	0verkill 0.16 - Game Client Multiple Local Buffer Overflow Vulnerabilities	Local	Linux
2003-09-30	Gamespy 3d 2.62/2.63 - IRC Client Remote Buffer Overflow	DoS	Linux
2003-08-25	BSD-Games 2.x - Monop Player Name Local Buffer Overrun (2)	Local	BSD
2003-08-25	BSD-Games 2.x - Monop Player Name Local Buffer Overrun (1)	Local	BSD
2003-05-09	Lgames LTris 1.0.1 - Local Memory Corruption	Local	FreeBSD
2003-02-05	Epic Games Unreal Engine 436 - URL Directory Traversal	Remote	Multiple
2003-02-05	Epic Games Unreal Engine 436 - Client Unreal URL Denial of Service	DoS	Multiple
2003-01-17	GameSpy 3D 2.62 - Packet Amplification Denial of Service	DoS	Linux
2002-07-03	Epic Games Unreal Tournament Server 436.0 - Denial of Service Amplifier	DoS	Multiple
2001-12-07	Volition Red Faction 1.0/1.1 - Game Server/Client Denial of Service	DoS	Windows
2000-04-20	RealNetworks Real Server 7.0 / GameHouse dldisplay ActiveX control 0 - Denial of Service	DoS	Windows
2000-03-08	GameHouse dldisplay - ActiveX control 0 / Real Server 5.0/7.0 Internal IP Address Disclosure	Remote	Windows
1999-11-04	Real Networks GameHouse dldisplay ActiveX control - Port Buffer Overflow (2)	Remote	Windows
1999-11-04	Real Networks GameHouse dldisplay ActiveX control - Port Buffer Overflow (1)	Remote	Windows
2012-01-18	DZCP (deV!L`z Clanportal) Gamebase Addon - SQL Injection	WebApps	PHP
2011-04-09	Real Networks Arcade Games - StubbyUtil.ProcessMgr ActiveX Arbitrary Code Execution (Metasploit)	Remote	Windows
2011-04-03	RealNetworks RealGames StubbyUtil.ProcessMgr.1 - ActiveX Control Multiple Remote Command Executions	Remote	Windows
2011-04-03	RealNetworks RealGames StubbyUtil.ShellCtl.1 - ActiveX Control Multiple Remote Command Executions	Remote	Windows
2011-01-26	PHPDirector Game Edition - 'game.php' SQL Injection	WebApps	PHP
2010-12-18	Mafia Game Script - SQL Injection	WebApps	PHP
2010-10-29	mygamingladder MGL Combo System 7.5 - 'game.php' SQL Injection	WebApps	PHP
2010-10-09	Chipmunk Pwngame - Multiple SQL Injections	WebApps	PHP
2010-06-19	Hacker Evolution Game: untold Mod Editor 2.00.001 - Buffer Overflow (PoC)	DoS	Windows
2010-06-13	Eyeland Studio Inc. - 'game.php' SQL Injection	WebApps	PHP
2010-05-17	PHP Gamepage - SQL Injection	WebApps	PHP
2010-04-16	Joomla! Component com_pandafminigames - SQL Injection	WebApps	PHP
2010-04-12	Joomla! Component FlashGames 1.5.0 - Local File Inclusion	WebApps	PHP
2010-04-12	Joomla! Component Arcade Games 1.0 - Local File Inclusion	WebApps	PHP
2010-03-26	Joomla! Component dcsFlashGames 2.0RC1 - 'catid' SQL Injection	WebApps	PHP
2010-02-25	GameScript 3.0 - SQL Injection	WebApps	PHP
2010-02-11	Video Games Rentals Script - SQL Injection	WebApps	Multiple
2010-02-11	GameRoom Script - Authentication Bypass / Arbitrary File Upload	WebApps	PHP
2010-01-22	Joomla! Component com_gameserver - SQL Injection	WebApps	PHP
2010-01-06	PHPDirector Game Edition 0.1 - Local File Inclusion / SQL Injection / Cross-Site Scripting	WebApps	PHP
2009-12-31	Freewebscript'z Games - Authentication Bypass	WebApps	PHP
2009-09-22	BPGames 1.0 - Blind SQL Injection	WebApps	PHP
2009-09-01	Joomla! Component com_gameserver 1.0 - 'id' SQL Injection	WebApps	PHP
2009-05-21	ChinaGames - 'CGAgent.dll' ActiveX Remote Code Execution	Remote	Windows
2009-05-14	My Game Script 2.0 - Authentication Bypass	WebApps	PHP
2009-04-20	fungamez rc1 - Authentication Bypass / Local File Inclusion	WebApps	PHP
2009-03-31	vsp stats processor 0.45 - 'gamestat.php?gameID' SQL Injection	WebApps	PHP
2009-01-28	gamescript 4.6 - Cross-Site Scripting / SQL Injection / Local File Inclusion	WebApps	PHP
2008-09-13	Sports Clubs Web Panel 0.0.1 - Remote Game Delete	WebApps	PHP
2008-09-05	Vastal I-Tech MMORPG Zone - 'game_id' SQL Injection	WebApps	PHP
2008-06-07	Joomla! Component GameQ 4.0 - SQL Injection	WebApps	PHP
2008-05-07	GameCMS Lite 1.0 - 'systemId' SQL Injection	WebApps	PHP
2008-04-26	PostNuke Module pnFlashGames 2.5 - SQL Injection	WebApps	PHP
2008-02-22	Quantum Game Library 0.7.2c - Remote File Inclusion	WebApps	PHP
2008-02-19	Ourgame GLWorld 2.x - 'hgs_startNotify()' ActiveX Buffer Overflow	Remote	Windows
2007-07-21	RGameScript Pro - 'page.php?id' Remote File Inclusion	WebApps	PHP
2007-07-08	FlashGameScript 1.7 - 'user' SQL Injection	WebApps	PHP
2007-07-07	GameSiteScript 3.1 - profile id SQL Injection	WebApps	PHP
2007-07-01	ArcadeBuilder Game Portal Manager 1.7 - SQL Injection	WebApps	PHP
2007-05-04	XOOPS Flashgames Module 1.0.1 - SQL Injection	WebApps	PHP

Date	Title	Type	Platform
2007-04-28	PostNuke pnFlashGames Module 1.5 - SQL Injection	WebApps	PHP
2007-03-08	GaziYapBoz Game Portal - 'kategori.asp' SQL Injection	WebApps	ASP
2007-02-22	FlashGameScript 1.5.4 - 'index.php?func' Remote File Inclusion	WebApps	PHP
2006-12-11	mxBB Module Activity Games 0.92 - Remote File Inclusion	WebApps	PHP
2006-11-23	PEGames - 'index.php' Remote File Inclusion	WebApps	PHP
2006-08-20	NES Game and NES System c108122 - Remote File Inclusion	WebApps	PHP
2006-06-09	0verkill 0.16 - ASCII-ART Game Remote Integer Overflow Crash (PoC)	DoS	Linux
2005-02-02	Painkiller 1.35 - in-game cd-key alpha-numeric Buffer Overflow (PoC)	DoS	Windows
2004-11-29	Orbz Game 2.10 - Remote Buffer Overflow (PoC)	DoS	Windows
2004-10-10	Monolith Games - Local Buffer Overflow (PoC)	DoS	Windows
2004-01-02	XSOK 1.02 - '-xsokdir' Local Buffer Overflow Game	Local	Linux
2003-08-01	xtokkaetama 1.0b (RedHat 9.0) - Local Game	Local	Linux
2003-07-31	XGalaga 2.0.34 (RedHat 9.0) - Local Game	Local	Linux

4. EXPLOIT OF STEAM ENGINE AS AN EXAMPLE

A zero-day vulnerability of steam engine for online gamers is a much known example. An elevation-of-privilege bug allows attackers to run any program on a target machine with high privileges[15]. Vasily Kravets, a self-described Windows Privilege Escalator, first publicized the exploit, which could affect millions of Windows users running Steam, earlier this week. In a blog post[16], he described in detail how manipulating some of the registry keys associated with the Steam Client Service could be used to mount an escalation of privilege attack on a PC, allowing one to run "any program with the highest possible rights on any Windows computer with Steam installed," according to Kravets. Kravets' disclosure was made only 45 days after his initial report. This is half of the standard 90-day limit that is usually adhered to in the industry. Kravets claimed this is due to Valve's lack of acknowledgement for his work.

The researcher first raised the issue with HackerOne, a bug bounty platform that Valve approves of. However, his report was marked as "not applicable" by HackerOne staff because they claimed it was an example of "attacks that require the ability to drop files in arbitrary locations on the user's filesystem". After some finagling with HackerOne, Kravets was finally able to get approval to send the report to Valve's security team but his report was rejected again for the same reason, with the added qualification that the attack "requires physical access to the user's device".

Figure 2. *Twitter Message for the Steam Engine Bug*

[15] M.Jarir Kanji, 2019, Neowin, https://www.neowin.net/news/valve-fixes-zero-day-exploit-for-steam-in-latest-beta/
[16] For details: https://amonitoring.ru/article/steamclient-0day/

A second researcher, Matt Nelson, has also published a proof of concept for the same exploit. Nelson also complained about a big company refusing to acknowledge his reports back in June. He then fingered Valve as the company back in July, remarking "Good luck reporting anything that doesn't fit their crappy bounty scope." Following Kravets' disclosure, he finally published his proof of concept this week as well. A Steam spokesperson pointed to the release notes for the latest Steam beta[17], which has fixed the issue.

5. SECURITY ISSUES AND COUNTERMEASURES

Different terms are being used for the security issues for games in the literature like cheat, exploit, hack, attack. Although they refer to different types of issues, sometimes they overlap; there is not a clear distinction between these terms. Simply we can divide them into two categories; cheat and attack. Cheat refers to gaining unjust advantage in game and attack is used for the attempts to give harm to another player or to whole system. In this study, we will focus on possible security issues, which are considered as cheats in our strategy game and point out possible countermeasures for them.

5.1. Abusing Game Procedures/Innocent Hack

Players can find the misuse of existing gameplay features. Sometimes these cases may be considered as creative use of existing game rules by the developers but if it is ruining the standard functionality of that feature and provides an unfair advantage over fair players it should not be tolerated. In this type of strategy games there can be a market functionality integrated into the game, where the players can trade off resources among each other. It adds a nice interactivity feature into the game but the players may find a way to exploit this feature in an unintended way. For example, a player can create a dummy account and transfer resources from that account to his main account for small prices. There is not a single solution for this type of problems. It can be solved by designing robust game procedures for the modules in the game. You need to be careful about unintended use-case scenarios for each feature you implement into the game.

5.2. Multi Accounts

In a cooperative online game, players not only fight against each other individually but also can cooperate with each other and fight against groups of other players as an alliance or against the computer. Thus, a player can help another player. In our game, a player may try get resource and army support by opening multiple accounts. Limiting to a single account per device is the clearest solution but technically, it is not a simple task. For desktops, the user may try to fool the game as if it is a new device in different ways such as using a virtual operating system. There are also similar programs for mobile devices, which allow multiple versions of a single program to exist in a single device. Another solution is again in the game design mechanics as we mention in "Abusing game procedures" section. We need to design game mechanics in a way that using a multiple account will not provide unfair advantage in the game. The problem about this solution is not about technical difficulty, but it requires an imaginative mindset. You need to preserve enjoyment while crippling down some features.

5.3. Client Modification

Game client is the application that the player interacts with on his own device. It provides a game interface to the player and sends the commands to server that it took from the player. Naturally, the game interface does not allow the user to send commands that he is not allowed to do in the game. For example, if the player does not have enough resources to build a specific building, the player should not be able to send a command to build that building. Since this client application runs on the player's device, he has the ability to manipulate client in different ways. He can modify the client infrastructure or modify the values in the ram. By this way, he can represent himself as if he is allowed to do some actions in the game although he does not met the requirements for those actions and the game client allows him to send those

[17] For details: https://steamcommunity.com/groups/SteamClientBeta/announcements/detail/1602638506845644644

requests to server. The most robust solution for this problem is to double check the requirements for actions in the server when you get a request from client.

5.4. Network Packet Modification

Another option for sending unauthorized requests to server is to modify the request packet content. For example, we can think of a situation where you have a certain amount of resource and it is enough for a cheap building "A" but not enough to build an expensive building "B". The game interface allows you to build "A" but does not allow you to build "B". You give order to build "A" in the game then by using packet sniffing tools you can interrupt the network request and modify the content of the package and change the building code for "A" to building code for "B". Then the server gets a request to build "B". The game requests are sent over TCP and it has its own protocol. Therefore, it is not as easy as changing the content of an HTTP packet, the data is in binary format but it is still possible.

The first solution that comes to mind is to encrypt the network requests. Another thing that comes to mind would be to attach the hash size of the packet but since the changes in our scenario are, micro-changes it is possible for the total hash size remain same after the modification. Therefore, it is not reliable. A third option is again making server-side controls for the incoming requests. Encrypting packages may resolve packet modification issue but it cannot provide a solution for client modification. Making server-side controls can provide a more robust solution for both the client modification and packet modification problems.

5.5. Information Gathering

Players can also use these client modification tools or packet sniffing tools to gather information about other player's private game information, like their soldier amounts, resources or the number of soldiers in an attacking army. This may not seem as a direct cheat at first sight but in this game, information is a key component to get advantage over your opponents. Players determine their strategy over this information. Therefore, for sake of fairness the game should protect private game information from other players. Again, here the main idea is not to rely on the client to hide confidential information about other players but to clear that information from each package that is sent to client. As an example, a moving army on the map is relevant information for both the attacker and the defender. That information is sent to both of them, but the number of the soldiers in the army must be invisible to defender. According to game design choice, it must remain unknown to the defender. Sending the same army package but hiding the soldier number information on the defender's game interface is not a reliable method because of the reasons we mentioned that information still exist in ram. It must be cleared out from the package when sending to defender, but that info should be included in the packet that is sent to attacker.

5.6. Denying Service to Other Players

This can be called as an attack but when someone use this against his opponent and try to gain an advantage over him it can also be categorized under the cheat category. The cheater can try this method in different ways, like flooding his opponent's network connection but more simply even existing game features can be used to exploit this attack. As an example, showing a popup when a player gets a message from another player may seem as an innocent feature but constantly sending messages to a player may prevent his gameplay. A player may use this against his opponent when his opponent needs to make some time critical actions in game. Someone can also use a bot for abusing this exploit in a more aggressive way for any target player. This is another usage of "abusing game procedures" section and solution is the same, you need to consider unwanted usages of each feature you put into the game.

5.7. Translating Source Code and Disassembling

It is quite easy to translate your Flash applications into source code with a third party tool, so putting static username and password into your application is not the right approach. When we cannot convert the SWF (Shockwave Flash) file of the Flash application to source code for analysis, we can convert the

bytecode to readable and perform our analysis. We can also use this path when we translate it into source code but cannot compile it again. In both of these cases, both RABCDasm[18]. For example, it is possible to translate a game into a source code, and it is in the SWF file of the management panel in the analysis, but it is not possible to reach this management panel by the application builders because the function related to this panel in ActionScript is not visible during the loading of the game. However, if you can disassembly this file (patching), you can make this panel visible during installation and access the admin panel [6].

5.8. Misleading Scanning Tools

Actionscript programming can lead to vulnerabilities such as XSS, XSRF if not used safely in a language. We can often encounter these weaknesses in parameters that use URLs that accept, process, and use HTML. For example, if the clickTAG used in Flash ads is not used safely, it can lead to XSS vulnerability. We need to analyze the Actionscript code in detail to identify these and similar issues, but we can benefit from this time consuming task and the programs that do it because most people may require expertise. Developed for this purpose, HP SWFScan is a useful tool that analyzes and reverses the target SWF file and can detect and report more than 60 security vulnerabilities. (SWFScan supports ActionScript versions 2 and 3. You can also use SWFIntruder for version 2.) Sometimes these tools may not be able to detect security weaknesses, so I can just say that it is useful to analyze ActionScript against it [6].

6. ONLINE GAME SYSTEM ARCHITECTURE

In computer games, there is a space or architectural structure in the development process from two-dimensional text-based games to three-dimensional photorealistic displays. Along with the new concepts that have been added to the game design process, architecture has begun to create its own place. In the field of computer games terminology, which is mostly called level design, architects have begun to realize the architectural dimension at the same time as they designed game sections and levels. In addition to the creation of episode maps, architectural design plays a major role in the game design process [7], with spatial playability, placement of spatial objects, intentions of maps and spaces, and additional tasks associated with them. The concept of the game player's need for architecture and perception in games is the result of learned and gained experiences from the real life role of the architect [8]. For the past decade, we've witnessed exploit after exploit targeting our favorite Web browsers, email clients, office productivity software, and operating systems [9]. Most research surrounding online games has been specific to cheating [10] and theft of virtual goods via external malware and overly permissive game scripts [11].

Architecture has become an important part of computer games until the existence of interactive three-dimensional spaces from text-based narratives with no visuals. Game design research has different approaches to the use of architecture and understanding of spatiality. It is an important step to understand the role of architects in games in order to examine these approaches, which can be linked to architectural concepts and environmental psychology. In this context, while architectural approaches are examined in game design studies, similarities between the theories presented for real architecture and mixed theories are presented.

6.1. Technical Design

Technical design of a game is of crucial importance. There should be specifications of a game server, game client, persistence layer and web server.

6.1.1. Game server

[18] For detailed information see https://github.com/CyberShadow/RABCDAsm

Game server is a standalone java application running on a server machine. This game server listens TCP connections from a specified port. Clients connect to game server through this port using TCP protocol. At first, there is an in-game authentication process. After authenticating successfully clients can send game related requests to game server and the game server responds to clients through the same channel. All of this communication is performed using TCP protocol.

6.1.2. Game client

A game client is an application on user side that allows the player to view the game and send his or her requests to game server. Unity, a popular game engine, is used to implement game client. In Unity, it is possible to create builds for different platforms, like desktop and mobile. When the client application is launched, it sends authentication information to server with TCP protocol. Client uses the specified domain name and server port to connect to game server. The client can communicate with the game server as long as it has an active TCP session with the game server. When the player quits client application, the connection ends.

6.1.3. Persistence layer

Game data persisted on a MySQL database. Most of the data also lies on RAM and used from there but for unexpected crashes or intentional shutdowns, it is needed to persist data on the disk. In addition, for design related issues, some of the data are not kept in RAM but only in MySQL.

6.1.4. Embedded web server

There is a java http web server called Jetty. Jetty can be used within software as a java library. In this game jetty is used as an internal web server. It is not another standalone server running on its own; it runs within game server java application and listens http requests on a specified port. It is used to reach game from the website for actions like creating a new player.

6.2. Game Design

Our case study is about an online strategy game. In this game, there is a single map and the players have a town at some location on this map.

6.2.1. Game mechanics

There are different types of resources around. The players can increase their borders and build different kinds of buildings within their borders. They can assign their population for different tasks as gathering resource, constructing a building, farming or training as a soldier. A player can attack another player's town and kill soldiers in that town, raid its resources and destruct buildings.

Figure 3. *Concept view of game map.*

Players can also form an alliance. The game is won as an alliance. When the total borders of an alliance have reached a specified amount that alliance wins the game at that game world and that is called a season.

6.2.2. MMO concept

There are different types of online games. MMO stands for massively multiplayer online games and in this type of games, large number of players, from hundreds to thousands, can play together on the same game server. MMOs usually provide a persistent game world. Even when the player is not connected to game server, the game world continues to live. Our case study fits into MMO type in this manner, all of the players are playing on the same map and when the players are not connected to server, their town stays on the game world. In addition, their actions take some time and after giving commands, they do not need to stay connected to server. For example, a player can send his army at a ten-hour distance and then he or she can safely disconnect from server. The army continues its way during the next ten hours in game server. To compare with other online games, you can think of a card game like poker. The game is played among a restricted amount of people, at most ten, in a table. In addition, when the player is not connected to game server, he or she has no relation with the game world.

7. CONCLUSION

Cheating in MMO has a significant impact on the gaming community, but the impact of attacks that fully compromise players' computers is much greater. Online game developers should identify such problem areas with security-conscious risk assessments and allocate their resources accordingly [12].

It is difficult to quantify for the entire games industry. The worldwide damage caused by cybercrime is estimated at $ 600 billion - not limited to the gaming industry. There will also be enormous damage in the games industry. We know that individual game publishers suffer considerable damage - providers of online games maintain entire departments to ward off hacks, and external software is also purchased. The costs are already very high to avoid hacks. Not to mention the costs if there were actually hacks - we're talking about IT-related costs, costs for community support and migrating players, especially when games are considered "cheat-infested". When a violent denial-of-service attack is averted, the costs for the company concerned can quickly reach six figures. Games that are considered "cheat-infested" quickly go down the drain. In this respect, however, we assume that these customers are not lost to the gaming industry as a whole, but only to the respective game - that is, they will then play other games. And companies that specialize in this also benefit from the costs of preventing hacks. As a new "damage item" there is also the fact that if personal data are affected, heavy fines can also be imposed according to the

General Data Protection Regulation. The English data protection authority has already announced two fines this year, each at over 100 million [13].

Online games have different security concerns than single player games, and massively multiplayer online games could be even more sensitive to these threats. We tried to point out the vulnerable points of MMO games and provide sufficient solutions to these problems by using an MMO strategy game as our case study.

If necessary security measures are taken, digital games have become an indispensable entertainment tool for children today. There is great responsibility for protection against risks, especially for parents. Families should demonstrate a friendly approach to their children with a strong communication channel, rather than an inquisitive and judgmental attitude. For example, what games did you watch today? How do you like to play? What is the most dangerous game in your mind? Although there can be no full assurance against threats in the online gaming environment but the following basic security precautions are recommended in general to ensure the safety of computers used for online gaming:

- User awareness and proper education on security concerns is of crucial importance
- Antivirus and anti-spyware programs should be used.
- Being careful when opening additional files that come in e-mail and instant messages.
- When downloading files and software from the Internet, secure sites should be preferred.
- The security settings of the internet browser used must be set.
- Firewall must be used.
- Personal and financial information must be secured and backed up.
- Strong passwords should be used.
- The software must be updated.

CONFLICT OF INTEREST

No conflict of interest was declared by the authors

REFERENCES

[1] Volk, P. (2016, September 13) *League of Legends Now Boasts over 100 Million Monthly Active Players Worldwide*. Retrieved from https://www.riftherald.com.

[2] PC online game market value worldwide from 2011 to 2019 (in billion U.S. dollars) (2016) Retrieved from https://www.statista.com/statistics/292516/pc-online-game-market-value-worldwide/.

[3] Yan, J., "Security design in online games," *19th Annual Computer Security Applications Conference, 2003. Proceedings.*, pp. 286-295 (2003).

[4] Jiankun, H. and Zambetta, F. "Security issues in massive online games", *Security and Communication Networks*, pp. 83-92 (2008).

[5] Woo, J. and Kim, H. K., "*Survey and research direction on online game security*". In Proceedings of the Workshop at SIGGRAPH Asia (WASA '12). ACM, New York, NY, USA, 19-25 (2012).

[6] Sarıca, M. (2011) "*Hackerlerin Gözünden Flash uygulamaları*" https://www.mertsarica.com/hackerlarin-gozunden-flash-uygulamalari/ .

[7] Ryan, M-L., "Immersion vs. Interactivity: Virtual Reality and Literary Theory" *SubStance*, 28(89), 110-137 (1999).

[8] Çatak, G. "*Bilgisayar Oyunlarında Mimarinin Kullanımı"*, Yüksek Lisans Tezi, Y.T.Ü. İstanbul. (2003).

[9] Hoglund, G. and McGraw, G. *"Exploiting Software: How to Break Code*", Addison-Wesley Professional, (2004).

[10] Hoglund, G. and McGraw, G., *"Exploiting Online Games"*, Addison-Wesley Professional, (2008).

[11] Muttick, I., "*Securing Virtual Worlds against Real Attacks*", McAfee, (2008).

[12] Bono, S., Caselden, D., Lansau G., Miller, C., "*Reducing the Attack Surface in Massively Multiplayer Online Role-Playing Games*", Published by the IEEE Computer society, (2009).

[13] Anderie, L., "*Game hacking: from pirated copies to cybercrime game hack Quick Guide Game*", Hacking, Blockchain and Monetization pp 1-21 (2020).