International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 1 (2017) pp. 91-94

A Study on the Profit Analysis of Online Game Hacking Programmes and Blocking Methods

[1] Chang-Seon Lee, [2]Dong-Hwan Jo and [3]Jinho Yoo

[1] *First AuthorDept. Sangmyung University, Korea.*

[2] *Second AuthorDept. Sangmyung University, Korea.*

[3] *Corresponding Author Dept. Sangmyung University, Korea*

[1] (ORCID- 0000-0003-0314-4934)

Abstract
South Korean government is applying legal limitations on selling online game hacking programmes. Although there are legal regulations, there are some cases that online game hacking programmes are still on sale. But, with the regulations, the direct sales disappeared and the money-making profit models, which used to be selling hacking programmes directly, have been transformed to indirect profit- targeting models. In this paper, we will find out the profits of an online game hacking programme when released, and then we will talk about the way to delay the continuously being updated and developed hacking programmes, not about just a simple patch.

Keywords: Revenue model of the hacking program, cost per click

INTRODUCTION

Game companies are making a lot of efforts to eliminate the online hacking programmes. Firstly, one of their main objectives is to prevent clients from being hacked by applying game client security solutions. The second main objective is to crack down on the online game hacking programme-selling companies through the legal action or by law. Lastly, the third one is to counteract the illegal abusers by utilising the gameplay information. Still, many online hacking programmes have been developed and made persistingly, and those programmes are already prevalent and are used by many ill-intentioned users. In this paper, we are going to figure out why the online game hacking programmes are being developed, and we will measure the advertising revenue which is the main source of income to programmers who developed them. Then, we will suggest an alternative on how we can block those online game hacking programmes.

This paper is made up of three chapters. Chapter 2 explains the revenue model of the online game hacking programmes. In chapter 3, we will demonstrate the prediction of the earnings from the revenue model from chapter 2. Chapter 4 is about a counteractive measure to delay the development of the online game hacking programmes.

THE REVENUE MODELS OF THE ONLINE GAME HACKING PROGRAMMES

In this chapter, we separated the online game hacking programme revenue models by a generation which can be differed by the method of making profits. The first generation is the programme-advertising period. In this first period, online game hacking programmers just focused on showing off their own programming skills or focused on the convenience in playing games with the macro code programmes, not really on the money. So, the hacking programmers just developed and distributed their own online game hacking programmes. After that, popular hacking programmes got many users, and the users had increased more and more. Therefore, those popular programmes transformed into a method to make money. This is the second generation. In this generation, those hacking programmes start making money, and there are two ways for this. One is the advertising revenue[1], another one is the sales profit of the hacking programmes[2]. However, selling these hacking programmes is now illegal by law[3]. So, there is the only one way to make money with the hacking programme, which is the money from the advertising revenue.

The 1st Generation, Advertising Programmes

As a hacking field, hacking programmers bypass the client security programmes, and they study and develop the easily usable hacking programmes. Then they post their programmes on their homepage or internet community web page. These free macro programmes or the hacking methods in the 1st generation were made to share and develop each hacker's personal programming researches and studies.

The 2nd Generation, Making Profits

Hardware Type Product Sales

As of the 2nd generation, the hardware type online game hacking programmes started selling. For about 5 to 6 months, these products gained sensational popularity and so the sales had shown rapid growth. That is because of the appearance of the new and lawful transactional websites. With these websites, online game users could sell the acquired game items. In other words, it is possible to make money with the game items. By using the hardware type macro products, many users could acquire lots of goods in games and made them into money[4]. The 'Auto Mouse' manufacturing company was charged with CRIMINAL ACT, art. 314, sec. 1 (interference with business), GAME INDUSTRY PROMOTION ACT, article 32 (Prohibition of Distribution of Illegal Game Products, etc.), ACT ON PROMOTION OF INFORMATION AND COMMUNICATIONS NETWORK UTILIZATION AND INFORMATION PROTECTION, ETC., Article 48 (Prohibition on

Intrusive Acts, etc. on Information and Communications Network), CRIMINAL ACT, Article 347-2 (Fraud by Use of Computer, etc.), ELECTRONIC FINANCIAL TRANSACTIONS ACT(Transferring or taking over a means of access). Therefore, companies that manufactured these kinds of products are no longer selling them[6].

Advertising Revenue (Adware)

Another way to make profits is the advertising revenue. Advertising revenue is not from the money by selling the hardware, but from the money by adding advertisements when the online hacking programmes were on sale or distributed[1]. There are some methods that AdWords applying (CPM, CPC, CPA). CPM means Cost Per 1000 impressions, CPC means Cost per Click, and CPA means Cost Per Action[7][8]. If the high traffic is predictable or if it is possible to introduce the hacking programme on the online game forums, the hacking programme uses CPM method to make profits, because there will be many visitors and they will surely watch the advertisement[9].

PREDICTION ON THE HACKING PROGRAMMES' ADVERTISEMENT REVENUE

Here are the conditions we followed for choosing a game and hacking programmes for this paper.

1. A game which has lots of online players
2. Hacking programmes that continuously execute updates

Based on these two conditions, we picked and utilised game 'A' for this paper. This game provides its service globally. The online game hacking programmes that used to calculate the advertisement revenue were from a specific website[12]. We calculated the advertisement revenue on the assumption that a user runs a hacking programme at least 1 time. The estimation of profits is based on how many times the hacking programmes were downloaded because a user should run the hacking programme at least over 10 times, or just directly click or watch an advertisement right away to run it. Also, we estimated the minimum amount of money because only the game administrator knows the maximum amount of money and the others do not.

On the matter of the amount of money from CPM, we applied for CPM money that 'adf.ly' announced. The 'adf.ly' pays for the advertisement fees via three ways.

Table 1. Ways of How 'adf.ly' Advertises

Advertising Method	Explanation
Interstitial	Skip to the main contents after 5 seconds
Banner	Show banners all the time
Pop/Ads	Cast pop ups

The online game hacking programmes, which were chosen for this paper, have been kept gaining the advertising revenue by applying the 'Interstitial' way. That is because there is a huge difference between the advertising methods, whether it is 'Interstitial' or it is 'Banner' or 'Pop/Ads.' If we calculate the advertisement revenue, the revenue made by the 'Interstitial' method is over 5 times more than the revenue made by the 'Banner' or 'Pop/Ads' method[10].

Also, every advertisement uses weighting method in accordance with the number of people trying to access and with the region or country where the access was initially made. For example, if someone makes the connection from the United States, the advertisement fee will be 1 dollar. On the other hand, the connection from South Korea only makes 50 cents. So, we calculated the profits by measuring the average amount of money. There are some reasons for this. Firstly, the difference of the advertisement fees depends on the region where the connection was made. Secondly, most games are servicing throughout the world. Lastly, the hacking programmes are also developed and distributed to all around the world without any regional or legal restriction. The advertisement fees on the 'adf.ly' change every day, so we picked an average amount of the advertisement fees among them. The formula for calculating average advertising revenue is:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$$

(※ x = Fee by Country, n = Count of Country)

It is $1.27(AR: Average of Rate) which was priced on September 7th, 2016. To make money by using 'adf.ly', you need to make an adf.ly account. After logging in, you should make a special adf.ly link by entering a website URL in a white box at the adf.ly's website. You should copy and paste the specially generated adf.ly link to a website. Then every time someone clicks on this link, your adf.ly account will accumulate lots of money after all[11].

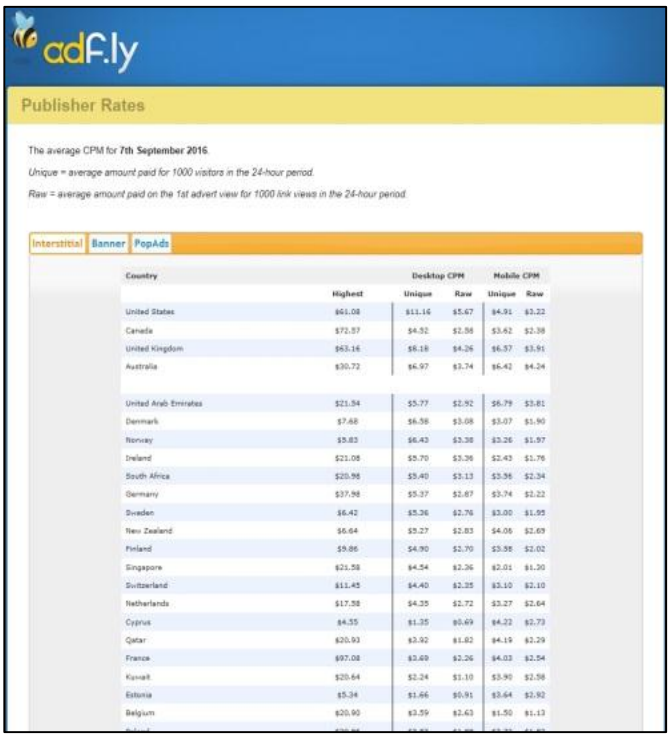

Figure 1. Average Amount of the Advertisement Fees on September 7th, 2016[10]

In order to estimate the profits collected when the online game hacking programme is released to the public, we classified the types of profit into three different types. 1) Revenue from publishing only on the website, 2) Revenue from creating and distributing the online game hacking programme, 3) Minimum revenue per day after publishing the online game hacking programme. The calculation formula for each profit type is as follows and the following three formulas were used to calculate the minimum projected revenue as shown in Table 2.

1) After checking the total number of updates(SUC: Site of Update Count) of online game hacking programmes currently shared on a particular website, we calculated the revenue earned(SI: Site Income) when sharing the online game hacking programme on this website.

$$SI = SUC * AR$$

2) After checking the number of updates of the selected online game hacking programme, we calculated the revenue earned over a month(MI: Month of Income) by calculating AR and the number of updates(MUC: Month of Update Count) and the average number of downloads in a month(ADC: Average of Download Count).

$$MI = MUC * ADC * AR$$

3) We divided the outcome of calculation(MI = MUC * ADC * AR) by 31 days(D). This refers to the daily profit for a single update.

$$DI = MI / D$$

Table 2. CPM Revenue by Programme (Unit: $)
P: Online Game Hacking Program, I: Items, UC: Update Count, DC: Download Count

P	I	UC	DC	CPM
SI		20	5,386	$6840.22
elpresidente024	MI	4	482	$2448.56
	DI	-	-	$78.98
ynt project esp aim v.6.3	MI	1	2,934	$3726.18
	DI	-	-	$120.19
FapCFro2.0	MI	1	1,250	$1587.5
	DI	-	-	$51.20

As shown in Table 2, even if you just upload and publish an online game hacking programme on a website, you can still earn $6,840. Furthermore, if you are a developer of a hacking programme, you can run your own homepage and make an additional return on your website(SI). That is, the profits earned by the hacking programme developer are the sum of SI and MI. Hacking programme developers keep producing and distributing online game hacking programmes because the revenue they earn is normally greater than the revenues earned by general office workers. In order to block such continuously updated online game hacking programmes, game companies often analyse hacking programmes and simply patch them. Game companies typically use signatures to detect online game hacking programmes for simple patches[13]. If a game company applies such a simple patch, it is easy for a hacking programme developer to bypass it because a game company cannot detect when the signature is modified. Therefore, the hacker will be able to earn a similar level of profit again in 2 to 3 hours.

SUGGESTION OF COUNTERACTIVE MEASURES TO DELAY HACKING PROGRAMME DEVELOPMENT

It is necessary to combine three different ways to continuously cut the online game hacking programmes.

1) Securing the memory address that the online game hacking programme uses

When a hacker distributes an online game hacking programme, the hacker usually uses a bug or changes the value of a variable that used in the game and he distributes it. The code in Figure 2 is actually the code that an online game hacking programme uses. This code is responsible for the speed at which you change weapons in the game. By changing the numerical value of the weapon change rate in this code, the user of this hacking program can attack and change weapons at a faster rate than other users.

```
/* First define No Weapon Change Delay's Offset & Weapon Pointer */

#define WeaponPointer 0x000000 (You have to find the Pointer)
#define WeaponChangeDelay 0x0000 (You have to find the Offset)

/*If you want to make your hack Auto On put this before it */

bool nochange = true; // Dont make it the same name with the defined one.

/* Then add these */

DWORD CShell     = (DWORD)GetModuleHandleA("CShell.dll"); // When CShell word will be used , it will understand it is
 CShell.dll
DWORD pWeaponMgr = *(DWORD*)(CShell + WeaponPointer); // Wehn pWeaponMgr word will be used , it will understand that
it is CShell with Weapon Pointer

/* Then add this */

if(nochange)
{
if (pWeaponMgr) // You already know what this is
{
for(int i=0; i<445; i++) // You already know what this is
{
if(*(DWORD*)((*(DWORD*)(CShell+pWeaponMgr))+(4*i)) != NULL)
*(float*)((*(DWORD*)((*(DWORD*)(CShell+pWeaponMgr))+(4*i))) + WeaponChangeDelay) = 100; // You already know what this
 is
}
}
}
```

Figure 2. Hacking Code to Weapon Change Delay[17]

When the situation occurs like above, the game company can detect that the client was hacked if it has the logic to check the attribute value corresponding to the weapon change.

2) Registering patterns about specific memory address in an online game hacking programme

Online game hacking programs are usually added to the game's memory in the form of DLL files. After this DLL file is added to memory, the online hacking program looks for a location to hack through a specific pattern.

```
void WriteText (int x, int y, DWORD color, char *text)
{
      if(!GetModuleHandle(eCShell)) return;
      if(!GetModuleHandle(eClient)) return;

      DWORD MoveECX = *(DWORD*)(Pointer);//<-Pointer

      __asm
      {
            push 01
            push FF000000                 //<-Shadow
            push 00
            push 00
            push x                              //<-X
            push y                              //<-Y
            push color                    //<-Color
            add ecx,-3F
            push 3E
            PUSH text                           //<-Write Text
            MOV ECX, MoveECX              //<-Pointer
            CALL DrawTextEngine1    //<-Call
      }
}

/*
DrawTextEngine: 8B 0D ?? ?? ?? ?? E8 ?? ?? ?? ?? 8B 96 ?? ?? ?? ?? ?? E8 ?? ?? ?? ?? 83 C4 ?? 85 ?? 74 ?? 8B 0D ?? ??
 ?? ?? 83 C1 ?? 89 4C 24 0C
```

Figure 3. Search Pattern to Draw Text Engine Function[18]

As shown in Figure 3, the hacking program uses the special character (8B 0D ?? ?? ?? ??) to find the base address of the function and DLL that you want to hack in online game memory. After searching Base Address, it inserts attribute and code through Offset to operate the online game hacking program. Therefore, if a memory pattern is registered and a detection operation is performed by using the unique strings or by utilising unique areas that are not changed after the DLL is registered, the update period of the online game hacking programme can be delayed.

3) Detection Methods Using Game Logs

A detection method that utilises game logs has financial issues that require analysis of large amounts of game logs [15]. If financial issues are not a problem or if you have a data analyst who can handle and analyse vast amounts of data, a detection model using game logs is more effective than the two detection models mentioned above. Since the two detection models mentioned above are detection methods according to the environment of the client, false detection may occur when the client environment is changed. However, the detection technique using the game log is a technique that analyses all the actions performed by the characters or the user's actions in the game, and then discovers useful correlations between collected data and extracts meaningful information for decision making[14]. This technique utilises the logs generated by the game regardless of the environment of the client so that it is possible to detect a specific game in detail level although this technique cannot be applicable to all games[16].

CONCLUSION AND FURTHER STUDY

In the near future, there will be limits on this method to delay the online game hacking programmes' update, because even if there are client security systems, hackers might find out bypassing ways in about less than a week by studying and analysing those client security systems. Therefore our main objective is to make a new detecting model by using and analysing the game logs on a game server. The detection model will be made by the data science method.

REFERENCES

[1] AdSense, https://en.wikipedia.org/wiki/AdSense
[2] http://www.coupang.com/vp/products/1105445?src=1032001&spec=10305201&addtag=400&ctag=1105445&lptag=3005816384
[3] http://www.law.go.kr/lsInfoP.do?docType=JO&lsiSeq=111964
[4] http://www.itemmania.com/
[5] https://ko.wikipedia.org/wiki/Criminal Laws 314 of Korea
[6] Ahn I.B, " '09, auto no game of the world 'expected'", http://www.khgames.co.kr/news/articleView.html?idxno=1225
[7] Kim, D.Y, "A Study on the Efficiency of Internet Keyword Advertisement According to CPM and CPC Methods by Analyzing Transactional Data", The Journal of Society for e-Business Studies / v.16, no.4, 2011 Year, pp.139-154
[8] http://eara.co.kr/457
[9] https://monetizepros.com/display-advertising/how-to-sell-display-ads-cpm-vs-cpc-vs-cpa/
[10] http://adf.ly/rates.php
[11] https://support.adf.ly/hc/en-us/sections/201733023-Service
[12] http://www.hotshotgamers.net/search/label/Crossfire%20Cheats?max-results=20
[13] LEE C.S, "A Study on Improved Detection Signature System in Hacking Response of One-Line Games", The Journal of Society for e-Business Studies / v.21, no.1, 2016 Year, pp.105-118
[14] Byung Il Kwak, Huy Kang Kim, "A survey and categorization of anomaly detection in online games" Journal of The Korea Institute of Information Security & Cryptology 25.5 (2015)
[15] Jin-Hee Song, Jung-Sook Kim. (2014.3). Analysis of the best practices big data services. The Korea Contents Association Review, 12(1), 32-37
[16] Jung kyu Park, Mee Lan Han, Huy Kang Kim, "A Study of Cheater Detection in FPS Game by Using User Log Analysis, Journal of Korea Game Society", Vol. 15, No. 3, pp. 177-188, 2015
[17] http://www.mpgh.net/forum/showthread.php?t=420300
[18] http://www.mpgh.net/forum/showthread.php?t=1178502
[19] Dongnam Seo, Jiyoung Woo, Kyung-moon Woo, Chong-Kwon Kim, Huy Kang Kim. (2012.6). Detecting gold-farmers group in MMORPG by analyzing connection pattern. Journal of the Korea Institute of Information Security & Cryptology, 22(3), 585-600