Virtual World Security Inspection

Nicholas C. Patterson and Michael Hobbs
Deakin University, Geelong, Australia
Email: ncp@deakin.edu.au
Email: {ncp, mick}@deakin.edu.au

***Abstract*— Virtual property theft is a serious problem that exists in virtual worlds. Legitimate users of these worlds invest considerable amounts of time, effort and real-world money into obtaining virtual property, but unfortunately, are becoming victims of theft in high numbers. It is reported that there are over 1 billion registered users of virtual worlds containing virtual property items worth an estimated US$50 billion dollars. The problem of virtual property theft is complex, involving many legal, social and technological issues. The software used to access virtual worlds is of great importance as they form the primary interface to these worlds and as such the primary interface to conduct virtual property theft. The security vulnerabilities of virtual world applications have not, to date, been examined. This study aims to use the process of software inspection to discover security vulnerabilities that may exist within virtual world software – vulnerabilities that enable virtual property theft to occur. Analyzing three well know virtual world applications *World of Warcraft, Guild Wars* and *Entropia Universe*, this research utilized security analysis tools and scenario testing with focus on authentication, trading, intruder detection and virtual property recovery. It was discovered that all three examples were susceptible to keylogging, mail and direct trade methods were the most likely method for transferring stolen items, intrusion detection is of critical concern to all VWEs tested, stolen items were unable to be recovered in all cases and lastly occurrences of theft were undetectable in all cases. The results gained in this study present the key problem areas which need to be addressed to improve security and reduce the occurrence of virtual property theft.**

***Index Terms*— virtual worlds, virtual property theft, real money trading, keylogging, vulnerability, software inspection**

I. Introduction

Virtual World Environments (VWEs) are computing simulation environments that allow users to socialize, play, compete and even work in an immersive on-line virtual world. VWEs have their heritage in the text-based multi-user computer games (MUDs) of the 1980s [1], while modern versions are commonly visually rich 3D, extensive environments that range from fantasy and space based realms, to life-like real world environments. The number of people actively participating in these environments has grown dramatically over recent years and current reports indicate the number of registered virtual world users exceed 1 billion world-wide [1]. It is common for users to pay a subscription fee to access these worlds and then, over a period of time, through completing various tasks are able to collect items that are owned by the VWE character, representing the player. It is also possible in many VWEs for users to spend real-world money to purchase items as well. The investment made by users in terms of their time, effort and real-world money, places a value on these virtual property items, which can then be traded with or sold to other users for either virtual-world or real-world currency. Virtual worlds expert Marcus Eikenberry estimates the market value of virtual property as high as US$50 billion dollars [2].

The ability to convert virtual property into real-world money has enabled the rise of a serious problem faced by many in VWEs, that of virtual property theft (VPT) [3]. The problem of VPT is complex as it envelopes many diverse areas, such as *legal issues* – lack of laws to support prosecution, especially in cases that span international borders; *social issues* – such as identity theft and harassment; and *technological issues* – the appropriate use of security methods and tools (within the software used to access VWEs) to protect resources.

Users access VWEs through software running on their computers. Commonly, this is in the form of a client-application that connects to a remote server application holding the data associated with the VWE. Although security in client-server based and other forms of distributed system applications has been researched extensively, to date, there has been no research on the security aspects of VWE software. It is this software (technology) that is used to access VWEs, and thus, it is important to understand the security vulnerabilities of such software. Having information on potential security vulnerabilities will help identify approaches needed to address the problem of VPT. This information can be used to form recommendations to VWE developers on how to improve their software and ultimately providing users with an appropriate level of trust to actively participate in VWEs.

The goal of this paper is to discover what vulnerabilities and threats may exist in software used to access popular VWEs that enable the problem of virtual property theft (VPT). One method to discover these vulnerabilities would be to analyze and test the actual source code for the VWEs. Due to the commercial nature of VWEs, companies will not provide the source code for research by an independent party; therefore vulnerabilities must be identified through an external examination of the executable VWE software. Therefore, to achieve our goal an operational software inspection technique [4] is employed as a method to externally

doi:10.4304/jnw.7.6.895-907

assess the quality of software (in this case VWEs) and to reduce the number of defects (security vulnerabilities).

There are many hundreds of VWEs that exist and since it is not feasible to analyze all, a representative selection of three VWEs was made. These included: *World of Warcraft*, *Guild Wars* and *Entropia Universe* were chosen based on their popularity among player (number of registered users), length of time they have been active, known issues with VPT, and examples of real money trading (RMT) of virtual property [5].

The inspection of the VWE software involves discovering functional and design problems related to VPT that may exist. These relate to issues including: authentication, virtual property trading and recovery. As far as can be discovered from published literature an inspection process of this nature has not been conducted on a collection of software of this nature. The results gained from this study can be used to identify problem areas that exist and the factors that cause them. This study provides a foundation to the development of a solution to the problem of VPT. It is envisaged that a solution to VPT could be incorporated into current and future virtual world software.

This paper is structured as follows. Section II provides background on the area of software inspection approaches and requirements. Section III present and details the three VWE selected for inspection, while Section IV describes inspection process and environment used. The set of inspection categories and objectives are presented in Section V. The inspection results are tabled and analyzed in Section VI. Section VII provides conclusions on this work.

II. Related Work

The general security issues associated with VWE have been examined in the past. However, this work was focused primarily on the game-play and social effects that activities like cheating and fraud have on users of such systems. These issues and their effects on the virtual economy within VWEs were discussed by Cikic et al. [6]. The security vulnerabilities of existing VWE client software has not been a focus of current research.

Livshits and Lam [7] conducted an analysis of nine popular open source applications, they used a method of static analysis to perform their inspection and testing. They found that there were a total of 41 potential security violations in the 9 benchmarks and 29 of those turned out to be security errors and 12 were false positives. However this study did not focus on VWEs specifically but provides support that this technique is applicable to the testing of VWEs.

A study to test the resilience of commercial virus scanning software packages was conducted by Christodorescu and Jha [8]. The aim of this study was to present architecture for detecting malicious patterns in executable files that are resilient to code-obfuscation attacks. To determine if an executable was resilient or not, they performed tests against three commercial virus scanners, the results showed that a combination of nop-insertion and code transportation was all that was required to render a malicious executable undetectable by these virus scanners [8]. This study was useful from a security analysis point of view, but it did not cover VWEs specifically.

Hole et al. [9] conducted a study on Norwegian internet banks from 2003 to 2004. Their aim was to determine if a false sense of security existed within bank customers and whether this contributes to an additional security risk in using online banking. They discovered that the customer authentication methods in many Norwegian Internet banks were weak, which allows simple but powerful attacks possible. This study presented an example of a successful attack that involved a PIN calculator (that was used by many Norwegian banks to generate new PINs for customers – based on certain period of time). In this attack it is possible to generate a timeline to associate a PIN number with a certain time interval and then employ brute force search to access customer accounts. This relates to our study in that one form of security may lead to users having a false sense of security. For example users of virtual world may get a false sense of security from their passwords, whereas their accounts can still be hacked into by the hackers/thieves.

A previous study conducted by Martin et al. [10] developed both static and dynamic techniques to find errors and security flaws to PQL (Program Query Language) queries. They found 206 errors in 6 large real-world open-source Java applications which contain a total of close to 60,000 classes [10]. Although this study did not focus on VWEs specifically more so on Java applications, it still relates and is useful to our study of analyzing applications (VWEs) for security flaws. Theory and techniques can be used from this study to flavor our analysis.

Newsome and song conducted a study [11] which looked at using a technique called taint analysis, which can be used for automatic detection of overwrite attacks, which are the most commonly used type of security exploit. They found that taint analysis regularly detected most types of exploits, producing no false positives in all the different commodity software programs they tested [11]. However for this study they did not discuss or analyze VWEs specifically, they tested several commodity programs (for example software available from a store, where the user does not know anything about the programs internet structure / source code).

From existing literature, it appears that no work has been conducted on the security vulnerabilities of the VWE client software. A study of this nature would assist in determining the potential for security problem to exist within VWE software. In doing so, this study will help identify if such flaws are common with other software testing results and potentially showing that no software package can be completely secure.

III. Virtual World Environment Choices

There exists many hundreds of VWEs available to the public. The choice of VWEs for this study was based upon their popularity in terms of users (relevancy in

terms of are they a good representation of the many available); does the VWE have virtual property available to users, which can be traded or potentially stolen. The three VWEs selected to be inspected for this study included: *World of Warcraft* [12], *Guild Wars* [13] and *Entropia Universe* [14]. They are online VWEs for personal computer (PC) compatible computers. Before we give broader details on these VWEs mentioned, a description of the system architecture used for VWEs is given, to help understand how they work from a technical point of view.

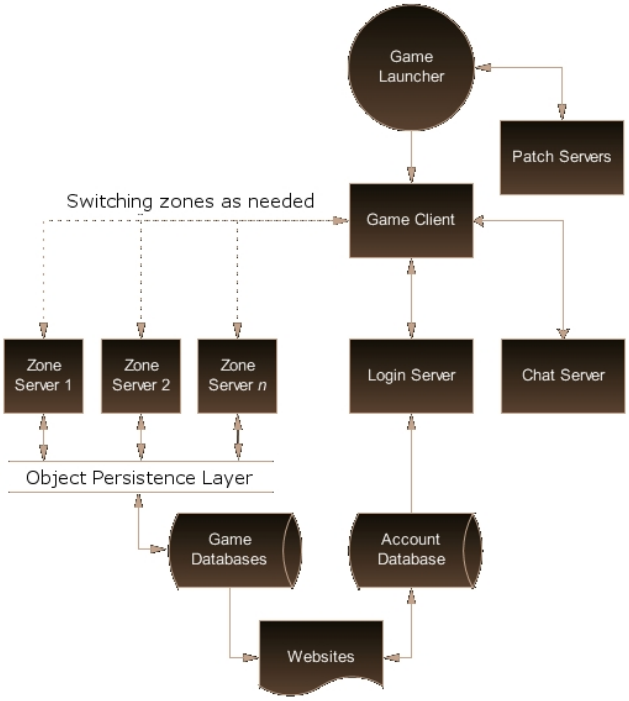

Figure 1. System architecture of VWEs [15]

In Figure 1 shows the architecture of a general VWE in the view of a massively multiplayer online role playing game based virtual world. VWEs are essentially client-server based and as a result rely heavily on servers for processing. In VWEs servers handle things such as interpretation of rules, maintenance of location and movement data [15]. The client actually does very little work in terms of processing of this data; often to prevent users on the client side from exploiting/cheating the VWE to give them a competitive advantage over other players [15]. A brief rundown of the architecture shown in Figure 1 is presented below:

Client: As mentioned VWEs are heavily reliant on a number of servers, however the user still needs some form of interface to interact with these servers and/or the VWE. This is where the client comes in, it includes a 3D engine for rendering the virtual world visual content and a control interface for manipulating an avatar [15].

Login server: In order for users to gain access to a VWE of their choosing they often need a login and password. The login server will handle the authentication of users, ensuring the user is who they claim to be. Once the user is authenticated by the login server, they will connect to the game server.

Game server: The purpose of this server is to enforce all the 'game' rules, collision detection, allow users to activate skills and attacks, as well as constantly recording their location and the movement of virtual property items [15].

Chat server: It is common in most VWEs to allow for text based communication between users. Commonly all users of VWEs will connect automatically to a chat server as they enter the virtual world, therefore regardless where they are in the world; they can still communicate with any other user that is logged in, even if they are not in the same location.

Databases: These are used to keep track of and save all the information, statistics, and inventory (virtual property items) with relation to avatars within the VWE [15].

Website: It is common for most VWEs to have an associated website which allows users to view such things as avatar profiles and/or armory, forums, news and we are now seeing web based virtual property auction systems linked into the VWE.

Patch server: Due to the complex nature of VWEs it is common for software updates to occur fairly often. These updates can be simple security fixes or content updates. When an update comes available the patching server often will require users to connect and download any updates before any further use can continue.

Zones: Due to VWEs being 'massive' in nature, it would be difficult and require a lot of data transfer to report to every client what is happening with every object and event in the virtual world [15]. This technique of zoning divides the actual virtual environment in different continents/areas each located on separate servers. So essentially only users located in area X will receive reports/updates for that specific area and not users who are in areas Y or Z.

A. *World of Warcraf (WoW)*

World of Warcraft (WoW) [12] is an online role playing VWE released for personal computers in 2004 with a user base consisting of 10's of millions of users worldwide. As of late 2010 WoW had over 12 million subscribers – when they launched their second expansion named Wrath of the Lich King [16]. This VWE utilizes a subscription based model where users pay approximately US$15 a month in order to access the world. This VWE uses a client-server based model; where the user through a client interface application executing on their personal computer connects to a WoW server over the internet. In this VWE users can take on the role of a fantasy based character, through which they can explore and quest across a large virtual world. *World of Warcraft* can allow thousands of users to interact with each other in the same virtual world. Users can form relationships with other users and compete against each other for virtual currency or virtual property such as armor or weapons.

B. *Guild Wars (GW)*

Guild Wars (GW) [13] is also an online role playing environment which was released on the personal

computer in 2005. In 2009 this VWE had sold over 6 million copies of the client software [17]. This VWE uses a client-server based model; where the user will utilize a client interface on their personal computer and then will connect to the server over the internet. A user is required to purchase the software but can play for free with no monthly subscription fee. *Guild Wars* is popular as it takes all the best aspects of other online games and combines them into a mission based design. *Guild Wars* allows users to create a fantasy based character in a virtual world and supports cooperative play. It also allows users to compete against each other for virtual currency and virtual property.

C. Entropia Universe (EU)

Entropia Universe (EU) [14] is a VWE that was designed by the company MindArk for the personal computer. It has grown to more than 1,000,000 registered accounts from over 200 countries or territories [18]. This VWE uses a client-server based model; where the user will utilize a client interface on their personal computer and then will connect to the server over the internet. The key reason for selecting this VWE is that the virtual economy is backed by real world money. It is currently the only VWE with a true Real Cash Economy (RCE) [18]. *Entropia Universe* employs a micropayment business model where players can play in the world for free but the company allows users to buy virtual currency, called Project Entropia Dollars or PED, which can then be traded back to the company for real world money. In *Entropia Universe* this means that virtual property and virtual currency has real world value, allowing users at any time to 'cash out' of the VWE. The better the user is at collecting virtual property in the world, the more money they can make outside of it. This VWE is an excellent choice for software inspection purposes since virtual property in this VWE has distinct value and many virtual property transaction occur in the VWE but also many real world transactions occur, in terms of 'cash-outs'.

IV. SOFTWARE INSPECTION PROCESS

The software inspection process has been used extensively as a common process for debugging and improving source code quality [19]. A related method is that of usability inspection, which is a popular way to evaluate user interfaces [19].

The method of inspection used in this paper is a combination of software and usability inspection techniques. The software inspection process used in this study will be that of *operational software inspection*, a process that typically involves a group of individuals. In this instance one individual (the first author) is used to examine the software to find defects or security vulnerabilities, which are often the result of one or many design or operational faults.

The test environment used for this operational software inspection process took place in the School of IT at Deakin University (Waurn Ponds campus). The test computer system (PC based system running a default Deakin University installation of Windows XP) is located on this campus, utilizing the universities network connection to the internet.

The VWEs being inspected were installed into the ~/Program Files/ directory on the test computer. Additional software based tools that were used for testing included: Sniphere, Wireshark and Actual Keylogger (discussed more in Section 5).

For inspection purposes no security measures will be in place apart from those inbuilt into the VWE's being inspected. All firewalls and antivirus software will be disabled on the inspection computer. This is to try and mimic the greater population of personal computers running VWE clients, where some may have security and some may not.

This operational software inspection consists of a clearly defined agenda and set of requirements for the inspection process. The set of inspection categories and the sets of tools utilized are presented next.

V. INSPECTION CATEGORIES

This section will outline the categories of testing that will occur on the selected VWEs. These categories were chosen as a result of a literature review conducted in the overall research project and determined as the most crucial areas of concern that relate to VPT and how it occurs. Firstly authentication will be focused on, as this is the users primary method of gaining access to a VWE, and in order to conduct theft; thieves have to break the authentication to gain access to victims accounts. Secondly detection will be focused on; in order to conduct theft there is often a specific signature of events that occur for this to be successful. If you can detect when theft is occurring you can stop it and prevent the virtual property from being stolen. Lastly recovery will be focused on, this is the last resort. If theft does occur and virtual goods are stolen; can users get them back effectively and in a timely fashion or are they lost forever?

A. Authentication

Authentication is the process of proving or confirming by the VWE server that an individual attempting to login to an environment is authentic and the actual owner of the account being accessed. Authentication is a vital component of most online or offline digital environments including VWEs, as it directly relates to being able to access a user's account and the virtual property within it. Authentication can come in the form of passwords, biometrics and digital signatures. The aspects we wish to inspect here are password sniffing, password robustness and keylogging.

1) Network Password Sniffing

This technique works by attempting to view the password as it is sent from the client to the server. This test will determine if encryption is used to send the username and password to the login server. The software used to do this testing will be Sniphere version 2.0 [20]. The process will involve launching the virtual world software and then starting the sniffing software, then proceeding to login to the server and analyzing if the password is sent unencrypted.

2) Password Robustness

This process will not be automatic and will work by manually entering in commonly used passwords until one is accepted. This is to determine the robustness required for client passwords in VWE's. This works due to people in general choosing easy to remember words as their password; for example pets name or family name. The technique used for this will be getting an assistant to set the password to the test account; making it unknown to the tester. The tester will then use a dictionary file of commonly used passwords and using human input to see if one is accepted. Risk can be determined in the results by looking at if the VWE operator requires the user to set a specific kind of password, for example 8 characters and combination of letters and numbers.

3) Keylogging

This technique is executed by utilizing a Trojan type program to monitor keystrokes on a user's computer system. This is a popular attack used to gain unauthorized access to user accounts in many popular VWEs. The testing application for this will involve the use of legitimate key logging software named Actual Keylogger [21], it will involve the monitoring of keystrokes as the authentication procedure is performed on each VWE, to determine if username and password can be captured.

B. Virtual Property Trading

Virtual property trading relates to the trade between users of virtual property within the VWE. Trade can occur in many different ways such as direct trade between two avatars (virtual characters), sending virtual property through an in-world mail system as well as buying and selling at an in-world auction house or multi user trade interface; which is a common feature in most of these VWEs. Once a computer criminal gains unauthorized access to a user account, these transactions allow them to steal virtual property from one account and send it off to another which they own. The aspects we wish to inspect here are mail trading, direct trading and aspects of multi-user trade mechanisms.

1) Direct Trading

Direct trading is the process whereby user X will open up a trade window dialog box with user Y and transfer virtual property directly. This requires both user X and Y to be online within the world at the same time and essentially provides a real time way of transferring virtual property. The aim of this test is to determine if virtual property items can be directly traded effectively without any security mechanisms or restrictions in place to ensure they are not being stolen.

2) Mail Trading

Mail trading involves user X wishing to send an item to user Y; commonly they send an electronic mail from within the VWE which often contains a message and the virtual property item/s. This mail system is often used when user Y is offline or not available for a normal trade window scenario. This is considered a quick and convenient option for players but can provide an easy avenue for unauthorized users to transfer virtual property to another account which they own without requiring them to be logged in on two different accounts at the same time. The aim of this test is to determine if virtual property items can be traded through mail based systems effectively without any security mechanisms or restrictions in place to ensure they are not being stolen.

3) Multi-user Interface Trading

In many of the VWEs looked at for this study; multi user trade interfaces such as an auction house are prevalent within them. An auction house is usually a virtual building within the world, where users can walk in and talk to a NPC (Non Player Character) avatar and place virtual property up for auction in order to sell and potentially make some profit. A thief might utilize this means; whereby they will have two accounts, one being their own account and one being a compromised account. The thief will then log into the compromised account; access one of the characters owned by that user and proceed to place virtual property up for auction for a small price and then buy it on their own legitimate account; providing a means to launder the property to make it seem like an innocent transaction. The aim of this test is to determine if virtual property items can be traded through multi-user trade interface based systems effectively without any security mechanisms or restrictions in place to ensure they are not being stolen.

C. Intruder Detection

Intruder detection is the process of detecting intruders or in this case potential thieves as they aim to gain access to unauthorized accounts. If a thief or hacker is not detected by the VWE software, they can break into many accounts and steal virtual goods without being noticed until the owner logs in to discover this theft has occurred. If an unauthorized user can be detected before entry to the VWE is permitted, many of the thefts can be stopped. The aspects we wish to inspect here will be looking at failed login attempts, unusual internet protocol addresses, unusual MAC addresses, and the software version the time of login and can they log onto an account at the same time as the user is currently logged in.

1) Login Attempts

This test will be performed during the authentication or login process, whereby the numerous failed attempts at logging in will be conducted and then determine if the VWE actually locks the user out of the account all together for conducting numerous failed attempts. Detection will be asserted TRUE if the account is locked after a certain amount of logins at that time.

2) IP Address

This test will conduct logging into a test account of a particular VWE numerous times from one class-B IP address, and then proceed to logon with a completely different class-B IP address using a foreign proxy address and then analyze if this is detected in any way. Therefore detection will be asserted TRUE if the account is locked at that exact moment or a number of days later.

3) MAC Address

This test will conduct logging into a test account of a particular VWE numerous times from a test computer with a specific MAC address and then proceed to logon from a completely different computer with a different MAC address and then analyze if that is detected in any

way. Therefore detection will be asserted TRUE if the account is locked at that exact moment or a number of days later.

4) Login Time / Zone

This test will be performed by logging into a test account of a particular VWE a number of times at a set time each day for a period of time, and after that period proceed to login at completely different times. Analysis will determine if the account gets suspended due to usage times being drastically different time zones.

5) Concurrent Access

One important factor is to determine, can a potential thief login to your account at the same time you are logged in? If so this could present some dangers in terms of having all your items stolen, while you're actually still logged in. This test will be conducted by logging into a test account of a particular VWE, then attempting to login to that same VWE with the exact same username and password. This is to determine if there are measures in place to stop two individuals from logging into the same account twice concurrently.

D. Recovery

Recovery relates to the reacquisition of stolen virtual property by the original owner. Virtual property often has great value associated with it by the owner; this is due to it often taking great amounts of time and effort to gather, not to mention the user is often paying a subscription fee to play within the VWE. When an item is stolen from a user's account, it is highly beneficial to return the property back to the original owner without a lengthy process as this involves human interaction by staff, costing money to the company since there could be many of these recovery sessions to do per day. The process of recovering stolen virtual property needs to be done accurately so that all individuals involved in the theft are compensated for any innocent transactions or example if a thief is selling stolen items on a virtual world auction house system and innocent users are spending virtual currency to buy these goods without knowing they are stolen.

The aspects inspected here relate to using a scenario based method where theft will be simulated and then requests will be issued to determine if VWE operators can recover the stolen property. More precisely, this series of tests looks at recovery of virtual property after it has been reported or detected as stolen. The only test here will consist of using VWE operators (administrators) to assist in the recovery of the stolen property. The aim of this test is to discover if virtual property can be recovered in the VWE or if it will remain stolen for good.

This experiment was designed to replicate a real VPT (and in need of recovery) situation as much as possible. Our aim was to design the experiments in a way that they would appear to the VWE operator to be a legitimate theft and recovery situation and not a mock scenario. The VWE operators from each individual VWE did not have any affiliation with the tester or knew of this experiment beforehand. This experiment breaks down to essentially, conducting a theft of a number of rare virtual property items between two individual unassociated accounts (thief and victim), then placing a request on the victims account using an online help system featured in-world; asking if the stolen virtual property items could be recovered and returned. Whereby the VWE operator would outright deny the request or conduct some investigation and recover the stolen virtual property items returning them back to the victim account.

As discussed above the inspection process for this test was essentially the same for all three VWEs. This involved having two independent user accounts for each VWE analyzed. These accounts were not related in any way to ensure that the act of VPT (performed entirely by the tester) was viewed as a legitimate act of theft between two separate entities by the VWE operator. This was achieved by registering these accounts under acquaintances of the tester.

These two individual accounts are logged into from two different Internet Protocol (IP) addresses. Each account will have an avatar created for it, and each avatar will be setup with a number of virtual property items of varying quality.

The process will involve the avatar from the first account trading; two to three virtual property items and virtual currency to the avatar from the second account using a direct trade mechanism. The first avatar will wait 24 hours and then report these items as stolen and request recovery. Due to the recovery process being textual conversing (through the online help feature) between mock victim and VWE operator entity, to validate if a test was successful or not, we utilized simple visualization to analyze whether or not the VWE operator was able to complete the recovery of our stolen virtual property items. This was answered by the operator entity either outright denying the request by using such phrases as "Sorry we are unable to complete your request" or accepting the request, doing some investigation and placing the stolen goods back into our inventory. An assertion value of TRUE or FALSE was recorded if the property is returned or not, respectively.

VI. RESULTS

There are two main methods of risk analysis and one hybrid method. First we have qualitative; this aims to improve the awareness of information systems security problems and the position of the system being analyzed [22]. Secondly we have quantitative; this is the identification of where security controls should be implemented, as well as the cost it will take to implement them [22]. Lastly the hybrid method is a combination of both the first two methods, and can be used to implement the components and use the available information all while minimizing the metrics to be collected and calculated [22]. The hybrid method is generally considered a less intensive and expensive method, compared to in-depth analysis.

This study will use qualitative analysis, it is considered much simpler and more widely used [22]. The goal of this study is to identify the parts of VWEs that are at risk and the vulnerabilities that might allow those threats to be realized, so this makes qualitative analysis perfect for this

situation. The analysis in this study will use simple calculations and procedures which will determine the impact, probability and overall risk evaluation associated with these threats.

Each test category was broken down into authentication, virtual property trading, intruder detection and virtual property recovery. Each test involved specifying what was evaluated, in which VWE, if the test was successful or not (assertion), what probability of this threat is, what the impact will be if it occurs, and then provide the overall outcome in terms of risk value (calculation of assertion, probability and impact) and evaluation. The tables of results in this study will list the assertion (Table I), probability (Table II), impact (Table III), risk (Table IV) and evaluation (Table V) outcomes.

A. Assertion

This is a simple test and is evaluated by determining if the aim of the test failed or was successful. A value of 'true' will be given if the test was successful and a value of 'false' will assigned if unsuccessful. The assertion results for each of the inspection categories: Authentication, Unauthorized Trade, and Intruder Detection; as presented in the previous section are shown in Table I.

B. Probability

A risk is an event that 'may' occur. This is evaluated by judging based upon knowledge or belief, how possible it is that a particular risk will occur. The probability of this risk occurring will be given a value of 'low', 'medium' or 'high' by the tester. The probability value is based upon knowledge or belief by the tester on how possible it is that a threat could occur. Note: It cannot be exactly certain or uncertain to occur or it wouldn't be classified as a risk.

- A value of 'low' will be assigned when the probability is at a point where it is so low that it is very unlikely to occur.
- A value of 'medium' will be assigned when the probability is at a point where it is considered possible to occur.
- A value of 'high' will be given when the probability is at a point where it is very certain that it will occur.

C. Impact

A risk by nature will always have a negative impact. The rating of the impact depends on factors such as in this case: value of the virtual property items stolen, chance of recovering the items and the victim's emotional impact. Impact will also take into consideration, what the potential result will be of this threat occurring? A value of 'low', 'medium' or 'high will be given by the tester which represents the impact.

- A value of 'low' will be assigned when the impact of the threat being successful will not result in the act of VPT. The result here can often be ignored as the threat will provide little or no impact.
- A value of 'medium' will be assigned when the impact of the attack being successful could result in VPT occurring. The result concluding from the impact of this threat can be managed effectively.
- A value of 'high' will be assigned when the impact of the threat being successful will almost certainly result in VPT occurring. The impact here is something you want to pay close attention too, as it will often result in misfortune in the form of VPT.

D. Risk

This is evaluated by how easily the assertion was achieved, combining the probability of the attack occurring and the impact if the attack occurs. A value of 'low', 'medium' or 'high' will be given. Figure 1 shows a risk matrix applicable in qualitative risk analysis; the threat level comprises of the probability or likelihood and impact of a risk. Probability is the chance that the risk will occur. Impact is the amount of damage that it would do were it to occur [23].

TABLE I. AUTHENTICATION TESTS – ASSERTION RESULTS

	Authentication Assertion			**Unauthorized Trade Assertion**			**Intruder Detection Assertion**				
	Password Sniffing	*Password Robust*	*Key-logging*	*Mail Trade*	*Direct Trade*	*MUT*	*Failed Logins*	*Multi-IPs*	*Multi-MACs*	*Login Time*	*Concurrent Access*
WoW	False	True	True	True	True	True	False	True	True	True	False
GW	False	False	True	True	True	True	False	True	True	True	False
EU	False	True	True	False	True	True	False	True	True	True	False

TABLE II. AUTHENTICATION TESTS – ATTRIBUTED PROBABILITY

	Authentication Probability			**Unauthorized Trade Probability**			**Intruder Detection Probability**				
	Password Sniffing	*Password Robust*	*Key-logging*	*Mail Trade*	*Direct Trade*	*MUT*	*Failed Logins*	*Multi-IPs*	*Multi-MACs*	*Login Time*	*Concurrent Access*
WoW	Low	Medium	High	High	High	Medium	Low	High	High	High	Low
GW	Low	Medium	High	High	High	Medium	Low	High	High	High	Low
EU	Low	Medium	High	Low	High	Medium	Low	High	High	High	Low

TABLE III. AUTHENTICATION TESTS – ATTRIBUTED IMPACT

	Authentication Impact			**Unauthorized Trade Impact**			**Intruder Detection Impact**				
	Password Sniffing	*Password Robust*	*Key-logging*	*Mail Trade*	*Direct Trade*	*MUT*	*Failed Logins*	*Multi-IPs*	*Multi-MACs*	*Login Time*	*Concurrent Access*
WoW	High	High	High	High	High	Medium	High	High	Medium	Medium	High
GW	High	High	High	High	High	Medium	High	High	Medium	Medium	High
EU	High	High	High	High	High	Medium	High	High	Medium	Medium	High

TABLE IV. AUTHENTICATION TESTS – ATTRIBUTED RISKS

	Authentication Risk			**Unauthorized Trade Risk**			**Intruder Detection Risk**				
	Password Sniffing	*Password Robust*	*Key-logging*	*Mail Trade*	*Direct Trade*	*MUT*	*Failed Logins*	*Multi-IPs*	*Multi-MACs*	*Login Time*	*Concurrent Access*
WoW	Medium	High	Critical	Critical	Critical	Medium	Medium	Critical	High	High	Medium
GW	Medium	High	Critical	Critical	Critical	Medium	Medium	Critical	High	High	Medium
EU	Medium	High	Critical	Critical	Critical	Medium	Medium	Critical	High	High	Medium

TABLE V. AUTHENTICATION TESTS – EVALUATION RESULTS

	Authentication Evaluation			**Unauthorized Trade Evaluation**			**Intruder Detection Evaluation**				
	Password Sniffing	*Password Robust*	*Key-logging*	*Mail Trade*	*Direct Trade*	*MUT*	*Failed Logins*	*Multi-IPs*	*Multi-MACs*	*Login Time*	*Concurrent Access*
WoW	Controllable	Secure	Needs Attention	Needs Attention	Needs Attention	Controllable	Secure	Secure	Controllable	Controllable	Controllable
GW	Controllable	Secure	Needs Attention	Needs Attention	Needs Attention	Controllable	Secure	Controllable	Controllable	Controllable	Controllable
EU	Controllable	Secure	Needs Attention	Secure	Needs Attention	Controllable	Secure	Controllable	Controllable	Controllable	Controllable

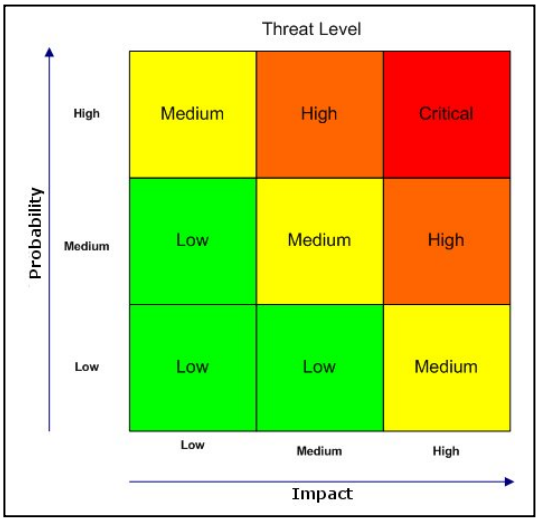

Figure 2. A decision matrix to determine risk rating [23]

- A value of 'low' will be assigned when the risk is at a point where it is so low that there is no apparent threat or danger to the user in terms of VPT. When determining a risk rating of low, both the probability and impact will go towards giving an ultimate value. In Figure 2 these are shown as "green" risks; these are insignificant and most likely will not result in VPT.
- A value of 'medium' will be assigned when the risk is at a point where it is considered a possibility that the user in danger of theft. When determining a risk rating of medium, both the probability and impact will go towards giving an ultimate value. In Figure 2 these are shown as "yellow" risks; they can have a major impact in terms of VPT but if they are managed well by the VWE operators, they can be mitigated.
- A value of 'high' will be assigned when the risk is at a point where it is considered very likely to occur and will provide an impact in terms of VPT. When determining a risk rating of high, both the probability and impact will go towards giving an ultimate value. In Figure 2 these are shown as "orange" risks, these are of high concern which can lead to virtual property theft, but not of critical concern.
- A value of 'critical will be assigned when the risk is of such a large scale that the user is very likely to be in danger of this threat. When determining a risk rating of high both the probability and impact will go towards giving an ultimate value. In Figure 2 these are shown as "red" risks and often occur when the VWE software or VWE operators are either unfamiliar with the risk or have no way of stopping it at all, so therefore resulting in a high frequency of VPT.

E. Evaluation

Each individual VWE that was analyzed will receive a security evaluation. This is evaluated by a combination of assertion, risk and the testers experience and knowledge. The evaluation will be given a determination of 'needs attention', 'manageable', or 'secure'.

- A determination of 'secure' will be assigned when the evaluation of the test is at a point where the problem or threat is considered negligible and improbable that it will be exploited and generally safe from the chance of VPT.
- A determination of 'controllable' will be assigned when the evaluation of the test is at a point where the problem is considered probable to happen and may be exploited.
- A determination of 'needs attention' will be assigned when the evaluation of the test is at a point where the problem is considered so high that it is very probable that it will be exploited and presents high concern regarding VPT.

F. Authentication Risk

Authentication deals with processes that relate to when the user is logging into a VWE. The tests associated with authentication, look at password sniffing, password robustness and keylogging and how much of a risk exploiting vulnerabilities in these areas present.

In Table II the authentication risk of each of these security concerns is shown, categorized with the VWE that the test was performed on. As you can see from this table the risk associated with password sniffing for each VWE is quite low, so there is very small chance that a users password could be intercepted between the client and server by a third party. The next risk being password robustness displayed that the password requirements and complexity required by each VWE is of mid level, thus reducing the chance a user's password could be gathered via brute force techniques. When it comes to the last risk being key-logging, as shown in the graph this is a large problem for all VWE's and users computers alike. Keyloggers fall under the category of malicious software and thus users should keep up to date antivirus and anti malware software on their personal computers as well as not visiting strange websites where key-loggers could be automatically downloaded. Overall the risk associated with authentication presents a high risk (medium probability and high impact) concern to all VWEs tested and can be dealt with some small provisions such as awareness and security software for user's personal computers such as anti-virus and anti malware

G. Trade Risk

Unauthorized trading deals with mechanics within the VWE which allow virtual property to be traded from one avatar to another. The tests associated with unauthorized trading involved mechanisms such as at mail trade, direct trading between avatars and multi-user trading.

In Table II the risk associated with each of these trading techniques is shown, categorized with each VWE that was analyzed. From the graph; mail trade is a critical risk (high probability and high impact) for *World of Warcraft* and *Guild Wars*, but is a low risk for *Entropia Universe*, due to the fact EU has no active mail trading mechanism. Mail trading can be used by for virtual property thieves to trade virtual property items without requiring a second avatar to be logged in at the same time to be used as a form of bank for stolen goods.

The next risk, direct trading, allows direct avatar to avatar trading and represents a critical risk (high probability and high impact) for all VWEs. This option in the testers belief is the most likely option to be used as a form of theft mechanics in VWE as it allows for real time trading of stolen goods, enabling thieves to log into a potential stolen account and at the same time be logged into a separate account they own, then send valuable virtual goods from the stolen account in real time and then log out. The last risk, multi-user trading, deals with looking at trade mechanisms which allow thieves to send virtual items to many different users, and be used as a form of laundering or attempting to bring legitimacy to the trade of stolen goods. Overall the risk of unauthorized trading presents a medium (medium probability and medium impact) concern for all VWEs looked at and represents unauthorized trading can occur quite easily once an unauthorized user has gained access to an account.

H. Intruder Detection

Intrusion detection deals with being able to detect unauthorized users that attempt to gain access to a legitimate users account for the purpose of virtual property theft. The tests associated with intrusion detection are failed login attempts, unusual internet protocol address (IP), unusual media access control address (MAC), unusual login times and concurrent login attempts.

In Table II the risk associated with each of these intrusion detection techniques is shown, correlated with each VWE that was analyzed. For all the VWEs tested it was shown that when an individual attempts a login a number of times and fails, it was detected and the account suspended for a time period, presenting medium risk from attacks such as automated brute force or a thief trying to guess a users password by hand. The next test was to determine if the VWE detected that a user had an unusual IP address than what had been used in the past and initializes any measures to accommodate that. The result of that test was that the risk is very high and no measures were taken by the VWE to stop a user from logging in from a completely different IP address.

The next test shown in Table II was similar to the IP address test but in fact looked at the MAC address, which is a unique identifier for network interfaces and each computer has a unique one of these. The result of the test was that no measures were taken by the VWE to stop a user from logging in from another computer with a completely different MAC address. Therefore the risk is classified as high. The next test as shown in Table II looked at if the VWE took any measures to detect if a user was logging in at an unusual time, differing than

what they usually are on at, say logging in at 3AM as opposed to what they normally log in at being 6 PM for example. The result of this test was the risk was of high concern and no measures were taken by the VWE to alert VWE operators of strange login times on a users account.

The final test as shown in Table II analyzed if two individuals could login to the same account on the same VWE, at the same time concurrently. In all VWEs this proved to be secure, no two accounts can be logged in at the same time. What occurs is simply the current person logged in, is disconnected once the second attempt is successful in authentication. The result here is therefore of medium risk, due to the fact it does actually stop two individuals from being logged in at the same time but does not prevent a hacker from logging into a stolen account, then whereby the system disconnects the owner off the account and the hacker can steal virtual property items of his choosing until getting disconnected. A better developers may implement would be to not disconnect the current active user if say a hacker is trying to login to the account in question. If the owner of an account is logged in and becomes disconnected from the internet, a timer of inactivity could be issued, whereby the account will become automatically logged out after say 6 minutes; then they could log back in.

Overall intrusion detection is for the most part a very high risk for all VWEs looked at; in most instances there are no detection mechanisms in place or there is no follow up when flags are triggered (such as unusual IP address, unusual MAC address, strange login times, concurrent logins); allowing thieves to freely venture in and out of stolen accounts without risk of being detected.

I. Scenarios: Virtual Property Theft Recovery

Recovery mechanics looks at the evaluation of the recovery tests and if they were able to be achieved or not. In Table III a series of recovery tests were performed on all VWEs chosen and as a result a success or failure measure was given. An evaluation result is given in this table also, which presents the authors evaluation as to whether the system being scrutinized is effective or ineffective.

TABLE VI. RECOVERY SCENARIO TESTS

	Virtual Property Recovery Scenarios		
	Success	*Failure*	*Evaluation*
WoW	(0 from 3) 0%	(3 from 3) 100%	Ineffective
GW	(0 from 3) 0%	(3 from 3) 100%	Ineffective
EU	(0 from 3) 0%	(3 from 3) 100%	Ineffective

As shown in Table III each of the virtual property recovery tests were performed at varying times and they all failed. This represents a high degree of inability for VWE operators to recover virtual property once it is stolen by thieves.

J. Scenarios: Virtual Property Theft Detection

A set of virtual property theft scenarios were performed to determine once a theft occurred, if the VWE software or VWE operator was able to detect it occurring, and stop it from resulting in theft. In Table IV a series of theft tests were performed on all VWEs chosen and as a result a success or failure measure was given. An evaluation result is given in this table also, which presents the authors evaluation as to whether the system being scrutinized is effective or ineffective. A measure of 'success' determines if the scenario was able to be achieved, a measure of 'failure' determines that the scenario was not able to be completed. Regarding the evaluation, a measure of ineffective determines that the VWE in question was not able to detect or stop the particular occurrence of VPT and a measure of effective determines that it was.

TABLE VII. VIRTUAL PROPERTY THEFT SCENARIO TESTS

	Virtual Property Theft Scenarios		
	Success	*Failure*	*Evaluation*
WoW	(4 from 4) 100%	(0 from 4) 0%	Ineffective
GW	(4 from 4) 100%	(0 from 4) 0%	Ineffective
EU	(4 from 4) 100%	(0 from 4) 0%	Ineffective

As shown in Table IV all the theft scenarios were successful, representing that theft was able to be performed without being detected by the VWE software or VWE operator whilst it is occurring.

VII. DISCUSSION

This section provides a discussion on results from our study and assists in determining such things as what are common security issues for other systems, how defective are VWEs compared to other systems and are the developers or the users at fault for security issues.

Livshits and Lam [7] conducted an analysis of nine particular Java based open source applications using the method of static analysis. Their aim was to show that the security of web applications is very important and that there is much vulnerability that still exists in these applications, which is of high concern. They found 29 security errors and vulnerabilities in the area of SQL injection, cross-site scripting, HTTP splitting attacks and other types of vulnerabilities such as tainted object propagation problems [7]. This relates to our study where we found a number of security vulnerabilities in the areas of authentication, unauthorized trade and intruder detection; in terms of the VWEs we tested, WoW had 8 security vulnerabilities, GW and EU both had 7 security vulnerabilities of varying degrees.

Christodorescu and Jha [8] conducted a study on the resilience of commercial virus scanning packages with the aim of presenting an architecture for detecting malicious patterns in executable files which specifically are resilient to code-obfuscation (obfuscated code is source code which has deliberately been made difficult to understand by humans) attacks. In this study they tested three commercial virus scanning applications using the obfuscated versions of four known viruses. Their results used a combination of nop-insertion and code transportation techniques in order to create obfuscated

versions of the four viruses, this proved to be enough to bypass detection by commercial virus scanners [8]. One specific result showed that a well known antivirus package (Norton antivirus) could not detect an obfuscated version of the devastating Chernobyl virus using the nop-insertion technique [8]. This relates to our findings where detection has proven to be a major concern, specifically in the three commercial VWEs we tested. These VWEs were unable to detect unwanted intruders in all cases (Table I) and unable to detect VPT when we performed our scenario tests (Table VII).

Hole et al. [9] conducted a study on Norwegian internet banks in the years 2003 to 2004 with the aim of determining if a false sense of security existed within the customers of banks, specifically with relation to online banking . This study they found that the authentication systems used to logon to net banking were quite weak, allowing a number of simple attacks to be conducted which provided fruitful results [9]. These results also concluded that many of Norway's Internet banking systems which consist of more than 1 million customers were in fact vulnerable to a combination of DDoS (distributed denial of service) and brute force attacks during the years 2003 to 2004 [9]. This relates to our study in the fact that one of the vulnerabilities that virtual world developers were actually able to secure, was the technique of brute forcing passwords. As shown in Table I, not one of the three commercial VWEs were susceptible to brute forcing authentication; most would lock the account after 5 login attempts. However Hole et al. [9] proved that even systems that close account access after 5 failed login attempts such as the 3 commercial VWEs in our study and similarly Norwegian banks (which on top of this also have two-factor authentication); can still be broken into. In their experiment with two Norwegian banks which locked accounts after 5 failed login attempts, they produced an average of 38 cracked accounts and the possibility of repeatedly attacking the bank which would crack new accounts each time [9].

Martin et al. [10] conducted an analysis on 6 large real-world open source Java applications, which contained close to 60,000 classes. The techniques by which they conducted this analysis was utilizing PQL (Program Query Language) queries (PQL queries allows developers to express a large class of application specific code patterns [10]) with a combination of static and dynamic techniques. They discovered through their analysis of these real-world Java applications utilizing these techniques that there were 206 errors which were categorized into security flaws, resource leaks and violations of consistency invariants [10]. They concluded that combining PQL, static and dynamic analysis can be effective at preventing errors such as security vulnerabilities at runtime (when the program is first launched). However, our findings were more effective at detecting errors or security flaws whilst the program was in operation. Through the use of the operational software inspection technique we found a total of 22 errors with the 3 commercial VWEs analyzed, which were categorized into authentication, virtual property trading and intrusion detection. Furthermore, we discovered 3 failed attempts at virtual property recovery (Table V) and 4 failed attempts at detecting VPT (Table VI).

Newsome and Song [11] conducted a study on software vulnerabilities and their effect on the internet. They state that internet attacks are quite fast and automatic in nature, so in order to stop them there would need to be fast detection and filtering mechanisms [11]. In their study they propose a technique called dynamic taint analysis, which automatically detects overwrite attacks (overwrite attacks are the most common type of exploit) [11]. The authors used taint analysis to detect a number of different attacks; which combined both synthetic and actual real exploits. Their technique successfully detected all the attacks and attempted exploitation they performed. To correlate our findings to this study, the 3 VWEs we analyzed were not as successful in detecting a number of attacks we performed such as the VPT scenario attacks we performed as shown in Table VII. On top of this in all 3 VWEs, the detection of intruders who had a different IP or MAC address and login/activity time than the owner of the account failed completely (Table I).

VIII. Potential Gain

The potential gain from addressing the problems that are persistent within the VWEs, which have been examined in this paper, is finding a solution to solve flaws existing in VWEs. And to achieve that this paper provides a vital discussion into determining why the flaws exist and how they are being used to conduct VPT. This paper Below is a more detailed list of potential gains that can come from fixing the flaws in VWE software.

- Limit or neutralize VPT from occurring: Fundamentally as a whole, stopping virtual property thieves from having free reign once an account is compromised, allowing theft of as many virtual property items as possible.
- Reduction in the black market of virtual property sales: Fewer occurrences of stolen virtual property items will result in thieves having less to sell on the black market.
- Account security more prevalent: Both users and virtual world operators become well versed in account security, reducing the occurrence of account theft and to less cases of VPT.
- Reduced costs of customer service, potential for development: This means virtual world operators will less load of customer service to deal with theft reports, allowing them to put more funding into development of content.
- Virtual world user-base will be safer and happier. The user-base can go about their virtual world activities, collect virtual property and currency without fear of it all being stolen by thieves. As a result subscriptions will continue and grow.

IX. CONCLUSION

Virtual world software is considered one of the most complex forms of software that exist today; it is essentially a large piece of enterprise software that consists of databases, specialized servers, client software, often millions of users and a huge amount of content. This complexity has presented points of vulnerability to many security problems that have existed for some time in VWE software for with the most part with no effective solutions being produced. People indulging in personal entertainment through buying VWE software and often paying a subscription fee per month should view these results, and then demand VWE operators improve security before the software is given global availability. Solutions to most of these problems can be moderately simple for a VWE development team. This small investment of time and effort and can quite potentially reduce VPT significantly.

There are some key intrinsic factors existing within VWEs which can lead to security compromise. One of these fundamental flaws with VWE software is primitive authentication features which allow key logging to be one off, if not the most fruitful, technique for stealing VWE accounts (which often then leads to VPT). From then on the ability for VWE software or operators to not only detect an account intrusion or detect a VPT situation is essentially nonexistent, allowing the theft to occur with no resistance. Lastly the ability for VWE operators to be able to recover and return victims (often hard earned) virtual property items is nonexistent according to our investigation results. Therefore we have in all areas of authentication, unauthorized trading, intrusion detection and recovery mechanisms we have inherent flaws ranging from medium to critical risk to users of these VWEs.

Until better security development practices are in place and thorough testing of VWE software (as shown in this study) from a security point of view occurs on VWE software by their creators, users should take valid measures to protect their 'investment'. Some of these measures that can be implemented by users to enhance security and avoid VPT are as simple as up to date anti-virus or anti malware software to prevent key-loggers and Trojans, changing your password frequently (bi-weekly, dependant on how much virtual property items you own) so if a potential thief does obtain a users login details, they will have changed the password hopefully before the thief gains access. Measures which can be utilized by VWEs to improve security and protect their community base are more up to date and secure authentication mechanisms along with effective intrusion detection and VPT detection systems, to not only avoid account intrusion but also if required detect VPT and stop it before it can occur. This study is crucial as it highlights the flaws and points out what needs to be fixed in VWEs. Overall the results gained from the set of tests presented in section 6 demonstrated that there are key areas of VWE software that require focus to improve security and reduce the chances of virtual property theft occurring.

To conclude we present a concise list of the most significant findings of this study, which represented a selection of three popular VWEs which were assessed in the year 2010 by a hybrid software inspection process.

- All three (100%) VWEs were highly susceptible to key logging methods, which is used in order to gain unauthorized access to user accounts.
- Mail and direct trading methods were to be the most likely method for intruders to transfer stolen virtual property items.
- Intrusion detection or lack thereof, is of critical risk to all three VWEs and is considered of extreme concern.
- Concurrent logins are not permitted but still don't prevent an account being compromised and virtual property stolen. Simply due to the fact, the current active user is disconnected upon a second successful login. This can potentially result in an ongoing loop of authentication between owner and potential thief.
- When virtual property was actively being stolen, this was not detected nor blocked.
- Stolen virtual property items were unable to be recovered in all scenarios tested – which is also of high concern.

REFERENCES

[1] A. Watters, "Number of Virtual World Users Breaks 1 Billion, Roughly Half Under Age 15," ReadWriteWeb, 2010.

[2] M. Eikenberry, "Real Money Trade is a Billions Dollar a year Industry," YouTube, 2011.

[3] N. Patterson and M. Hobbs, "A Multidiscipline Approach to Governing Virtual Property Theft in Virtual Worlds," in *What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience*. vol. 328, J. Berleur, M. Hercheui, and L. Hilty, Eds.: Springer Boston, 2010, pp. 161-171.

[4] B. Cathal, "Prioritizing Software Inspection Results using Static Profiling," 2006, pp. 149-160.

[5] DFC.Intelligence, "Virtual Property and Real Money Trade: A Business and Legal Survey," DFC Intelligence, San Diego, California2009.

[6] S. Cikic, S. Grottke, F. Lehmann-Grube, and J. Sablatnig, "Cheat-prevention and -analysis in online virtual worlds," in *1st International Conference on Forensic Applications and Techniques in Telecommunications, Information, and Multimedia and Workshop (e-Forensics '08)* Adelaide, Australia: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1-7.

[7] V. B. Livshits and M. S. Lam, "Finding security vulnerabilities in java applications with static analysis," in *Proceedings of the 14th conference on USENIX Security Symposium - Volume 14* Baltimore, MD: USENIX Association, 2005.

[8] M. Christodorescu and S. Jha, "Static analysis of executables to detect malicious patterns," in *Proceedings of the 12th conference on USENIX Security Symposium - Volume 12* Washington, DC: USENIX Association, 2003.

[9] K. J. Hole, V. Moen, and T. Tjostheim, "Case study: online banking security," *Security & Privacy, IEEE,* vol. 4, pp. 14-20, 2006.

[10] M. Martin, B. Livshits, and M. S. Lam, "Finding application errors and security flaws using PQL: a program query language," *SIGPLAN Not.,* vol. 40, pp. 365-383, 2005.
[11] J. Newsome and D. Song, "Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software," in *The 12th Annual Network and Distributed System Security Symposium* San Diego, California: Internet Society (ISOC), 2005.
[12] Blizzard.Entertainment, "World of Warcraft Community Site," California: Vivendi, 2004.
[13] NCSoft, "NCSoft Corporation," Korea: NCSoft, 2009.
[14] MindArk, "Entropia Universe," Gothenburg, 2009.
[15] J. Radoff, "Anatomy of an MMORPG, by Jon Radoff," 2007.
[16] W. Yin-Poole, "World of Warcraft hits 12m subscribers ": EUROGAMER, 2010.
[17] NCsoft, "Guild Wars Surpasses Six Million Units Sold," in *Guild Wars Press Release*, 2009.
[18] C. Donatello, "'Entropia Universe' Boasts Improved Land Grab System," Science Fiction, 2011.
[19] J. Nielsen, "Usability inspection methods," in *Conference companion on Human factors in computing systems* Boston, Massachusetts, United States: ACM, 1994.
[20] SecureSphere, "SecureSphere - Free IT Security Software.," 2010.
[21] Actual.Spy.Software, "Actual.Key.Logger," 2010.
[22] J. W. Meritt, "A Method for Quantitative Risk Analysis," Wang Global, 1999.
[23] B. Witzel, "Bad things that can happen to good people: Identifying project risks," CharityVillage, 2005.