Norsk Regnesentral
ANVENDT DATAFORSKNING
Norwegian Computing Center/Applied Research and Development

**Note**

# Cheating in Online Games – Threats and Solutions

Version 1.0

Knut Håkon T. Mørch

**Tittel**/Title:
Cheating in Online Games – Threats and Solutions

**Dato**/Date: January 8th
**År**/Year: 2003
**ISBN:**
**Publikasjonsnr.:** DART/01/03
Publication no.:

**Forfatter**/Author:
Knut Håkon Tolleshaug Mørch

**Sammendrag**/Abstract:
A survey for a suitable taxonomy and the selection of one used in this report has been done. Publicly known cheats for some selected online games has been surveyed. Some possible solutions against cheating have been proposed.

**Emneord**/Keywords: IT-security, Cheat Detection, Online Games

**Tilgjengelighet**/Availability: Open

**Prosjektnr.**/Project no.: 320041

**Satsningsfelt**/Research field: Security

**Antall sider**/No. of pages: 17

# Table of Contents

# Introduction

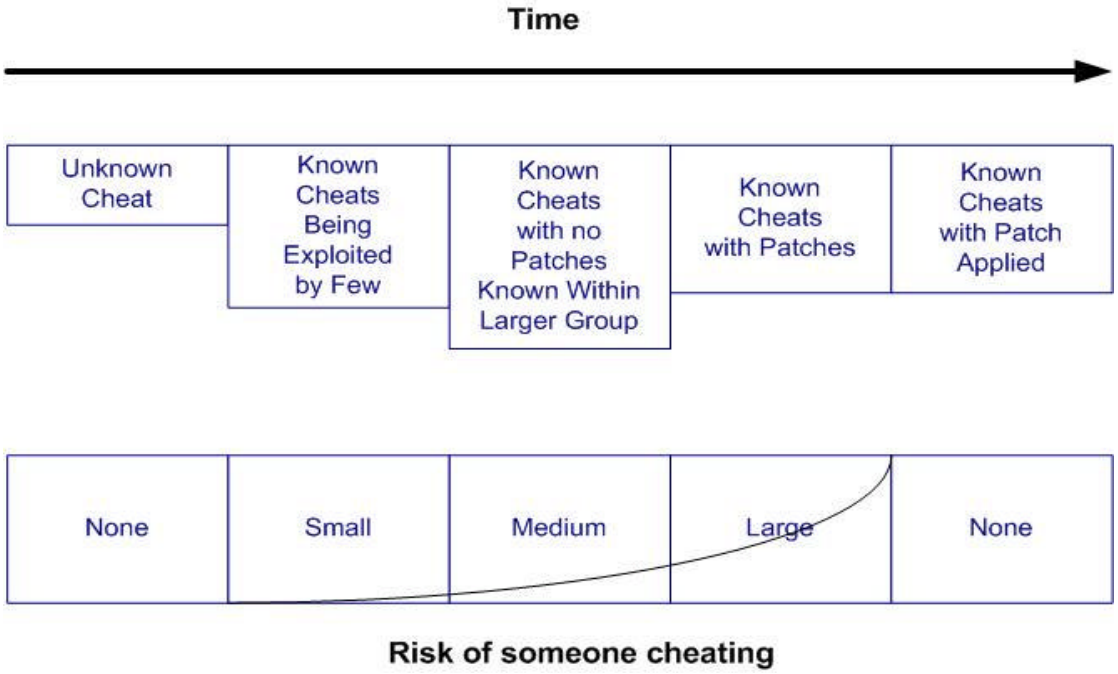
Cheating is an old concept in game playing. Card, board games and other none electronic games all have to deal with the possibility of someone not playing by the rules, or cheaters. Some none electronic gametypes have potential economic revenue for cheaters. One example is the card game poker when using money to bet. Others have only the thrill of winning. Electronic online games often have both.

Online games, like Warcraft III [6] or Quake III [7], must be purchased in order to use them. Cheating to get someone's player ID might save some money for the cheater. For some kind of games, like the Massive Multiplayer Online Game (MMOG) Anarchy Online [5], there is also potential revenue for items gathered within the game. A search on e-bay [4] for Anarchy Online revealed 164 items and/or characters for sale with start bid ranging from US$ 5 to US$ 800. If cheating could be used to build powerful characters or perhaps produce special items, the cheater may actually earn real life money on cheating in a virtual world. It has also been reported frauds where players have sold but not delivered virtual items and earned up to US$ 11.000 [10].

Other players just want to be the one with the highest score. This might be accomplished with talent, training and/or cheating. Cheating when playing a real world board game is usually not socially accepted. When playing an online game this changes. Players are often not on the same physical location and do not know each other. The social structures preventing cheating are no longer in place.

From the above it is clear that there are many reasons why one would like to cheat. It is also an important point that reasons not to cheat are often not present. This makes online games attractive targets for cheating. Although it's some years old, a survey from 1997 [8] shows that 35% of online gamers had cheated. Cheating is clearly nothing only a few players are willing to do. For game developers, cheating is a large threat to potential income, as cheaters will make the game less attractive to non-cheaters. A popular Massive Multiplayer Online Role Playing Game such as EverQuest has been reported to reap more than $50 million a year in revenue [21].

Cheats are in some ways similar to security holes in other software. First, only a limited number within a group knows about the cheat-program or possibility. Then more people within a subgroup get access to the information and start using it. The software vendor will usually at some time make a patch and make it known to everybody that such a vulnerability exists. Unlike the rest of the SW industry the game industry has here an advantage. When a patch is made, one can easily force everybody to upgrade in order to play on servers. However, should someone play and not use the patch they will be highly vulnerable. When all clients/players and/or servers have applied the patch, the risk is reduced to none. This is illustrated in the following figure.




In the following, we will try to present possible methods to cheat in online games, as well as possible solutions.

# 1 Taxonomy

In order to discuss cheats and countermeasures, one needs taxonomy to discuss the subject. Some people have made attempts to establish a taxonomy for cheating in online games. Pritchard [1] suggest six different categories for online cheating:

1. **Reflex Augmentation**: Exploiting a computer program to replace human reaction to produce superior results
2. **Authoritative Clients**: Exploiting compromised clients to send modified commands to the other honest clients who will blindly accept them
3. **Information Exposure**: Exploiting access or visibility to hidden information by compromising client software
4. **Compromised Servers**: Modifying server configurations to get unfair advantages
5. **Bugs and Design Loopholes**: Exploiting bugs or design flaws in game software
6. **Environmental Weaknesses**: Exploiting particular hardware or operating conditions

This categorization was according to Yan and Choi [9] made ad hoc, and many cheats couldn't fit into either of the categories. Yan and Choi [9] made an improved taxonomy based on Pritchard's [1]. They ended up with 11 different categories of cheating for online games. We've found this to be the most complete taxonomy published. The taxonomy will be briefly presented below and used in our research for known methods for cheating in online games.

1. **Cheating by Collusion:** It's not a new thing that players collude to cheat. This might occur in games where players are expected to not know what other players know. Consider a game of online poker. If three players participate in a game, two of them could cheat the third player by exchanging information. Another possibility is one player controlling more than one online player profile. In a single player game that would enable a player to possibly see areas of the virtual world he could not possible see at the same time with only one online player profile.
2. **Cheating by Abusing Procedure or Policy:** In some games statistics is use to record every wins and losses for a player profile. In order to avoid a loss being recorded, the player might disconnect before the game ends and the loss is recorded, thus abusing the procedure or policy.
3. **Cheating Related with Virtual Assets:** Trading of virtual characters and items (e.g. weapons or magical spells) acquired in games is a new and real business created by online games. With real money available cheating is attractive. With no digital signatures or other techniques to help claim possession of virtual items, cheaters may offer a virtual item, receive real money and not deliver the virtual item as agreed.
4. **Cheating by Compromising Passwords:** A password used by a player is often the key to all or much of the game data and authorization in a game system. Some players are interested in competing for the highest rank. The ranking system is often only protected by a password. Such systems might be protected for password guessing by only allowing 3 wrong passwords before locking the account. Due to user friendliness and the possibility of Denial of Service, such protection is often ignored.
5. **Cheating by Denying Service from Peer Players:** As discussed above, a system using passwords and lock-out protection might be misused to prevent legitimate players from playing, e.g. in a tournament. More experienced players might be blocked this way, thus increase the cheaters chance of winning. Another common cheat is flooding (or other techniques) one of the other players in such a way that the targets responses to the server or other players are significantly delayed. In real-time strategy games the other players might then vote the slower player out of the game. The cheater could use this to kick more experienced players out of the game.

6. **Cheating due to Lack of Secrecy:** Since many online games send packets with payload in cleartext over the network, a player can easily cheat by eavesdropping packets or inserting data. Encryption will possibly solve some of these problems.
7. **Cheating due to Lack of Authentication:** Cheats related to lack of authentication of one or both part in the communication between client and server.
8. **Cheating Related with Internal Misuse:** System administrators for an online game server have the power to do what cheaters want to do. This could include generation of special items that could be sold to other players for real money, or modifying some character for own use or because some player paid the system administrator.
9. **Cheating by Social Engineering:** Social engineering is often used to steal passwords. There are many variations of this scam but all of them aim at the same: to trick players to happily reveal their ID-password pair. One might try to trick the player into believing something attractive has happened to the player's account and that the ID-password pair is needed.
10. **Cheating by Modifying Game Software or Data:** This has been a traditional cheating since the beginning of the PC game area [9], and there are many tools available to enable cheaters to modify either program file or memory. Cheaters may use debuggers to re-engineer programs to get some unfair advantage. Advantages may include but are not limited to increased speed or other actions for objects or seeing what other players can't see.
11. **Cheating by Exploiting Bug or Design Flaw:** Some online cheats exploit bugs or design flaws found in a game software to get an unfair advantage.

In the rest of this report, the taxonomy suggested by Yan and Choi [9] will be used.

# 2 Some Publicly Known Cheats

With the increased popularity of online games, the number of cheats and the players using them has increased too. In order to understand the problems a survey has been conducted to find the cheaters state-of-the-art techniques. We do not intend to find and present all publicly known cheats, but have focused on finding cheats for some of the most popular online games.

Below we present some publicly known cheats within the categories defined by the taxonomy we chose to use.

## *2.1 Cheating by Collusion*

There are numerous examples of these kinds of cheats in the real world. In many online games there is a mode of playing called 'Capture The Flag' (e.g. Unreal Tournament [16]), where two teams compete to get a flag from the other and bring it to another place. A player on one team will usually be attacked if he approaches the enemy base. Two players or one player using two PC's might gain an advantage for one of them by joining one character for both the teams, e.g. blue and red. If the player on the red team tells the one on the blue team where his red teammates are located the blue team gets an advantage by cheating.

## *2.2 Cheating by Abusing Procedure or Policy*

In some tournaments, the games are meant to be played as released from the developers. A common cheat on such occasions is the use of scripts or macros. Such a script might decrease the time needed to buy items on the beginning of a round like the script Nextwish [11].

Another "popular" way of cheating is disconnecting the client to avoid the loss to be recorded. This has been reported for Warcraft III [14] and Starcraft [9] among others. In Warcraft III the wood and gold values are locked, thus forcing the game to kick the client without recording the loss. This hack resulted in Blizzard deleting all the accounts that had been using the hack [18].

## *2.3 Cheating Related with Virtual Assets*

It has been reported frauds where players have sold but not delivered virtual items and earned up to US$ 11.000 [10]. Especially role-playing games not limited by short "rounds" of gameplay are vulnerable to these kinds of cheats.

## *2.4 Cheating by Compromising Passwords*

Blizzard has a rather strict policy [15] for choosing a password for the account a player create to play online where certain passwords are not allowed, like those found in a dictionary. This indicates that stealing passwords has been a problem.

## *2.5 Cheating by Denying Service from Peer Players*

Where statistics are used there will always be someone who doesn't want a possible loss against a superior player recorded. One possible method is described in 2.2. Another possibility is forcing the other player to disconnect, thus avoid the loss being recorded and in addition let the other player be recorded as the one who left before the game was over. In Starcraft a lag hack was used to disconnect the other player [9]. Warcraft III has also been exposed to disconnect cheats [14], where one player forces another player to get disconnected.

## 2.6 Cheating due to Lack of Secrecy

Networked games without the use of cryptography might be vulnerable to inspection, modification and insertion of packets sent between players. A computer between two players playing e.g. a first person shooter game might act as a proxy, inspecting packets and insert the right coordinates to make the players' bullets or similar hit the target.

Aiming proxies has been built for Quake [17]. The first aiming proxies didn't take into account the player's field-of-view (FOV), thus shooting out of the back could occur. Later version only aimed at objects within FOV, compensated for lag and added enough randomness to stay below threshold values [1,17]. Other similar games are also vulnerable to this cheat.

## 2.7 Cheating due to Lack of Authentication

Players normally have to identify themselves in order to use online servers. Hence, the unique ID is valuable. A well-known threat to such systems is using a fake server or machine-in-the-middle-attack, which in some cases could steal or duplicate the unique ID. Worst case is a fake server stealing players CD-key that proves they bought the game. This CD-key might be resold for real money to other cheaters. Solving this problem will include some kind of two-way authentication. A fake server might also use the auto update feature to install a backdoor on clients machine, thus gaining full access to all the data on that machine.

It is also necessary to re-authenticate users who want to change their password. If not, someone temporarily leaving their machine, e.g. in an Internet café, could be exposed to someone else changing their password.

## 2.8 Cheating Related with Internal Misuse

Game servers are vulnerable for internal misuse as they are run by administrators who might change the server's behavior. It is reported that system administrators has been fired [1] due to misuse of administers right. A clan might also set up their own server, publish it on the website for the game and use their administrator rights to gain an advantage.

## 2.9 Cheating by Social Engineering

A common scam is pretending to be some authority person in an attempt to get ones CD-key, or the single thing that uniquely distinguish that a player actually bought a copy of the game. Blizzard [20] has published 8 rules to avoid such scams, indicating this kind of cheating is widely used and known.

## 2.10 Cheating by Modifying Game Software or Data

There are many ways to modify game software or data. Many games have a game engine with different modifications (Mods). These games can communicate internally between the game engine and the Mod, and that communication might be intercepted and values changed [13].

Such values might be found like reported in [1], where a player paused the game and searched for known values seen on the paused screen, i.e. the stock of wood and stones. Another approach for finding where to hook into the game or modify exe-files is looking at the single-player cheats many vendors provide [1].

Some cheat might change values to increase targeting or firing rate, like in Battlefield 1942 where cheaters could get 50% less off target for bullets and double firing rate [2]. There are two subcategories of this kind of cheating that are very common and well documented: object correlation and object exposure. Because of the popularity of these cheats they will be explained more detailed below.

### 2.10.1 Object Correlation

Cheats related to object correlation is cheats that try to correlate e.g. a bullet and the target. Such correlation might be done by providing the user with information to visually or automatically better correlate two objects.

***Extrasensory Perception (ESP)***

ESP cheats try to provide the player with visual enhancements to better enable the player to correlate between two objects. One such ESP cheat is improved or new radar and map for a game. Many games run a full copy of the game on each client, i.e. the client knows where all the other objects are. When the objects are positioned a place the player is meant to see them, they are shown on the screen. A modification might add a minimap or similar on the screen showing where all the other objects are at any time, like it's done for Counter Strike and Warcraft III [17,14].

Some games have built-in features allowing players to choose easy to spot colors for the enemies. This is not considered cheating. However, some cheats might remove textures, like it has been done for Quake III by modifying CVAR values [12]. This will give the cheater a better possibility to spot the enemies although this feature was not intended from the game developers.

Another variant of ESP is "glow" [17]. Glow ESP helps a player see enemies better. Either the whole object is changed to a color that is light or "glowing", or only the sides of the objects are "glowing". This makes it easer to distinguish friend from foe or spot objects in dark areas.

In some first player shooting games a crosshair or aiming object is used target the object to shoot. This object might be improved or in some cases where such an object is not already present be inserted to help the player aim. Sometimes the aim is only a dot, and an improved aiming object could significantly increase the chance of hitting the target.

***Automatically aim***

As described above, some games use a game engine and a MOD with internal communication. It is possible to hook into such solutions and modify the data to e.g. always hit the head of nearby enemies, like one aim hack for Counter Strike [17]. Such cheats correlates the attack object (e.g. a bullet) with the most vulnerable spot on the target.

Another automatically aim cheats changed the color of all the enemies in Counter Strike [17] to one color. The mouse pointer was then changed to automatically move to that color, making it easier to aim. The drawback with this technique is that the mouse pointer would move to any part of the body. Before cheat detection for this hack was issued, other players tried to beat the cheaters by spraying similar colors on the walls, a feature possible in Counter Strike [17]. Some advanced automatically aiming cheats has the possibility to only activate the gathered data (e.g. for a head shot) when the player press a certain button, making it easier to avoid being spotted as a cheater.

### 2.10.2 Objects Exposure

The goal for these cheats is to show something to the player that he shouldn't be able to see. This can be done be removing objects or making it possible to see objects meant to be hidden behind other objects.

***Removed objects***

In some games certain groups of objects clouds the visibility of other objects – and they are meant to. In Counter Strike smoke grenades release clouds of smoke that are very opaque, making them hard to see through. Cheats remove these clouds [17]. Warcraft III uses something called fog of war. This is visible as a layer over the terrain that only lets the player view the contour of the landscape below. Some cheats remove this, giving the player the opportunity to explore the whole map without moving objects around to remove the fog of war [14].

***Transparent objects***

This kind of cheat is often known as wall hack [17]. The following group of cheats are also known as wall hack, but this is probably the one most players think of when a wall hack is mentioned, i.e. the ability to see through walls. Sometimes these cheats makes all walls in a map 50% transparent [17]. In other words, it is possible to make some group of objects more or less transparent, making objects behind them visible.

***Changed object layers***

This method aims at showing objects a player wants to see without modifying objects like walls. A cheater can put some objects that are located behind others in front of those hiding them, making them visible. A solution might be to put all objects of a certain kind in front of any other object. This might – like in Counter Strike [17] – be a player model, enabling a player to know where other players are located. This wall hack is more efficient than 'transparent objects' because the cheater now only sees what is important and not everything else behind the wall. Seeing everything can limit ones ability to navigate on the map.

***Textual information***

Text can be shown about an object to reveal details about it not visually easy to spot for humans [17]. This might be items a player is carrying or the players' health or strength. It is possible to make the text visible on top of other objects, e.g. a player model. Further it's possible to show the text on top of other objects,

in effect creating a wall hack with only the text visible through walls. This has the effect that the player models are located where they should, but the cheater will get an early warning when text is shown on a wall but no player model is visible, i.e. the player model is behind the wall.

## *2.11 Cheating by Exploiting Bug or Design Flaw*

Online games have thousands of lines of code. Errors are likely to happen. Some recent bugs are the human farm bug in Warcraft III and cancel request bug in Ages of Empire. The first bug allows "a human player to build a farm in a very small clearing of trees, and continually build and cancel it. For unknown reasons, the player gains the cost of building a farm, without the original cost being deducted. Essentially, this bug can generate endless amounts of resources for a human player" [14].

The bug in Ages of Empire used the fact that when the network was lagging, it was possible to issue several cancel requests for a building of a building. When the network worked again – or was no longer attacked by the cheater – all the cancel requests for the same building would be processed, returning more resources than initially spent when starting the building [1].

# 3 Cheating Mitigation: Prevention, Detection and Management

Well-known security mechanisms such as cryptographic protocols have many applications in online games. However, beating the cheaters has no single solutions. A process-oriented approach is required for the best possible solution. Yan and Choi [9] treated the subject 'Cheating Mitigation' in their paper. Here, we've used theirs grouping of possible solutions and extended them with others.

## *3.1 Built-in Cheating Detection*

The battle between cheaters in online games and the anti-cheaters are quite similar to the virus battle. However, with cheats in online games the anti-cheaters have an advantage: the possibility to build cheating detection into the games from the beginning. Many game developers, such as Joe Wilcox from Epic Games [27], supports such an approach and are working on implementation in online games.

Yan and Choi [9] argues that "in case game providers cannot guarantee good security for game client software, the built-in detection should be implemented in a game server, which is typically installed in a protected environment where it is difficult for cheaters to tamper the software".

Such built-in cheat detection for the server software might be reused, thus it's a cost-effective alternative. Cheat detection could be related to such parameters as suddenly increase in hits for a player in a shooter game, or detection of actions that lead to duplication of items.

The United Admins software HLGuard [26] relies heavily on setting threshold levels for hit rates within some time window to detect automatically aiming or correlation between objects, i.e. statistical analysis. The method of statistically identify players whose aim is too good to be true is also recommended by Pritchard [1]. This has the effect that some players claim that they are banned just because they are good players [25]. This might be true, and illustrates a pitfall using such techniques.

However, as we have seen from chapter 3, quite a lot of the known and used cheats today are based on modification of the client software. Techniques to reduce such possibilities would hit the cheaters harder.

One such technique used by several game servers using is demanding the installation of software for and right to scan client machines in an attempt to find known cheats or altered CRC for game files. Such solutions could easily be built into a game. However, invading players' privacy, e.g. looking at other files than those related to the game itself, is controversial. Sony attempted this approach with their popular online game EverQuest. The result was a massive negative response from the player community, with the result that Sony backed away from the proposed solution, and didn't search for software on client machines [24].

Pritchard [1] argues that all clients might and for many online games already do run a full copy of the game simulation. This enables cheaters to find information about the other players, but it might be used against them. A cheat detection system might answer questions like "Can a player see the object he just clicked on?" [1]. The system might do this by randomly requesting a report of the game simulation copy on the clients, thus finding those tampered with and that are out of synch.

Such system might discover a machine between a server and a client modifying data. If the data received at the server is not compliant with those on the client, something weird is going on. Another application is for

peer-to-peer networks where authoritative clients might fool others to blindly accept commands or data from them. By moving from issuing commands to issuing requests, one can look for invalid request that should not have been sent. Another cheat detection mechanism is using what is proposed above, running a full copy of the game simulation on all clients and sending out a request as a status check to find cheaters with conflicting CRC or similar [1].

## 3.2 Encryption of Sensitive Game Data

Cheaters often find their targets easy prey because data can bee seen in clear text, either between the client and server or internally in the machine under the cheaters control. An approach to reduce this threat is encrypting important data kept in memory or sent between the client and server or between the game engine and a modification (Mod).

There are many encryption schemes useful for online games, like the Advanced Encryption Algorithm [28]. However, such cryptographic algorithms might be to resource consuming. Another solution that might be sufficient is using XOR for important values. That would make it more difficult to e.g. search for known values in memory e.g. for a paused game. Each new patch for a game could include a new XOR key to make a cheaters life more difficult.

Pritchard [1] suggest another kind of "encryption", known as security through obscurity. By dynamically changing the command syntax, hackers will have more trouble finding the desired data in memory. Such solutions are not popular in the academic world, but from a practical approach, everything that increase the pain of the attacker at a low cost for the defender has some clear advantages.

## 3.3 Reduce Client's Information

Some games run a full game simulation locally, enabling cheaters to find information they were not supposed to see, like for Counter-Strike [19]. For that game, United Admins has created a protection called HLGuard [26]. The anti-cheat program has features against the wall hack: it removes objects the client cannot possibly see on a certain map. Thus, it stops a cheater from receiving information about other living players that player was not meant to see.

Similar techniques to reduce the client's information about other objects in a game could be implemented from the beginning of a games lifecycle. The downside is the heavier load on the server to make such decisions.

## 3.4 Make Players be Security Aware

Many cheats could have been avoided if the players were more security aware. Many cheats include some kind of social engineering, like the scams mentioned by Blizzard [15,20], e.g. to get ones unique ID that prove a player bought the game. Game providers need to educate their customers about potential threats and ways to behave, much like banks have done to learn their customers about security related to the use of e.g. VISA card. Players must known what kind of scams or cheats to look out for and be provided with possible solutions, like whom to contact if they believe they lost their unique game ID. Such education may take long time, and must be a continuing process.

## 3.5 Good Password Practice and Management

Passwords are a commonly used security technique to restrict access to e.g. a players account on a server. However, using easy to guess passwords makes such an account easy prey for a cheater. Some users might even choose a password vulnerable to a dictionary attack. Due to usability issues such a think as locking out an account with three wrong password guesses are often not implemented and dictionary attacks are then possible. A proactive password checker is needed to not allow weak passwords. Such a checker might deny all passwords found in a dictionary. It's also necessary to educate the players and guide them when choosing a password, like Blizzard has done [15].

## 3.6 Fair Trading

Trading with virtual items is a little business itself. A researcher estimated that Norrath, a virtual world owned by Sony as part of their game EverQuest, is the 77$^{th}$ richest country in the world [23]. Some companies do not want such trading to occur. But if such activity is accepted, it should also be supported. A common cheat or scam is to advertise something in a virtual world, collect the payment and not give away any virtual items. Having the game provider acting as a trusted third party could assure the buyers to actually get the virtual items they bought.

## 3.7 The Bug Patching Approach

"No developer can fix all bugs before software release. The traditional bug patching approach in security still works here" [1].

## 3.8 An Active Complain-response Channel

A complain channel is necessary to enable players to report new bugs, cheats and possible cheaters. Game providers must provide fast responses to ensure players will continue to use the channel and believe they're taken seriously.

## 3.9 Logging and Audit Trail

Logging and audit trail is necessary to find certain cheats, in particular evidence of new cheats. This is also necessary to deal with the insider threat. Logging and audit trails ensure that the game providers have a better chance of proving someone is cheating.

## 3.10Post-detection Mechanisms

In order to beat the cheaters, some kind of disciplinary punishment is needed. If cheating doesn't result in anything negative for the cheater, they will likely continue cheating. One such punishment is disabling the

user account associated with the cheat. At the same time, those offended by the cheat should have some possibility of restoring e.g. something that was taken from them.

# Conclusion

Cheating in online game is a serious problem with potentially large economic losses for the game industry. Those cheating expose themselves to malicious software, e.g. Trojans [22]. It's clear that implementing measures to mitigate cheating is necessary. Such measures must be seen as a process, starting with the design of the architecture.

# References


1. Matt Pritchard, *How to Hurt the Hackers: The Scoop on the Internet Cheating and How You Can Combat It*, Available: http://www.gamasutra.com/features/20000724/pritchard_01.htm, July 24 2000, Accessed 12-16-2002,
2. Paladin X, *Battlefield 1942 Cheats & Hacks*, Available: http://www.counter-hack.net/content.php?page=bf1942, Accessed 12-12-2002
3. *Battlefield 1942 North American 1.2 Patch*, Available: http://www.ea.com/eagames/official/battlefield1942/editorial/bf1942_12patch.jsp, Accessed 16.12.2002
4. *E-bay*, www.ebay.com, Accessed 12-17-2002
5. *Anarchy Online*, www.anarchyonline.com, Accessed 12-16-2002
6. *Warcraft*, www.warcraft.com, Accessed 12-16-2002
7. *Quake III Arena*, www.idsoftware.com/games/quake/quake3-arena, Accessed 12-16-2002
8. Greenhill, R., *Diablo, and Online Multiplayer Game's Future*, Available:http://www.gamesdomain.com/gdreview/depart/jun97/diablo.html, Accessed 12-17-2002
9. Yan, J. J. & Choi, H. J, *Security Issues in Online Games*, The Electronic Library: international journal for the application of technology in information environments, Vol. 20 No.2, 2002, Emerald, UK, Available: http://www.cl.cam.ac.uk/~jy212/TEL.pdf, Accessed 01-02- 2002
10. Hankyoreh, *Online Cheating is ubiquitous*, Available: http://www.hani.co.kr/section-005100025/2001/05/005100025200105091907004.html, Accessed 12-17-2002
11. *Nextwish*, www.nextwish.org, Accessed 12-16-2002
12. *Cheats & Hacks: Quake III* www.counter-hack.net/content.php?page=quake3, Accessed 01-02-2003
13. *Why Cheating-Death is Different*, Available: www.cheating-death.com/cddiff.htm, Accessed 01-12-2002
14. *Cheats & Hacks: Warcraft 3*, Available: www.counter-hack.net/content.php?page=warcraft3, Accessed 01-02-2003
15. Blizzard Support, *Battle.net Account Password Security*, http://www.blizzard.com/support/?id=asi0505p, Accessed 01-02-2003
16. *Unreal Tournament*, www.unreal.com, Accessed 12-16-2002
17. *Cheats & Hacks: Half-Life & Mods*, Available: www.counter-hack.net/content.php?page=halflife, Accessed 01-03-2003
18. *Counter Hack: Game Specific News: Warcraft III*, Available: http://www.counter-hack.net/index.php?game=warcraft3, Accessed 01-03-2003
19. *Counter-Strike*, www.counter-strike.net, Accessed 01-07-2003
20. Blizzard support, *Simple rules to follow on protecting your CD-Key*, Available: http://www.blizzard.com/support/?id=asi0635p, Accessed 01-03-2003
21. Carter, M., *Hacking of Web game EverQuest Linked to local teen*, 08-31-2001, Available: http://seattletimes.nwsource.com/html/localnews/134335724_hacker31m.html, Accessed 01-07-2003
22. Paris, W. S., *RE: Increase in Sub7 scans*, 06-12-2001, Available: http://online.securityfocus.com/archive/75/190653, Accessed 01-07-2003
23. Dodson, S., *Lord of the ring*, 03-21-2002, Available: http://www.guardian.co.uk/computergames/story/0,11500,670972,00.html, Accessed 01-07-2003
24. Borland, J., *Online game backs away from privacy threat*, 04-05-2000, Available: http://news.com.com/2100-1017-238900.html, Accessed 01-07-2003

25. rq60, *HLGuard aimbot detector*, May 2002, Available: http://forums.unitedadmins.com/showthread.php?s=&threadid=19661, Accessed 01-07-2003
26. *United Admins HLGuard*, Available: http://www.unitedadmins.com/HLGuard.aspx, Accessed 01-07-2003
27. Unreal Playground, *Exclusive Interview with Dr. Sin of Epic Games*, 10-03-2002, Available: http://www.unrealplayground.com/interview.php?id=1, Accessed 12-17-2002
28. FIPS-197 Advanced Encryption Standard, Available: http://csrc.nist.gov/encryption/aes/, Accessed 01-07-2003