

# A Systematic Classification of Cheating in Online Games

Anonymized for review

## Abstract

*Cheating is rampant in current game play on the Internet, as a new major security concern. However, it is not as well understood by security experts as one might expect. In this paper, we first identify common forms of cheating as they have occurred or might occur in online games. We then define a taxonomy of online game cheating with respect to the underlying cause (namely what is exploited?), consequence (what type of security failure can be achieved?) and the cheating principal (who can cheat?). The four traditional aspects of security – confidentiality, integrity, availability and authenticity – are insufficient to explain cheating and its consequences in online games. We argue that fairness is a vital additional aspect, and the problem of its enforcement provides a convincing perspective for understanding the role of security techniques in developing and operating online games.*

## 1 Introduction

While online games are fast becoming one of the most popular applications on the Internet [10], cheating has emerged as a notable phenomenon in current game play on the Internet. Recent research has suggested that cheating is in fact a new, major security concern for online computer games [14, 15, 16]. Therefore, a careful investigation of online cheating can benefit the study of security in this representative Internet application.

However, cheating has not been studied as thoroughly as one might expect. For instance, although online cheating is rampant in games, there is no generally accepted definition for it.

Three reasons may explain this fact. First of all, cheating is a relatively new topic for security re-

searchers, although many online game players have been familiar with it for a considerable time. Second, the variety of online games now in existence has made cheating a complicated phenomenon. For example, there are a number of entirely different game genres, and each may give rise to varied forms of cheating. Third, many novel cheats have been invented that are different from but often entangled with ordinary security attacks.

In this paper, we systematically examine cheating in online games while adopting the following definition for it, which is a refined version of our previous definition used in [15].

*Any behavior that a player uses to gain an advantage or achieve a target in an online game is cheating if, according to the game rules or at the discretion of the game operator (i.e. the game service provider, who is not necessarily the developer of the game), the advantage is unfair to his peer players or the target is one that he is not supposed to have achieved.<sup>1</sup>*

Specifically, we present a classification scheme for online game cheating, in the expectation that by categorizing various online game cheats, our understanding of this phenomenon will be extended, and useful patterns and conclusions can be established, and that it will be possible to protect online game systems against cheating using these knowledge. It is intentionally reminiscent of the dependability taxonomy provided in [8].

Our classification scheme provides a three dimensional taxonomy for online cheating, in which the clas-

---

<sup>1</sup>At present the preponderance of cheating in online games is carried out by male game players, so for linguistic convenience in the rest of this paper we will appear to imply that all cheaters are male.

sification is made with respect to the underlying cause (what is exploited?), cheating consequence (what type of security failure can be achieved?) and cheating principal (who can cheat?) respectively.

Our taxonomy is aimed at being comprehensible and useful to security experts, game developers, operators as well as game players. For example, both security experts and game developers can learn how game systems have failed to prevent cheating, and how they can design their systems that can eliminate or minimize the possibility of being exploited by online cheaters. On the other hand, game operators and players can learn to recognize cheating and manage the risks of encountering cheaters.

Our classification of cheating consequences is derived from the traditional aspects of computer security such as confidentiality, integrity, availability and authenticity. However, we find that these traditional four aspects are insufficient to explain cheating in online games. Fairness is another important perspective in understanding security in applications such as online games. This echoes the result of [16], namely that the most important new security concern in online game design is about fairness enforcement.

This paper extends our previous work in [15, 16], and is organized as follows. Section 2 reviews the related work in this field. In Section 3, we identify common cheating forms as they have occurred or might occur in online games, including cheats that we have ignored. Misconceptions in [15] will also be corrected. Section 4 describes our three dimensional taxonomy. All common cheating forms identified in the previous section are classified using this taxonomy. Section 5 presents some results deduced from our taxonomy, and finally, Section 6 provides some brief concluding remarks.

## 2 Related Work

A number of authors have attempted to define a framework for classifying and understanding online game cheating. For example, Davis [5] categorized traditional forms of casino cheating and discussed their potential counterparts in online games. However, a casino is not representative enough to reflect all forms of online game settings, where cheating may occur with different characteristics.

Pritchard [14] proposed a six-category framework as follows.

- *Reflex augmentation*: exploiting a computer program to replace human reaction to produce superior results in action games
- *Authoritative clients*: exploiting compromised clients to send modified commands to the other honest clients who blindly accept them
- *Information exposure*: exploiting access or visibility to hidden information by compromising client software
- *Compromised servers*: modifying server configurations to get unfair advantages
- *Bugs and design loopholes*: exploiting bugs or design flaws in game software
- *Environmental weaknesses*: exploiting particular hardware or operating conditions

However, this is an ad hoc framework, and a lot of online cheating does not readily fit into any of these six categories.

Author 1 et al [15] reported a more thorough effort, which identified eleven common cheating forms in online games, and structured them as to help security specialists understand the threats underlying online game cheating, as well as to look for countermeasures. In addition, Author 1 [16] thoroughly examined cheating that has occurred or might occur in online Bridge systems, and organized them into a simple framework.

There is also a large amount of literature investigating the definition of taxonomies for security vulnerabilities, attacks or intrusions in a general setting. For example, Landwehr et al constructed a classification of security flaws in software with respect to genesis (how did the flaw enter the system?), time of introduction (when did it enter the system?) and location (where in the system is it manifested?) [7]. Krusl conducted his PhD research on software vulnerability analysis and taxonomy construction [6]. Neumann et al gave a taxonomy of attacks with respect to the technique used to launch a given attack [12]. The

MAFTIA project [3] proposed a taxonomy for intrusion detection systems and attacks. Lindqvist and Jonsson [9] conducted a brief but useful survey on desired properties of a taxonomy, and defined a taxonomy of intrusions with respect to intrusion techniques and results. All these studies are relevant. In online games, a player may cheat by exploiting a “vulnerability”, or by launching an “attack” or “intrusion”. However, as will be discussed later, online game cheating also has some unique manifestations.

### 3 Cheating in Online Games: Common Forms

Before defining our taxonomy, we identify all cheating forms known to us, as they have occurred or might occur in online games.

Eleven common cheating forms were identified in our previous work [15]. While furthering our study on game cheating, however, we have seen the need of refining our previous framework, and now present a revised listing, which classifies cheats into 15 categories. (Those that are new, or are significantly revised version of the categories listed in [15] are marked with asterisks.).

**A:\* Cheating due to Misplaced Trust.** Much cheating involves modifying game code, data, or both on the client side. A cheater can modify his game client program, configuration data, or both, and then replace the old copy with the revised one for future use. Alternatively, the modification or replacement of code and data can be done on the fly.

This form of cheating is really due to misplaced trust. Too much trust is placed on the client side, which in reality cannot be trusted at all because a cheating player can have the total control over his game client. Countermeasures based on security by obscurity approaches such as program obfuscation will eventually fail to fight against this form of cheating, because they try to protect the wrong thing.

**B: Cheating by Collusion.** Players collude to gain unfair advantages. (Representative cases, including various collusion in online Bridge and the

“win trading” collusion in the WarCraft game, were discussed in detail in [16].)

#### C:\* Cheating by Abusing Game Procedure.

This form of cheating may be carried out without any technical sophistication, and a cheater simply abuses the operating procedure of a game. One common case is *escaping*: a cheater disconnects himself from the game system when he is going to lose [15, 16].

Another example is *scoring cheating* [15] in online Go games, which abuses the scoring procedure as follows. When a game is finished, “dead” stones must be identified and then removed by hand before the system can determine which side wins this game. During this scoring process, however, a cheating player may stealthily remove “alive” stones of his opponent, and then “overturn” the game result. (When the size of territory occupied by each side is close, this cheating may easily escape the awareness of the cheated player, especially when he is not a strong player.)

**D: Cheating Related to Virtual Assets** Virtual characters and items acquired in online games can be traded for real money. Lots of cheating related to these virtual assets can then occur.

**E:\* Cheating due to Machine Intelligence.** Artificial intelligence techniques can also be exploited by a cheating player in some online games. For example, the advancement of computer chess research has produced many programs that can compete with human players at the master level. When playing chess online, a cheater can always look for the best candidates for his next move by running a strong computer chess program.

This is in fact cheating due to the superiority, in this particular situation, of machine intelligence over that of an ordinary human being. It can happen in many other online games, depending on two factors: 1) properties of the game: whether the game can be modeled as a computable problem, and 2) the maturity of AI research into such games. For example, online Go players do not worry about this form of cheating, since the state of the art of AI research can produce only very

weak computer Go programs (the strongest one at present can be easily beaten by an amateur human player [11]).

**F:\* Cheating via the Graphics Driver.** By modifying the graphics driver installed in his operating system, a cheating player can make a wall transparent in some online games so that he can see through the wall and locate other players who are supposed to be hidden behind the wall [1].

**G: Cheating by Denying Service to Peer Players.** A cheating player gains advantages by denying service to his peer players. For example, a cheater can delay the responses from one opponent in a real-time game by flooding his network connection. Other peer players will then be cheated into believing that there is something wrong with the network connection of the victim, and agree to kick him out from the game in order to avoid the game session being stalled.

**H:\* Timing Cheating.** In some real-time online games, a cheating player can hold his own move until he knows all the opponents moves, and thus gain a huge advantage [4]. This *look-ahead cheat* is one kind of *timing cheating*.

Other timing cheating includes the *suppress-correct cheat*, which allows a cheater to gain an advantage by purposefully dropping update messages at the “right” time [4].

**I: Cheating by Compromising Passwords.** A password is often the key to much of or all the data and authorization that a player has in an online game system. By compromising a password, a cheater can have access to the data and authorization that the victim has in the game system.

**J: Cheating due to Lack of Secrecy.** When communication packets are exchanged in plain text format, one can cheat by eavesdropping packets and inserting, deleting or modifying game events or commands transmitted over the network.

**K: Cheating due to Lack of Authentication.** If there is no proper mechanism authenticating a game server to clients, a cheater can collect many

ID-password pairs of legitimate players by setting up a bogus game server. Similarly, if there is not a proper mechanism authenticating a client, a cheater can also exploit this to gain advantages. For example, it is critical to re-authenticate a player before any password change is executed for him. Otherwise, when a player leaves his computer temporarily unattended and his game session unclosed – in countries such as China and Korea, many people play online games in internet cafes – a cheater who can physically access the player’s machine may stealthily change his password, and exploit the changed password afterwards.

**L:\* Cheating by Exploiting a Bug or Loophole.** This form of cheating exploits a bug or loophole in game programs, without involving any modification of game code or data. Once discovered, such a bug/loophole will give knowledgeable players a major advantage. The cheat exploiting a farm-stopping bug discussed in [14] is such an example. In fact, the first case of such cheating can be traced back to an old incident occurred in Lucasfilm’s Habitat [2], one of the first multi-user virtual environments.

If a player has to modify the game program or data in order to exploit a bug or design loophole to gain unfair advantages, according to our definition, his cheating behavior will not be covered by this form, but by *cheating due to misplaced trust* or the following form of *cheating by compromising game servers*.

**M:\* Cheating by Compromising Game Servers.** A cheater can tamper game server programs or change their configuration once he has obtained access to the game host systems.

**N: Cheating Related to Internal Misuse.** A game operator usually has the privileges of a system administrator. It is easy for an insider – an employee of the game operator – to abuse this privilege. For example, he can generate super characters by modifying the game database on the server side.

**O: Cheating by Social Engineering.** Often cheaters attempt to trick a player into believing something

attractive or annoying has happened to him and that as a result his ID and password are needed.

### 3.1 Generic vs. Specific Cheats

Table 1 summarizes all cheating forms into two divisions. The “generic” division includes seven forms of common cheating in online games, which are also generic to all network applications but may appear with different names such as “attacks” or “intrusions” in different contexts. The “specific” division includes both cheating specific to online games, and cheating that may also occur with different names in other network applications but has some interesting features or implications in the context of online games.

In fact, some cheating forms even appear to be unique to specific game genres. For example, *cheating due to machine intelligence* is unique to online versions of the traditional board or card games, and *cheating related to virtual assets* has occurred only in multiplayer role-playing games. This can be explained by the unique characteristics of such game genres.

### 3.2 The Non-Atomic Nature of Some Cheats

Although each form included in Table 1 can be an independent cheat, an actual case of cheating may be complex and involve multiple cheating forms. An example is the Pogo cheat discussed in [16]. It is a collusion cheat, which abuses the game procedure, and at the same time also exploits a loophole in the game system design.

Another example is the *hit-then-run* cheat [15] in some Internet Go games, which can occur as follows.

Go is a time critical game played between two people. The Go server counts the time spent by each player in a game, and the player who runs out of his time will automatically lose the game. Many online players choose to play 25 moves in 10 minutes or less time, and it is usual for one to play 5 stones in the last 10 seconds. Therefore, a cheating player can easily defeat one opponent by timing him out with a well timed flooding attack. This is a form of cheating by denying service to peer players.

The above *timeout* cheat can be used together with cheating by abusing the game procedure. Some Internet Go services implemented a penalty rule to fight

against the *escaping* cheat: players who disconnect themselves will lose their unfinished game unless they return to finish it within a limited period. A cheater can take advantage of this rule in the following way. He floods one opponent so that the game is recorded as disconnected by the opponent. Then he does not log on until the penalty period has passed. The game cannot be finished in time, and the opponent will automatically lose points for it.

### 3.3 Coverage

Our refined framework covers all cheating forms known to us. For example, it is easy to redesignate the cheats discussed by Pritchard [14] in our framework.

For instance, the “aiming bot” was an interesting example of “reflex augmentation” cheating discussed in [14]. As a popular cheat in shooting games, an aiming bot worked as a proxy sitting between a game server and a cheater. It tracked the movements and locations of all other players by monitoring packets passed. When the cheater issued a Fire command, the aiming bot would automatically pick a target for him, and then insert a Move/Rotate command packet into the stream going to the server in front of the Fire command packet that pointed the cheater straight at the selected target.

Aiming bots are in fact largely a form of cheating due to misplaced trust. The locations of other players constitute sensitive information in this type of games, so they should not be sent to each client, or at least they should be delivered in a secure way.

Similarly, the “information exposure” cheats discussed by Pritchard are also forms of cheating due to misplaced trust. Furthermore, the cheat exploiting “authoritative clients” can be largely designated as *cheating due to misplaced trust* and *cheating due to lack of secrecy or authentication*. In addition, among the three cheats exploiting “environmental weaknesses”, the first two are in fact *cheating by exploiting a bug or design loophole*, and the third is of *cheating by denying service to peer players*.

## 4 A Taxonomy of Online Cheating

In this section, we define a taxonomy for online game cheating. This is a three dimensional taxon-

| <i>Type</i>              | <i>Label</i> | <i>Cheating Form</i>                            |
|--------------------------|--------------|---|
| Specific to online games | A            | Cheating due to Misplaced Trust                 |
|                          | B            | Cheating by Collusion                           |
|                          | C            | Cheating by Abusing Game Procedure              |
|                          | D            | Cheating Related to Virtual Assets              |
|                          | E            | Cheating due to Machine Intelligence            |
|                          | F            | Cheating via the Graphics Driver                |
|                          | G            | Cheating by Denying Service to Peer Players     |
|                          | H            | Timing Cheating                                 |
| Generic                  | I            | Cheating by Compromising Passwords              |
|                          | J            | Cheating due to Lack of Secrecy                 |
|                          | K            | Cheating due to Lack of Authentication          |
|                          | L            | Cheating by Exploiting a Bug or Design Loophole |
|                          | M            | Cheating by Compromising Game Servers           |
|                          | N            | Cheating Related to Internal Misuse             |
|                          | O            | Cheating by Social Engineering                  |

**Table 1. Common cheating forms in online games**

|                          |                                    |   |
|--------------------------|------------------------------------|---|
| System Design Inadequacy | In the Game System                 | Cheating due to Misplaced Trust                 |
|                          |                                    | Cheating due to Lack of Secrecy                 |
|                          |                                    | Cheating due to Lack of Authentication          |
|                          |                                    | Timing Cheating                                 |
|                          | In the Underlying Systems          | Cheating by Exploiting a Bug or Design Loophole |
|                          |                                    | Cheating by Denying Service to Peer Players     |
|                          |                                    | Cheating via the Graphics Driver                |
|                          |                                    | Cheating by Compromising Game Servers           |
| Operational Failure      |                                    | Cheating by Denying Service to Peer Players     |
|                          |                                    | Cheating by Collusion                           |
|                          |                                    | Cheating Related to Internal Misuse             |
|                          |                                    | Cheating by Abusing Game Procedure              |
|                          |                                    | Cheating by Compromising Passwords              |
|                          |                                    | Cheating by Social Engineering                  |
|                          |                                    | Cheating due to Machine Intelligence            |
|                          | Cheating Related to Virtual Assets |   |

**Table 2. Online game cheating taxonomy: by Cause**

omy, and online cheating is classified by the underlying cause (what is exploited?), the cheating consequence (what type of security failure can be caused?) and the cheating principal (who can cheat?).

Tables 2 – 4 shows the details of the taxonomy by cause, consequence and cheating principal. Note that the same cheating form will appear at least once in each of these categories. Divisions and, where appropriate, subdivisions are provided within the categories; these and their motivations are described in detail later.

#### 4.1 By Cause

Online cheating may or may not exploit system design inadequacies. For example, *cheating by exploiting a bug or loophole* exploits inadequacies in the game design, implementation or both. However, social engineering does not involve exploitation of any technical design inadequacies. Therefore, we classify the causes of online cheating to two divisions: *system design inadequacy* which concerns technical design failure arising in the process of system development, and *operational failure*, which is largely due to failure of human-computer interaction during the operational phase of a game system. (Some operational failures can be ultimately a design failure: they arise due to “the inability to foresee all the situations of the system will be faced with during its operational life, or the refusal to consider some of them” [8] for reasons such as a concern for time-to-market.)

There are two subdivisions in system design inadequacy: *inadequacy in the game system* and *inadequacy in the underlying systems*. Online games are applications running on top of the underlying networking and operating system. A cheater can exploit a flaw in a game system, a flaw in its underlying networking or operating system, or both.

*Cheating due to misplaced trust, lack of secrecy or authentication, timing cheating, cheating by exploiting a bug or design loophole* exploit technical inadequacies in the game system, and they belong to the first subdivision.

Two common cheating forms, namely *cheating via the graphics driver* and *cheating by compromising game servers*, belong to the second subdivision. Specifically, the first cheating form occurs on the game client side. However, rather than exploit the

game system itself, it modifies a system driver that is part of the operating system. Similarly, a cheater compromising a game server usually breaks into the server by exploiting an operating system or network flaw on the server side<sup>2</sup>.

In addition, *cheating by denying service to peer players* usually exploits some inherent weakness of the network layer, but it can also be committed by exploiting a design inadequacy in the game system alone. For example, a cheat having occurred in the Firestorm game [13] exploited a buffer-overflow condition in the game program to disconnect all players. Therefore, this form of cheating is included in both subdivisions.

A lot of cheating techniques in online games, such as collusion, social engineering, game procedural abuse, password compromising, cheating related to internal misuse or virtual assets, are only weakly related to any technical design inadequacy. Instead, they largely exploit “the human side” of computer security [13]. Therefore, they are classified as operational failures.

#### 4.2 By Consequence

We largely base our classification of cheating consequences on the four traditional aspects of computer security: confidentiality (prevention of unauthorized disclosure of information), integrity (prevention of unauthorized modification of information), availability (prevention of unauthorized withholding of information) and authenticity (the ability to assure the identity of a remote user regardless of the user’s host). A breach of confidentiality results in *theft of information or possessions*, a breach of integrity results in *code or data modification*, a breach of availability results in *service denial* and a breach of authenticity results in *masquerade*.

*Cheating by collusion, compromising passwords or social engineering, or cheating due to lack of secrecy* results in theft of information or possessions in a game. *Cheating due to lack of authentication* results in *masquerade*. *Cheating by denying service to peer players* involves selective service denials, but *cheating by compromising game servers, due to misplaced trust,*

---

<sup>2</sup>A game server program may have flaws that can be remotely exploited by a cheater, but we have not yet seen such cases in real life.

|                                     |   |
|-------------------------------------|---|
| Theft of Information or Possessions | Cheating by Collusion                           |
|                                     | Cheating by Compromising Passwords              |
|                                     | Cheating due to Lack of Secrecy                 |
|                                     | Cheating by Social Engineering                  |
| Service Denial                      | Cheating by Denying Service to Peer Players     |
| Code or Data Modification           | Cheating due to Misplaced Trust                 |
|                                     | Cheating via the Graphics Driver                |
|                                     | Cheating by Compromising Game Servers           |
|                                     | Cheating Related to Internal Misuse             |
| Masquerade                          | Cheating due to Lack of Authentication          |
| Fairness Violation                  | Cheating by Abusing Game Procedure              |
|                                     | Timing Cheating                                 |
|                                     | Cheating by Exploiting a Bug or Design Loophole |
|                                     | Cheating Related to Virtual Assets              |
|                                     | Cheating due to Machine Intelligence            |

**Table 3. Online game cheating taxonomy: by Consequence**

or related to internal misuse usually involves integrity failure.

However, these traditional aspects of computer security are insufficient to cover all the consequences of online game cheating. For example, the cheat exploiting the farm-stopping bug in [14] violated none of the issues of confidentiality, availability, integrity or authenticity. And the list goes on.

We introduce “fairness” between peer players as an additional aspect for understanding online game cheating, and a breach of fairness results in a *fairness violation*. Either *cheating by abusing game procedure*, *timing cheating*, *cheating by exploiting a bug or design loophole*, or *cheating due to machine intelligence* can result in *fairness violation*. Although *cheating related to virtual assets* may result in theft of possessions, it is hardly the result of confidentiality failure. Therefore, *cheating related to virtual assets* is also categorized as a *fairness violation*.

### 4.3 By Cheating Principal

A player can cheat independently either in single player or multi-player online games, whereas in multi-player games two or more players can cheat via malicious cooperation. Furthermore, a player can also collude with an insider to cheat. The identity of the cheating principal is used as the third dimension in our

classifications, and it provides a way of distinguishing cooperative cheats from their independent counterparts.

Regarding the cheating principal, there are three divisions: by *player*, by *game operator* and by *operator-player* (i.e. the cooperation of player and game operator).

The division of by operator-player accommodates cheating committed through the cooperation of a player and an insider, which typically involves collusion as well as internal misuse that are specific to the game.

The division of by game operator accommodates cheating related to internal misuse, where no collusion between player and insider is involved, however. One example is that of an insider who is also a player. As discussed in [16], house cheating orchestrated by a game operator alone is likely to occur. However, it is beyond the scope of our online cheating definition used in this paper.

There are two subdivisions in the category cheating by player, namely by *single player* or by *multiple players*. Collusion between players is covered by the second subdivision, whereas, as indicated in Table 4, 13 other cheating forms belong to the first subdivision.

|                 |   |   |
|-----------------|---|---|
| Player          | Single Player   | Cheating due to Misplaced Trust                 |
|                 |   | Cheating by Abusing Game Procedure              |
|                 |   | Cheating Related to Virtual Assets              |
|                 |   | Cheating by Compromising Passwords              |
|                 |   | Cheating by Denying Service to Peer Players     |
|                 |   | Cheating due to Lack of Secrecy                 |
|                 |   | Cheating due to Lack of Authentication          |
|                 |   | Timing Cheating                                 |
|                 |   | Cheating by Exploiting a Bug or Design Loophole |
|                 |   | Cheating by Compromising Game Servers           |
|                 |   | Cheating by Social Engineering                  |
|                 |   | Cheating due to Machine Intelligence            |
|                 |   | Cheating via the Graphics Driver                |
|                 |   | Multiple Players                                |
| Game Operator   | Cheating Related to Internal Misuse (No collusion involved) |   |
| Operator-Player | Cheating Related to Internal Misuse (Collusion involved)    |   |

**Table 4. Online game cheating taxonomy: by Cheating Principal**

|   | Info Theft | Service Denial | Code or Data Modification | Masquerade | Fairness Violation |
|---|------------|----------------|---------------------------|------------|--------------------|
| Design inadequacy in the game system        | J          | G              | A                         | K          | E, H, L            |
| Design inadequacy in the underlying systems |            | G              | F, M                      |            |                    |
| Operational failure                         | B, I, O    |                | N                         |            | C, D               |

**Table 5. Distribution of cheating forms in the cause-consequence matrix**

## 5 Discussion

Our taxonomy brings out a systematic view of online cheating, from which a number of observations can be made.

First, it is interesting to examine the distribution of each common cheating form in the two orthogonal dimensions of causes and consequences.

Table 5 constructs such a distribution matrix, where the cheating cause and consequence are displayed in rows and columns respectively, and cheating forms in the cells are represented with their labels assigned in Section 3. The matrix in Table 5 shows that most types of online game cheats have been about information theft, code or data modification, or fairness violation, and they largely exploit either operational failures or flaws in the game systems.

However, the distribution of cheating forms in the cause-consequence matrix may not be stationary while online game and the cheating phenomenon coevolve. Therefore, any observation based exclusively on this matrix may have to remain tentative. For example, it is not yet clear whether cheats exploiting the flaws in the underlying networking and operating systems will increase in the future.

Second, as the classification by cheating principal in Table 4 shows, the majority of current game cheating can be committed by a single player independently, although some others involve collusion between one and his peer player(s) or an insider. Similarly due to the above reason, this observation also remains tentative.

Third, re-examining the taxonomy by consequence in Table 3, in fact, no matter whether a cheating form results in either information theft, service denial, code or data modification, or masquerade, a fairness violation is caused and it gains a cheater some advantages over his peer players in the game. Therefore, the perspective of fairness appears to be essential in understanding security in applications such as online games. This echoes the result of [16] and can be easily explained as follows. On the one hand, fair play is essential to any game. Online gaming is not an exception, and fairness should be an inherent concern in its design. On the other hand, online players usually do not know each other, and they are often scattered in different physical locations. Therefore, the social structures preventing cheating in the non-electronic world are no

longer in place for online games. It is security that can provide an alternative mechanism for fairness enforcement.

Nonetheless, some game cheating problems, such as collusion in online Bridge [16], cannot be solved by security techniques alone. Instead, other technologies such as artificial intelligence can also contribute. Therefore, security plays an important but non-exclusive role in enforcing the fair play in online games.

In addition, it is also interesting to note that as a result of taxonomic analysis using Table 5, we have corrected a mistake in a previous version of this paper. Namely, we found that we carelessly missed a type of cheating by denying service to peer players, which involves exploitation of design inadequacies in the game system only.

It appears that we can also use this table to suggest novel additional forms of cheating that will likely occur in the future while arguing why some blank squares in the table are and will remain empty. For example, it appears that cheating leading to service denial due to operational failure will never occur, since seemingly there is no other way to deny peer players to service rather than exploit technical design inadequacies in the game system, the underlying systems, or both. However, it is very likely for cheats, which lead to masquerade, information theft or fairness violation and are due to design inadequacies in the underlying systems, to occur in the future, although it is not yet clear in which forms they will be manifested.

## 6 Conclusion

Online games open themselves to a wide spectrum of cheating, in addition to those found as “attacks” or “intrusions” in other networked applications. We have presented a classification scheme for online game cheating, in which the classification is made with respect to the underlying causes, consequences and the cheating principals. This scheme is intended to be comprehensible and useful not only to security specialists, but also to game developers, operators and players who are less knowledgeable and experienced in security. Although there is room for further refinement, we offer it as a candidate for a general taxonomy of cheating in online gaming, a fast-growing represen-

tative Internet application.

## References

- [1] Christopher Choo, "Understanding Cheating in Counterstrike", Nov. 2001. Available at <http://www.fragnetics.com/articles/cscheat/print.html>.
- [2] C Morningstar and FR Farmer, "The Lessons of Lucasfilm's Habitat", in *Cyberspace: First Steps*, M Benedikt (ed.), MIT Press, Cambridge, 1990.
- [3] D Alessandri (ed.), "Towards a Taxonomy of Intrusion Detection Systems and Attacks", MAFTIA deliverable D3, Version 1.01, September 6, 2001. Available at <http://www.newcastle.research.ec.org/maftia/deliverables/D3.pdf>.
- [4] N Baughman and B Levine. "Cheat-proof Play-out for Centralized and Distributed Online Games", in *Proc. of the Twentieth IEEE INFOCOM Conference*, Apr. 2001.
- [5] SB Davis, "Why Cheating Matters: Cheating, Game Security, and the Future of Global On-line Gaming Business", in *Proc. of Game Developer Conference 2001*, 2001.
- [6] IV Krsul, "Software Vulnerability Analysis", Ph.D. Thesis, Purdue University, Computer Sciences Department, 1998.
- [7] CE Landwehr, AR Bull, JP McDermott and WS Choi, "A taxonomy of computer program security flaws", *ACM Computing Surveys*, Vol.26 No.3, Sept. 1994. pp211-254.
- [8] JC Laprie (ed.), *Dependability: Basic Concepts and Terminology*, Springer-Verlag, Vienna, 1992.
- [9] U Lindqvist and E Jonsson, "How to Systematically Classify Computer Security Intrusions", in *Proceedings of the 1997 IEEE Symposium on Security & Privacy*, Oakland, California, May 4-7, 1997. IEEE Computer Society Press. pp154-163.
- [10] S McCreary and K Claffy, "Trends in Wide Area IP Traffic Patterns: A View from Ames Internet Exchange", in *Proceedings of the ITC Specialist Seminar on IP Traffic Modeling, Measurement and Management*, Monterey, CA, USA, Sept. 2000.
- [11] M Müller, "Computer Go", *Artificial Intelligence*, Vol.134, No.1-2 (Special issue on Games, Computers and AI), January 2002, pp145-179.
- [12] PG Neumann and DB Parker, "A Summary of Computer Misuse Techniques", in *Proc. of the 12th National Computer Security Conference*, Baltimore, MD, 1989, pp. 396-407.
- [13] K Poulsen, "Mitnick to Lawmakers: People, Phones are Weakest Links", *SecurityFocus.com News*, March 2000. Available at <http://www.politechbot.com/p-00969.html>.
- [14] M Pritchard, "How to Hurt the Hackers: The Scoop on Internet Cheating and How You Can Combat It", *Information Security Bulletin*, February 2001.
- [15] Author 1 et al, "Security Issues in Online Games", *The Electronic Library*, Vol. 20, No.2, 2002. A previous version appears in *Proc. of International Conference on Application and Development of Computer Games*, City University of Hong Kong, Nov. 2001.
- [16] Author 1, "Security Design in Online Games", in *Proc. of the 19th Annual Computer Security Applications Conference*, IEEE Computer Society, December, 2003.