

Network security isn't all fun and games: An analysis of information transmitted while playing Team Fortress 2

Brett Turner and Andrew Woodward
Secau
Edith Cowan University
b.turner@ecu.edu.au

Abstract

In the world of online gaming, information is exchanged as a matter of course. What information is exchanged behind the scenes is something that is not obvious to the casual user. People who play these games trust that the applications they are using are securely written and in this case, communicate securely. This paper looks at the traffic that is transmitted by the game Team Fortress 2 and incidentally the supporting authentication traffic of the Steam network. It was discovered through packet analysis that there is quite a lot of information which should be kept private being broadcast in the clear. Information discovered as a result of traffic capture and analysis included users IDs, and of greater concern, the remote console password. While this information may seem trivial, discovery of such information may lead to compromise of the game server, leaving it open to be controlled by someone with malicious intent.

Keywords

Network security, Team Fortress 2, packet analysis

INTRODUCTION

Team Fortress 2 (TF2) is a popular online, multiplayer game created by VALVe Software and distributed over their content delivery system, Steam (VALVe, 2008a). While TF2, currently with over 14,000 concurrent players today is not as popular as some of their other Steam offerings, such as Counter Strike: Source (CSS), with a peak today of over 95,000 concurrent players, it is still a significant number of users and a rich target vector. By any stretch of the imagination, Steam is a successful platform with currently over 1.3 million concurrent users at the time of writing (VALVe, 2008a).

TF2 is played over a network, most commonly over the Internet and uses the internet protocol (IP) suite (TCP/IP). Public servers are hosted by many different communities and can be open to the public or kept private through the use of passwords (Steam_Support). These games require Steam to authenticate with the online authentication servers before they can be played. Although a high speed network connection is required for full game-play, anyone can download a TF2 client for either Linux or Windows for free and setup their own internet server that anyone can connect to (VALVe 2008b).

Although the idea of playing games or investigating game security may seem frivolous at first glance, there are some very real security implications if the traffic exchanged between client and server is not secure. For example, it is common for commercial organisations that host game servers to site multiple instances on the one physical server (Gameservers, 2008). As a consequence of this practice of server hosting, it is necessary to give game server administrators remote access to the server. It is also common practice to make multiple regular users of a game server what is known as administrator privileges to allow them to reset the game, or to evict players who do not comply with any stated rules or regulations. Password reuse, ie the practice of using the same password for multiple systems, is a significant security issue (Ives *et al*, 2004). If the password is sent in clear text by the game, this does present a risk to network security, as it may be used to compromise a server, and not just the game service.

There is also a financial incentive to want to hijack a users account, or to steal their online identity. There have been numerous reports of virtual property theft which resulted in real financial loss (Chen *et al*, 2004).

It is the aim of this paper to analyse the IP traffic involved in the playing of TF and investigate the possible impacts of the contents of this traffic on network security. The traffic will also be investigated to determine whether there is any information that may allow for identity theft or other such malicious activity.

METHOD

The process Steam games go through to enable game play requires communication between many servers other than just the client and game server. To ensure all authentication and game data was captured the following scenario was devised. A server running the Vista Service Pack 1 was used as a platform. A dedicated Windows version of Team Fortress 2 server, which is based on the HalfLife dedicated server (HLDS), was then installed and configured for internet use. Wireshark was installed on both client and server and was used to capture the network data. The data capture was performed on the client during the Steam authentication process and then again once TF2 was started until the game was finished. The server data was captured from when the TF2 server was started until when the process was terminated. This allowed all traffic from all hosts, including third party hosts, associated with playing TF2 to be captured. The client procedure was then replicated on a separate, live machine belonging to a genuine player of TF2 on Windows XP Service Pack 2. This second client connected to a live internet based server while wireshark was again used to capture the network data so the results could be validated.

RESULTS

The obvious information that can be seen in the packet headers is the IP addresses of the hosts involved (Figure 1). This isn't anything surprising but it does provide a target. Once the Steam client is started, initial communications are established. At this point something a little more interesting happens in that the host name of the computer being used is sent in clear text (Figure 2). This is all common information that could be gleaned after listening to a host for long enough but could be used for careful selection of a target.

2080	23.251284	72.165.61.185	10.77.25.128
UDP	Source port: 27017	Destination port: 50910	

Figure 1: Packet header captured during authentication process.

0000	00 00 5e 00 01 03 00 1a 6b 4e 25 ef 08 00 45 00	..^.....kN%...E.
0010	00 4b 4f 54 00 00 80 11 00 00 0a 4d 19 80 8b e6	.KOT.....M....
0020	52 3c fd d1 00 35 00 37 02 38 31 23 01 00 00 01	R<...5.7.81#....
0030	00 00 00 00 00 00 0e 63 73 30 30 31 61 36 62 34cs001a6b4
0040	65 32 35 65 66 03 61 64 73 03 65 63 75 03 65 64ed
0050	75 02 61 75 00 00 01 00 01	u.au.....

Figure 2: Packet data payload containing hostname sent during authentication process

Once a connection is established, the authentication process begins. While these packets are encrypted they appear to be easily identifiable by the first 4 bytes of the data payload, VS01, which is highlighted in yellow in the illustrated captured packet (Figure 3.) These packets are passed not only during the authentication process but also all the while the game is being played. While they are readily identifiable by their data header and the host they are associated with IP address 69.28.145.172. While this IP address does not resolve to a domain name, whois reports this back as a limelight server belonging to and with a technical contact email at VALVE Software (Figure 4.) Little else is obvious in these packets.

0000	00 1a 6b 4e 25 ef 00 80 2d 29 f2 08 08 00 45 00	..kN%...-)...E.
0010	00 48 63 4e 00 00 6d 11 40 2c 48 a5 3d b9 0a 4d	.HcN..m.@,H.=..M
0020	19 80 69 89 c6 de 00 34 c0 13 56 53 30 31 08 00	..i....4...VS01..
0030	02 00 00 00 00 00 00 02 00 00 01 00 00 00 01 00
0040	00 00 00 00 00 00 00 00 00 00 00 00 00 00 8a 61a
0050	3e f7 09 00 00 00	>.....

Figure 3: Authentication data packet beginning with the "VS01" double word.

What is surprising is that the steam user name, the name used to identify the legitimate user (as opposed to the user's screen name or tag) is sent in clear text (Figure 4). The server this is sent to resolves to a Qwest server which whois once again clearly identifies as belonging to and operated by VALVe Software, located in Washington State, USA. (Figure 5).

```

1215 13.216494 10.77.25.128 72.165.61.141 TCP 53733 > 27039
[PSH, ACK] Seq=14 Ack=6 Win=65692 [TCP CHECKSUM INCORRECT] Len=35

0000 00 00 5e 00 01 03 00 1a 6b 4e 25 ef 08 00 45 00 ..^.....kN%...E.
0010 00 4b 51 56 40 00 80 06 00 00 0a 4d 19 80 48 a5 .KQV@.....M..H.
0020 3d 8d d1 e5 69 9f 1a 84 87 e2 de 73 ff ba 50 18 =...i.....s..P.
0030 40 27 aa 3c 00 00 00 00 00 1f 02 00 0d 73 73 30 @'.<.....ss0
0040 31 38 38 30 30 31 70 63 32 39 00 0d 73 73 30 31 188.....ss01
0050 38 38 30 30 31 70 63 32 39 88.....

```

Figure 4: Steam user name passed to VALVe server in clear text.

```

Whois: 72.165.61.141

OrgName: VALVE CORPORATION
OrgID: VALVE-2
Address: 6101 S. 180TH
        STREET
City: TUKWILA
StateProv: WA
PostalCode: 98188
Country: US
NetRange: 72.165.61.128
        - 72.165.61.191

CIDR:
72.165.61.128/26

```

Figure 5: whois data

Once you load up TF2 and enter the server browser, your “favourites” list is updated. Every server you have in any of your steam “favourite” lists is updated in clear text. This means every favourite list from every steam game you have played is updated despite the fact you are playing TF2. From this information it can be seen what game preferences the user has other than simply TF2 (Figure 6). In this case we can see from the illustrated captured packets that this user also plays CSS (Figure 7) and the Left 4 Dead demo (Figure 8). Apart from the IP address of the server the user plays on in the packet header, the server name can be seen in red, the current map name in green, the steam game name in cyan, the game long name in yellow and the server flags in blue.

```

0000 00 11 11 75 94 da 00 04 ed 38 09 c4 08 00 45 00 ...u.....8....E.
0010 00 8c 00 00 40 00 3b 11 49 86 ca 48 bf 8b ca 48 ....@.;.I..H...H
0020 a1 be 69 87 12 04 00 78 e9 74 ff ff ff ff 49 0e ..i.....x.t....I.
0030 33 46 4c 20 57 41 20 2d 20 54 46 32 20 23 30 31 3FL, WA - TF2 #01
0040 20 32 66 6f 72 74 20 6f 6e 6c 79 20 2d 20 49 6e 2fort only - In
0050 73 74 61 53 70 61 77 6e 21 00 63 74 66 5f 32 66 staSpawn!.ctf 2f
0060 6f 72 74 00 74 66 00 54 65 61 6d 20 46 6f 72 74 ort.tf.Team Fort
0070 72 65 73 73 00 b8 01 00 18 00 64 6c 00 01 31 2e ress.....dl..1.
0080 30 2e 34 2e 32 00 a0 87 69 63 74 66 2c 72 65 73 0.4.2...ictf,res
0090 70 61 77 6e 74 69 6d 65 73 00 pawntimes.

```

Figure 6: Team Fortress 2 “favourite” server update packet

```

0000 00 11 11 75 94 da 00 04 ed 38 09 c4 08 00 45 00 ...u.....8....E.
0010 00 80 00 00 40 00 3c 11 a1 f5 cb 18 65 58 ca 48 ....@.<.....eX.H
0020 a1 be 69 87 12 04 00 6c 60 47 ff ff ff ff 49 07 ..i.....l`G....I.
0030 45 47 4e 20 43 53 53 23 30 38 20 5b 44 75 73 74 EGN CSS#08 [Dust
0040 32 5d 20 52 61 6e 6b 65 64 20 28 31 30 30 54 69 2] Ranked (100Ti
0050 63 6b 29 00 64 65 5f 64 75 73 74 32 00 63 73 74 ck).de_dust2.cst
0060 72 69 6b 65 00 43 6f 75 6e 74 65 72 2d 53 74 72 rike.Counter-Str
0070 69 6b 65 3a 20 53 6f 75 72 63 65 00 f0 00 18 18 ike: Source.....
0080 00 64 6c 00 01 31 2e 30 2e 30 2e 33 34 00 .dl..1.0.0.34.

```

Figure 7: CSS “favourite” server update packet

```

0000 00 11 11 75 94 da 00 04 ed 38 09 c4 08 00 45 00 ...u.....8....E.
0010 00 cf 75 2c 00 00 7c 11 2c 75 cb 18 65 5d ca 48 ..u,..|. ,u.e].H
0020 a1 be 69 87 12 04 00 bb fe 59 ff ff ff ff 49 24 ..i.....Y....I$
0030 45 47 4e 20 57 41 20 4c 34 44 23 32 20 5b 65 67 EGN WA L4D#2 [eg
0040 6e 2e 63 6f 6d 2e 61 75 5d 00 6c 34 64 5f 64 65 n.com.au].l4d de
0050 6d 5f 68 6f 73 70 69 74 61 6c 30 31 5f 61 70 61 m hospital01 apa
0060 72 74 6d 65 6e 74 00 6c 65 66 74 34 64 65 61 64 rtment.left4dead
0070 00 4c 34 44 20 2d 20 43 6f 2d 6f 70 20 2d 20 48 .L4D - Co-op - H
0080 61 72 64 00 12 02 00 04 00 64 77 00 01 31 2e 30 ard.....dw..1.0
0090 2e 30 2e 30 00 a0 87 69 65 6d 70 74 79 2c 72 65 .0.0...iempty,re
00a0 73 65 72 76 61 62 6c 65 2c 6b 65 79 33 36 2c 53 servable,key36,S
00b0 65 72 76 65 72 20 42 72 6f 77 73 65 72 20 4a 6f erver Browser Jo
00c0 69 6e 20 45 6e 61 62 6c 65 64 2c 73 76 5f 73 65 in Enabled,sv se
00d0 61 72 63 68 5f 6b 65 79 5f 33 36 2c 00 arch key 36, .

```

Figure 8: Left 4 Dead “favourite” server update packet

When joining a server, part of the information transmitted in clear text is the users screen name or tag and the contents of the password variable (Figure 9). This is the field used to access password restricted servers. More interesting is the remote console (rcon) password. The rcon password is used to authenticate a user authorized to control and issue commands to a remote game server. This password is also sent in clear text preceding any rcon command. In the illustrated packets (Figure 10) the rcon password for the test server, wibble, is seen in the packet immediately preceding the issued rcon command. (Steam_Support) In this case the command was a simple status command. In game the data is either encrypted or in raw binary as it makes no readily apparent sense when viewed as ASCII. When the test runs were done, the user used the in game chat and team chat to leave easily identifiable messages. The packets were searched for these text strings but were not found in any

```

0000 00 04 ed 38 09 c4 00 11 11 75 94 da 08 00 45 00 ...8.....u....E.
0010 03 1c a3 01 00 00 80 11 9e f4 ca 48 a1 be ca 48 .....H...H
0020 bf 8b 69 7d 69 87 03 08 cf d7 ff ff ff ff 6b 0e ..i}i.....k.
0030 00 00 00 03 00 00 00 ae e8 20 0f 54 4f 47 20 7c .....TOG |
0040 20 4e 69 67 68 74 66 69 72 65 00 69 72 6f 6e 76 N.....e.ironv
0050 6f 6c 74 00 d4 02 2a 00 00 00 14 00 00 00 f0 fa olt...*.....
0060 61 25 90 58 16 72 66 ff 02 00 01 00 10 01 9d 10 a%.X.rf.....
0070 15 49 00 01 00 80 b5 e0 3b 55 1d d6 01 77 15 35 .I.....;U...w.5

```

Figure 9: Screen name and server password variable being passed to the server.

```

0000 00 1a 6b 4e 25 91 00 1a 6b 4e 25 ef 08 00 45 00 ..kN%...kN%...E.
0010 00 3d 0a f8 40 00 80 06 a8 22 0a 4d 19 80 0a 4d .=...@....".M...M
0020 19 87 c6 44 69 87 0a a2 3b 41 dc f5 23 ee 50 18 ...Di...;A..#.P.
0030 40 29 54 f7 00 00 11 00 00 00 02 00 00 00 03 00 @)T.....
0040 00 00 77 69 62 62 6c 65 00 32 00 ..wibble.2.

0000 00 1a 6b 4e 25 91 00 1a 6b 4e 25 ef 08 00 45 00 ..kN%...kN%...E.
0010 00 3c 0a f9 40 00 80 06 a8 22 0a 4d 19 80 0a 4d .<...@....".M...M
0020 19 87 c6 44 69 87 0a a2 3b 56 dc f5 23 ee 50 18 ...Di...;V..#.P.
0030 40 29 54 ea 00 00 10 00 00 00 00 00 00 00 02 00 @)T.....
0040 00 00 73 74 61 74 75 73 00 00 ..status..

```

Figure 10: rcon password and rcon “status” command being passed in clear text.

clear text form.

Once the game was done and the process exited steam had one more set of useful clear text to offer. This final packet is uploaded from the client to 67.132.200.140. This IP address resolves to 67-132-200-140.valvesoftware.com, a server once again on Qwest’s network and owned by VALVe Software. In this packet (Figure 11) we can see the srcid that corresponds to the PseudoUUID registry key, both marked in yellow. The CPU type and speed is marked in red, the graphics card type in blue, the graphics card driver version in green and the DirectX version in magenta. Finally the Steam application ID, coloured in dark red, which corresponds to the Apps registry key (Steam_Support, 2008).

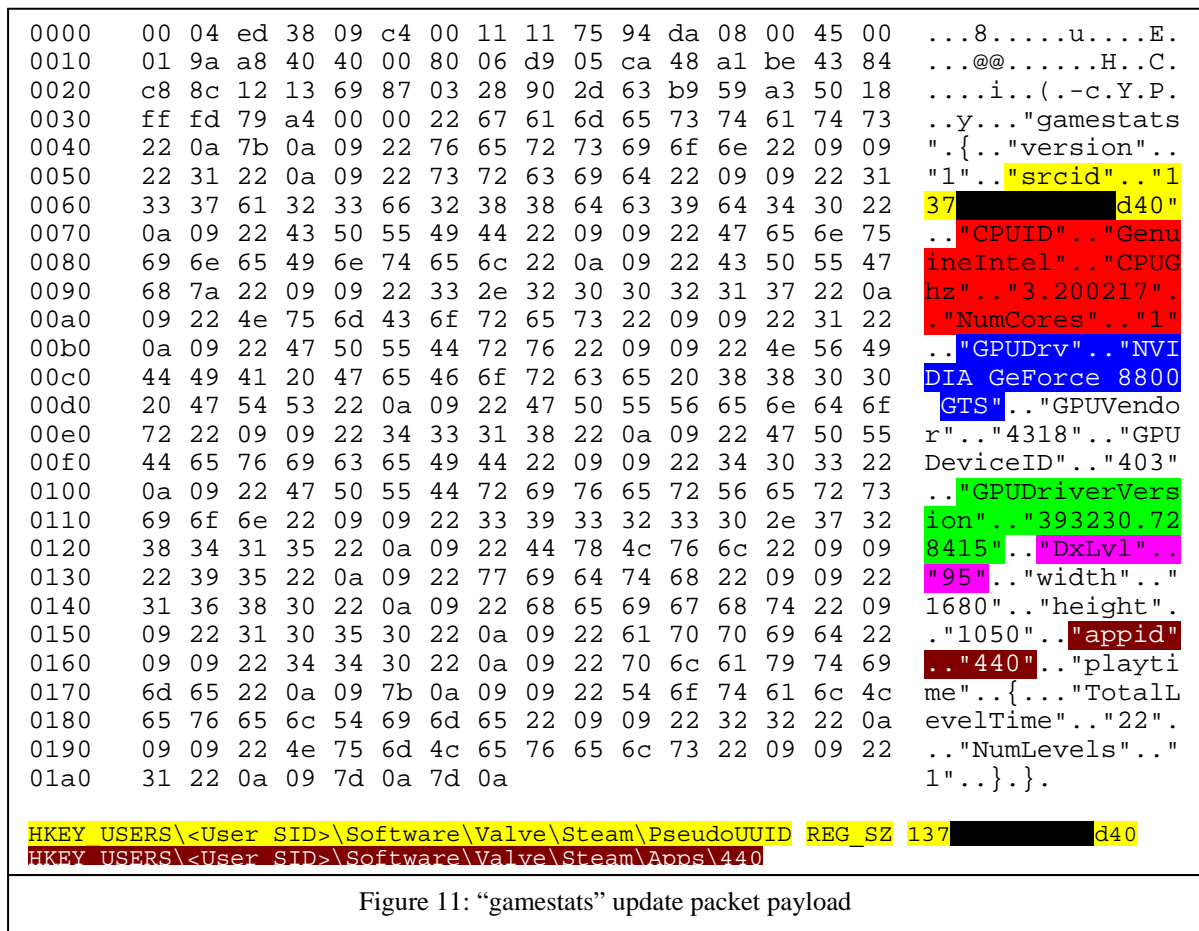


Figure 11: "gamestats" update packet payload

DISCUSSION

Whilst game designers certainly take security into consideration security from the perspective of preventing cheating (Knutson *et al*, 2004), there appears to be little consideration given to overall network security in the form of protecting the exchange between client and server.

The data that is transmitted through the Internet while playing TF2, viewed individually may seem trivial. When they are put together however, there is a significant amount of personal information that can be gathered. A source IP address and a host name by themselves appear harmless. However if the names returned by a DNS query on the source IP address don't match the hostname in the TF2 packets, this may be an indicator that the host the user is playing the game on is behind a NAT device.

The server browser automatically updating the user's favourite servers in all their Steam games may seem to be pure convenience, but this provides a profile of the user's activities and preferences. It also provides avenues and opportunities for further intelligence gathering on hosts that the user has been known to frequent and is likely to visit again. It is possible that one of the user's favourite servers is easier to compromise, providing possible attack vectors.

Sending the user's screen name is not significant since this is easily found in game and is simple to copy. There is nothing preventing another user taking an already used tag or even impersonating the original user of that screen name. Sending the password variable contents in clear text is unusual though. This would allow a malicious user to join password protected servers that they would otherwise not have access to. This does not allow control over anything but can allow a "griefer" to impersonate a player and then proceed to destroy the user's reputation or generally cause annoyance. A griefer is a player who derives enjoyment from causing distress, annoyance or the disruption of other people who are playing a game (Lin & Sun, 2005). There are a large number of players that for a variety of reasons, their on-line persona is important (Yee, 2006). For them to lose that persona through someone stealing it and having it banned, or have it defamed through someone else using it, can cause psychological distress (Wan & Chiou, 2006).

Of more concern is the rcon password. This allows a player to send commands to the game process and otherwise control it as if they were on the machine and owned it. This ranges from changing maps, changing environment settings through to anything the process will allow them to otherwise do. Further research could be

done into determine if the HLDS process (the executable that runs the TF2 game server process) allows shell access. If so, this could be a potentially serious compromise vector on the part of the server.

By themselves these pieces have varying points of significance; together they reveal a good deal of information. The more information that can be gathered, the better an attempt at internet identity theft can be made, as has been seen already in Steam games (sgtbane, 2007). A simple impersonation could be accomplished by assuming the user's screen name frequenting the user's known servers. Armed with passwords, knowledge of personal computer specifications and the steam account name, this would provide someone familiar with this social environment with a very good opportunity to pass themselves off as the impersonated user. If this is achieved, close friends of the user may reveal information as a result of carefully crafted stories (such as an account password).

The final packet sent to 67-132-200-140.valvesoftware.com after TF2 has exited contains some interesting information. It is obviously data mining on the part of VALVe, determining the general hardware and software environment the game is running on, the game that was played and how long was spent playing it. Some of these statistics can be seen in the player's Steam profile. To an individual of malign intent, this could be valuable intelligence. Specific hardware details could well help evaluate a target's desirability, or point to specific vulnerabilities. Specific driver versions made by particular manufacturers could be vulnerable and expose specific attack vectors. (Rapid7, 2006) The srcid is a semi-unique ID that appears to be unique from installation to installation. This could have some relation to authorization to use certain software.

Finally the Steam ID, transmitted in clear text. This is one half of the authentication process. With one half of the puzzle solved all that is needed from here is a password and this user could lose their steam account. Since a steam account is free to begin with this may not seem like a large issue to some, until one realises that Steam is a digital content delivery system. The Steam account is how VALVe associates a user with valid purchases and authorization to use specific products. Once this is compromised, it can be difficult to reclaim. In the case of a stolen account being used to cheat and subsequently banned by the VALVe Anti Cheat system (VAC), the account will permanently be banned from VAC public servers for that product (Steam_Support, 2007). Of greater concern from a network security point of view is that if the game server traffic is sending passwords clear text, thus revealing a password used by a user of the game system, capture of this password may allow for privilege escalation and compromise of the game server. Further issues arising from the compromise of a server are that it could be used to launch spam or even coordinate a DDoS attack.

CONCLUSION

While the difficulty of intercepting this sort of traffic should be acknowledged, it should also be noted that it is far from impossible. Any number of techniques could be used to gain this information from Trojan viruses on the users own machine to compromised network infrastructure through which the traffic passes. It is obvious that some of this information is already obscured in some aspect since the Steam account password and the in game chat messages were not found in clear text in the data captures. The significance of the information that can be seen in clear text should not be underestimated. A general, unsuspecting user will have no knowledge of the exact data that leaves their computer while they play TF2 and trusts that it is appropriately secured. While a user might have that trust and believe trapping data off the wire is too difficult a task for anyone to either do or do to them, it does not change the fact that it can be done.

As discussed, what is easily readable can easily lead to several malicious outcomes such as griefing, identity theft, stolen Steam accounts and even the compromise of a server should an rcon password be used. Being aware of these risks allows an individual to take appropriate measures such as changing passwords on a regular basis. If an individual has control of a server, they may be able to protect themselves by adding further layers of defence such as VPN encryption but this is unfeasible for playing on general public servers. The only entity that can definitively address this issue is VALVe Software. Ultimately VALVe must address this issue or the average user will remain at risk. Since most rcon users are general users many will not even be aware of what measures they could be taking, let alone should.

Since this is the first in a proposed series of game traffic analysis, it is hard to say if VALVe are doing anything better or worse than the general game community, or if the security they have in place is standard for the game development community. All that can be said is that in general, Steam traffic is reasonably secure but there is room for improvement and most definitely a reason to be concerned.

REFERENCES

- Chen, Y.-C., Chen, P., Song, R., & Korba, L. (2004). Online Gaming Crime and Security Issue - Cases and Countermeasures from Taiwan. In Proceedings of the 2nd Annual Conference on Privacy, Security and Trust. Fredericton, New Brunswick, Canada. October 13-15, 2004

- Gameservers (2008). Dedicated servers – game servers. Retrieved 10th June 2008 from <http://www.gameservers.com/dedicated/>
- Ives, B., Walsh, K.R. & Schneider, H. (2004). The domino effect of password reuse. *Communications of the ACM*. **47(4)**: pp75-78
- Knutsson, B. , Honghui L, Wei X. & Hopkins, B. (2004). Peer-to-peer support for massively multiplayer games. *In* INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies
- Lin, H., & Sun, C.-T. (2005). *The 'White-eyed' Player Culture: Grief Play and Construction of Deviance in MMORPGs*. Paper presented at the DiGRA 2005.
- Rapid7. (2006). Buffer Overflow in NVIDIA Binary Graphics Driver For Linux. Retrieved November 10, 2008, from <http://www.rapid7.com/advisories/R7-0025.jsp>
- sgtbane. (2007). Thiefs. Retrieved November 10, from <http://forums.steampowered.com/forums/showthread.php?t=610982&highlight=Identity+Theft>
- Steam_Support. Setting up a Steam Source Dedicated Server. Retrieved November 11, 2008, from https://support.steampowered.com/kb_article.php?ref=7017-UJBN-6785
- Steam_Support. (2007). Reclaiming a Hijacked Steam Account 2008, from https://support.steampowered.com/kb_article.php?ref=2347-QDFN-4366
- Steam_Support. (2008). Application ID's for Steam Games. Retrieved November 10, 2008, from https://support.steampowered.com/kb_article.php?ref=3729-WFJZ-4175
- VALVe. (2008a). Steam & Game Stats. Retrieved November 11, 2008, from <http://store.steampowered.com/stats/>
- VALVe. (2008b) Game Servers Overview. Retrieved 11th November 2008 from https://support.steampowered.com/kb_article.php?ref=5410-USFG-9239#types
- Wan, CS. & Chiou, WB. (2006). Psychological Motives and Online Games Addiction: A Test of Flow Theory and Humanistic Needs Theory for Taiwanese Adolescents. *CyberPsychology and Behavior*. **9(6)**: pp317-324
- Yee, N. (2006). Motivations for Play in Online Games. *CyberPsychology and Behavior*. **9(6)**: pp772-775

COPYRIGHT

Brett Turner and Andrew Woodward ©2008. The author/s assign Edith Cowan University a non-exclusive license to use this document for personal use provided that the article is used in full and this copyright statement is reproduced. Such documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. The authors also grant a non-exclusive license to ECU to publish this document in full in the Conference Proceedings. Any other usage is prohibited without the express permission of the authors.