A Traffic Analysis of a Small Private Network Compromised by an On-line Gaming Host

Ron McLeod, BCSc, MCSc.
Director - Corporate Development Telecom Applications Research Alliance
Doctoral Student, Faculty of Computer Science, Dalhousie University

Abstract

In the early months of 2006 a small private network (the Network) suffered a noticeable degrading of its network performance. A network traffic capture and analysis was conducted and used to investigate the network performance issues. This paper presents partial results of that analysis. The network traffic capture formed part of an experimental use of the SilkTools ™ [1] capture and analysis suite developed by CERT personnel at Carnegie Mellon University. During the first analysis of the captured data it was discovered that the Network contained a host that had been compromised at some time in the past and was currently being used to support the on-line gaming activity of over 174,000 distinct player source addresses around the globe. These players were believed to be participating in the Half-life™ [2] first-person shooter game (the Game). The initial finding was the result of a manual investigation of unusual time and volume traffic spikes from arbitrarily chosen time slices. Subsequent work was conducted on searching for a traffic signature which could be representative of the presence of the Game such that future discovery of Game activity could be automated. Gaming traffic is predominantly UDP traffic of high byte volumes, typically targeted at a given range of destination ports. This analysis also searches for a specific TCP traffic pattern that is suggestive of a Game signature. Network traffic patterns that emerge after access to the compromised host has been closed are labeled as SCAR traffic, for **S**evered **C**onnection **A**nomalous **R**ecords

1. Methodology and Experimentation

1.1. The Network Sampling Environment

On February 3, 2006 an ongoing traffic capture was initiated within the Network. This was accomplished by instructing the primary edge router to construct netflow [3] records and to deliver those records to a single collection point within the Network. The Network used for experimentation is comprised of four /24's, one of which has been divided into /27's. In total, the entire Network consists of approximately 40 user assigned hosts, although the actual number of hosts is subject to minor fluctuations over time. Approximately an additional 40 special purpose hosts also exist within the address space although these hosts are not assigned to individual users. Many of the hosts are the property of separate owners and are subject to separate administration. However, traffic from each passes through a single edge router and it is at this router that the author did his data collection. For reasons of privacy, payload data was neither collected nor examined. It was further understood that the author would not have access to the content of specific hosts for further investigation purposes. Although, the owners of each host for which anomalous activity (if any) was discovered would be informed immediately of any observed anomaly in their machines and full disclosure of the analysis results would be made upon request. For confidentiality reasons the identity of the Network is not specified in this document. For purpose of analysis, only the non-port 80 traffic and non-null traffic was initially considered.

1.2 Network Analysis – The Discovery of an Intruder

On February 11, 2006 the first sample of network traffic was extracted for analysis. The time period from midnight to 7:00 AM local time on February 8 was chosen for the first data slice. This was partially a random choice and partially due to the fact that the author expected minimal traffic volumes during this time. The analysis tools were instructed to access each flow record for the time in question and to extract Source IP address, Destination IP address, Source Port, Destination Port, Protocol, Bytes (the number of bytes in the flow record), TCP Flags, Start Time (of the flow record) and End Time (of the flow record).

Figure 1 shows a profile of the sampled traffic data that corresponds to a three tuple consisting of the Flow Record Number (in order of appearance), Protocol Number and Bytes per flow record for the hour of 12:00 – 1:00 AM.

Figure 1

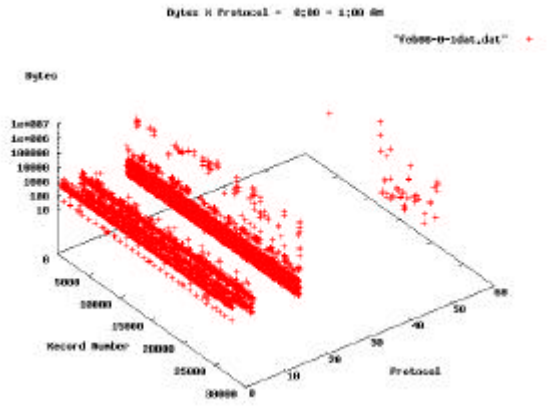

Figure 1 shows a clear grouping of the data into four distinct bands corresponding to five separate protocol clusters. They are:

1. Protocols 50 and 53 used by IPSEC and SWIPE respectively.
2. Protocol 17 used by UDP traffic.
3. Protocol 6 used by TCP.
4. Protocol 1 used by ICMP.

Table 1

Port	Flows
53	260596
123	16139
137	37586
138	26875
161	40799
500	28151
1027	10170
1031	18241
1954	13445
2008	11777
2967	51571
5060	81821
6346	16320
25383	141890
26900	72348
27000	13173
27001	13342
27002	13174
27003	13233
27005	34933
27010	13039
27015	**6061263**
27243	64616

Within these bands the largest consistent byte volume is within the UDP (Protocol 17) Band.
The first point of interest was the volume of flow records. There were approximately 27,000 records between 12 midnight and 1:00 AM, when no users were present. The records were then ordered by byte size. This was the first attempt to search for outliers within the data.

From this sorting it was discovered that a small group of SourceIPs using protocol 17 appeared to be responsible for a large portion of the traffic bytes. These were referred to as the Heavy-Traffic-Hosts. However, given the size of the database it was not immediately apparent if there was a subset of the Heavy-Traffic-Hosts that were unusually heavier than the rest. The traffic was then sorted by Source IP and the total bytes over all flow records were accumulated for each SourceIP. The result was striking. One SourceIP accounted for more than 56% (79,865,126 bytes) of the traffic volume measured in bytes during the hour in question. This SourceIP was labeled as the Suspicious Host

The next step in the analysis was to examine the remaining available data about Suspicious Host. It was found that out of 27,477 flow records for the hour, the Suspicious Host was the SourceIP in 9, 235, or 34% of all flow records during the hour. It was found that the Suspicious Host communicated with 5,987 separate DestinationIP's during the hour. These DestinationIP's were distributed around the globe. Further analysis of the ports targeted by more than 10,000 flows revealed that almost all traffic from SourceIP's that targeted the Suspicious Host as the DestinationIP was using protocol 17 and destination port 27015 (Table 1). Also, a significant amount of the traffic to and from Suspicious Host was directed at university campuses in the United States and consumer ISP's around the world. Although not elaborated on herein due to space limitations, the reader should note the somewhat uniform distribution of flows using ports 27,000 - 27,005 and 27,010 in Table 1.

Figure 2

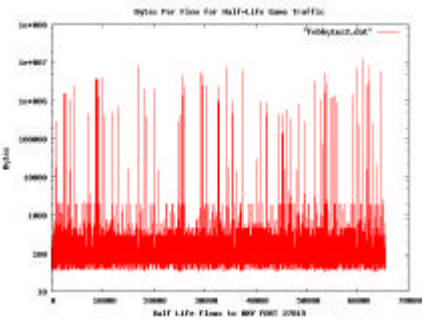

Figure 2 shows the bytes per flow for traffic where either source or destination port was set to 27015.The slice presented in Figure 2 covers a period of approximately 2 hours, or approximately 65,000 consecutive flow records. The repeating pattern of low, medium and high byte volumes is indicative of the presumed

application protocol. This behavior is manifest as low followed by medium volume for all external SourceIP's followed by regular high volume by a smaller set of SourceIP's. Finally, the Suspicious Host was identified as an experimental development machine that had been part of a development and testing project in the previous year. Although it was still connected to the network it was not supposed to have any active users. The known facts about the Suspicious Host are summarized below.

Suspicious Host
Responsible for 56% of Network Non-Port 80 Byte Volume
Responsible for 34% of Network Non-Port 80 Flow Volume
Some Preference for University Campuses and Consumer ISP's
Primarily uses UDP Port 27015
Should Have Little or No Traffic

A search of the Internet revealed that (with the exception of the last feature) this is the behavior pattern of a server involved the Half-Life[tm] on-line first person shooter game. It appeared that at some time in the past this host had been compromised and was now being used as part of a worldwide on-line gaming community.

It is important to point out that since the experimenter had no access to the actual machine or payload data this conclusion is simply conjecture. However, it is one with which the author is exceedingly comfortable. Furthermore, it is important to point out that Valve Software, the maker of Half-Life[tm] is a legitimate company that would never knowingly allow its products to be part of an unauthorized network compromise. Indeed, in these circumstances, companies such as Valve Software are as much a victim as the owner of the compromised network. The Half-life[tm] game, like other such on-line games, contains Client and Server software. The player installs the client software on their own machine and then searches for an available server. These servers are run by other players or by server hosts. This particular game uses a third component known as Meta-Servers [4] or Master Servers [5] which provide a list of known game servers. Without access to the suspect machine to search for installed software we are left to speculate as to whether a Server or Meta-Server was installed. Further analysis of the traffic pattern of the compromised machine and comparison to the traffic that one would see from actual known Servers and Meta-Servers would most likely resolve this ambiguity. Such a comparison does not form part of this analysis.

On February 13, the owner of the compromised host was advised of the infection. The owner indicated his intention to immediately bock subsequent access to the machine by the gaming community. The author recommended against this action. However, the machine owner decided to proceed with shutting down access. This action proved unsuccessful and gaming traffic continued for another month.

2. The Search for a Behavioral Signature

An investigation was undertaken to attempt to discover a TCP signature that could be associated with game traffic.

2.1. Separating Normal from Infected Traffic

Each action listed below is accompanied by an example Silktools Command used to achieve the desired result. In order to discover an un-infected model of the Network traffic it was necessary to remove the traffic which could be attributed to those hosts involved in the gaming traffic, thus creating an artificial normal traffic sample. This normal traffic would then be compared to the infected traffic in an attempt to find distinguishing characteristics. To achieve this separation, a filtering of the data was done to extract all SourceIP's that specified either the source or destination port as 27005, 27014 or 27015 within a 24-hour period when infection was known to be present. The resultant file was labeled as the half-life traffic (**rwfilter --aport=27005,27014,27015 --pass=hltraffic.f out* % out* will open each flow record file for the day in sequence).**

A set of unique SourceIP's was then created using the half-life traffic file as input and labeled hlsipfile.set (**rwset --sip-file=hlsipfile.set hltraffic.f).**

We now had a list of unique SourceIP's allegedly involved in the game traffic. To remove the ambiguity of action which is naturally associated with UDP traffic it was decided to see how much (if any) TCP

traffic these game SourceIP's participated in, and what was the nature of that traffic. A set of unique SourceIP's for all TCP traffic was created. This was done in two steps. First the TCP traffic was isolated from the rest (**rwfilter - -proto=6 - -pass=tcptraffic.f out*).** Next a set of unique SourceIP's was created (**rwset - - sip-file=tcpsipfile.set tcptraffic.f).**

The Game SourceIP's involved in TCP traffic are given by the intersection of the two previous sets and placed in the file hltcp.set (**setintersect - -add-set=hlsipfile.set - -add-set=tcptraffic.f - -set-file=hltcp.set).**

Upon completion hltcp.set contained only four unique SourceIP's, one of which was the Suspicious Host. The other three were not within the address space of the Network. By removing traffic involving these SourceIP's from the total TCP traffic we are left with a file of TCP transactions where the participating game players and the Suspicious host have been removed. This TCP traffic we label as normal, or without-infection. Figure 3 shows two images of the data. The figure on the left shows a small area of the plot of destination ports versus Bytes per flow in the presence of the infection. The image on the right of the figure shows the same area of the solution space when the infecting traffic has been removed.

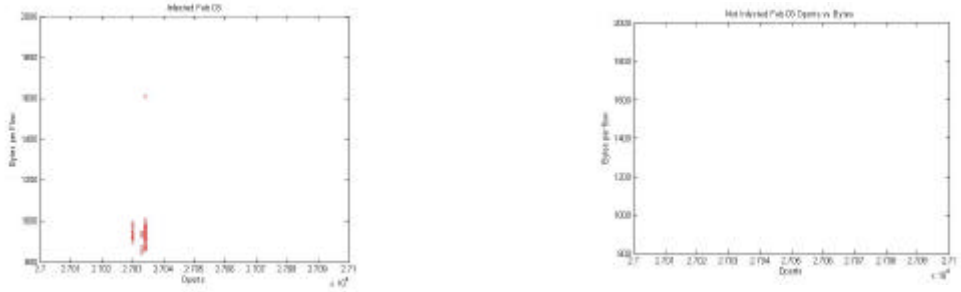

Figure 3

From the two images one can discover a set of TCP traffic contained in the Destination Port range 27,030, 27,033 and 27,034 with Bytes Per Flow sizes ranging from the low 800's to slightly more than 1000 with a noticeable outlier at approximately 1600. This traffic is absent in the artificial normal data set. This traffic was labeled as possible game Signature Traffic. It was originally the author's hypothesis that this data set represents the possible presence of a Game Server in a network (i.e. a signature). This hypothesis has yet to be examined fully and tested.

3. Future Work

Work on this dataset will continue in August 2006, at which time further searching for a TCP signature indicative of the administrative layer of the game network will be sought. This activity will expand the search to include Port 80 and Null traffic. In addition, the author hypothesizes that the loss of the game server from the network will create a continuing repeating pattern of attempted logins by players. The author has labeled this type of traffic as SCAR traffic, for Severed Connection Anomalous Records. This Scar traffic may indicate the recent presence of a game server (unauthorized) on a network.

References

[1] SilkTools, http://silktools.sourceforge.net/, as accessed March 28, 2006.

[2] Valve Software http://www.half-life.com as accessed on March 28, 2006

[3] CISCO IOS Netwflow, http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html, as accessed on March 28, 2006.

[4] Weaver, N, Paxson, V., Staniford, S., Cunningham, R. "Large Scale Malcious Code: A Research Agenda", The International Computer Science Institute (ICSI), a DARPA Sponsored Report, 2003.

[5] HL Master Game Server configuration,http://hlmaster.sourceforge.net/documentation/hlmaster.gameserver.php as accessed on March 31, 2006.