Svision: Visual Identification of Scanning and Denial of Service Attacks

Iosif-Viorel Onut, Bin Zhu, and Ali A. Ghorbani

***Abstract*—**We propose a novel graphical technique (SVision) for intrusion detection, which pictures the network as a community of hosts independently roaming in a 3D space defined by the set of services that they use. The aim of SVision is to graphically cluster the hosts into normal and abnormal ones, highlighting only the ones that are considered as a threat to the network. Our experimental results using DARPA 1999 and 2000 intrusion detection and evaluation datasets show the proposed technique as a good candidate for the detection of various threats of the network such as vertical and horizontal scanning, Denial of Service (DoS), and Distributed DoS (DDoS) attacks.

***Keywords*—**Anomaly Visualization, Network Security, Intrusion Detection.

I. Introduction

DATA visualization represents a fundamental part of the current network security practices, providing the network administrators with important information regarding the state of the network as well possible threats that exist. Frost and Sullivan [5], recently reported that only 11.6% of the available Intrusion Prevention Systems (IPSs) in 2003 were set to prevention mode by the administrators. Consequently, in all the other cases, the network administrator is the one that decides upon the proper response that has to be enforced. In order to do that, he/she has to have a deep understanding of the current state of the network, and this is mostly achieved through different network visualization techniques. Thus, despite all the existing criticisms against the visualization techniques as a detection method, we do not anticipate its possible replacement in the near future.

We propose a network visualization technique that allows the security personnel to easily identify potential anomalies in the network. The network is depicted as a community of hosts that are roaming inside a three dimensional space. Since a network might have hundreds of hosts, the proposed view highlights only the ones that might represent a potential threat to the network, while the normal hosts overlap near the center of the view.

Our experimental results conducted on two of the well known intrusion detection and evaluation datasets (i.e., DARPA 99 [6] and DARPA 2000 [7]) empirically proved the technique to be successful against main types of Denial of Service (DoS) attacks, Distributed DoS (DDoS) attacks, as well as vertical and horizontal scanning attacks.

This paper is organized as follows: Section II presents some of the important existing visualization techniques. Section III describes the proposed visualization technique presenting the main outcomes and drawbacks of the representation. Next, Section IV presents the empirical results against the common attacks such as DoS, DDoS, and probing. Finally, the last section summarizes the conclusions and presents possible future improvements.

II. Background Review

Visualization techniques are some of the pioneers approaches successfully applied in the network area. Network administrators tend to be very comfortable with network data presented in the form of charts, functions, and tables. The network visualization techniques do target most aspects of the network security including topology representation, protocol communication, and congestion control, to name a few.

M. Spencer [11] proposed a visualization technique that displays the network topology, assisting the security personnel in detecting possible failure points and checking the availability of the devices within the network. R. F. Erbacher [1] proposed a similar technique that uses a glyph based approach in order to represent not only the topology of the network but also its load. In the same line of work, D. Estrin et. al. [2] proposed a visualization system that shows network topologies animations, measuring packet loss rates for various links in order to detect potential connectivity problems.

The most common visualization technique remains the two dimensional graphs where one dimension represents the time coordinate (e.g., usually x axis), while the second axis represents a particular feature of the network. Moreover, by the use of colors, multiple graphs can be mixed in a single view [9], [2], [8], [10]. Such visualization tools have been widely used by network administrators to monitor the network links and identify abnormal external behavior such as DDoS, DoS, Scanning, and Worms, as well as improper internal activity such as P2P file sharing.

As more powerful computation capability becomes available, visual representation of network has evolved from

Manuscript received May 15, 2005.

Iosif-Viorel Onut is pursuing Ph.D., in Computer Science at Faculty of Computer Science, University of New Brunswick, Canada. (e-mail: onut.viorel@ unb.ca).

Bin Zhu is pursuing M.Sc., in Computer Science at Faculty of Computer Science, University of New Brunswick, Canada. (e-mail: bin.zhu@ unb.ca)

Ali A. Ghorbani is with Faculty of Computer Science, University of New Brunswick, Canada. (e-mail: ghorbani@unb.ca).

2D to 3D in order to incorporate more complex information. In the 3D graphical technique proposed by M. Fisk et al. [3], internal and external IP addresses are mapped into a 3D space, the connection between an internal host and an external host is presented by a line with particular color and length, representing information such as service type, duration of that connection, and source/destination IPs. This visualization technique proved to be very efficient in the case of scanning attacks. The CICHLID Data Visualization Software [4] is an example of 3D visualization tool for network data processing. It consists of various views with colored bars representing different network characteristics such as IP Address Utilization, Packet Length Distributions for major IP protocols, and bytes vs. time for major TCP/UDP port numbers, to name a few.

III. The SVision Visualization Technique

The proposed visualization technique (SVision) pictures the network as a community of hosts that independently roam in a 3D space defined by the set of services that they use. The aim of SVision is to graphically cluster them into normal and abnormal ones. Circles are used as graphical elements for representing the hosts' position, colors are used to distinguish between external and internal hosts (i.e., red and blue, respectively), and color intensities are used to discriminate between recently detected hosts (i.e., dark red/blue color) and previously detected hosts (i.e., light red/blue color); that is, as the time passes by the colors turn from dark to light.

The visualization technique assumes that the system administrator is able to identify the set of critical services that are to be monitored (e.g., HTTP, FTP, DNS, to name a few). Let Ψ represent this particular set of services. Furthermore, let *sparsely-active* (*constantly-active*) represent a host who is seldomly (constantly) using any number of the Ψ services in a predefined time window interval τ.

Our proposed graphical model uses a two dimensional plane (i.e., *Service Usage Plane*) to discriminate between *sparsely-active* and *constantly-active* hosts with respect to the selected set Ψ of services. Let *service point* represent the place in the view where a particular service is displayed (see Fig. 1). Furthermore, let all the *service points* of the services in Ψ be equally-distant placed on a circle centered in θ called the *Attraction Circle*. Thus, the more a host is using a particular service during τ. the closer it will be from that *Service Point*.

Let us define the *attraction force* as the force that a particular service $S_k \in \Psi$ attracts a host H_j.

$$\vec{F}_{k,j} = A_k \cdot L_{k,j} \begin{bmatrix} \cos(\alpha_{k,j}) \\ \sin(\alpha_{k,j}) \end{bmatrix} \qquad (1)$$

where $L_{k,j}$ is the load of the host H_j with respect to the service S_k, and A_k is a predefined *anomaly factor* for the service S_k.. The *anomaly factor* reflects traffic load expectations from service to service. For instance, the load of the FTP protocol is expected to be higher than the load of ICMP protocol. Consequently, the same bit rate might be a sign of normal or intrusive behavior from case to case, the *anomaly factor* being intended to minimize this difference.

Furthermore, the position of a host inside the *Service Plane* is not only dependent on the usage of one service, or on its current load, but is computed as a spatial and temporal equilibrium.

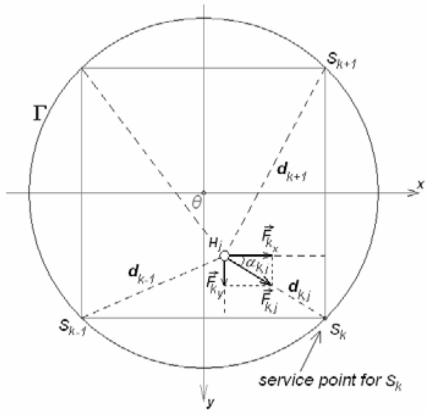


Fig. 1 The *Service Usage Plane* of the SVision

The spatial equilibrium of host H_j is calculated by taking into consideration all the *attraction forces* between the host and $S_k \in \Psi$:

$$\sum_{\forall s_k \in \Psi} \vec{F}_{k,j} \cdot d_{k,j} = 0 \qquad (2)$$

where $d_{k,j}$ represents the distance between the host H_j and the *Service Point* of the S_k. Furthermore, replacing (1), (2) and expressing both $\cos(\alpha_{k,j})$ and $\sin(\alpha_{k,j})$ as a function of H_j and S_k positions, the final coordinates of the host can be computed.

The temporal equilibrium for a host H_j is obtained by splitting the *time window interval* τ into x equally sized subintervals, and computing for each subinterval an *attraction force*. Let $\vec{F}_{k,j,t}$ be the *Attraction Force* for the t^{th} time slot computed for j^{th} host with respect to the k^{th} service. Thus, the *Attraction Force* for the current moment n is computed as:

$$\vec{F}_{k,j,n} = \sum_{k \in \tau} \left(\vec{F}_{k,j,t} \cdot e^{-|n-t|} \right) \qquad (3)$$

where e^{-1} is the *unit delay operator*; that is, e^{-1} operating on $\vec{F}_{k,j,t}$ at time t yields its delayed version $\vec{F}_{k,j,t-1}$. Thus, the closer t is to the current time, the more influence will have its correspondent $\vec{F}_{k,j,t}$ over the computation of the host's position; therefore, simulating a *short term memory mechanism*.

Showing the inbound and outbound activity of each host is an important issue in characterizing the host's active and passive behavior[1]. Consequently, two sets of coordinates can be identified for each host by separately using the inbound load and outbound load when computing (1), resembling its passive and active behavior. We call this points *inbound extreme* and *outbound extreme* of a host.

[1] The inbound (outbound) activity of a host is defined as the number of bites it receives (sends) in the time window interval.

The previously presented 2D representation does not include any information about the real traffic load of the hosts, this being justified by its inability to distinguish between a host that is constantly using service S_k with, lets say, 10 Kb/s and another host that is constantly using the same service with, lets say, 100Kb/s. The solution to this problem is the introduction of a third dimension that will represent the hosts' load. Moreover, because the load cannot have a negative value, and in order to distinguish between inbound and outbound activity, the 3D view is split into two spaces (i.e., the inbound activity space, and the outbound activity space) as depicted in Fig. 2. Thus, if a host is receiving a lot of inbound traffic its *inbound extreme* will migrate to the top of the view, denoting a possible victim of an attack. Conversely, if the host is producing a lot of outbound traffic, its *outbound extreme* will migrate to the bottom of the view, denoting a possible attacker. Throughout our experiments, we noticed that the most majority of the hosts will remain and overlap close to the center of the view near the *Service Usage Plane*, which is desirable since they are considered to be normal. On demand, the view is displaying a graphical line between the two points in order to show that they belong to the same host (see Fig. 3).

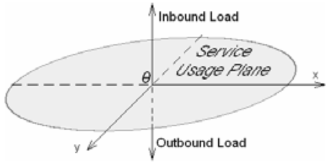


Fig. 2 The 3D space of SVision

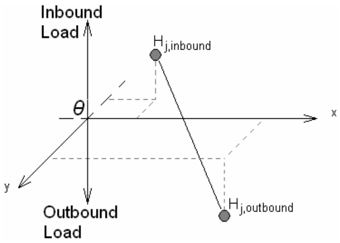


Fig. 3 The inbound and outbound extremes of a host

Finally, in order to depict vertical and horizontal scanning attacks, the proposed visualization technique is using rays attached to each host. The semantics of these rays differs from internal to external hosts. Consequently, in the case of an internal host the rays depict the number of distinct ports that have been open or scanned by other hosts (resembling a potential victim of a vertical scanning attack or DDoS), while in the case of an external host the rays represent the number of distinct IPs that it connects to (resembling a potential attacker of a horizontal scan).

IV. Experimental Results

Two well known intrusion detection and evaluation datasets, provided by Lincoln Laboratory, have been chosen (i.e., DARPA 99 and DARPA 2000) for evaluating our proposed graphical technique. Since in both cases the intrusions and attack scenarios are known, the behavior of the view can be studied and possible improvements can be identified.

A. Pign of Death

The Ping of Death (PoD) attack is a denial of service attack that appeared in early 1997 as an exploit of a flaw in the networking implementation of some operating systems.

The ping command is implemented as a tool to send ICMP messages for troubleshooting purposes; in particular, it checks the routing between the system which initiated the ping command and the target system. Since the maximum length of IP packets is 65,535 bytes, by carefully crafting an oversized IP packet the target host can be compromised (e.g., slowed down, crashed).

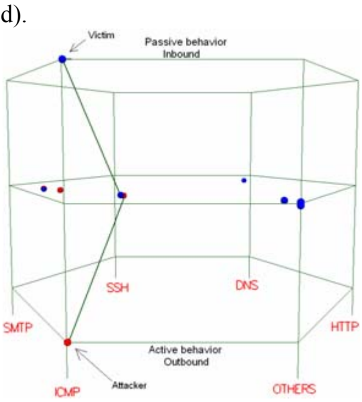


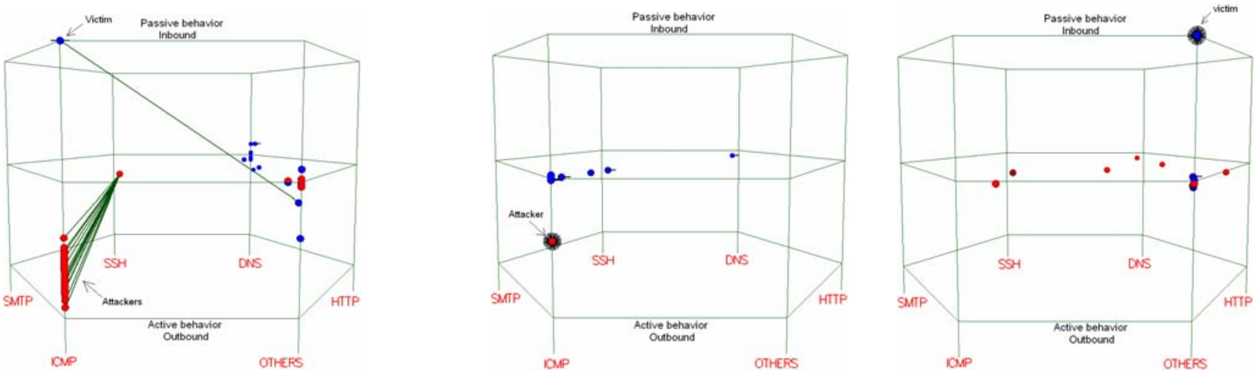
Fig. 4 The PoD attack

Fig. 4 depicts the PoD attack during the first day of the fifth week from the DARPA 99 database. As expected, both the attacker and the victim are close to the *service point* of the ICMP protocol. While the attacker's outbound extreme can be clearly identified in the lower part of the view showing an evident active behavior (i.e., its outbound is high because it sends the attack packets), the victim's inbound extreme is highlighted close to the ceiling of the view confirming its passive behavior (i.e., its inbound is high because it receives the attack packets).

B. Smurf

A typical scenario of a DDoS attack is the case where the attacker remotely controls several daemons over the internet in order to compromise a victim. This type of attack is very efficient due to its distributed nature. Moreover, most of the times, it is very difficult to trace the initiator of the attack.

The *Smurf* attack simulated in DARPA 99 is a classic example of a similar scenario, where the attacker sends ICMP 'echo request' messages to the broadcast address with the source address spoofed to be that of the victim. In this way it transforms all the hosts in that network into daemons that will send a large number of ICMP 'echo reply' messages to the victim in a short time interval (see Fig. **5**(a)). The similar active behavior of the daemons makes their outbound extreme to reside very close to the ICMP's access point. Furthermore, since they exhibit almost no passive behavior, their inbound extreme is close to the origin of the view. Conversely, the victim's passive behavior makes its inbound extreme to reside



(a) The *Smurf* attack

(b) The phase 1 of the LLDOS 1.0 scenario consisting in a IPsweep attack

(c) The phase 5 of the LLDOS 1.0 scenario consisting in a DDoS attack

Fig. 5 Different attack scenarios from DARPA 99 (see Fig. 5(a)), and DARPA 2000 (see Fig. 5(b) and Fig. 5(c))

very close to the ceiling of the view highlighting the anomaly of the attack.

C. *LLDOS 1.0 attack scenario*

The following attack scenario (i.e., LLDOS 1.0) is implemented by Lincoln Laboratory in DARPA 2000 dataset. It consists of a multistage five phases attack as follows:

1) IPsweep of three internal networks from a remote site;
2) Probe of IPs to look for the vulnerable Solaris hosts;
3) Breakins via the sadmind buffer overflow;
4) Installation of the mstream DDoS software on compromised hosts; and
5) Launching the DDoS attack.

Our system successfully highlights the attacker of the first phase as well as the victim of the last phase of the scenario. In the first phase of the scenario the attacker scans multiple subnets of the Air Force Base for identifying the possible candidate hosts for the second phrase (i.e., sending ICMP echo-request packets). This is clearly depicted in Fig. **5**(b), where the external host manipulated by the attacker has multiple rays around it representing the number of distinct victims that it scans.

After braking in several hosts and creating the daemons (i.e., phases 2, 3, and 4 of the scenario), in the final phase the attacker manually lunches via a telnet a DDoS attack of 5 seconds duration. The DDoS attack consists of many connection requests to a variety of ports on the victim. Each request has a spoofed random IP source address, which makes impossible the daemons identification.

Fig. 5(c) depicts the victim's anomalous behavior during the last phase. The high volume of inbound traffic that it receives makes its inbound extreme close to the ceiling of the view. Furthermore, the high number of rays around the victim (i.e., number of distinct ports that are being scanned) is justified by the nature of incoming packets representing connection requests to various ports, and enables the network administrator to successfully identify the victim.

V. Conclusions and Future Work

We presented a technique that combines both anomaly and graphical techniques for network intrusion detection. The view graphically clusters the existing hosts in the network with respect to the services that they use. The set of services can be selected by the network administrator and may vary from network to network. Experimental results on DARPA 99 and DARPA 2000 show the proposed technique as a possible solution for graphical detection of DoS, DDoS, and Scanning attacks (e.g., Ping of Death, Smurf, UDP storm, IPsweep).

Our future work will focus on transforming the proposed view from the passive visualization into an active one by integrating a detection engine that will analyze the movement of the hosts in the view.

References

[1] R. F. Erbacher, *Visual traffic monitoring and evaluation*, Conference on Internet Performance and Control of Network Systems II (Denver, CO, USA), August 2001, pp. 153–160.

[2] Deborah Estrin, Mark Handley, John Heidemann, Steven McCanne,Ya Xu, and Haobo Yu, *Network visualization with the vint network animator nam*, Tech. Report 99-703, University of Southern California, Los Angeles, March 1999.

[3] Mike Fisk, Steven Smith, Paul Weber, Satyam Kothapally, and Thomas Caudell, *Immersive network monitoring*, The Passive and Active Measurement Workshop (PAM2003) (SDSC at UC San Diego 9500 Gilman Drive La Jolla, CA 92093-0505 U.S.A.), April 2003.

[4] National Laboratory for Applied Network Research (NLANR)'s Measurement & Operations Analysis Team (MOAT), *CICHLID data visualization software*, http://moat.nlanr.net/Software/Cichlid/, 09 May 2005,last access.

[5] Frost and Sullivan, *World intrusion detection and prevention systems markets*, Tech. report, Frost and Sullivan, 7550 West Interstate 10, Suite 400 San Antonio, Texas 78229-5616. USA, 25 June 2004.

[6] Lincoln Laboratory, *Intrusion detection evaluation data set DARPA 1999*, http://www.ll.mit.edu/IST/ideval/data/1999/1999 data index.html, 1999.

[7] Lincoln Laboratory, *Intrusion detection evaluation data set DARPA 2000*, http://www.ll.mit.edu/IST/ideval/data/2000/2000 data index.html, 2000.

[8] Tobias Oetiker and Dave Rand, *Multi router traffic grapher (mrtg)*, http://ee-staff.ethz.ch/oetiker/webtools/mrtg/, May 9, 2005 last access.

[9] D. Plonka, Flowscan: *A network traffic flow reporting and visualization tool*, USENIX Fourteenth System Administration Conference LISA XIV (New Orleans, LA), December 2000.

[10] Q1Labs, *QRadar*, http://www.q1labs.com/, May 9,2005, last access.

[11] Mark Spencer, *Cheops network user interface*, http://www.marko.net/cheops/, May 9, 2005 last access.