Existence Plots: A Low-Resolution Time Series for Port Behavior Analysis

Jeff Janies

CERT Network Situational
Awareness Group
4500 Fifth Avenue
Pittsburgh, PA 15213
janies@cert.org

Abstract. An existence plot is a low-resolution visualization that concurrently represents the activity of all 2^{16} ports on a single host. By doing so, we are able to show patterns of port usage which can indicate server activity and demonstrate scanning. In this work we introduce the existence plot as a visualization and discuss its use in gaining insight into a host's behavior.

Keywords: Network traffic visualization, Low-resolution visualization, Time series

1 Introduction

An *existence plot* is a time-series visualization of traffic of *all* the active ports of a single monitored host. Existence plots summarize activity for a single host in a limited space, regardless of the number of unique sources with which the host communicates. For example, Figure 1 is the existence plot for an SMTP server. Ports are listed on the y-axis, while the x-axisrepresents time. The box-and-whisker diagram on the right of the plot shows a color coding of byte magnitude: blue for the 1st quartile, green for the 2nd and 3rd quartiles, and red for the 4th quartile. We also see two families of lines on the plot - a constant, horizontal red line at port 25, indicating the host's SMTP server activity, and a collection of lines with similar slope in the ephemeral port range (1024-65535) indicating client activity.

Existence plots provide useful, high-level summaries of traffic from a particular host, due to their coarse representation of activity. Using the existence plot, we provide a high-level view of *all* activity originating from a host. While we cannot provide exact information on the magnitude of activity from a particular port and maintain readability, we note that the majority of network traffic contains noise generated by automated scanning, bots and other hostile activity [1][5][7]. Because of this, it is not unusual for simple magnitude-based indicators to include a large amount of trivial data due to prevalent, but meaningless low-volume interactions caused by hostile activity.

This paper is a tutorial on the construction and use of existence plots. We demonstrate that existence plots provide a method for analysts to rapidly identify aggregate host behavior such as *hidden servers* (hosts that are operating as public servers which the administrator may not be aware of), scanning and scan response.

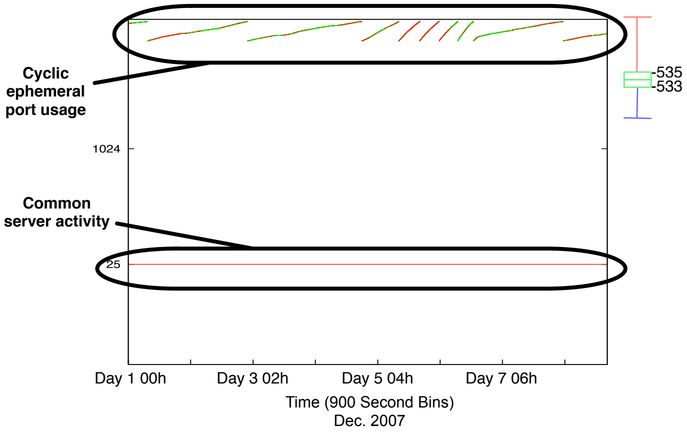


Fig. 1. Existence plot showing activity from December 1st to 7th, 2007 from an SMTP server.

The remainder of this paper is structured as follows: §2 describes the composition and construction of existence plots from traffic data; for our examples we use the SiLK toolkit[1], but plots can also be constructed using raw `tcpdump` data. §3 shows how to interpret results from an existence plot, in particular the identification of ephemeral port activity and hidden servers. §4 discusses other visualizations that center on representing individual hosts. §5 concludes this work with a discussion of future application.

2 Constructing Plots

In this section, we outline the data requirements and method for generating existence plots, as follows: §2.1 describes our source data and its format and §2.2 describes the process of plotting formatted source data.

2.1 Source Data Format

Existence plots represent a unidirectional count of bytes transfered on individual ports. We format network traffic summaries as a series of values, $X_{p,\tau}$, where

[1] Available at `http://tools.netsa.cert.org/silk/`

p is a port and τ represents a time interval. As with other time series, τ is a discrete interval of time, in this case measured in seconds. Since we format the data as unidirectional traffic summaries, each host, A, can be represented with two non-equal data sets `inbound` and `outbound`, where `inbound` is the set of byte counts of all packets destined to A, and `outbound` is the set of byte counts of all packets from A.

2.2 Plotting From Data

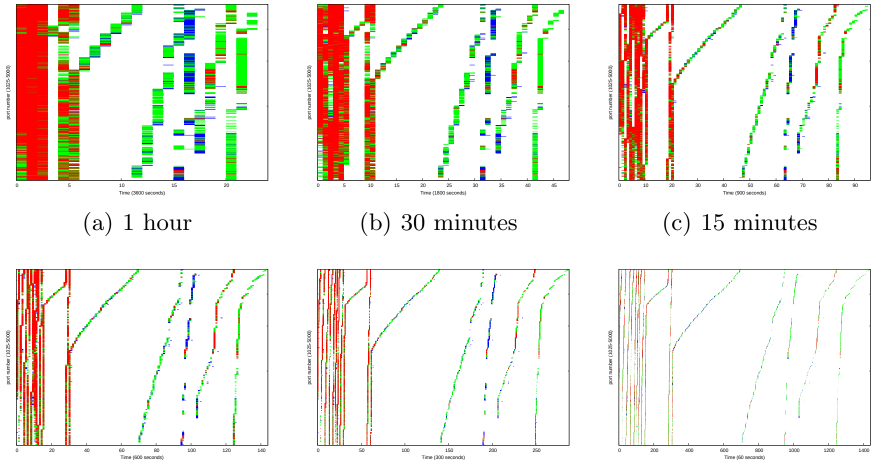
For the images in this paper, the y-axis represents the 2^{16} TCP/IP and UDP ports plotted in log scale, unless otherwise specified. This representation is a *low-resolution* time-series; that is, instead of a precise delineation of every discrete value (as is the case with MRTG), we bin the values into broad categories in order to provide a complete view of the data. We denote drastic changes in magnitude with color. We are able to compress more information into each image by using the y-axis to represents unique variables instead of a shared scale measuring magnitude.

$$color(p,t) = \begin{cases} none & X_{p,t} = 0 \\ blue & 0 < X_{p,t} < S_0 \\ green & S_0 \le X_{p,t} \le S_1 \\ red & S_1 < X_{p,t} \end{cases} \tag{1}$$

Equation 1 shows the mapping of magnitude to colors using the data dependent values: $X_{p,t}$, S_0 and S_1. We set S_0 and S_1 as the 1st and 3rd quartile of all non-zero values of $X_{p,t}$ (*i.e.* the values change per data set). However, alternative approaches are viable: for example, if the magnitude of traffic is predictable, S_0 and S_1 can be fixed across images to provide consistency.

With the existence plot, we are equally interested in periods of activity *and* inactivity. By representing both, patterns emerge. The most common ephemeral port usage pattern is a series of lines with similar slopes, depicting *port cycling* in client interaction (*i.e.* the host sequentially uses a set of ports in a finite range, and this sequence is repeated). Servers consistently use ports common to the service they provide. This results in a horizontal line of activity.

Figure 2 shows how variations in the size of τ affect the shapes in existence plots. In this figure, we represent one day of `outbound` traffic from ports 1025 through 5000 of a frequently used Microsoft Windows machine. We vary τ sizes to be 1 hour, 30 minutes, 15 minutes, 10 minutes, 5 minutes and 1 minute and only display the port range 1025 through 5000. With the largest τ size of one hour, the port cycles with a longer period are discernible, but the port cycles with shorter periods are completely indiscernible. As the τ size decreases the shorter period port cycles become discernible. At 10 minutes the lines are distinct; we, therefore, use a resolution of 10 minutes for a majority of the images in this paper.



(a) 1 hour

(b) 30 minutes

(c) 15 minutes

(d) 10 minutes

(e) 5 minutes

(f) 1 minute

Fig. 2. The x-axis represents time and the y-axis represents the port range 1025 through 5000. As the size of τ decreases the port cycles become more distinct.

3 Interpreting Plot Data

Existence plots display aggregate port activity. In this section, we demonstrate how this aggregated view can be used to identify various phenomena, specifically servers (in particular *hidden* servers), and scanning. This section is divided as follows: §3.1 explains how existence plots can be used to identify hidden servers. §3.2 shows how existence plots can represent scans.

3.1 Hidden Server Identification

We define a *hidden* server to be any host that provides a service to hosts outside of the network, without the administrator's knowledge or consent. Specifically, we show a misconfiguration causing a client to function as its own mail relay. We compare this behaviors with a known mail relay.

Here, we represent one day of a host's activity with two existence plots, which we refer to as *existence plot pairs* (an example is shown in Figure 3). The existence plot on the left represents the destination ports of packets in the data set `inbound` for the host in question (dport inbound). The existence plot on the right represents the source ports of packets in the data set `outbound` for the host in question (sport outbound). In both plots we use a τ size of ten minutes (600 seconds).

Figure 3 shows an example of a misconfigured host. Nominally, the host should use an internal mail server for mail inspection and distribution. However, due to a misconfiguration, the host began forwarding mail to external hosts itself.

As the figure on the left shows, the host receives a great deal of traffic to a limited number of ports but does not respond to these connection attempts (*e.g.* there is a lack of activity in the figure on the right). Instead, a short quick burst of activity occurs in the ephemeral port range. This burst encapsulates connections to 75 unique mail servers and lasts for approximately one minute, after which no further mail activity is observed from the host. This visualization provides us with two key insights. First, the activity occurred in a very short time and is inconsistent with the host's *modus operandi*. Second, the host receives consistent scans to ports associated with well-known vulnerabilities but does not respond. Therefore, we are able to rule out the possibility of this being a mail relay for external hosts.

Contrary to the activity demonstrated in Figure 3, Figure 4 shows an existence plot pair of a known mail relay. Similar to Figure 3, this host receives a large number of connection attempts from external hosts, and likewise, does not respond. However, unlike Figure 3 the host has a consistent pattern of ephemeral port activity, with no reserved port activity. Additionally, the ephemeral port activity does not show drastic deviations, *e.g.* the port cycling continues through the course of the day.

3.2 Scan Detection

Figure 5 shows an existence plot pair representing the port usage of a compromised host. Upon inspection, we find that the host appears to have two different instances of port cycling. The first instance is in the ephemeral range, as is expected from a client. The second is the port range 54 through 499. Since the second instance only occurs in the image on the right, we can exclude scan response as a possible reason. Upon inspection of the host's connections, we find

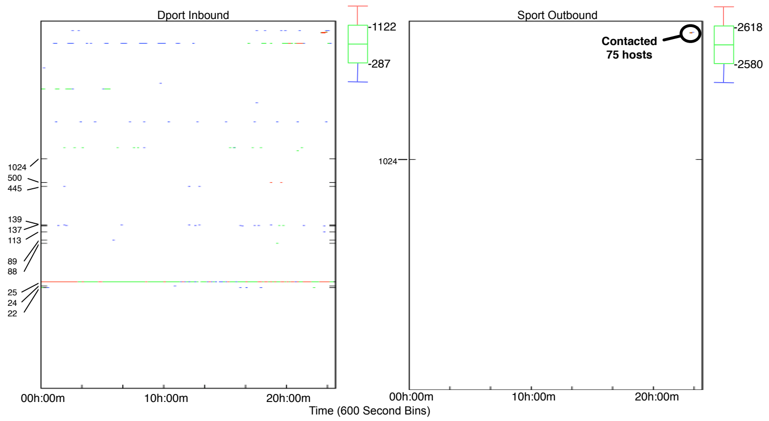


Fig. 3. Existence plot pair of a misconfiguration. The host contacts 75 mail servers in a short time.

that this activity is directed at external hosts listening on port 53 (DNS), 67 (DHCP) and 137 (NetBIOS). All of the packets observed are of similar size and have a greater than 99% fail rate over an observation period of 5 days (*e.g.* at no time in the observation period did the victims of the scan attempt to complete a connection with the host). From this we conclude that this is an internal host scanning external hosts (presumably as the result of compromise).

In addition to being able to visualize monitored hosts' scan attempts, existence plots can represent external hosts' scan successes. By concurrently repre-

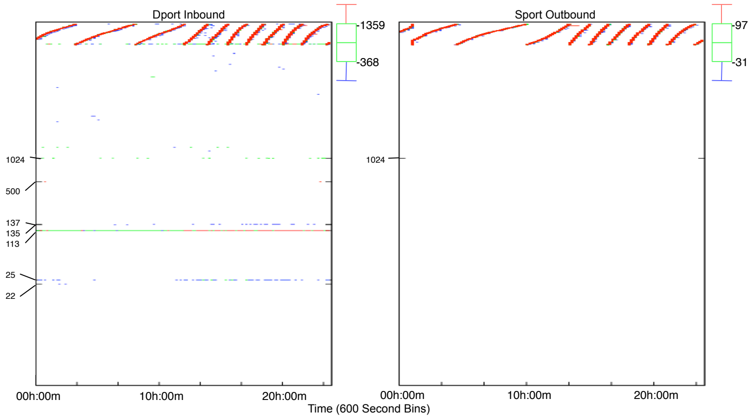


Fig. 4. Existence plot pair of a legitimate SMTP relay. The host only acts as a client, forwarding mail to external mail servers.

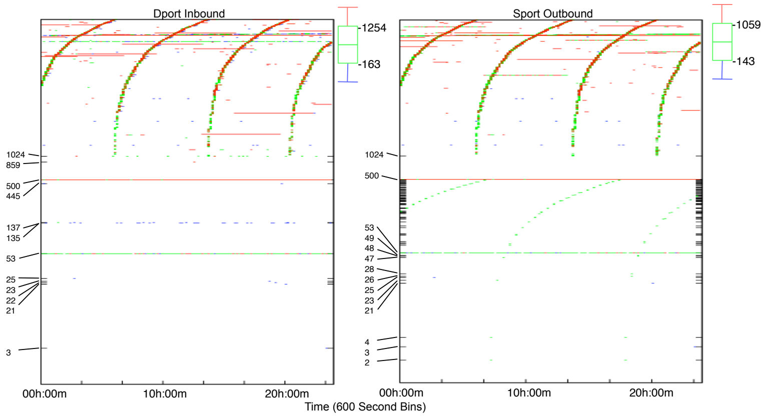


Fig. 5. Existence plot pair of a host using reserved ports to scan. Port cycling is present in the reserved port range.

senting the ports, existence plots are well-adept at demonstrating vertical scanning (*i.e.* every port on the host was contacted by the scanner) and scan response. Figure 6 represents the source port utilization of `outbound` traffic for a host over one week. During this time the host was vertically scanned twice, which is represented by two vertical bars of activity.

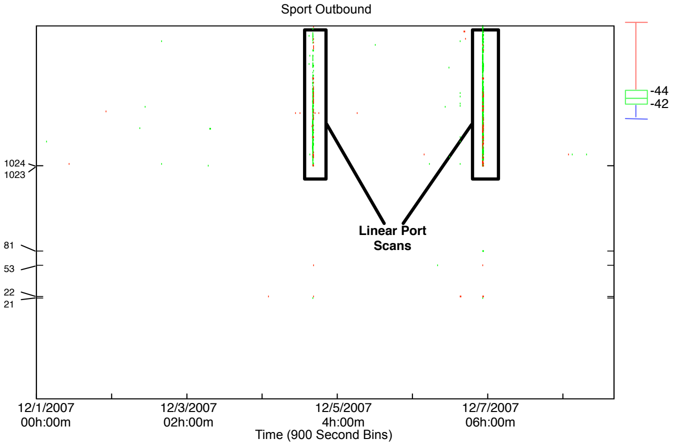


Fig. 6. Existence plot showing rampant scan response. The vertical bars denote scan response.

4 Related Works

Work centering on visualizing a single host's activities is limited in comparison to work on large-scale visualizations of network communication. Some work [3][8] centers on visualization of networks at multiple levels, but the focus is still heavily skewed to higher-level views of network interactivity, providing only rudimentary measurements of individual hosts.

Two network visualization methods centered on representing individual hosts are graphlets and heat maps. Graphlet approaches focus on visualize protocol usage patterns [4]. Mansman et al. [4] abstracts expected server behavior into gravitational entities that affect a host's position on a plane; hosts are drawn to their most prevalent activity. Heat maps aid in detecting obfuscation by comparing the commonalities among patterns of communication [2][6]. Wright et al.[6] use heat maps to classify encrypted connections. By plotting intensities of byte counts of outgoing connections versus incoming connections and duration of flows, the visualization helps to identifies "hot spots", which are attributed to specific behaviors. Existence plots are similar to heat maps insofar as the images are not based on preconceived notions of activity and are open to interpretation. Hernandez-Campos et al. [2] use heat maps for broader-scale representation of network traffic. Unlike heat maps, existence plots use only four discrete states to display magnitude instead of a continuous color map. Since varying time resolution can greatly affect the smoothness of magnitude changes (causing jarring color shifts in heat maps), we find the existence plot to be more informative.

5 Conclusions

In this work, we have demonstrated the use of a low-resolution visualization, which we call the existence plot, in representing the port usage of individual hosts, particularly in detecting hidden services and representing scanning activity. In both cases, the existence plot provides useful insight into the host's activities by concurrently representing ports' usage over time. In the future, we intend to use this visualization to further discussion about port interactivity and systematic patterns of port usage.

References

1. M. Collins, T. Shimeall, S. Faber, J. Janies, R. Weaver, M. De Shon, and J. Kadane. Using uncleanliness to predict future botnet addresses. In *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 93–104, 2007.
2. F. Hernandez-Campos, A. Nobel, F. Smith, and K. Jeffay. Understanding patterns of tcp connection usage with statistical clustering. In *MASCOTS '05: Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, pages 35–44, Washington, DC, USA, 2005. IEEE Computer Society.
3. K. Lakkaraju, W. Yurcik, and A. Lee. Nvisionip: netflow visualizations of system state for security situational awareness. In *VizSEC*, pages 65–72, 2004.
4. F. Mansman, L. Meier, and D. Keim. Graph-based monitoring of host behavior for network security. In *VizSEC*, 2007.
5. R. Pang, V. Yegneswaran, P. Barford, V. Paxson, and L. Peterson. Characteristics of internet background radiation. In *Proceedings of the 2004 Internet Measurement Conference*, 2004.
6. C.. Wright, F. Monrose, and G. Masson. Using visual motifs to classify encrypted traffic. In *VizSEC '06: Proceedings of the 3rd international workshop on Visualization for computer security*, pages 41–50, New York, NY, USA, 2006. ACM.
7. V. Yegneswaran, P. Barford, and J. Ullrich. Internet intrusions: Global characteristics and prevalence. In *Proceedings of the 2003 ACM SIGMETRICS Conference*, 2003.
8. X. Yin, W. Yurcik, M. Treaster, Y. Li, and K. Lakkaraju. Visflowconnect: netflow visualizations of link relationships for security situational awareness. In *VizSEC/DMSEC '04: Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, pages 26–34, New York, NY, USA, 2004. ACM.