ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

NETWORK TRAFFIC ANALYSIS FOR TCP COMMUNICATION

Author: Gopalkrishnan.R1 ,Msc Computer Science

Co-Author: SandhyaS2., Assistant Professor

Department of Computer Science,

Sri Krishna Arts and Science College,

Coimbatore-641008, Tamilnadu.

Email; gopskrish4@gmail.com , sandhyas@skasc.ac.in

ABSTRACT

The TCP protocol provides a standard general-purpose method for reliable delivery of data. For applications TCP provides a standard way of accessing remote computers on unreliable internetwork. This reliability is provided by adding services on top of IP. IP is connectionless and does not guarantee delivery of packets.TCP is the primary transport protocol used to provide reliable, full-duplex, virtual circuit connections. The most common use of TCP is to run it over IPv4 or IPv6, although several experimental projects have been done to run TCP on other Network layer protocols .TCP provides a means for the receiver to govern the amount of data sent by the sender. Computers that send and receive TCP data segments can operate at different data rates because of differences in CPU and network bandwidth.TCP enables many processes within a single host computer to use TCP communications simultaneously. Different processes may be communicating over the same network interface. Thus they must be separated from each other. This separation is done by using different port numbers for each process.

The most reliable way to sniff traffic is to use a network tap. A network tap is a "bump-in-the-wire" device designed only to copy traffic passing through it to a monitor port. A technique commonly used by hackers and penetration testers for getting hold of traffic in a switched environment is to use ARP poisoning. Basically ARP poisoning is a technique where two hosts on a network are tricked into sending packets destined for each other to a sniffer machine on the network.

Index Terms : ARP, Sniffing, Penetration Testing, Packet Analysis.

1. INTRODUCTION

Anonymous communication networks hide the actual source (or destination) address of Internet traffic, preventing the server (or client) and other entities along the network from determining the actual identities of the communicating parties. There are two broad types of

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

anonymity-preserving systems: low latency and high latency. Low-latency systems are designed primarily for semi-interactive applications, such as web browsing and instant messaging. Among others Tor is probably the most widely used proxy-based low- latency anonymous communication network. In Tor, clients establish circuits through a chosen set of proxies, beginning with an entry node and reaching the final destination through an exit node.

Much of the work in network traffic analysis so far has focussed on studying traffic on a single link in isolation. How-ever, a wide range of important problems faced by network researchers today require modeling and analysis of traffic on all links simultaneously, including traffic engineering, traffic matrix estimation, anomaly detection, attack detection traffic forecasting and capacity planning

Unfortunately, whole-network traffic analysis – i.e., modeling the traffic on all links simultaneously – is a difficult objective, amplified by the fact that modeling traffic on a single link is itself a complex task. Whole-network traffic analysis therefore remains an important and unmet challenge.

One way to address the problem of whole-network traffic analysis is to recognize that the traffic observed on different links of a network is not independent, but is in fact determined by a common set of underlying origin destination (OD) flows and a routing matrix. An origin destination flow is the collection of all traffic that enters the network from a common ingress point and departs from a common egress point. The superposition of these point-to-point flows, as determined by routing, gives rise to all link traffic in a network. Thus, instead of studying traffic on all links, a more di- rect and fundamental focus for whole-network traffic study is the analysis of the network's set of OD flows.

However, even though OD flows are conceptually a more funda- mental property of a network's workload than link traffic, analyzing them suffers from similar difficulties. The principal challenge presented by OD flow analysis is that OD flows form a high dimensional multivariate structure. For example, even a moderate-sized network may carry hundreds of OD flows; the resulting set of time- series has hundreds of dimensions. The high dimensionality of OD flows is in fact a prime source of difficulty in addressing the whole- network analysis problems listed above. Thus the central problem one confronts in OD flow analysis is the so-called "curse of dimensionality"

In general, when presented with the need to analyze a high- dimensional structure, a commonly-employed and powerful approach is to seek an alternate lower-dimensional approximation to the structure that preserves its important properties. It can often be the case that a structure that appears to be complex because of its high dimension may be largely governed by a small set of in- dependent variables and so can be well approximated by a lower- dimensional representation. Dimension analysis and dimension reduction techniques attempt to find these simple variables and can therefore be a useful tool to understand the original structures.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

1.1 SYSTEM SPECIFICATION

The amount of resources Wireshark needs depends on your environment and on the size of the capture file you are analyzing. The values below should be fine for small to medium-sized capture files no more than a few hundred MB. Larger capture files will require more memory and disk space.

Microsoft Windows

- The current version of Wireshark should support any version of Windows that is still within its extended support lifetime. At the time of writing this includes Windows 10, 8, 7, Vista, Server 2016, Server 2012, Server 2008 R2, and Server 2008.
- Any modern 64-bit AMD64/x86-64 or 32-bit x86 processor.
- 400 MB available RAM. Larger capture files require more RAM.
- 300 MB available disk space. Capture files require additional disk space.
- 1024×768 (1280×1024 or higher recommended) resolution with at least 16 bit color. 8 bit color should work but user experience will be degraded. Power users will find multiple monitors useful.

1.2 SOFTWARE SPECIFICATION

The specifications of the software used for analysing the traffic are as follows. And we only have front end software's which has to be installed in our system. Wireshark is a Free and Open packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

FRONTEND OS

- LINUX / UBUNTU / BACKTRACK5

BACK END SYSTEM

- ANY OS (TARGET)

SOFTWARE PACKAGES

- NMAP, WIRESHARK

EXISTING SYSTEM

System administrators, network engineers, security engineers, system operators, and programmers all use network analyzers, which are invaluable tools for diagnosing and troubleshooting network problems, system configuration issues, and application difficulties. Historically, network analyzers were dedicated hardware devices that were expensive and difficult to use. However, new advances in technology have allowed for the development of software-based network analyzers, which make it more convenient and affordable for administrators to effectively troubleshoot a network. It also brings the capability of network analysis.

The art of network analysis is a double-edged sword. While network, system, and security professionals use it for troubleshooting and monitoring the network, intruders use network analysis for harmful purposes.A network analyzer is a tool, and like all tools, it can be used for both good and bad purposes.

A network analyzer is used for:

■ Converting the binary data in packets to readable format

■ Troubleshooting problems on the network

■ Analyzing the performance of a network to discover bottlenecks

■ Network intrusion detection

■ Logging network traffic for forensics and evidence

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

- Analyzing the operations of applications

■ Discovering faulty network cards
■ Discovering the origin of virus outbreaks or Denial of Service (DoS) attacks
■ Detecting spyware
■ Network programming to debug in the development stage
■ Detecting a compromised computer
■ Validating compliance with company policy
■ As an educational resource when learning about protocols

Reverse-engineering protocols to write clients and supporting programs
Sniffing programs are included with most rootkits that are typically installed on compromised systems. Rootkits are used to cover the tracks of an intruder by replacing commands and utilities and clearing log entries. Intruders also install other programs such as sniffers, key loggers, and backdoor access software. Windows sniffing can be accomplished as part of a Remote Admin Trojan (RAT) such as SubSeven or Back Orifice. Intruders often use sniffing programs that are configured to detect specific things (e.g., passwords), and then electronically send them to the intruder (or store them for later retrieval by the intruder). Vulnerable protocols for this type of activity include Telnet, File Transfer Protocol (FTP), Post Office Protocol version 3 (POP3), Internet Message Access Protocol (IMAP), Simple Mail Transfer Program (SMTP), Hypertext Transfer Protocol (HTTP), Remote Login (rlogin), and Simple Network Management Protocol (SNMP).
One example of a rootkit is "T0rnKit," which works on Solaris and Linux.The sniffer that is included with this rootkit is called "t0rns" and is installed in the hidden directory /usr/srec/.puta.Another example of a rootkit is Linux Rootkit 5 (Lrk5), which installs with the linsniff sniffer.
Intruders may also use sniffer programs to control back doors (This practice isn't quite"common," but it isn't unheard of). One method is to install a sniffer on a target system that listens for specific information and then sends the backdoor con- trol information to a neighboring system.This type of backdoor control is hard to detect, because of the passive nature of sniffers.
cd00r is an example of a backdoor sniffer that operates in non-promiscuous mode, making it even harder to detect. Using a product like Fyodor's Nmap (http:// insecure.org/nmap) to send a series of TCP synchronize (SYN) packets to several predefined ports will trigger the backdoor to open up on a pre-configured port.

PROPOSED SYSTEM

One of the most interesting challenges in today's world is how to measure the performance of computer network infrastructures, when different types of networks are merged together. In the last few years, the data-oriented networks evolved into converged structures, in which the real-time traffic, like voice calls or video conferences, is more and more important. The structure is composed of traditional data cable or more modern fiber links, existing Plain Old Telephone Service (POTS) lines used to provide analog services (voice telephony), or digital services (ADSL, PBX, ISDN), and nowadays also of mobile and wireless networks. There are numerous

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

methods for the measurement of Quality of Service (QoS) in the current use, which provide the measurements both on the user side and in the core of the network. Internet Service Providers are interested in centralized measurements and detecting problems with particular customers before the customers start complaining about the problems, and if possible, before the problems are even noticed by the customers.

Each network carries data for numerous different kinds of applications. QoS requirements are dependent on the service. The main service-specific parameters are bandwidth, delay, jitter, and packet loss. Regarding delay, we can distinguish strict real time constraints for voice and video conferences, and interactive services from delivery in relaxed time frame. In a conversation, delay of about 100 ms is hardly noticeable, but 250 ms of delay means an essential degradation of the transmission quality, and more than 400 ms is considered as severely disturbing.

Therefore, in order to provide detailed information about the quality level for the given service in the core of the network, we need to know, what kinds of data are flowing in the network at the present time. Processing all the packets flowing in a high-speed network and examining their payload to get the application name is a very hard task, involving large amounts of processing power and storage capacity. Furthermore, numerous privacy and confidentiality issues can arise. A solution for this problem can be the use of Machine Learning Algorithms (MLAs), which use previously generated decision rules, which are based on some statistical information about the traffic. In our research, we used one of the newest MLAs – C5.0. MLAs need very precise training sets to learn how to accurately classify the data, so the first issue to be solved was to find a way to collect high-quality training statistics.

In order to collect the necessary statistics and generate the training sets for C5.0, a new system was developed, in which the major role is performed by volunteers. Client applications installed on their computers collect the detailed information about each flow passing through the network interface, together with the application name taken from the description of the system sockets. The information about each packet belonging to the flow is also collected. Our volunteer-based system guarantees precise and detailed data sets about the network traffic. These data sets can be successfully used to generate statistics used as the input to train MLAs and to generate accurate decision rules.

The knowledge about the kind of application to which the traffic belongs obtained from MLAs can be used together with traffic requirements for the given application to assess the QoS level in the core of the real network. The real traffic needs to be sampled to obtain the necessary raw statistics. Parameters like jitter, burstiness, download and upload speed (and delay and packet loss for TCP traffic) can be assessed directly on the basis of the information obtained from the captured traffic. To assess the delay and packet loss for UDP traffic, active measurement techniques must be involved (like ping measurements in both directions).

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

FEATURES

A simple search on SecurityFocus (www.securityfocus.org/tools/category/4) shows the diversity and number of sniffers available. Some of the most prominent are:

- WiresharkWireshark is one of the best sniffers available and is being developed as a free, commercial-quality sniffer. It has numerous features, a nice graphical user interface (GUI), decodes over 400 protocols, and is actively being developed and maintained. It runs on UNIX-based systems, Mac OS X, and Windows.This is a great sniffer to use in a production environment, and is available at www.wireshark.org.

WinDumpWinDump is the Windows version of tcpdump, and is avail- able at www.winpcap.org/windump. It uses the WinPcap library and runs on Windows 95, 98, ME, NT, 2000, and XP.

- Network General Sniffer A Network General Sniffer is one of the most popular commercial sniffers available. Now a suite of enterprise network cap- ture tools, there is an entire Sniffer product line at www.networkgeneral.com.
- Windows 2000 and 2003 Server Network Monitor Both the Windows 2000 Server and the Windows 2003 Server have a built-in pro- gram to perform network analysis. It is located in the "Administrative Tools" folder, but is not installed by default; therefore, you have to add it from the installation CD.
- EtherPeekEtherPeek is a commercial network analyzer developed by WildPackets. Versions for both Windows and Mac, and other network anal- ysis products can be found at www.wildpackets.com.
- TcpdumpTcpdump is the oldest and most commonly used network sniffer, and was developed by the Network Research Group (NRG) of the Information and Computing Sciences Division (ICSD) at Lawrence Berkeley National Laboratory (LBNL). It is command line-based and runs on UNIX-based systems, including Mac OS X. It is actively developed and maintained at www.tcpdump.org.
- Snoop Snoop is a command-line network sniffer that is included with the Sun Solaris OS.
- Snort Snort is a network IDS that uses network sniffing, and is actively developed and maintained at www.snort.org. For more information, refer to Nessus, Snort, & Ethereal Power Tools: Customizing Open Source Security Applications (Syngress Publishing: 1597490202) and Snort Intrusion Detection and Prevention Toolkit (Syngress, ISBN: 1597490997).
- DsniffDsniff is a very popular network-sniffing package. It is a collection of programs that are used to specifically sniff for interesting data (e.g., pass- words) and to facilitate the sniffing process (e.g., evading switches). It is actively maintained at www.monkey.org/~dugsong/dsniff.
- EttercapEttercap was specifically designed to sniff a switched network. It has built-in features such as password collecting, OS fingerprinting, and character injection, and runs on several platforms including Linux, Windows, and Solaris. It is actively maintained at ettercap.sourceforge.net.

Analyzer Analyzer is a free sniffer that is used for the Windows OS. It is being actively developed by the makers of WinPcap and WinDump at Politecnico di

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 11, March 2016

Torino, and can be downloaded from analyzer.polito.it.

■ PacketyzerPacketyzer is a free sniffer (used for the Windows OS) that uses Wireshark's core logic. It tends to run a version or two behind the cur- rent release of Wireshark. It is actively maintained by Network Chemistry at www.networkchemistry.com/products/packetyzer.php.MacSniffer MacSniffer is specifically designed for the Mac OS X envi- ronment. It is built as a front-end for tcpdump.The software is shareware and can be downloaded from personalpages.tds.net/~brian_hill/macsniffer.html.

CONCLUSION

Our understanding of the threat that traffic analysis attacks represent on public networks is still fragmented, and research in this growing field is still very active. The results we have presented should act as a warning call against ignoring this threat: traffic analysis not only can be used to collect more information in general but can also be used to bypass security mechanisms in place. A range of traffic analysis attacks have been used to degrade the security of anonymous communications networks. Long term intersection attacks (also referred to as disclosure attacks) reply on long term observations of input and output messages to detect communicating parties. Stream traffic analysis has been used to trace web requests and replies through low-latency networks. Finally the attacker can infiltrate the network or try to influence the way in which honest nodes chose paths to anonymize their traffic. Lately attacks have focused on weaker adversaries, and it has been shown that some forms of traffic analysis can be performed even without any access to the actual data streams to be traced. So little importance has been payed to securing public networks against traffic analysis that the information leaked can be detected and abused even far away from its source.