International Journal of Research Studies in Computing
2012 April, Volume 1 Number 1, 11-20

Establishing a valuable method of packet capture and packet analyzer tools in firewall

Kumar, P. Senthil
Nandha College of Technology, Erode, India (psenthilnandha@gmail.com)
S., Arumugam
Nandha College of Technology, Erode, India (dotearumugam@yahoo.co.in)

ISSN: 2243-772X
Online ISSN: 2243-7797

OPEN ACCESS

Received: 29 September 2011 **Revised**: 15 October 2011 **Accepted**: 16 October 2011
Available Online: 18 October 2011 **DOI:** 10.5861/ijrsc.2012.v1i1.43

Abstract

Packet capture is the act of capture the data packets across a computer network. Packet captures is used by the network administrators and security engineers for the purposes of Monitor network traffic, analyzes traffic patterns, Identify and troubleshoot network problems. Problem statement: The Conventional firewall is performed the packet capture followed by allowing or disallowing the packet as per user specified policy. Approach: our approach is to implement the Deep packet capture (DPC), Deep packet Inspection (DPI) and also analyze the packet in effective manner. This approach is helpful for monitor the all activates in the public or private network. Deep packet capture (DPC) is the act of capturing, at full network speed, complete network packets payload, crossing a network with a high traffic rate. Deep packet inspection (DPI) to review network packet data, perform forensics analysis to uncover the root cause of network problems, identify security threats, and ensure data communications and network usage complies with outlined policy. Some DPCs can be coupled with DPI and can result as, inspect, and analyze all networks traffic in real-time.

Keywords: packet capture; packet analyzer; deep packet capture; deep packet inspection; network traffic

Establishing a valuable method of packet capture and packet analyzer tools in firewall

1. Introduction

1.1 Packets

In Internet all traffic travels in the form of packets. A packet is a quantity of data of limited size. The entire file downloads, Web page retrievals, email, all these Internet communications always occur in the form of packets. The packet is a formatted unit of data carried by a packet mode in computer network. In computer communications links that do not support packets, such as traditional point-to-point telecommunications links, simply transmit data as a series of bytes, characters, or bits alone. Such data is formatted into packets, the bitrate of the communication medium.

A packet is a series of digital numbers basically, which conveys the following:

- The source IP address and port;
- the destination IP address and port;
- error checking information;
- Usually some sort of information about the type and status of the data being sent.

1.2 Packet framing

A packet consists of two kinds of data such as control information and user data. The control information provides data, to deliver the user data, for example: source and destination addresses, error detection codes like checksums, and sequencing information. Typically, control information is found in packet headers and trailers, with user data between them.

1.3 Packet filtering

The Packet Filtering consists of examining the incoming or outgoing packets and allowing or disallowing their transmission or acceptance on the basis of a set of specified rules. The packet filter examines the header of each packet based on a specific set of rules (Ries, 2005).

Packet filtering policies may be based upon any of the following:

- Allowing or disallowing packets based on source IP address.
- Allowing or disallowing packets based on the destination port.
- Allowing or disallowing packets according to protocol.

There are two ways in which a packet filter can be configured. In the first way, the filter accepts only those packets that it is certain are safe, dropping all others. This is the most secure mode, but it can cause inconvenience if legitimate packets are inadvertently dropped. In the second way, the filter drops only the packets that it is certain are unsafe, accepting all others. This mode is the least secure, but is a cause less inconvenience, particularly in casual Web browsing.

1.4 IP packet filter firewall

The IP packet filter firewall is to create a set of rules that either discards or accepts traffic over a network connection. The firewall itself does not affect this traffic in any way. Most packet filters have an implicit *deny all rules* at the bottom of the rules file.

Packet filters usually permit or deny network traffic based on:

- Source and destination IP addresses.
- Protocol, such as TCP, UDP, or ICMP
- Source and destination ports and ICMP types and codes
- Flags in the TCP header, such as whether the packet is a connect request
- Direction of inbound or outbound.

1.5 Methods of Packet filtering

The various method of packet filtering as follows.

A. Egress Filtering.

B. Ingress Filtering.

C. Network Ingress Filtering.

1.5.1 Egress Filtering

Egress filtering is monitoring and potentially restricting the flow of information outbound from one network to another. Packets that do not meet security policies are not allowed to leave - they are denied ***egress***. Egress filtering helps ensure that unauthorized or malicious traffic never leaves the internal network. Egress filtering may require policy changes and administrative work whenever a new application requires external network access (Ettercap, n.d.).

1.5.2 Ingress Filtering

In computer networking, ingress filtering is a technique used to make sure that incoming packets are actually from the networks that they claim to be from.

Problem in Ingress Filtering

Networks receive packets from other networks. Normally a packet will contain the IP address of the computer that originally sent it. This allows devices in the receiving network to know where it came from, allowing a reply to be routed back. However, a sender IP address can be faked. This disguises the origin of packets sent, e.g., in a Denial-of-service attack.

Solution

Filtering a packet is when the packet is not processed normally, but is denied in some way. The computer processing the packet might simply ignore the packet completely, or where it is possible it might send a packet back to the sender saying the packet is denied. In ingress filtering, packets coming into the network are filtered if the network sending it should not send packets from IP addresses of the originating computer. In order to do ingress filtering, the network needs to know which IP addresses each of the networks it is connected to may send. This is not always possible. For instance, a network that has a single connection to the Internet has no way to

know if a packet coming from that connection is spoofed or not.

1.5.3 Network Ingress Filtering

Network ingress filtering is a packet filtering technique used by many Internet service providers to try to prevent source address spoofing of Internet traffic, and thus indirectly combat various types of network abuse by making Internet traffic traceable to its source. Network ingress filtering is a ***good neighbor*** policy which relies on cooperation between ISPs for their mutual benefit.

1.6 Filtered capture

Packet capture devices may have the ability to limit capture of packets by protocol, IP address, MAC address, etc. With the application of filters, only complete packets that meet the criteria of the filter (header and payload) are captured, diverted, or stored.

2. Packet capture

The packet capture is the act of capture at network speed, network packets crossing a network with a high traffic rate. Once captured and stored, either in short-term memory or long-term storage (Deri, n.d.). Partial packet capture can record headers without recording the total content of datagram. This can reduce storage requirements, and avoid legal problems, but yet have enough data to reveal the essential information required for problem diagnosis.

2.1 Deep Packet Capture (DPC)

Deep Packet Capture has the ability to capture packet data from the data link layer on up of the ISO-OSI Reference model. This includes headers and payload. Headers include information about what is contained in the packet. The payload includes the actual content of the packet. The Deep Packet capture encompasses every packet that crosses a network segment, regardless of source, protocol or other distinguishing bits of data in the packet. Deep packet capture is the unrestricted, unfiltered, raw capture of all network packets. Our system is to implement the method of Deep Packet Capture (Bendrath, 2009; Office of the Privacy Commissioner Of Canada, n.d.; Porter, 2010).

2.2 Deep packet Inspection (DPI)

Deep Packet Inspection is a form of network packet filtering that examines the data part of a packet and it passes an inspection point, searching for protocol non-compliance, viruses, spam, intrusions or predefined criteria to decide if the packet can pass or if it needs to be routed to a different destination, or for the purpose of collecting statistical information (Bendrath, 2009; Office of the Privacy Commissioner Of Canada, n.d.; Porter, 2010). Deep Packet Inspection enables network management features, user service, and security functions as well as internet data mining, eavesdropping. DPI is currently being used by the enterprise and governments in a wide range of applications.

3. Packet analyzer

A packet analyzer sometimes called as network analyzer, protocol analyzer or sniffer, or Ethernet sniffer or wireless sniffer (Spangler, 2003). The packet analyzer is a computer program or a piece of computer hardware that can intercept and log traffic passing over a part of a network (Chan, 2002; Wikipedia, 2010).

3.1 Packet analyzer tool: Capsa

Capsa is a packet analyzer tool; it was by developed by Colasoft. This tool is used for network administrators to monitor troubleshoot and analysis wired & wireless networks. Currently, there are four editions

available: Capsa Enterprise Edition, Capsa WiFi, Capsa Professional Edition, and Capsa Free Edition. Our system is to implement the Capsa Packet analyzer tool.

Main Functions in Capsa packet analyzer tool:

- Wired & wireless network real-time packet capturing.
- Traffic & bandwidth monitoring.
- Advanced protocol analysis.
- Multiple network behavior.
- Network activity logging.
- In-depth packet decoding.
- Captures packets from a single or multiple network adapters.
- Analyzes the header & contents of each packet.
- Provides statistics on MAC & IP address.
- Presents statistics in graphs.

3.2 Capsa Enterprise Edition

Capsa Enterprise Edition is supports the both Ethernet and networks. It mainly performed the real-time packet capturing and analysis as well as supporting past-events analysis. It is a very useful tool for enterprise network administrator to help them deal with daily network work.

3.3 Capsa WiFi

Capsa WiFi is a WiFi wireless network analyzer developed by Colasoft for IEEE 802.11 a/b/g/n wireless network monitoring, analyzing and troubleshooting.

3.4 Packet analyzer tool: Omni Peek

Omni Peek is a packet analyzer software tool. It is used for network troubleshooting and protocol analysis. It supports a plug-in API. Our system is also to implement the Omni Peek Packet analyzer tool.

4. Related discussion

4.1 Traffic generation model

A traffic generation model is a stochastic model of the traffic flows in a network. A packet generation model is a traffic generation model of the packet flows or data sources in a packet-switched network. These models are useful during the development of telecommunication technologies, in view to analyze the performance and capacity of various protocols, algorithms and network topologies.

4.2 Packet erasure channel

The packet erasure channel is a communication channel model where sequential packets are either received or lost. An erasure code can be used for forward error correction method.

4.3 Measuring network throughput

Throughput of a network can be measured using various tools available on different platforms.

Reasons for measuring throughput in networks

The Human People are often concerned about measuring the maximum data throughput in bits per second of a communications link or network access. A typical method of performing a measurement is to transfer a ***large*** file from one system to another system and measure the time required to complete the transfer or copy of the file. The throughput is then calculated by dividing the file size by the time to get the throughput in megabits, kilobits, or bits per second.

4.4 Packet square

Packet Square is a free and open-source pcap-based network protocol testing tool. It is used for testing devices, network troubleshooting.

Features in Packet square

- Protocol field value modification.
- Packet deletion.
- Packet duplication.
- Packet reordering.
- Fragmentation of packets.
- Interface selection for sending packets.
- Option for sending a single selected packet or all packets.

4.5 Packet crafting

Packet crafting is a technique that allows network administrators or hackers to probe firewall rule-sets and find entry points into a targeted system.

The act of packet crafting can be broken into four stages:

A. Packet Assembly

B. Packet Editing

C. Packet Play

D. Packet Decoding

4.5.1 Packet Assembly

Packet Assembly is the creation of the packets to be sent. Some popular programs used for packet assembly are Hping, Nemesis, Ostinato, Cat Karat packet builder, Scapy and Yersinia. Packets may be of any protocol and are designed to test specific rules.

4.5.2 Packet Editing

Packet Editing is the modification of created or captured packets. This involves modifying packets in

manners which are difficult or impossible to do in the Packet Assembly stage, such as modifying the payload of a packet. These modified packets can be saved in packet streams which may be stored in pcap files to be replayed later.

4.5.3 Packet Play

Packet Play is the act of send a pre-generated or captured series of packets. The packets may come from Packet Assembly and Editing or from captured network attacks. This allows for testing of a given usage or attack scenario for the targeted network.

4.5.4 Packet Decoding

Packet Decoding is the capture and analysis of the network traffic generated during Packet Play. In order to determine the targeted network's response to the scenario created by Packet Play, the response must be captured by a packet analyzer and decoded according to the appropriate specifications.

5. Sample implementation

Our system is to implement the Deep packet capture and Packet analyzer using the software tools called Colasoft and smart sniffer. The figures (Figures 1 to 4) represent the packet analyzer and the figures 5 and 6 represent the packet capture.

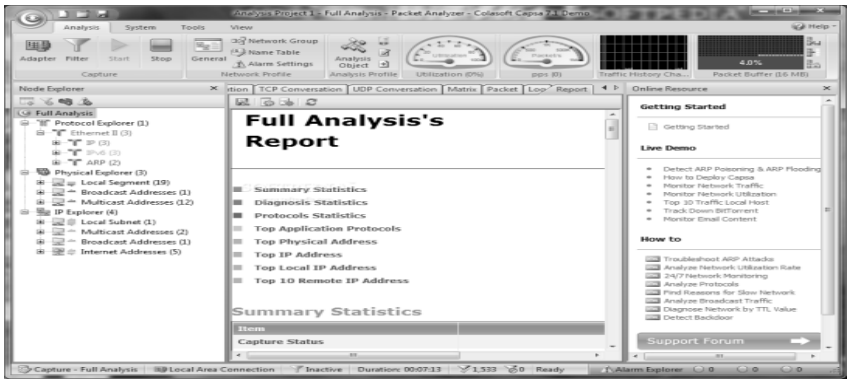

Figure 1. Full analysis report

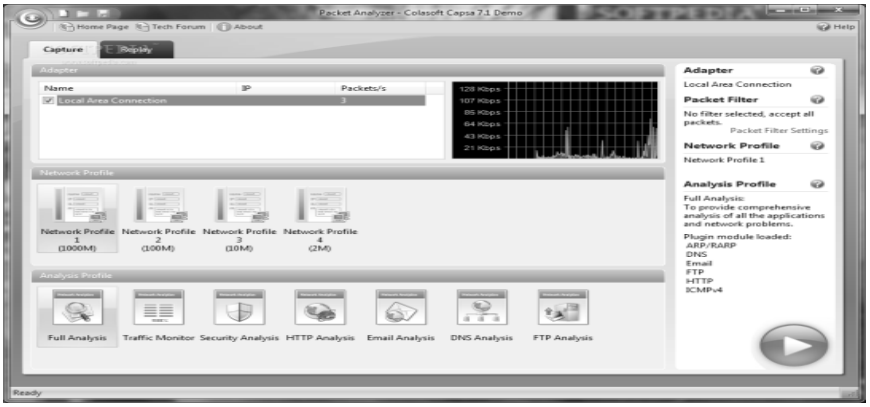

Figure 2. Full analysis report

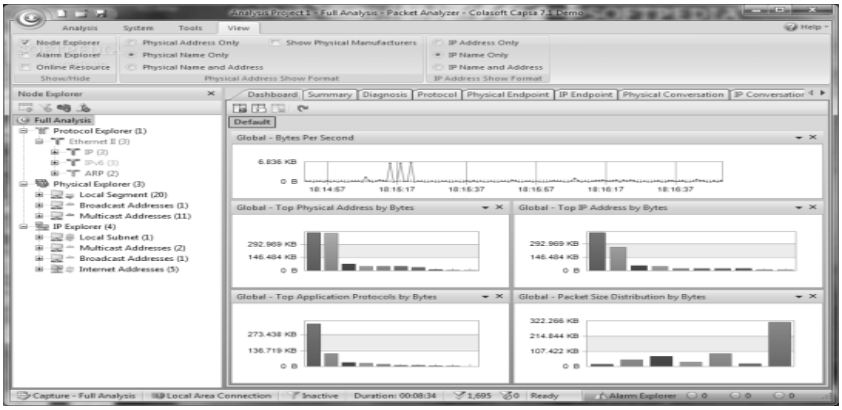

Figure 3. Complete analysis graph

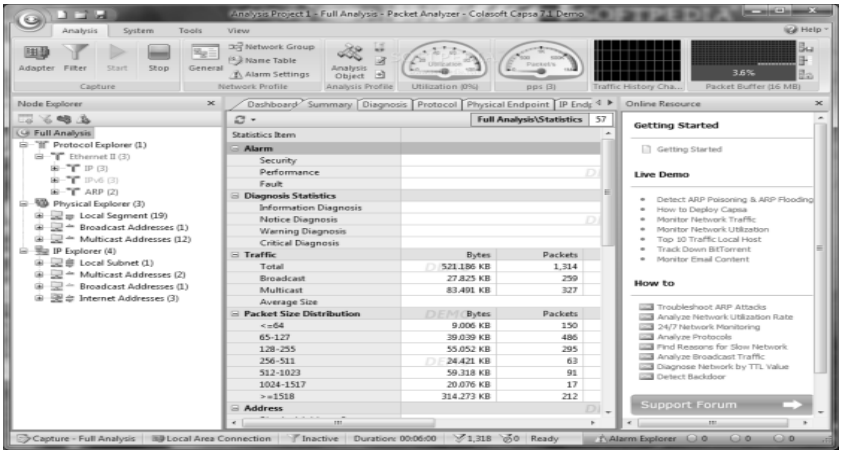

Figure 4. Full analysis in traffic packets

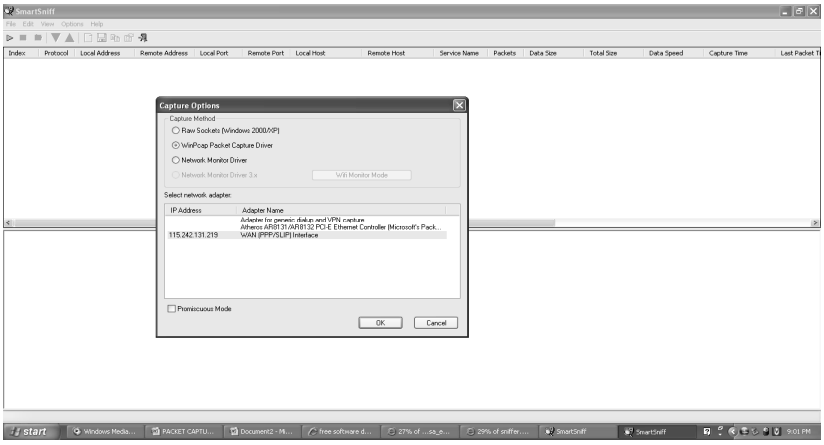

Figure 5. Capture option

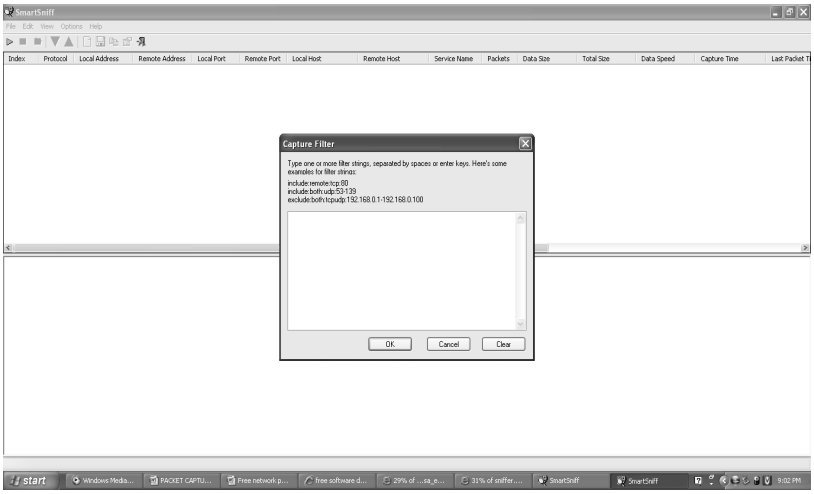

Figure 6. Capture filter

6. Conclusion and future work

The many tools used for Packet capture in network traffic that researcher used in their work, but there is a limitation in their work. Our system is to implement the packet capture and packet analyzer in the firewall traffic. In firewall concept, Packet capture approach is to monitor the all activates in the network and also stored the each incoming and outgoing packet. Our system is to implement the Deep packet capture (DPC) that helps to capturing at full network speed, complete network packets payload, crossing a network with a high traffic rate. Once captured and analyzed means stored in memory. Deep Packet Inspection (DPI) is to review the network packet data, identify the security threats, and ensures data communications and network usage complies with outlined policy. Some DPCs can be coupled with DPI and can result as, inspect, and analyze all networks traffic in real-time.

Packet Capture is also very helpful tool for troubleshooting difficult IP network issues. This field technician can gather all data for the technical expert who can perform detailed analysis of the problem and find the right strategy for resolving it. The computers communicate over networks; they normally listen to the traffic in the networks. However, Packet analyzer has the ability to enter promiscuous mode, which allows them to listen to all network traffic regardless of if it's directed to them.

About the authors: *P. Senthilkumar* is a faculty member in the Department of Computer Science and Engineering, Nandha College of Technology, Erode, Tamil Nadu, India. He received his M.E. (CSE) in 2008 from Anna University, Chennai, Tamil Nadu and Pursing Ph.D. in Anna University – Coimbatore, Tamil Nadu. His area of research interest includes Networks and Network Security, Mobile Ad-HOC networks. He has Published 4 International Journals and Conferences.

Dr. S. Arumugam received the PhD Degree in Computer Science and Engineering from Anna University, Chennai in 1990. He also obtained his Bachelor in Science, Major in Electrical and Electronics and Masters in Science, Major in Applied Electronics Engineering Degrees from P.S.G College of Technology, Coimbatore, University of Madras in 1971 and 1973 respectively. He worked in the Directorate of Technical Education, Government of Tamil Nadu from 1974 at various positions from Associate Lecturer, Lecturer, Assistant

Professor, Professor, Principal, and additional Director of Technical Education. He has guided 4 PhD scholars and guiding 10 PhD scholars. He has published 70 technical papers in International and National journals and conferences. His area of interest includes network security, Biometrics and neural networks. Presently he is working as Chief Executive Officer, Nandha Engineering College Erode.

7. References:

Bendrath, R. (2009). *Global technology trends and national regulation: Explaining variation in the governance of deep packet inspection*. Retrieved October 1, 2011, from http://userpage.fu-berlin.de/~bendrath/Paper_Ralf-Bendrath_DPI_v1-5.pdf

Chan, C. Y. (2002). *A network packet analyzer with database support*. Retrieved October 2, 2011, from http://www.cs.rpi.edu/~szymansk/theses/chan.ms.02.pdf

Deri, L. (n.d.). *Improving passive packet capture: Beyond device polling*. Retrieved October 1, 2011, from http://www.net-security.org/dl/articles/Ring.pdf

Ettercap. (n.d.). *Ettercap*. Retrieved October 4, 2011, from http://ettercap.sourceforge.net/

Office of the Privacy Commissioner Of Canada. (n.d.). *Just deliver the packets*. Retrieved October 2, 2011, from http://dpi.priv.gc.ca/index.php/essays/just-deliver-the-packets/

Porter, T. (2010). *The perils of deep packet inspection* Retrieved October 2, 2011, from http://www.symantec.com/connect/articles/perils-deep-packet-inspection

Ries, C. (2005). *Defeating windows personal firewalls: Filtering methodologies, attacks, and defenses*. Retrieved October 5, 2011, from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.117.5306

Spangler, R. (2003). *Packet sniffer detection with antisniff*. Retrieved October 4, 2011, from http://www.linux-sec.net/Sniffer.Detectors/snifferdetection.pdf

Wikipedia. (2010). *Packet analyzer*. Retrieved October 4, 2011, from http://en.wikipedia.org/wiki/Packet_analyzer