Global Information Assurance Certification Paper

Tracking DDoS Attacks by cooperating packet recording system (CPRS)
Jia-Chyi Wu
January 8, 2002

Abstract

By studying the paper of 'Tackling Network DoS on Transit Networks' [1], I try to deliver a related concept of an approach to the similar scheme. I also suggests some conceptual diagrams for reference. DDos attacks, which employ forged source address, have proven to be a troublesome issue for Internet Service Providers (ISPs). DDos attacks often cross multiple ISPs. This paper describes a concept to develop a system called co-operating packet recording system (CPRS) to detect DDos attacks. With cooperation among the ISPs, the CPRS can cross network domains to track attacks. With CPRS, ISPs can quickly detect the source of IP spoofing and notify the administrator of the domain of resource network. Functions of the CPRS include data recording, data querying and data sharing. In order to maintain the backbone security especially in tracking DDos attacks, extensive use of the CPRS by ISPs may prove beneficial.

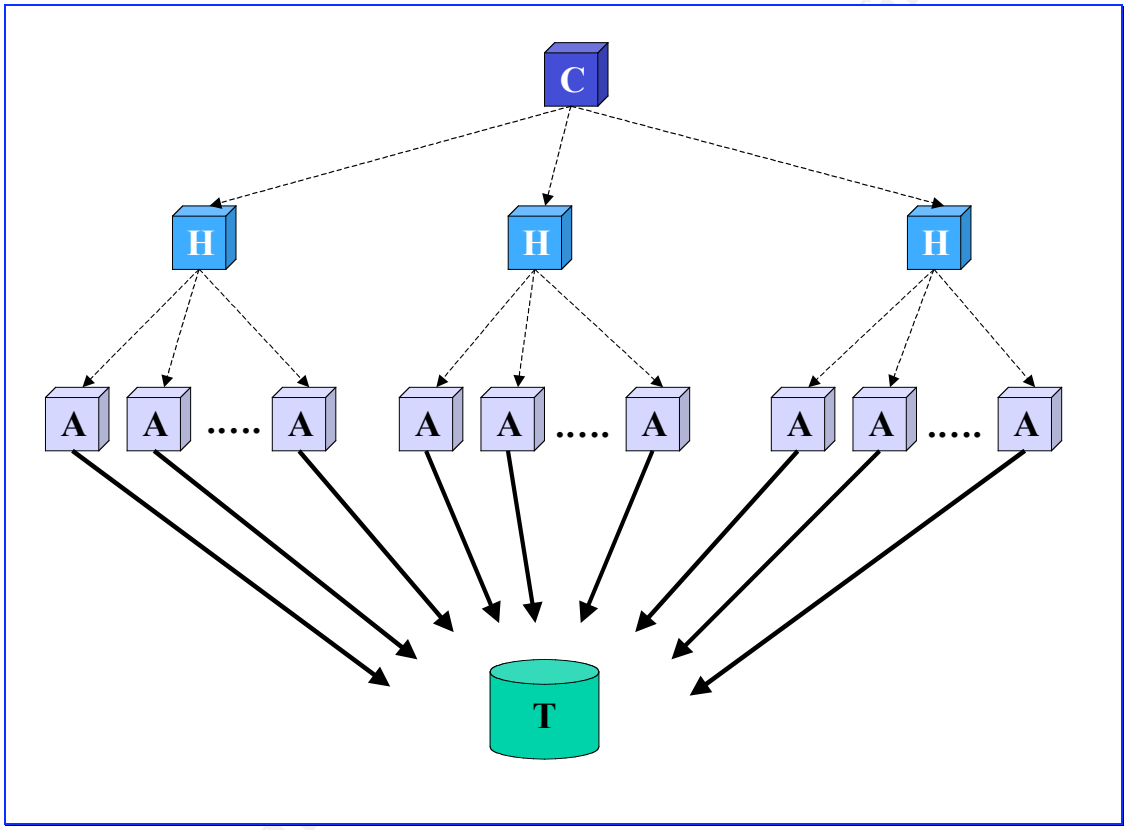
Keywords: Distributed Denial of Service (DDos), co-operating packet recording system (CPRS), Internet Service Providers (ISPs), agent, console, secure socket layer (SSL), SYN Flooding, ICMP packet, IP Spoofing, TCP Flag

1. Introduction

With the advent of the broadband era, network users are demanding increasingly, thus resulting in an ever-expanding communication infrastructure. ISPs are facing an out-channel boom with versatile internet applications prevailing over the world. The complicated network infrastructure and the tremendous data flow have made the prevention of attacks on network and incident handling even more difficult. Moreover, These years have seen the emergence of new attacks, In particular, has caused vest damage to the distributed denial-of-service (DDoS) attack famous web servers such as Yahoo and Amazon. All these attacks posed problems to network security.[2]

DDoS attack is a mutation of DoS, an attack on a network or computer. Its primary aim is to disrupt access to a given service. DDos utilizes distributed sources to achieve its DoS purpose(Fig. 1)[3]. The attacker on host C(Client) intrudes multi hosts H(Handler) to install the handler program. Intruding more hosts A(agent), the attacker installs the real attack program on A(agent). Each handler host controls some agent hosts when attacking the target host T. When an attack starts, a lot of packets are sent from many agents to victim host T. A denial-of-service (DoS) attack floods a network with requests, which slows or halts normal traffic. In a distributed DoS attack, a malicious hacker hijacks multiple computers, turning them into "zombies" that flood a network with bogus requests.

Because the packets of DDoS use IP spoofing technique[4][5], it is difficult to judge which is the correct source packet. The network must trace back one by one via gateways to determine the real source. This kind of method consumes a lot of effort and requests immediate tracking when an attack occurs; otherwise, it is incapable of handling the incident in time. Moreover, blocking the bandwidth is the main purpose of DDos attack. Even when the victim of DDos can detect the source packet of attack and block it with some mechanisms, if the upstream ISPs are not willing to cooperate in detecting such attack packet, continuous arrival of packets from the upstream can not be retarded.



[fig. 1 DoS Concept]

In view of the above, a system called co-operating packet recording system (CPRS) that fosters cooperate among the ISPs is developed. The CPRS can cross different network domains to track attacks. With CPRS, ISPs can quickly detect the source of IP spoofing and notify the administrator of the resource network to take action. The system is described in detail as follows.

2. Objectives and features of CPRS

Objectives

- Coordinate different network domains to track attacks simultaneously.
- Alert when specific patterns are detected.

- Foster cooperation among ISPs.
- Monitor packets when bad behavior pattern and notify administrators when necessary.
- Acquire and store packets with special pattern and administrators can then analyze these packets to see if they constitute an attack.
- Provide the administrators with information of the source and content of packets for prompt detection of the attacker’ s location.
- Enable different networks or ISPs to share stored tracking data so as to save time in tracing attack packets.

Features
- Tracking system across ISPs.
- Distributed architecture.
- Communication within the system need to be encrypted as Secure Socket Layer (SSL) .

3. Functions of CPRS

CPRS contains three primary functions as follows.
- Data recording

This function acquires data including packet header information such as source IP/port, destination IP/port and protocol etc that can represent uniquely that packet [6]. Data structure in memory can be transferred and then recorded in hard disc as database or file type. Administrators can decide how long the data are to be kept.

- Data querying

This function not only can compare the suspicious packets with attack pattern to check if there is any attack packet going through the gateway of network, but can also query the upstream gateway of network that installed the same system. Tracking the source of suspicious packets requires tracing the route of the attack packet. Secured mechanisms like digital certificate or SSL has to be used for remote query between different ISPs in order to prevent unauthorized data being acquired or compromised.

- Data Sharing

Administrators can analyze and filter the behavior of the attack packet and share the information obtained with other network domains. CPRS can be regarded as a defense mechanism of the network.

4. Advantages of CPRS

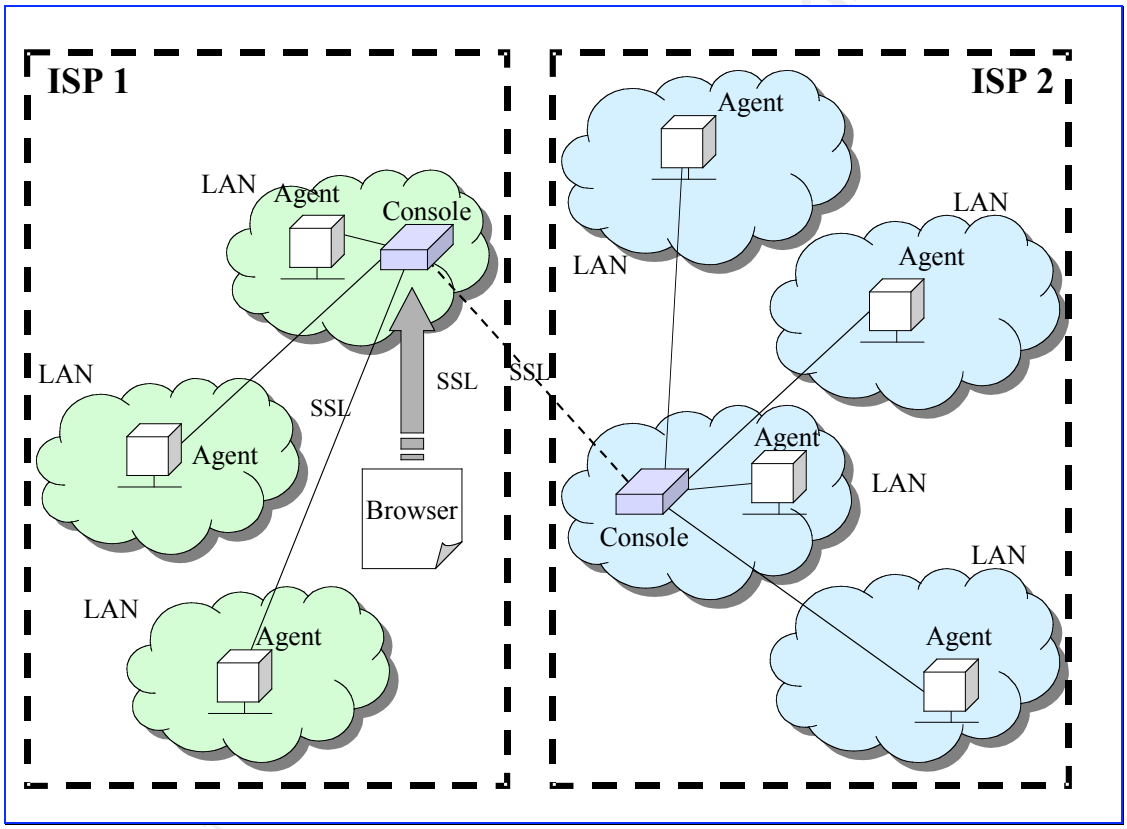
There are three main advantages as follows.
- Shorten the time needed to inspect IP spoofing when an attack occurs.
- Achieve tracking among different networks. Few existing network security systems have such a function.

- Alert different ISPs when an attack occurs.

Because CPRS relies on coordination among different networks, ISPs must install CPRS on an important nodes on their backbone and allow limited right to access data of one another. The more the ISPs adopt the CPRs, the more useful the system will be, offering greater advantages to the ISPs.

5. System configuration

One important issue of the CPRS is that how the total system achieves security via effective authentication mechanism and access right. It is the concept picture of CPRS as follows.



[fig. 2 CPRS Concept]

The concept first describes how to track any attack packet in the network itself and query other networks. It can alert CPRS other networks when special conditions such as Nimda pattern are detected. CPRs consists of three major components, namely agent, console and secure socket layer (SSL).

- Agent

An agent mainly acquires packets using filter rule to examine suspicious packet and then stores them in memory. Then data are recorded on hard disk as files. Accumulated history data can offer query for other coordinate users. An agent should have the following functions..

I. Digital Certificate: This digital certificate can be used to verify the identity of agents and consoles. This function also ensures confidentiality and integrity among different nodes of agents and consoles.
II. Filter Rules: Filtering rules should be applied to traffic received at agents. Each filter rule corresponds to one specific action such as dropping, storing or alerting. For easy management, the filter rules should be performed by the console.
III. Filter and capture mechanism: An agent captures the primary information such as Source IP, Source Port, Destination IP, Destination Port and Protocol . It also compare the traffic using filter rules and converts data structure to record these data.
IV. Alert: An agent not only determines which kind of traffic to be recorded, but also triggers alert to console via sending alert message.
V. Storing: An agent can transfer the memory data to hard disk in specific format. This data can analyzed later by the console.
VI. Searching: An agent can search the history data and process queries from the console.

- Console

The console mainly provides the interface for comparing the suspicious packet pattern with the history data the agent stored. This function can check sure if an attack packet has gone through the network or not and it is useful to track the upstream network continuously. In addition, the console also provides data sharing between ISPs to allow some query right to other networks. So it is possible to trace the routes of the attack packet. The console should have the following function.

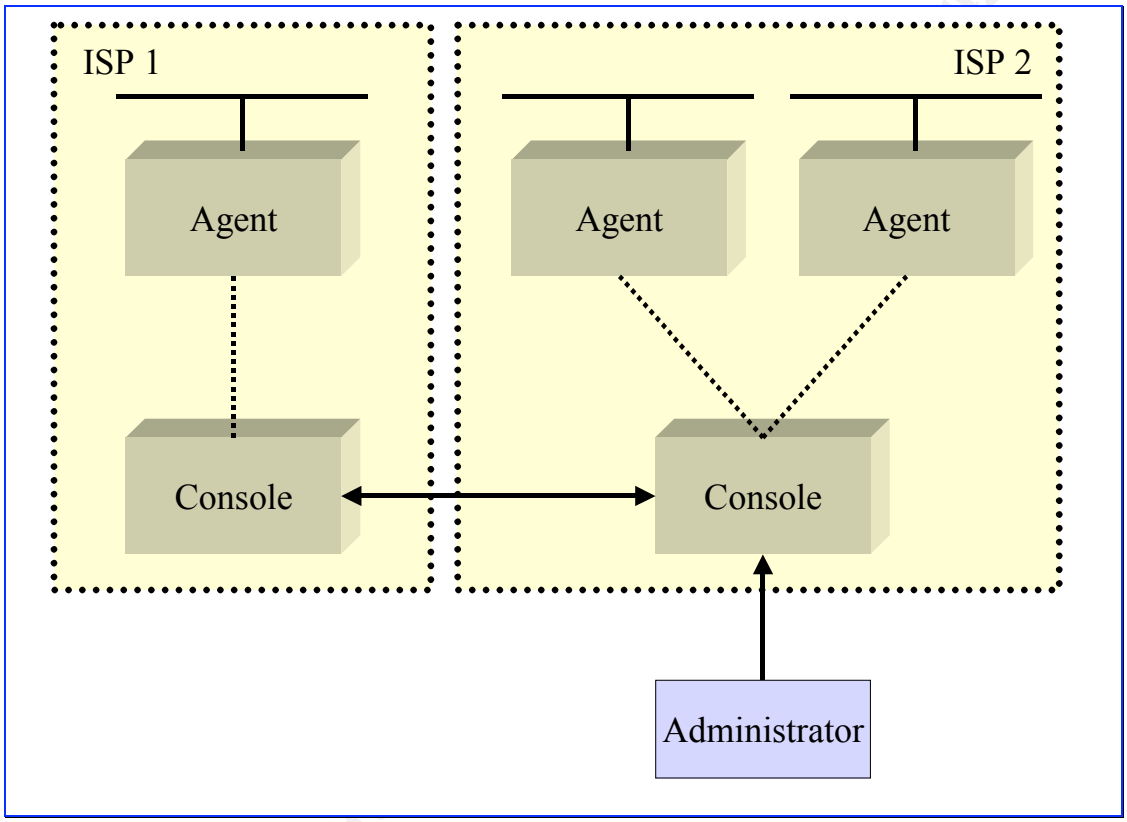
I. Digital Certificate: This digital certificate function has the same function as described above.
II. Access Rule: This rule defines the name list of the agents within the controlled domain. It also defines the name list of allowed consoles and what kind of permission is allowed. Administrators can adjust these rule.
III. Centralized Management Interface: The console can first start or stop the operation of all or specific agents. It can then monitor the performance of the agents and modify the filter rules of agents. Finally, it can deal with the alert message and inform the right person responsible to take action.
IV. Inquiry Interface: The console allows the user interface to query the history packets of the controlled domain. If there are no attack packets detected within the controlled domain, the console can use the information exchange proxy to query other consoles to judge the source of an attack.
V. Information Exchange Proxy: A console needs the information exchange proxy to query other consoles that are not controlled.

- Secure socket layer (SSL)

The application of SSL must be encrypted on network within CPRS to ensure the security

during data communication. In upper picture [fig. 2 CPRS Concept] there are three portions that required SSL protection. The first is the connection between the user node of Web and the Console, the second is the connection between administration node of console to other agents, and the third is the connection between the local console and the coordinated console during data sharing.

6. Scenarios of CPRS usage



[fig. 3 CPRS architecture]

The CPRS architecture is shown in figure 3. The following scenario is suggested for effective use of CPRS.

- Set up agents at the network that need monitoring in order to capture data.

A console may monitors several agents called controlled agent list (CAL). The console can also add a new agent into CAL.

I. Nomenclature: This nomenclature can be used to represent the agent.
II. IP address: This IP address represents the real IP of the host that the agent is installed.
III. Port: This port represents the port of service that the agent is running.
IV. Duration of packets stored: The duration of packets stored represents the stored time of captured packets. If it is over the stored time limit, the history packets will be erased.
V. Detecting period: The detecting period can check sure if an agent has been alive or not.

VI. Space size occupies limit when alert: CPRS need a big space of hard disc to store the captured packets. So it need a mechanism to monitor the space of occupied by stored data for alerting by email to the administrator. The mechanism may overwrite the history packets or stop recording packets when it reaches the limit..

- Set the filter rules and alert rules in agents.

With different agents in different domain networks, users need capture packets in different ways, thus resulting in an different filter rules. The more details the filter rules be specified, the precise the packets will be captured. The filter rules should have the following attributions.

I. Nomenclature of filter rule: This usage is used to identify the filter rule such as Nimda which means that the filter rule for capturing the packets of Nimda attack patterns.
II. Range of IP: An agent can regard the source IP or destination IP as a kind of filter rule.
III. Port: An agent can regard the source port or destination port as a kind of filter rule.
IV. Length of Packet: An agent may turn on the filter rule of length of packet accompany the filter rule of the ICMP protocol. For example, the packet traffic of ping command using ICMP protocol which should cause not many bytes of length. It is especially useful to turn the ICMP and input the length of packet.
V. Protocol Type: It general has TCP, UDP and ICMP protocols. Most popular attack patterns are used by these protocols.
VI. TCP Flag: Check the six flags of the TCP packet. These flags are URG, ACK, PSH, RST, SYN and FIN.
VII. Check the content of packet: This filter rule allows to check the specific patterns in content. For example, a type of Nimda attack has the "winnt/system32/cmd.exe?" in the content of that packet. A type of Code RED attack has the "GET /default.ida" in the content of that packet.

- Start the agent to capture data.

The console can first start or stop the operation of all or specific agents.

- Query the data in agents through the consoles.

The console should have the following query rules.

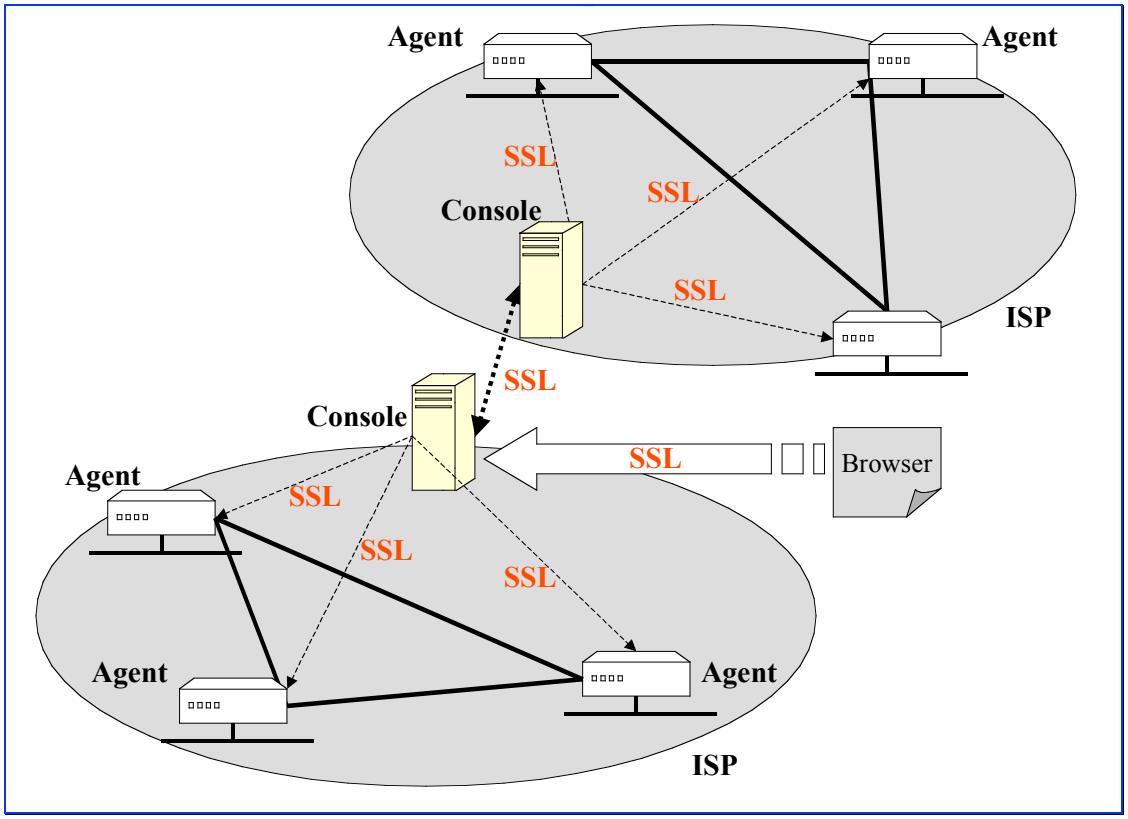
I. Default filter rule: Some default filter rule are available for quickly setting for capture packets.
II. Query traget: A console can monitor several agents. An administrator can select one or more of them to view the recorded packets.
III. Data for display: An administrator can select header of packet to be displayed only. Otherwise, the content of packet will be displayed with header.
IV. Time of packed captured: Because the recorded traffic is usually very large, so it saves time by using this condition.

V. IP scope: A console can define the source IP or destination IP as a kind of query condition.
VI. Port: A console can define the service port as a kind of query condition.
VII. Packet Length: A console can define the same packet length as described above.
VIII. Protocol: A console can define the same protocol rule as described above.
IX. TCP flag: A console defines the six flags of TCP protocol as a kind of query condition.

- Query the data in other consoles.

A console may query its controlled agents. It also can query another consoles which are allowed some data be viewed. A console holds a list of allowed consoles to share its recorded packets.
Two consoles in different networks use the public key system to process certificate authentication (CA). Two consoles share their public key and use their private keys to trust each other.

7. Network architecture for CPRS

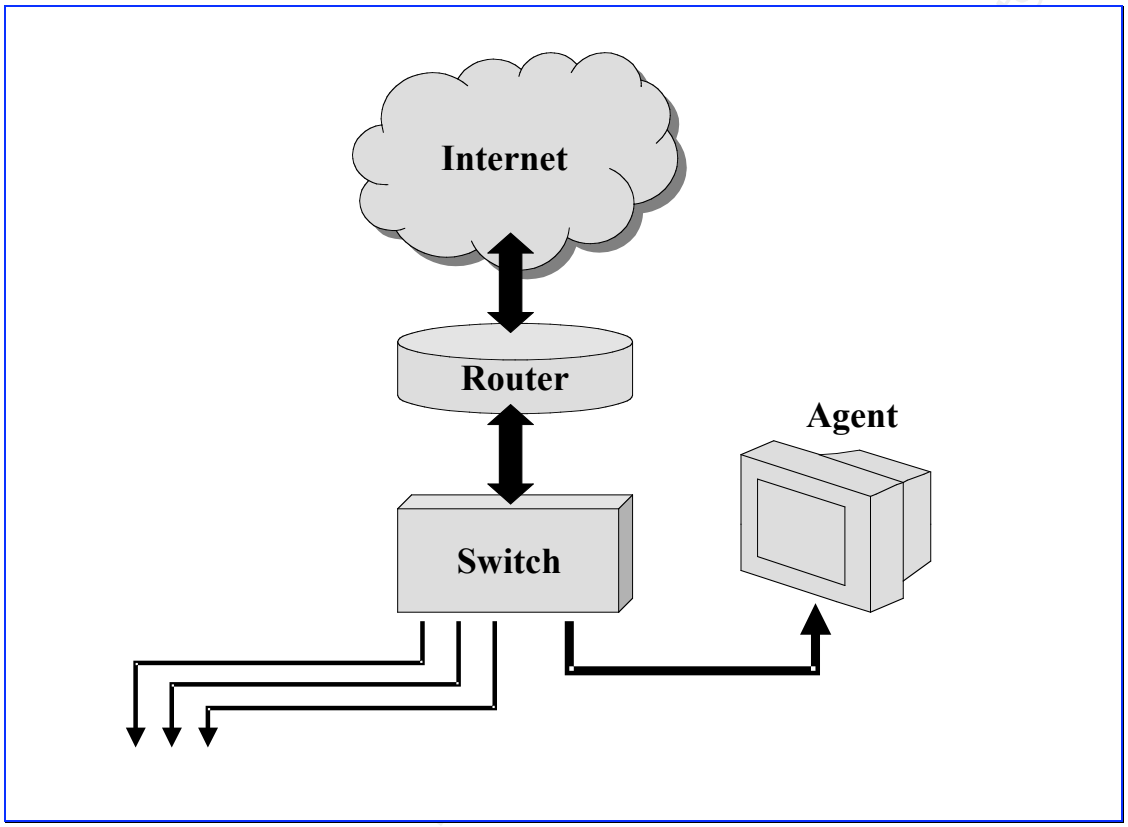


[fig. 4 The Network architecture for CPRS]

The network architecture for CPRS is shown in figure 4. An agent should be installed at the outlet of each subnet to monitor the incoming packets and store them in its host. Each network is made up of some subnets and the console is employed to control and monitor

the status of agents within that network. As to the structure of the network, a star topology with the console in the center and surrounded by several agents is recommended. The administrator communicates through the local console with consoles of other networks to query the authorized data.

8. Physical network architecture concept for CPRS



[fig. 5 The physical network architecture concept for CPRS]

The physical network architecture concept for CPRS is shown in figure 5.

- Agent

The agent is installed on the connection of the network. The administrator need to set all packets through the switch mirror to the port of the connected agent. This enable the packets to be analyzed through the switch. The agent itself must set the related filter mechanisms to be performed by the specific console.

Because the agent play a monitor role, the little impact on the network the better it is. In order to avoid the load on the router, the agent does not connect to router directly. The agent connects to L2 switch which defined a monitor port for capturing all the packets through the network interface. The agent then stores the packets.

With making sure the security of integrated CPRS, all agents must be strictly controlled by secured process before added into internet.

- Console

The console can be installed in any machine. It allows permission of access to other networks by SSL connection for querying stored tracking data. The console itself must

set the related filter mechanisms to be managed by the administrator only.

9. Popular patterns to be set into the filter rules[7]

For the IP spoofing ability, the ICMP protocol design and implementation, the UDP protocol design and implementation, and the general TCP connection management implementation in most operating system, one can easily launch a DoS attack by generating a large mount of network traffic of different protocol（e.g. TCP, UDP, or ICMP）with spoofed IP address. The victim hosts or networks that receive a lot of annoying packets may be confused and busy to process all the packets. This will degrade the performance of the services, hosts or networks of the victims.

- SYN Flooding[8]

By the agent of CPRS captures and records SYN flooding packets and then uses the console to query the actual source network. The capture rule of SYN flooding can check on the TCP protocol and the SYS flag. When the SYS flag occurs over a limit, an agent start to capture the packets to be analyzed.

- ICMP packet[9][10]

By the agent of CPRS, to capture and record tremendous spoofing source IP of ICMP packets and then uses the console to query the actual source network. ICMP may cause DDos attack. The capture rule checks on the ICMP protocol and the length of packet. When the echo request occurs over a limit, an agent starts to capture the packets to be analyzed.

- Packets made up of abnormal TCP flags

There are some abnormal TCP flags. The SYN+FIN in packet is a case. The other case, X’ mas Tree packet, containing mixed of six TCP flags may cause some bad results in communication.

- UDP packet[11]

UDP provides no reliable message delivery, flow control and error recovery. These features make UDP easy to address spoofing. Thus malicious users can easily spoof UDP-based applications. The CERT Coordination Center has received reports of programs that launch denial-of-service attacks by creating a "UDP packet storm" either on a system or between two systems. An attack on one host causes that host to perform poorly. An attack between two hosts can cause extreme network congestion in addition to adversely affecting host performance. The capture rule checks on the UDP protocol . When the UDP booms, an agent starts to capture the packets to be analyzed.

- Specific content packet can cause DDos attacks.

The agent captures and records the content of packets with specific pattern such as Nimda or CodeRed and then alerts the console.

Nimda patterms:[12]

The scanning activity of the Nimda worm produces the following log entries for any web server listing on port 80/tcp. So, these may be the packet contents of Nimda attack.
GET /scripts/root.exe?/c+dir

GET /MSADC/root.exe?/c+dir
GET /c/winnt/system32/cmd.exe?/c+dir
GET /d/winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET /_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET
/msadc/..%5c../..%5c../..%5c/..\xc1\x1c../..\xc1\x1c../..\xc1\x1c../winnt/system32/cmd.exe
?/c+dir
GET /scripts/..\xc1\x1c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0/../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0\xaf../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x9c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%2f../winnt/system32/cmd.exe?/c+dir

Code Red patterns:[13]

Code Red worm infect target machine by sending special form of HTTP request which contains code of worm. The request may look like:
/default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNN%u9090%u6858%ucbd3%u7801%u909%
u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c
3%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a

10. Special issues

- Privacy

ISPs have the right to protect the data of their clients. A lot of data of enterprises or users are usually confidential. Therefore, if the CPRS is developed and utilized, the legal issue has to be defined. Some principles must be laid down clearly. An administrator must be punished for wrong usage of CPRS. The issues concerning rights and privilege have great importance in CPRS.

- Effectiveness

The more the networks adopting the CPRS, the more effective it is in tracing the attacker. If only a few networks use the CPRS, it is the like the Tcpdump only. So CPRS ought to emphasize coorporation.

- Hardware

The disk of an agent host must be large enough to store the history data.

- Performance

An agent should be installed on the connection of the network. The administrator has to set all packets going through the switch mirror to the port of the connected agent. There is no doubt that the agent of the CPRS will impact the traffic flow of networks.

- Programming

An agent needs the effort to program.[14]

11. Conclusion

In summary, the development of CPRS is likely to provide an ideal scheme for tracking the DDos attacks among the ISPs. It attempts to capture malicious patterns of traffic through the node of ISPs. Implementing this kind of distributed approach to detecting and dealing with DDos attacks would require considerable cooperation among different networks and routers over which illegitimate traffic might pass. This papers describes how to detect and track attacks among different ISPs, which has been rarely reported in literature.

List of Reference

[1]Tackling Network DoS on Transit Networks
http://www.dante.net/pubs/dip/42/42.html
[2]Denial-of-service threat gets IETF's attention
http://www.nwfusion.com/news/2000/0724itrace.html
[3]Distributed Denial of Service Defense Tactics
A Quick Review of the Attack Model
http://razor.bindview.com/publish/papers/DDSA_Defense.html
[4]Consensus Roadmap for Defeating Distributed Denial of Service Attacks
A Project of the Partnership for Critical Infrastructure Security
http://www.sans.org/ddos_roadmap.htm
[5]CERT Coordinate Center-Denial of Service attacks
http://www.cert.org/tech_tips/denial_of_service.html
[6]Egress Filtering v 0.2
http://www.sans.org/y2k/egress.htm
[7]Backtracking Spoofed Packets
http://www.epm.ornl.gov/~dunigan/oci/bktrk.html
[8] CERT Advisory CA-1996-21 TCP SYN Flooding and IP Spoofing Attacks
http://www.cert.org/advisories/CA-1996-21.html
[9] CERT Advisory CA-1998-01 Smurf IP Denial-of-Service Attacks
http://www.cert.org/advisories/CA-1998-01.html
[10]ICMP
http://www.icir.org/vern/papers/reflectors.CCR.01/node5.html
[11]CERT Advisory CA-1996-01 UDP Port Denial-of-Service Attack
http://www.cert.org/advisories/CA-1996-01.html
[12]CERT Advisory CA-2001-26 Nimda Worm
http://www.cert.org/advisories/CA-2001-26.html
[13].ida "Code Red" Worm

http://www.eeye.com/html/Research/Advisories/AL20010717.html
[14]Writing a Basic Packet Capture Engine
http://www.cse.nau.edu/~mc8/Socket/Tutorials/section3.html

Last Updated: July 24th, 2017

Upcoming Training

SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event