INMON CORP.
1404 IRVING ST., SAN FRANCISCO, CA 94122
WWW.INMON.COM

Using sFlow and InMon Traffic Server for Intrusion Detection and other Security Applications

1 Introduction

With organizations becoming more and more dependent on network infrastructure for business critical applications, unauthorized access to networks can have a significant impact on the ability of a business to function. An organization must continuously defend against external and internal security threats. A continuous onslaught of denial of service attacks, port scans, and system infiltration requires constant vigilance. While external threats are serious, studies have shown that insiders do the most damage.[1] Monitoring for internal threats requires much broader coverage than monitoring a well defined perimeter for external security threats.

This application note describes how the sFlow network monitoring technology and InMon Traffic Server can be used to detect and protect against unauthorized network access, with the main focus on the monitoring capabilities to detect an intrusion. It also compares sFlow to other technologies for network security.

2 Overview of network security threats

Network administrators are usually concerned about four main network security threats:

- **Reconnaissance:** probing or mapping the network to identify targets, examples are ping sweeps and port sweeps; both of which are usually a precursor to an actual exploit attempt.
- **Denial of Service (DoS):** attempts to consume bandwidth or computing resources in order to prevent a host communicating on the network, common examples are Smurf attack or SYN floods. A Distributed DoS (DDoS) attack is very similar, except that the attack appears to originate from multiple machines. These machines ("zombies") have usually been compromised and have been stealthily programmed to start an attack on a signal from the real attacker.
- **Exploits:** attempts to gain access to or compromise systems on the network, often seen as repeated failed login attempts, or TCP hijacking.
- **Misuse:** attempts to violate organizational policy, for example using disallowed services or including unauthorized content in e-mail or ftp transfers.

3 Requirements for an intrusion detection system

In order for attacks to be detected and responded to promptly, an intrusion detection system should meet certain requirements:

- Network-wide, continuous surveillance
- Reliable, timely availability of data, especially during network overload, which is common during an attack
- Interpretation of traffic patterns
- Alerts to violations or threats
- Provision of sufficient information to take action

4 sFlow and its application to network security

4.1 sFlow overview

sFlow was designed to provide continuous network-wide monitoring of L2-L7 flows in high-speed switched and routed environments. sFlow works by monitoring accurate interface counters and statistically sampling forwarding decisions in a switch. Both of these types of data are immediately forwarded to a central data collector (e.g. InMon Traffic Server) that performs analysis, resulting in detailed, real-time information on application level data flows. This data is used to identify security threats and determine appropriate actions.

4.2 sFlow and its application to network security

Attacks and security threats will come from unknown sources. For effective security monitoring, complete network surveillance, with alerts to suspicious activity is required. sFlow provides a blanket audit trail, for the whole network. No configuration of the monitoring system is required. This complete and continuous network surveillance coupled with InMon Traffic Server's ability to correlate traffic with routing information allows security threats to be traced rapidly. In addition, Traffic Server provides a history of traffic patterns and supports historical queries. This allows a baseline to be established, from which anomalies can be detected and suspicious activity identified.

[1] SecurityPortal, October 27, 2000 <http://securityportal.com/articles/abuse20001027.html>

Some forms of security threat require protocol following to fully identify and diagnose. Sampling-based monitoring systems are not appropriate for this application.

4.2.1 Detecting and diagnosing threats with sFlow and InMon Traffic Server

There are some basic techniques for detecting and diagnosing security threats by looking at the traffic patterns:

- Look for the Top N hosts associated with suspicious traffic
- Look for changes in traffic patterns, for example use of new services or new users of services
- Use of historical traffic patterns to explore the extent of a threat

This section gives some examples of how sFlow and InMon Traffic Server use these techniques to detect security threats and determine appropriate action.

4.2.1.1 Reconnaissance

A reconnaissance attack is characterized by a single source accessing an abnormally large number of destinations. To identify this threat, Traffic Server can be used to generate a list of the top sources of traffic by the number of peers. Peers may be host addresses and/or UDP/TCP ports.

4.2.1.2 Denial of Service

A DoS attack is characterized by a large number of spurious requests sent in short period of time to overload network resources. Typically, this traffic appears to be no different from legitimate traffic, but the volume is far greater.
sFlow and Traffic Server can be used to identify and determine controls for DoS attacks. The following example of a Smurf attack illustrates the general techniques.

A Smurf attack occurs when an attacker sends an ICMP echo request (also referred to as a "ping") packet to the broadcast address of a subnet containing a large number of host machines. The source address of the packet is altered to that of the intended victim, typically a web server. The hosts in the subnet respond with ICMP echo replies to the victim. This quickly overwhelms the network and the victim, effectively denying service.

To detect and defend against this attack:

- **Identify the victim**: use the real-time traffic monitoring capability to identify top destinations of ICMP echo response
- **Identify the subnet sourcing the ICMP echo response**: use the real-time display, custom filter to show top subnets sourcing ICMP echo responses
- **Identify port through which ICMP echo response enter site:** use the real-time display, custom filter to show top (router/switch) ports receiving ICMP echo response packets destined for the victim.
- **Block the attack:** install access control filter on router/switch port that will block ICMP traffic from the source subnet to the victim.

In a distributed DoS attack, it is important to understand which hosts have been compromised. After Traffic Server real-time monitoring capability has been used to address the immediate concerns of quenching a DoS attack, the historical traffic data can be used to identify the compromised machines. This can be done by querying the Traffic Server to generate a list of the top sources of traffic to the identified victim, during the period of the attack. Top sources will be the compromised machines that should be examined for the illicit software that generated the attack traffic.

4.2.1.3 Exploits

Whilst sFlow and Traffic Server cannot be used to identify failed login attempts, because the complete packet sequence between end node is not captured, it is possible to identify exploit attempts by looking at top sources of login-type traffic. The compromised machines can then be identified by looking at the top destinations of the login-type traffic, sourced by the identified exploiter.

4.2.1.4 Misuse

Unauthorized access, for example outside access to local machines, can be identified using Traffic Server's audit trail and baselining with anomaly detection functionality. For example, a report can be set up to look at all external access to local hosts, reporting daily on additions to the traffic patterns. This report will identify external hosts, volume of traffic transferred and top local hosts accessed. From here suspects can be identified and Traffic Server can be used to investigate further:

- Which systems have been compromised and on which days: Top talkers to suspect by day over the last week.
- At what time of day were systems accessed: Top talkers to suspect by hour on the busiest day.

- Was there any system hopping: Correlate traffic going to the compromised local host with traffic going to the suspect - Top talkers to accessed local host during the time when the suspect was also accessing the local host.

Another example of unauthorized access is the use of unauthorized services (e.g. doom or napster). Traffic Server can be used to identify top sources and destinations of any undesired service.

However, sFlow only looks at packet headers. This means that it cannot determine if unauthorized content is being transferred in ftp or email for example.

4.2.2 Detail and accuracy

A monitoring system based on sampling, by its nature, will not provide absolutely accurate data. However, sampling provides the detailed data required to detect and understand most security threats.
In order to handle most security threats, having data that indicates the existence of certain types of traffic is more important than having accurate information on traffic volumes. Most security threats generate enough traffic to be seen by sampling and hence the existence will be reported. With the sFlow system, in which there is no up front aggregation or filtering of traffic data, detailed information is available. For example, using sFlow, Traffic Server reports on anomalous traffic with, characterization of end-nodes, services, ports, routes, interfaces etc. This level of detail is required to take effective action to defend against an attack.

4.2.3 Monitoring and analysis system behavior during a DoS attack

Typically large volumes of traffic are generated during an attack. This places a heavy load on the network infrastructure. At these times, it is especially important for traffic data to be available in a timely manner. A well-designed monitoring system will continue to operate effectively under heavy network load.

sFlow places no appreciable load on network device processing and does not cache data or buffer packets. So even when a switch or router is heavily loaded with forwarding large volumes of traffic, the sFlow monitoring system will continue to operate. sFlow samples will continue to be taken and forwarded to a central data collector. In addition, the central data collector is unlikely to become overloaded with traffic data in this situation since the volume of data has been reduced through sampling.

sFlow has been designed to use an inherently unreliable data transport (UDP) to forward samples to the central data collector. Sample packet loss results in a slight reduction of the effective sampling rate. In network overload conditions, although some samples will be dropped, some will arrive at the central data collector and the data collector will still be able to build a representative picture of the network traffic.

Aggregation-based traffic monitoring systems for example NetFlow, in which traffic data is aggregated at source and then unreliably forwarded to an analysis station, are very sensitive to packet loss. The loss of a NetFlow datagram means that the analysis station does not receive data on traffic flows. Under heavy load, loss of NetFlow packets is common, because of buffer overflow at the device generating NetFlow, or the NetFlow collector being unable to keep up with the volume of NetFlow packets being generated, or because of packet loss in the network. In any of these cases, the data loss is unrecoverable and the NetFlow collector will be unable to build a representative view of network traffic.

5 Other monitoring methodologies for network security

Traditional intrusion detection solutions involve either an external monitor attached to a SPAN or monitor port of a switch (e.g. Cisco Secure Intrusion Detection Sensor), or a module that is installed in a switch slot (e.g. Cisco Secure Intrusion Detection Module). In either case the network administrator configures the intrusion detection monitor to inspect certain types of traffic and downloads traffic recognition patterns or signatures. The intrusion detection module operates by filtering the traffic stream according to the configuration and then using an internal pattern recognition engine to match the filtered packet stream to the downloaded signatures. Matches will trigger events that are logged to a central management console (eg Cisco Secure Intrusion Detection Director).

In addition to displaying and managing alarms from the intrusion detection monitors, the management console can manage configurations for multiple monitors, archive data associated with attacks, and forward events to other systems.
This technique is effective for detecting Exploits or some Misuse attacks that require deep packet analysis of protocol following where it is necessary to decode a sequence of packets.

Since traffic data is filtered at the monitor, if a security violation occurs that either involves filtered out traffic or does not match a signature, then the network administrator must use another monitoring tool to diagnose and rectify the situation.

With this scheme, analysis of security violations is performed at the data collection source. There is no automatic correlation between traffic flows through other devices. This makes it much more difficult to understand the "big picture" of an attack and defend against it at source.
Both of these draw backs make this technique less effective for detecting and protecting against Denial of Service and Reconnaissance attacks.

System log file analysis can be used to identify security violations. However, log files tend to contain huge amounts of data and sophisticated, and often, manual analysis is required. In addition, it is often difficult to correlate log files, and time synchronization is required. Log file analysis can be very effective if there are hints on where to look. A continuous monitoring system that highlights anomalies (e.g. sFlow and Traffic Server) can enable directed log file analysis.

6 Automatic action on detection of security violation

On detection of a security violation or a known attack pattern, some steps can be automated.

6.1 Shunning

When a security violation or intrusion is detected, it is possible for the system to be configured to take action automatically to defend against the attack. This is often referred to as shunning. In this case, on detection of a specific traffic pattern a network device's access control lists are reconfigured and reloaded to deny access to a specific host or entire network. This type of automated response should only be configured for attack signatures with a low probability of false positive detection, such as an unambiguous SATAN attack. Shunning can itself be a security problem. It is possible for a denial of service attack to use the shunning mechanisms to maliciously deny service.

6.2 Alarms and events

Given the likelihood of false positive attack detection and the vulnerability of shunning itself to be used to deny service, it is often better to respond to attack detection manually. In this case the system raises an alarm or logs an event that provides enough information for the situation to be understood and appropriate manual action taken.

7 Conclusion

sFlow provides a cost effective mechanism for complete network surveillance at media speed. The interpretation of flows across the whole network quickly outlines the "big picture" of an attack, allowing the network administrator to implement the most effective controls.