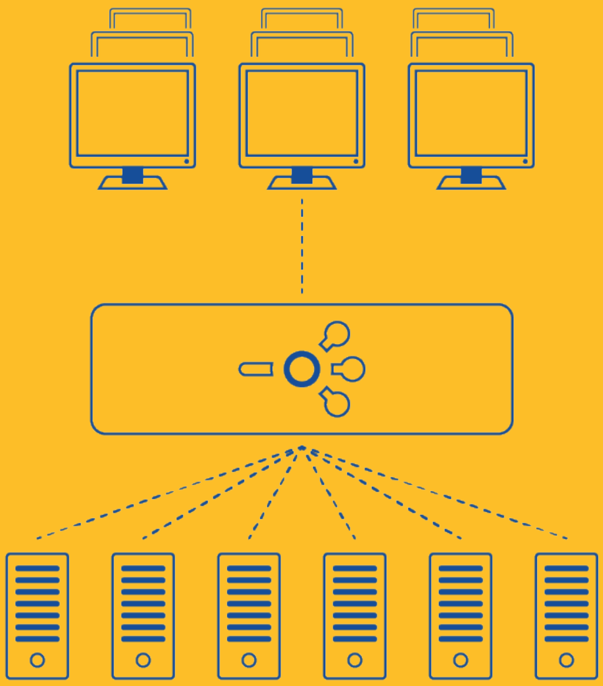
ADVANCED ANALYSIS
OF DATA NETWORK TRAFFIC

Infrastructure Monitoring vs. Network Traffic Monitoring



by Pavel Minarik, Chief Technology Officer at Flowmon Networks

The traditional understanding of network monitoring covers server and service availability, CPU and RAM utilization, the status of a particular network interface, or the number of transferred packets. It is a foundation for the work of any network administrator. However, there is more to the matter than just the red/green status. This paper describes the difference between traditional infrastructure monitoring and next-generation network monitoring (or NPMD - Network Performance Monitoring and Diagnostics), which helps to tackle the many diverse issues that arise in modern IT environments.

The traditional concept of monitoring, as mentioned above, is performed by SNMP (Simple Network Management Protocol), which delivers an overview of the IT infrastructure to provide network administrators with information about the availability of each component. This level of information detail is the barest and most essential minimum for network monitoring and troubleshooting.

Imagine a situation where an unexpected anomaly causes a sudden traffic spike. The administrator learns about the increased number of packets and volume of data transferred through network interfaces, but that is all. They get nothing on the origin of this anomaly, what device lies at the root of it, or what protocols and services may be involved - in other words, actionable information they can actually use for troubleshooting. Traditional infrastructure monitoring cannot provide an answer to these questions Because it does not analyze the network traffic as such, it cannot provide information about its structure.

Network Traffic Monitoring

Modern corporate infrastructure is too complex to be efficiently managed by simple infrastructure monitoring. It stands to reason - how does red/green status help deliver a consistent end-user experience? This is why NPMD tools have been developed. They provide answers to problems that are far more complex and multifaceted than those of the past and ask for insight that is more granular, yet noise-free.

One method that provides enough granularity is packet analysis. It inspects all the information contained in the exchanged packets, including the content, and extracts relevant information for the user. While the level of detail it provides is tremendous, it has to handle very large volumes of data, making it very performance-intensive and also somewhat inflexible.

Another method is NetFlow/IPFIX analysis. It generates statistics both on the underlying data transfers abstracted from packets, and the overall subject of the communication (though the actual content of the communication is not stored). These statistics are known as network flow data and can be thought of as a list of telephone calls. It shows who communicates with whom, when, how long and how often, but the subject of the conversation is hidden. To put it in the language of data networks, flow data bears records of IP addresses, data volumes, time, ports, protocols and other characteristics of TCP/IP communication at the third and fourth network layers.

Returning to the anomaly mentioned above, a simple query of data flows immediately reveals that it occurred during communication with an FTP service (TCP protocol, port 20; see Figure 1) when a local PC (with the IP address 192.168.34.78) uploaded a large amount of data to a public server.

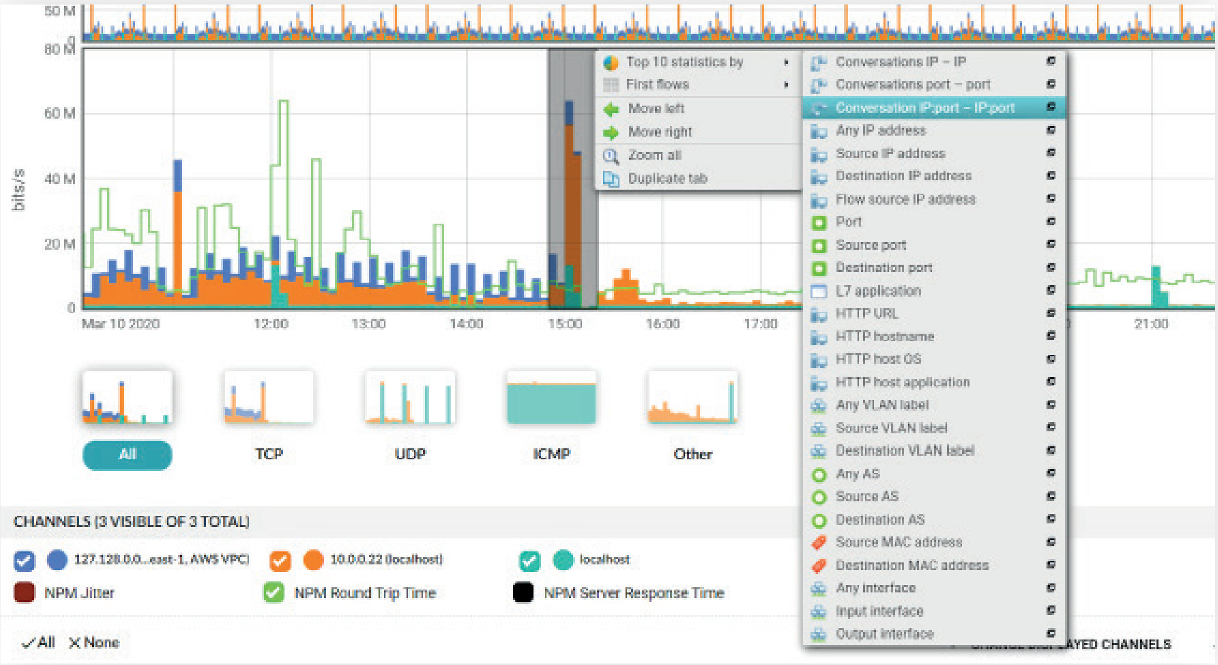


FIGURE 1: MONITORING CAPABILITIES AND ADVANCED TRAFFIC ANALYSIS

Flow Data

There are numerous different flow data standards available, the most important one being NetFlow developed by Cisco. NetFlow is available in several versions. The original NetFlow v5 is now considered obsolete, as it lacks the support for some essential traffic details, specifically IPv6 traffic, VLAN numbers or MAC addresses. These shortcomings were corrected in NetFlow v9, which is template-based and enables flexible settings of the monitored traffic information. The latest standard is known as IPFIX (or NetFlow v10), and its development is the outcome of an effort to standardize NetFlow by IETF (RFC 5101, RFC 5153).

NetFlow statistics are either generated by network elements (routers, switches) or by dedicated probes. The probes are transparently connected to the network as passive appliances, producing a precise and detailed stream of statistics. Cloud platforms such as AWS or Azure can deliver statistics called FlowLogs, which are similar to NetFlow. This process does not impede network performance because the data is generated from a copy of network traffic. This approach is used to overcome various performance and feature limitations of router-based NetFlow monitoring.

It is always important to check the router/switch documentation to make sure it supports NetFlow and if so, which version. Sometimes they require testing because older nodes may suffer from performance issues, fail to provide precise enough statistics or have a limited scope of the network traffic characteristics they can monitor.

Flow data is not the sole domain of Cisco; in fact, there are a number of compatible alternatives out there, e.g. jFlow, cFlow, or NetStream.

Processing Flow Data

Full utilization of flow data requires a tool that is capable of collecting, storing, displaying and analyzing the network statistics. One such tool is the NetFlow collector. It can be a specialized piece of software or a hardware appliance with the appropriate software installed. Collectors ensure that data traffic statistics are stored, reported and analyzed centrally. Thanks to this, network administrators are provided with an immediate overview of traffic structure, receive reports on network utilization, and possess a powerful troubleshooting tool.

Consider the following scenario. A user at a remote branch experienced slowly loading websites and poor response from internal systems located at the company headquarters. They contact the administrator at the headquarters, complaining that “something is wrong.” After a glance at the monitoring system, the administrator learns that one end station has been downloading large amounts of data from the internet. Based on this finding, they can proceed to ask the user to exercise more restraint or stop downloading completely.

Behaviour Analysis

The major benefits of network traffic monitoring technologies are data protection and IT security. The analysis of network statistics provides a completely new overview of the monitored infrastructure, which allows detecting infections, malicious activities, attacks, or other network anomalies. This technology is known as Network Traffic Analysis (NTA).

Unlike signature-based approaches, which check the traffic against a database of known threats, NTA is able to detect unknown, insider, and advanced threats because it analyzes the traffic for deviations from normal baseline traffic and calculates the probability of the anomaly being malicious. In this way, NTA can pick up indicators of compromise in the early stages, and help avoid breaches and denial of service.

For illustration, here is another scenario. A curious user opens the attachment of an email they received. This attachment contains a script, which installs malicious code onto their computer. This code then seeks out internal network servers running Windows and begins trying different combinations of credentials for Remote Desktop Services. From time to time, it encrypts the sensitive data it collected and sends it to a server somewhere on the other side of the planet. How do security tools respond to such a situation? An antivirus will certainly be able to stop many such threats, but it is powerless against targeted attacks and custom-made malware. Once the antivirus has allowed the user to run an infected attachment, there is hardly anything it can do to stop the infection from progressing. A secured perimeter and advanced firewall do not help here since the above-described activity will not appear at the perimeter at all.

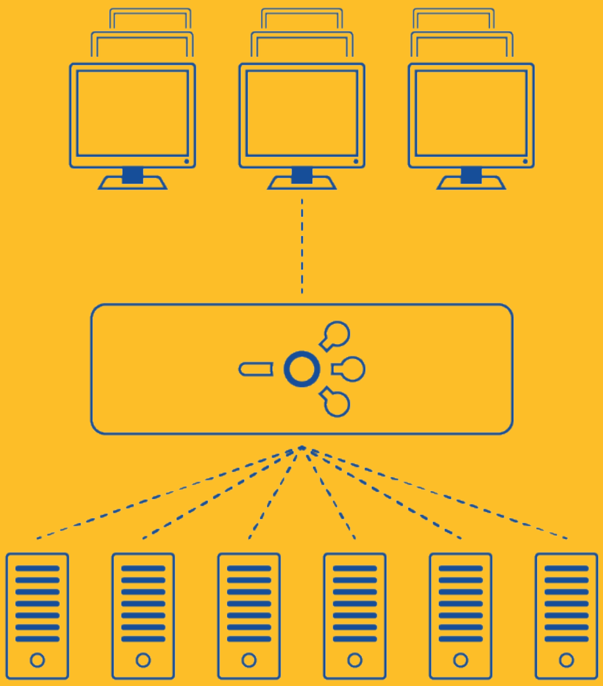
A monitoring system equipped with NTA responds differently. It automatically reveals the attack the moment the malicious code starts seeking servers in the internal network (i.e. a horizontal port scan) and when it attempts to connect to Windows Remote Desktop (a dictionary attack). It also detects and reports the malware’s communication with the control center located elsewhere in the world.

Summary

This is an introduction to the technology of network traffic monitoring using flow data. This technology is designed to deliver detailed network visibility and enable effective network administration, troubleshooting, and the detection of security issues and threats. It is therefore fair to say that it is a technology that brings company networks under control.

ADVANCED ANALYSIS
OF DATA NETWORK TRAFFIC

Full Packet Capture and Analysis



by Pavel Minarik, Chief Technology Officer at Flowmon Networks

When it comes to network traffic monitoring, troubleshooting, or threat detection, there are two approaches available - the NetFlow-based traffic monitoring described in Part 1 and full packet capture.

First, a reminder of what flow-based (NetFlow, IPFIX) network traffic monitoring means. Flow data is an abstraction representing the network traffic. Flow data statistics are created as an aggregation of the network traffic using the source IP address, destination IP address, source port, destination port, and protocol numbers as attributes that identify the individual flow records. The content of the communication is not included, producing an achievable aggregation of about 250 to 500:1. Thanks to flow data, administrators can analyze traffic structure, identify end-stations transferring large amounts of data, or troubleshoot network issues and wrong configurations.

Flow data can also be utilized in security. Such technology, known as Network Traffic Analysis, approaches detection differently than signature-based methods. Instead of looking through a database of malicious payloads, it observes patterns of behavior, detects anomalies and deviations, and calculates their probability of being malicious. This allows it to recognize and respond to undesirable behaviors or yet unknown threats for which no signature is available.

Packet Analysis

Packet analysis inspects the content of communication to decide whether it is legitimate or not. By nature, there is no aggregation, compression or trimming involved - the data is stored in its original size. Therefore, this method is extremely performance- and storage-intensive. Just imagine capturing traffic in a network running at 250 Mbps on average. That is equal to a data load with more than 31 MB per second, 1.8 GB per minute, 108 GB per hour, and 2.6 TB per day. In the case of 10 Gbps networks, the numbers become astronomical, equalling 100 TB of data stored per day. However, large volumes of data are not the only downside. The principal limitation of packet analysis is encrypted traffic. Without the encryption key, the content of any of the transferred data, and often not even the transfer protocol or application, cannot be accessed. And still, the amount of encrypted traffic grows, reaching almost 90 % of all traffic in the US today.

There are two different approaches to packet analysis. The first involves continuous, full-scale traffic recording (full-packet capture). It requires suitable technical equipment, especially high-speed storage arrays with enough capacity. This is a very expensive approach and is therefore viable only for critical infrastructure and networks that have been designed for a specific purpose. It should be noted that data storage is not the only problem since effective analysis and digging out the relevant information brings challenges of its own.

The other approach is the so-called on-demand packet capture. As the name suggests, it captures packets only when needed - typically when dealing with system compatibility issues - upon discovering that some packets are missing or are damaged, etc. On-demand packet capture is very simple and affordable, but it does have its pros and cons. A major downside of this approach, at least as far as traditional tools are concerned, is the fact that the administrator has to determine in advance which traffic should be stored. Therefore, there is no possibility to reach the traffic archive and get appropriate information for analysis if an incident occurs.

Packet Capture Tools

Two well-known packet capture tools are tcpdump for Linux and WinPcap for Microsoft Windows. Equipped with these tools, network administrators usually arrive at a place with their laptop, connect to a mirror port or TAP, and record the traffic. Problems may arise with remote locations, optical network interfaces, or 10Gpbs infrastructures – issues that could hardly be overcome with a laptop.

These problems can be easily avoided by the installation of standalone probes throughout the network to perform on-demand packet capture. Probes allow capturing traffic in high-speed networks as well (10 Gbps or more) via different types of interfaces. They can also receive remote inquiries. Professional probes for traffic recording often offer advanced packet filtering functions beyond L3 and L4 filter capabilities. As a result, network probes allow recording of, e.g. an entire VoIP communication or traffic based on the application protocol. It should be noted that modern firewalls allow full-packet capture as well, but they are usually restricted to perimeter network traffic.

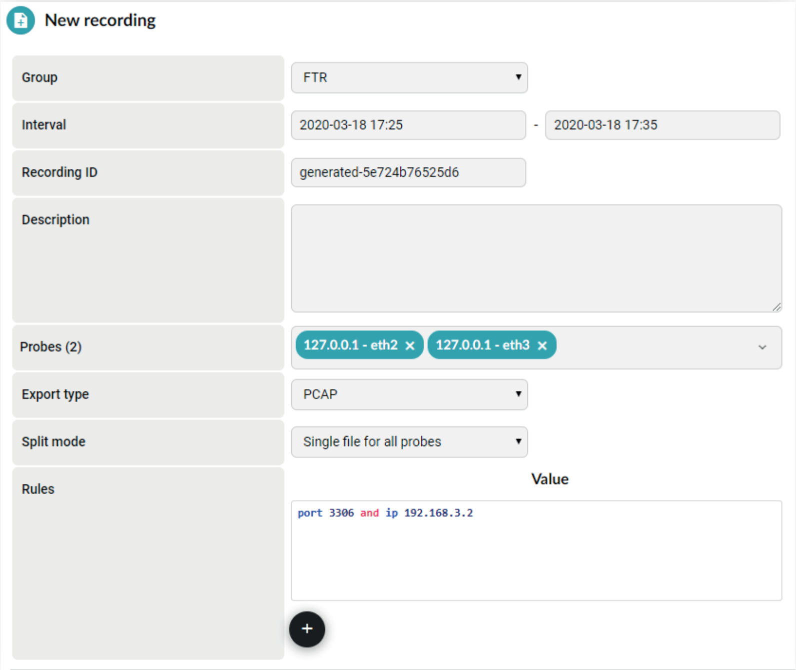
Analyzing the Traffic

Once the traffic is recorded and stored in a file, it needs to be analyzed. There are many commercial analytical solutions available that cover a broad spectrum of user needs. Apart from those, there is the very popular open-source tool called Wireshark.

Wireshark can recognize and decode hundreds of protocols. Moreover, it possesses analytical functions for very deep traffic inspection, such as filtering, reconstruction of TCP connections or phone calls, traffic decryption, or data extraction. Even though there are plenty of online courses, manuals, and example data sets available on the Internet, Wireshark demands advanced knowledge of TCP/IP protocols, awareness of the principles of data networks, and the user's own analytical skills.

Here is a simple example to show the benefits of packet analysis. A user complains that a language character set is not displaying properly and provides a screenshot as proof. However, the end-station administrator claims that application clients are installed and configured correctly. Likewise, the database administrator reports no issues whatsoever.

The answer lies inside the packets. The first step is to set up a filter to capture the communication between the clients and the database server. In this case, the application is based on a MySQL database server running on default port 3306. The IP address of the client is 192.168.3.2. With this in mind, the administrator needs to set up a filter for traffic recording. Storing the traffic of a given client in full and afterward filtering out particular packets is also an option, but on-demand recording allows downsizing data volumes intended for analysis at the moment of recording.

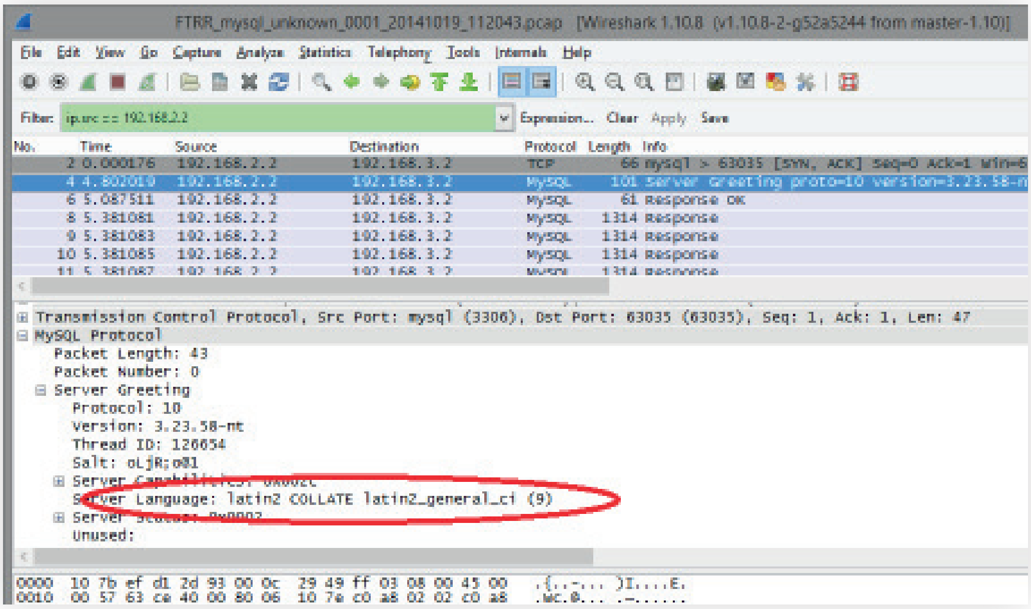


New recording

Group	FTR
Interval	2020-03-18 17:25 - 2020-03-18 17:35
Recording ID	generated-5e724b76525d6
Description	
Probes (2)	127.0.0.1 - eth2 ✕ 127.0.0.1 - eth3 ✕
Export type	PCAP
Split mode	Single file for all probes
Rules	Value port 3306 and ip 192.168.3.2

FIGURE 1: SETTING UP A FILTER TO CAPTURE THE COMMUNICATION BETWEEN ONE OF THE CLIENTS AND THE MYSQL SERVER.

The capture stored in a PCAP file is then opened in Wireshark. It immediately shows the character set announced by the server to the client. Where it should be the Windows 1250 charset, it says LATIN2 instead. This proves that there is a problem with server configuration and it is the administrator's task to put things in order again.



FTRR_mysql_unknown_0001_20141019_112043.pcap [Wireshark 1.10.8 (v1.10.8-2-g52a5244 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: ip.src == 192.168.2.2
Expression... Clear Apply Save

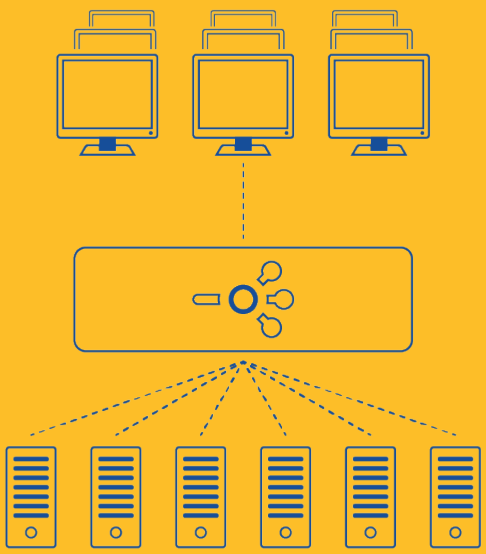
```
No.   Time        Source         Destination    Protocol Length Info
    2 0.000176    192.168.2.2    192.168.3.2    TCP        66 mysql > 63035 [SYN, ACK] Seq=0 Ack=1 Win=6
    4 4.802019    192.168.2.2    192.168.3.2    MySQL     101 Server Greeting proto=10 version=3.23.58-n
    6 5.087511    192.168.2.2    192.168.3.2    MySQL      61 Response OK
    8 5.381081    192.168.2.2    192.168.3.2    MySQL    1314 Response
    9 5.381083    192.168.2.2    192.168.3.2    MySQL    1314 Response
   10 5.381085    192.168.2.2    192.168.3.2    MySQL    1314 Response
   11 5.381087    192.168.2.2    192.168.3.2    MySQL    1314 Response

⊞ Transmission Control Protocol, Src Port: mysql (3306), Dst Port: 63035 (63035), Seq: 1, Ack: 1, Len: 47
⊟ MySQL Protocol
    Packet Length: 43
    Packet Number: 0
  ⊟ Server Greeting
      Protocol: 10
      Version: 3.23.58-nt
      Thread ID: 126654
      Salt: oLjR;o81
    ⊞ Server Capabilities: 0x002C
      Server Language: latin2 COLLATE latin2_general_ci (9)
    ⊞ Server Status: 0x0002
      Unused:

0000  10 7b ef d1 2d 93 00 0c  29 49 ff 03 08 00 45 00   .{..-... )I....E.
0010  00 57 63 ce 40 00 80 06  10 7e c0 a8 02 02 c0 a8   .Wc.@... .~......
```

FIGURE 2: A WIRESHARK WINDOW SHOWING THE CHARACTER SET ANNOUNCED BY THE SERVER (HIGHLIGHTED IN RED).

Summary

This simple example offers a glimpse into the capabilities of full packet capture and traffic analysis using the open-source tool Wireshark. Even though monitoring based on flow data provides answers to many questions and helps to detect the causes of network issues, packet analysis remains irreplaceable in situations where it is necessary to look inside the content of a particular communication.

Gartner analytics estimate that flow analysis should be done 80 % of the time while packet capture using probes should take up 20 %. The need to capture packets is not going anywhere, although, given the greater depth of insight that flow technologies provide, the demand is definitely shrinking. Solutions for full-scale traffic recording and analysis are costly and run into technological limits in high-speed network environments and when dealing with encrypted traffic. The roles are therefore divided so that flow-based infrastructure monitoring provides the majority of insights and is supported by on-demand deep packet analysis.



ADVANCED ANALYSIS
OF DATA NETWORK TRAFFIC

Network Traffic Monitoring or Packet Analysis? Get the Best of Both

by Pavel Minarik, Chief Technology Officer at Flowmon Networks

The previous two parts examine two different approaches to network traffic monitoring and analysis. They describe the basic principles of flow-based traffic monitoring and complete analysis of full packet traces. But these two approaches are not mutually exclusive - quite the contrary. This part describes a modern solution that combines the strengths of both approaches.

First, a brief recap of how the two technologies differ. Flow data represents an abstraction of the network traffic; an aggregation based on the source IP address, destination IP address, source port, destination port, and protocol number. The content of the communication is not stored, and the achievable aggregation rate reaches 250 to 500:1. On the other hand, packet analysis is focused on recording and analyzing full-scale network traffic, including the application layer. It is therefore very performance- and disc capacity-intensive.

TABLE 1 COMPARISON OF FLOW-BASED MONITORING AND PACKET ANALYSIS

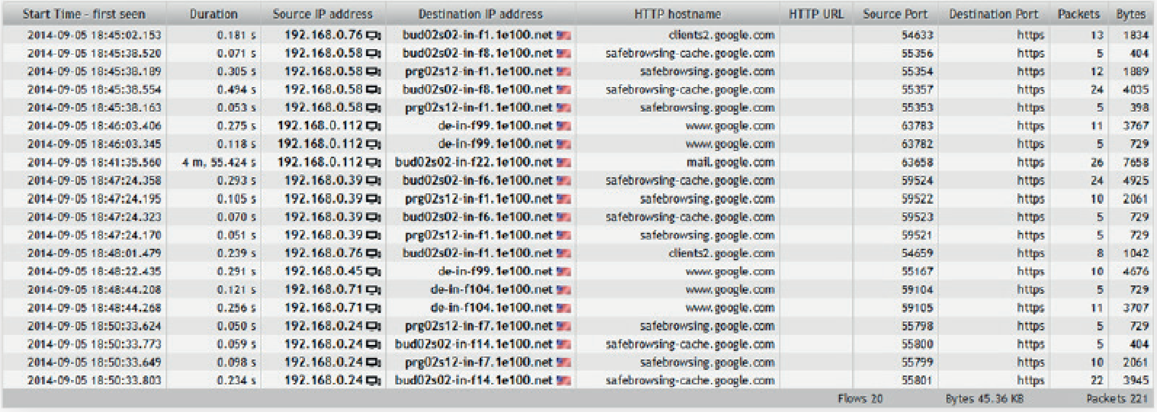
	Strong points	**Weak points**
Flow Data	Works in high-speed networks Unaffected by encrypted traffic Traffic visibility and reporting Network Traffic Analysis (security)	No application layer data available with some formats Not enough detail for some tasks Sampling (routers, switches)
Packet Analysis	Full-scale network traffic capture Sufficient detail for troubleshooting and analysis Supports forensic analysis Signature-based detection	Inefficient in encrypted traffic Very resource-intensive Surplus of information for most tasks

Let's now explore how the two approaches fit together. It is clear that flow data by itself does not provide enough detail for some operations. By contrast, packet analysis usually provides an overwhelming amount of information, where relevant data needs to be filtered away from the noise. Combining both technologies by extending traditional flow data with application layer information yields just the right amount of detail that provides network administrators with insights into data communication, flexible reporting, effective troubleshooting, and the automatic detection of security incidents.

This extension is made possible by the international flow data standard called IPFIX. IPFIX has introduced a lot of new attributes based on information from the application layer. Different vendors can expand this standard thanks to the so-called enterprise extensions, which broaden the scope of information that IPFIX provides. One of the most important innovations is the signature-based identification of applications. Application ID becomes part of network traffic statistics, which is determined by the application protocol classification mechanism. The first few bytes of the application layer, therefore, make it possible to recognize hundreds of applications.

The best-known implementation of this technology is Cisco's NBAR2 (Next Generation Network-Based Application Recognition). It combines flow data monitoring with continuous packet analysis that extends the traffic statistics by application name or protocol. Based on this information, modern flow collectors are able to deliver traffic reporting and analysis.

One of the most widespread communication protocols is HTTP or its encrypted version HTTPS. Today, it is mostly used to provide access to websites, but this is not its sole function. The protocol is also the basis of communication between the components of business systems, or applications working with sensitive data (e.g. electronic banking). Identifying this transfer protocol allows extending flow data statistics by fundamental HTTP request attributes – a hostname or URL information. Thanks to SNI (Server Name Indication), it is possible to get hostname information even when HTTPS is used. SNI is a mechanism by which a client indicates which hostname they are attempting to connect to at the start of a handshake.



Start Time - first seen	Duration	Source IP address	Destination IP address	HTTP hostname	HTTP URL	Source Port	Destination Port	Packets	Bytes
2014-09-05 18:45:02.153	0.181 s	192.168.0.76	bud02s02-in-f1.1e100.net	clients2.google.com		54633	https	13	1834
2014-09-05 18:45:38.520	0.071 s	192.168.0.58	bud02s02-in-f8.1e100.net	safebrowsing-cache.google.com		55356	https	5	404
2014-09-05 18:45:38.189	0.305 s	192.168.0.58	prg02s12-in-f1.1e100.net	safebrowsing.google.com		55354	https	12	1889
2014-09-05 18:45:38.554	0.494 s	192.168.0.58	bud02s02-in-f8.1e100.net	safebrowsing-cache.google.com		55357	https	24	4035
2014-09-05 18:45:38.163	0.053 s	192.168.0.58	prg02s12-in-f1.1e100.net	safebrowsing.google.com		55353	https	5	398
2014-09-05 18:46:03.406	0.275 s	192.168.0.112	de-in-f99.1e100.net	www.google.com		63783	https	11	3767
2014-09-05 18:46:03.345	0.118 s	192.168.0.112	de-in-f99.1e100.net	www.google.com		63782	https	5	729
2014-09-05 18:41:35.560	4 m, 55.424 s	192.168.0.112	bud02s02-in-f22.1e100.net	mail.google.com		63658	https	26	7658
2014-09-05 18:47:24.358	0.293 s	192.168.0.39	bud02s02-in-f6.1e100.net	safebrowsing-cache.google.com		59524	https	24	4925
2014-09-05 18:47:24.195	0.105 s	192.168.0.39	prg02s12-in-f1.1e100.net	safebrowsing.google.com		59522	https	10	2061
2014-09-05 18:47:24.323	0.070 s	192.168.0.39	bud02s02-in-f6.1e100.net	safebrowsing-cache.google.com		59523	https	5	729
2014-09-05 18:47:24.170	0.051 s	192.168.0.39	prg02s12-in-f1.1e100.net	safebrowsing.google.com		59521	https	5	729
2014-09-05 18:48:01.479	0.239 s	192.168.0.76	bud02s02-in-f1.1e100.net	clients2.google.com		54659	https	8	1042
2014-09-05 18:48:22.435	0.291 s	192.168.0.45	de-in-f99.1e100.net	www.google.com		55167	https	10	4676
2014-09-05 18:48:44.208	0.121 s	192.168.0.71	de-in-f104.1e100.net	www.google.com		59104	https	5	729
2014-09-05 18:48:44.268	0.256 s	192.168.0.71	de-in-f104.1e100.net	www.google.com		59105	https	11	3707
2014-09-05 18:50:33.624	0.050 s	192.168.0.24	prg02s12-in-f7.1e100.net	safebrowsing.google.com		55798	https	5	729
2014-09-05 18:50:33.773	0.059 s	192.168.0.24	bud02s02-in-f14.1e100.net	safebrowsing-cache.google.com		55800	https	5	404
2014-09-05 18:50:33.649	0.098 s	192.168.0.24	prg02s12-in-f7.1e100.net	safebrowsing.google.com		55799	https	10	2061
2014-09-05 18:50:33.803	0.234 s	192.168.0.24	bud02s02-in-f14.1e100.net	safebrowsing-cache.google.com		55801	https	22	3945
						Flows 20	Bytes 45.36 KB	Packets 221	

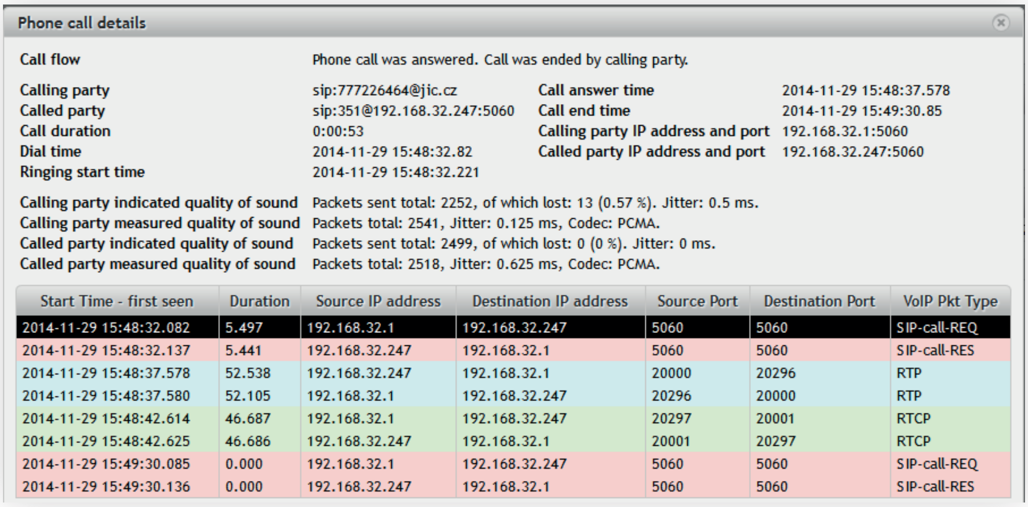
FIGURE 1: A LISTING OF FLOW DATA EXTENDED BY HTTP HOSTNAME INFORMATION. THE URL IS NOT SHOWN SINCE THE TRAFFIC IS ENCRYPTED.

Other information can be obtained from HTTP communication as well; for example the operating system and its version, the browser and its version, or device type (in the case of mobile phones). This information is part of an attribute known as the User-Agent - a textual string included in the client's request. This piece of information enables devices in the network to be detected.

The User-Agent can also be used to detect security incidents. When running DNS traffic, the type of query and domain name or the DNS server response can be monitored. An end station that receives a significant number of "non-existing domain" responses is suspicious and warrants the attention of the system administrator. Moreover, domains are related to reputation databases, making the efficient detection of suspicious communication quite easy (e.g. communication with known botnet command and control centers). Integration with IP reputation databases and hostname reputation databases should be part of every modern network traffic analysis solution. In addition, many excellent databases are available for free, such as the database of known attackers available at D-Shield.org.

Use Case:
Protection against Attacks on a VoIP System

Voice over Internet Protocol (VoIP) is seeing wide use by businesses today. An analysis of the Session Initial Protocol (SIP), which is responsible for signaling and controlling voice communication sessions, provides an overview of actual telephone calls. The quality of the calls can then be measured by analyzing the flow data itself. Monitoring SIP signalization enables control of client registration requests on specific gateways and detection of security threats. This means that the overview of telephone calls is part of regular network traffic monitoring and can be used for troubleshooting and resolving call quality issues.



Phone call details

Call flow	Phone call was answered. Call was ended by calling party.		
Calling party	sip:777226464@jic.cz	Call answer time	2014-11-29 15:48:37.578
Called party	sip:351@192.168.32.247:5060	Call end time	2014-11-29 15:49:30.85
Call duration	0:00:53	Calling party IP address and port	192.168.32.1:5060
Dial time	2014-11-29 15:48:32.82	Called party IP address and port	192.168.32.247:5060
Ringing start time	2014-11-29 15:48:32.221		
Calling party indicated quality of sound	Packets sent total: 2252, of which lost: 13 (0.57 %). Jitter: 0.5 ms.		
Calling party measured quality of sound	Packets total: 2541, Jitter: 0.125 ms, Codec: PCMA.		
Called party indicated quality of sound	Packets sent total: 2499, of which lost: 0 (0 %). Jitter: 0 ms.		
Called party measured quality of sound	Packets total: 2518, Jitter: 0.625 ms, Codec: PCMA.		

Start Time - first seen	Duration	Source IP address	Destination IP address	Source Port	Destination Port	VoIP Pkt Type
2014-11-29 15:48:32.082	5.497	192.168.32.1	192.168.32.247	5060	5060	SIP-call-REQ
2014-11-29 15:48:32.137	5.441	192.168.32.247	192.168.32.1	5060	5060	SIP-call-RES
2014-11-29 15:48:37.578	52.538	192.168.32.247	192.168.32.1	20000	20296	RTP
2014-11-29 15:48:37.580	52.105	192.168.32.1	192.168.32.247	20296	20000	RTP
2014-11-29 15:48:42.614	46.687	192.168.32.1	192.168.32.247	20297	20001	RTCP
2014-11-29 15:48:42.625	46.686	192.168.32.247	192.168.32.1	20001	20297	RTCP
2014-11-29 15:49:30.085	0.000	192.168.32.1	192.168.32.247	5060	5060	SIP-call-REQ
2014-11-29 15:49:30.136	0.000	192.168.32.247	192.168.32.1	5060	5060	SIP-call-RES

FIGURE 2: THE LIST SHOWS DETAILS OF THE PHONE CALLS, INCLUDING THE LISTING OF FLOW DATA. INFORMATION PROVIDED BY PACKET ANALYSIS INCLUDES CALLING PARTIES, TIME STAMPS, AN AUDIO CODEC AND INFORMATION ABOUT AUDIO QUALITY.

Let us now examine a case from IP telephony security using information from application analysis of the SIP protocol. One of the most popular attacks in this field is toll fraud, which gets money via fraudulent phone calls. It's a serious kind of fraud causing total annual losses of up to an estimated 72 billion dollars worldwide.

Here is how it works. An attacker establishes a company in a foreign country and rents a premium service to run legitimate high-cost phone numbers. Next, they find a poorly-configured SIP gateway in the victim organization. Using a compromised machine, they make a large number of calls to their premium numbers via this gateway. At the end of the month, the organization receives an enormous telephone bill, but since the service was delivered to the organization by the operator, there is no way to avoid paying it.

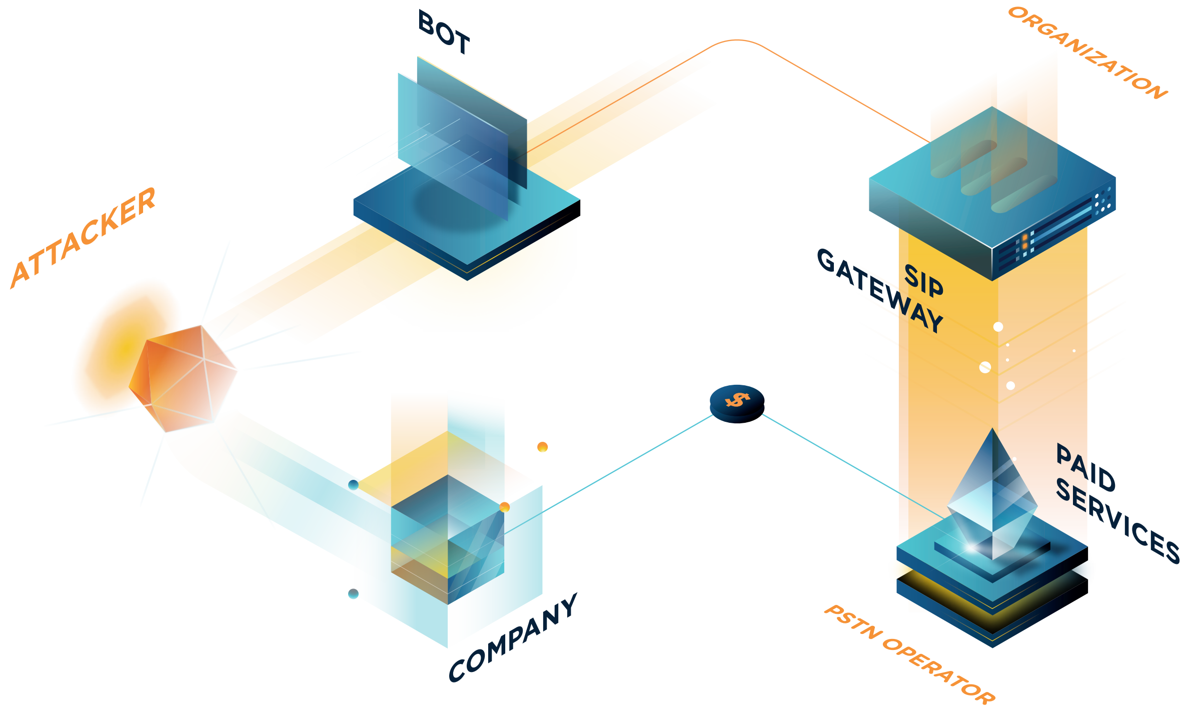


FIGURE 3: MECHANISM OF UNSECURED VOIP GATEWAY ABUSE. THE ATTACKER PERFORMS FRAUDULENT PHONE CALLS TO A PREMIUM HIGH-COST SERVICE NUMBER UNDER THEIR CONTROL.

The attack starts with a number of unsuccessful INVITE or REGISTER scans. These scans are part of the SIP protocol signalization. Monitoring these incidents and setting up an appropriate notification in case they spike enables detecting and preventing the attack early even before a financial loss is incurred.

However, VoIP gateways are not the only target of toll fraud. Attackers can also take advantage of bot-net-infected machines in the local network. Such end-stations can usually communicate with the gateway without any restrictions and may attempt to gain access to its configuration by an SSH dictionary attack.

Detecting attacks that utilize SIP signalization is based on the principles of Network Traffic Analysis; i.e. the automatic analysis of information entropy (e.g. IP addresses which communicate with gateways, identification of calling parties, or INVITE and REGISTER messages) that enables the detection of suspicious activities and unauthorized connection attempts.

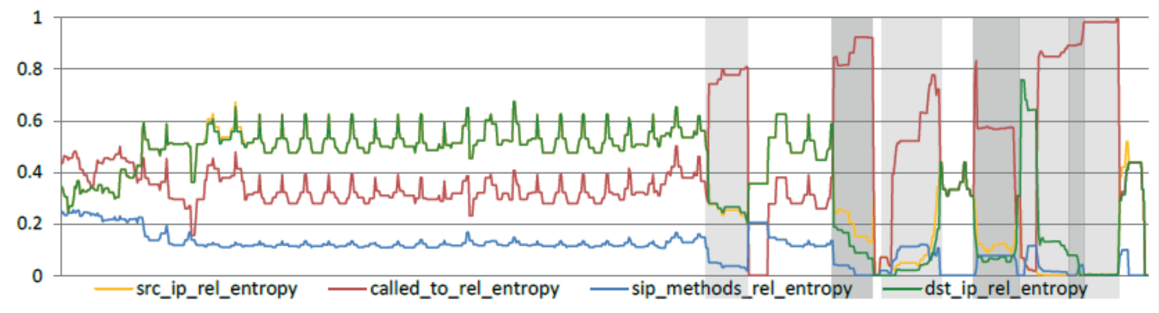


FIGURE 4: CHANGES IN THE ENTROPY OF SIP TRAFFIC CHARACTERISTICS DURING A RUNNING ATTACK. ATTACKS ARE MARKED BY THE GREY FIELDS.

Get the Best of Both

This paper shows how the benefits of flow data monitoring and application layer analysis can be combined for real-life purposes. The result is more detailed information about data communication, better traffic analysis capabilities, and anomaly detection. At the same time, it preserves the excellent compression/aggression rate of network traffic statistics of the original traffic volume. It is always possible to carry out full-scale traffic recording where necessary.

RNDr. Pavel Minarik, PhD.

Pavel Minarik has worked in the field of cybersecurity since 2006. During this time he took part in several research projects as a senior researcher at the Institute of Computer Science at Masaryk University. He is the author of multiple publications on behavior analysis and numerous algorithms for traffic processing and anomaly detection.

As Chief Technology Officer at Flowmon Networks, Pavel is responsible for the technology roadmap, product design, and development, as well as technical support and customer projects worldwide.

kemp

kemp.ax