Journal of Network and Systems Management, Vol. 6, No. 2, 1998

Report
Edited by Paul Brusil

Network Management and Realtime Traffic Flow Measurement

Nevil Brownlee[1]

An understanding of the traffic flows in a network is vital for network management. One needs to collect traffic flow data from many points throughout the network for performance monitoring and capacity planning, and be able to analyze it in terms of users and traffic types for security and congestion management. The Realtime Traffic Flow Measurement (RTFM) Architecture provides an effective method for obtaining traffic flow data, and for these purposes it offers some advantages over current remote monitoring (RMON) techniques. The IETF's RTFM Working Group has developed the architecture and is refining it within the IETF Standards process [1–4].

Managing Network Traffic Flows

In a connectionless network, traffic appears as streams of packets moving from one end-point to another, e.g., from a user's desktop computer to a server in another country. Such streams are usually described as *flows* of network traffic. Depending on how their end-points are defined, flows may have various degrees of granularity, from coarse-grained (e.g., from one network to another) to fine-grained (e.g., between specified ports on individual hosts). They may be large (carrying many packets) or small, bi-directional (carrying packets in both directions) or unidirectional, etc.

When a network begins operation there is usually plenty of capacity available. Over time the number of users grows, along with the demands they each place on the network. We can describe this as an increase in the number and size of the network's traffic flows. To effectively manage this growth, it is useful to

[1]IT Systems & Services, The University of Auckland, New Zealand. E-mail: *n.brownlee@auckland.ac.nz*

1064-7570/98/0600-0223$15.00 /0

develop and maintain a good understanding of the traffic flows [5]. Traffic flow data can be used in several ways, as illustrated later.

Troubleshooting

From time-to-time there may be abnormally high levels of network traffic, e.g., a flood of packets from a faulty network card, a server being backed up across the network during working hours instead of overnight, etc. Tools such as stand-alone network analyzers are generally well suited to finding single high-volume flows of this kind.

Performance Monitoring

Most of the time, however, one needs to look at more than just total volume. One really needs to know the network's normal usage patterns, in terms of user or traffic type. For security management it is essential to detect activities such as repeated *login* attempts from unusual hosts and sequences of packets sent to many different ports on a few hosts.

Once *normal* patterns are known one can determine whether, over time, traffic flows change their character; for example average flow durations may lengthen, some protocols may increase in popularity while others decline, and so on.

Congestion Management

It is of particular interest to know whether, in the short or long term, there are flows which consume an unusually large portion of the network resources. If there are, steps can be taken to improve the situation. Many approaches are possible, including 'social' methods—persuading users to change their work patterns—and/or 'economic' methods—charging users for their network traffic.

Upgrading the Network

Eventually it becomes necessary to increase the network's capacity. In some networks it may be possible to simply increase the bandwidth everywhere, e.g., one might change an Ethernet LAN from 10 to 100 Mbps. Usually, however, one must decide which parts of the network to upgrade. At this stage it becomes vital to understand the makeup of the traffic flows through the network so as to determine where the traffic is heaviest, and where upgrades will provide the most benefit.

RTFM ARCHITECTURE

The RTFM Architecture includes four components: meters, meter readers, managers and analysis applications. The Architecture does not specify the method used for communication between these components, but most implementation experience (see later) has been gained using SNMP carried over UDP.

A *meter* is a self-contained logical entity which observes network traffic flows and builds up a table of flow data records for them. It can be implemented as a stand-alone device, observing flows at the point where it is connected to the network. Alternatively it could be implemented within a network device (such as a multiport switch or router), in which case it could observe traffic passing through the device. The flows which a meter is to observe, and the flow data records to be created for them are specified by a *rule set*, as described later.

A *meter reader* collects traffic flow data from one or more meters, producing files of traffic flow data. It would normally be implemented as a process running within a multi-user computer system, writing its flow data files to disk.

An RTFM *manager* is a computer program which oversees the operation of meters and meter readers. Its responsibilities include downloading configuration files to meters and setting parameters for meter readers. It should also verify that they continue to operate properly, and notify operations staff if a meter or meter reader fails.

Analysis Applications process flow data files so as to provide information and reports as required by Network Operations staff. Because these differ widely between networks, the RTFM Working Group has not attempted any standards work in this area. Analysis Applications are, nonetheless, an essential part of any on-going network management effort, and the requirement for on-going development and maintenance of them should not be underestimated.

Specifying Flows: Attributes and Rule Sets

The notion of a traffic flow was introduced earlier. In more detail, a traffic flow is an abstract entity with a set of 'attributes.' A flow's data record in an RTFM meter is a data structure holding its current attribute values. These attributes include its end-point addresses and other information about it, such as its start and stop times and the number of packets and bytes it has carried in each direction.

End-points are specified by a set of address attribute values, one for each network layer, e.g., network X, host H, port P. 'Wildcard' values and/or lists of values are allowed at each layer, which provides a simple but very powerful way to specify end-points, e.g., all FTP traffic from networks A, B, or C which passed through local router R.

While a meter is running it uses a *rule set* to specify which flows are of interest; this is very similar to the use of 'filters' in other network analysis systems. The rule set is more general, however, in that it also specifies exactly what information is to be recorded for each flow of interest. As an example, we might regard all packets from network A as a single flow, but (at the same time) want to have separate flows for each subnet of network B, and so on. Essentially, one may regard a rule set as a program for recognizing the flows of interest and doing preliminary reduction of their data.

Implementation Experience

The first implementation of the RTFM Architecture was *NeTraMet*, which was first released (as free software) in October 1993 [6]. It has a combined manager and meter reader (NeMaC) and a meter (NeTraMet) which runs as a process on a Unix system, or stand-alone on a DOS PC. NeTraMet is in widespread use in many countries. It is most commonly used to collect flow data from 10BaseT Ethernet network segments, but it can also be used with FDDI and 100BaseT Ethernet. Work is well advanced on porting it to run as a 'statistics gathering module' within the OC3MON platform [7]. (OC3MON is a PC-based system for monitoring traffic flows on ATM links.)

The NeTraMet distribution includes a few simple analysis applications, including 'nifty,' an X/Motif realtime flow analyzer which uses data from a NeTraMet meter to pinpoint 'unusual' flows.

A second, independent, implementation has been produced within IBM Research (Hawthorne, New York); this version runs in an AIX-based network management environment.

RMON AND RTFM

Remote network monitoring (RMON) capability has become widely available in recent years, via stand-alone RMON probes and RMON agents built into network devices. RMON provides a good way to determine traffic levels in network segments, total traffic loads to/from busy hosts and (with RMON2) traffic loads between host pairs, for different protocols, and so on. Overall, RMON is an effective day-to-day network monitoring tool.

RMON does not, however, provide a general way to specify traffic flows, and it has no notion of bi-directional flows. For example, it would be very difficult to specify flow end-points as lists of networks in RMON. Furthermore, RMON probes have limited ability to pre-process the traffic data they collect; this is left to the management systems which oversee them.

In situations where detailed information about the fine structure of traffic

flows is needed (e.g., for usage logging), RTFM should provide a more effective approach. This is particularly so where good time resolution is needed—an RTFM meter records flow start and stop times in centiseconds, allowing very short-lived flows to be observed even when flow data is only read from the meter at long (10 minutes or more) intervals.

CONCLUSIONS

The RTFM Architecture provides a simple, consistent scheme for describing network traffic flows and for collecting flow data. The RTFM Working Group is advancing it on the Internet Standards Track. As it becomes more widely used, vendors will be able to implement it in network devices, making it possible to measure flows in switched networks.

One common approach to flow analysis is to collect (perhaps by using an RMON probe) traces of all packets of interest, then analyze them off-line. This approach can quickly generate very large amounts of data, and off-line analysis does not lend itself to real-time network management.

RTFM, in contrast, provides a very general way to specify the flows of interest and the amount of detail required for each flow. It effectively uses RTFM meters to do preliminary reduction of the data before it is collected and stored in flow data files. This can dramatically reduce the volume of data to be retrieved across the network. Furthermore, the reduced flow data is available as it is collected, making it possible to develop near-real-time analysis applications.

Another strength of the architecture is that it can easily be extended by adding new attributes. The RTFM Working Group is working on new attributes, and is particularly interested in measurements relating to packet arrival times, such as the delay and jitter of packets within a flow.

REFERENCES

1. C. Mills, G. Hirsch, and G. Ruth, Internet accounting background, RFC 1272, Bolt Beranek and Newman Inc., Meridian Technology Corporation, November 1991.
2. N. Brownlee, C. Mills, and G. Ruth, Traffic flow measurement: Architecture, RFC 2063, The University of Auckland, Bolt Beranek and Newman, Inc., GTE Laboratories, Inc, January 1997.
3. N. Brownlee, Traffic flow measurement: Meter MIB, RFC 2064, The University of Auckland, January 1997.
4. N. Brownlee, Traffic flow measurement: Experiences with NeTraMet, RFC 2123, The University of Auckland, March 1997.
5. Realtime Traffic flow measurement working group home page, http://www.auckland.ac.nz/net/Internet/rtfm
6. NeTraMet home page, http://www.auckland.ac.nz/net/NetraMet
7. OC3MON: Flexible, affordable, high performance statistics collection, http://www.nlanr.net/NA/Oc3mon

Nevil Brownlee is responsible for The University of Auckland's campus network (which has about 6,500 connected hosts), and is manager of Kawaihiko, the New Zealand Universities' network. He has been active in the IETF since 1992, and is currently co-chair of the Realtime Traffic Flow Measurement Working Group.