L-045

Sifting through Monitored Network Data: the Difficulties and the Workaround

Kazuhide Koide[†‡] Masahiro Nagao[†] Satoshi Utsumi[†] Glenn Mansfield Keeni[‡§] Norio Shiratori[†‡]

1. Introduction

Effective network management is a growing challenge as the network bandwidth, size and geographical distribution continues to grow. Network managers have to take care of what happens in their networks for safety network operation. For the purpose, they will have to analyze various data from network monitoring applications, but generally, these data are too much for network managers to deal with all over the data.

We propose the event-based network management scheme for effective traffic monitoring and analysis. With the result of our experiment, our system provided about only 0.1 percent of whole data as described in Table 2 of 5.4, and we figured out details of some events from the data. This result shows that our system can provide worthful data effectively from vast amount of data.

We consider an interesting occurrence in the network as an *event*. For example, faults and intrusions are significant events. Network managers need to analyze the cause and the detail of each event. In our management scheme, only when an event detected, the detailed data related the event is provided to network managers. This can help the managers to deal with massive data effectively, without losing important data for network management. We have designed a system that can be used to define many types of events, detect these events from monitored data, and analyze the data to diagnose the cause of the events.

We have also taken into consideration the necessity of exchanging information about events. Information sharing is important though it has difficulties in terms of data capacity, security policy and so on. Our proposed design can be applicable to share small data about events effectively based on a standardized data format.

2. Concept of Event from Network Management Information

2.1 Problem: Too much network data

Fine-grained data, packet traces, etc. will be referred to as *first order* data or, "raw" data. Generally, this type of data is not suitable for daily monitoring and reporting as the volume of data is large and/or the format is not user-friendly, so this data is stored and referred to only when needed.

In general, network managers monitor time series coarse-grained data routinely. This *second order* data is in general aggregated form of the *first order* data. So a network manager will monitor the *second order* data, look for interesting parts and then refer to the relevant stored *first order* data for analysis and diagnosis.

The essential problem in effective network monitoring is the massive volume of data which the network administrator is expected to "use" to understand the operational status of the network. In most cases the volume of data itself renders the data practically useless. The data is just logged for later day usage.

[†]Research Institute of Electrical Communication/Graduate School of Information Science, Tohoku University
[‡]NICT Tohoku Research Center
[§]Cyber Solutions Inc.

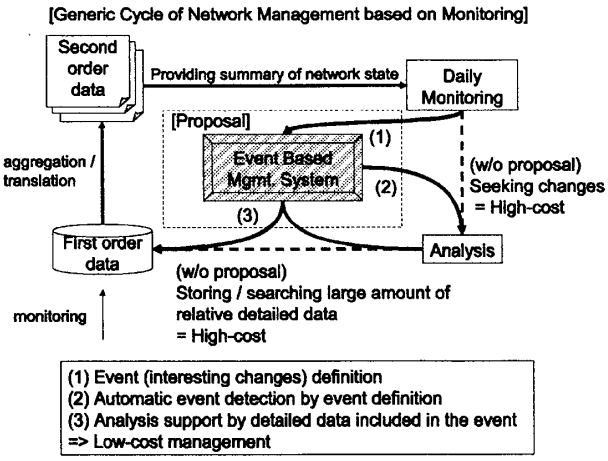

Figure 1: Event-based monitoring.

A general solution is to deal with only a few types of second order data. It is easy to deal with, but it is not enough to monitor a network sufficiently.

2.2 Concept: Event-driven information monitoring

Our approach is to simplify the general network 1) monitoring and 2) analysis.

In our model, network monitoring is controlled by an *event*. An event has (1) a name, (2)duration and (3)relevant data. Usually an event is defined by the pattern of interesting changes in time series data. An event should be defined by the interest of the network manager. It does not necessarily indicate actual faults or intrusions. Given the patterns of interest, a monitor can detect potential events and notify the manager. The manager monitors detected potential events instead of observing the voluminous time series data. This simplifies the task of network monitoring.

Detection of events is not the end in our model. We envisage that a manager will want to analyze and diagnose the event. Event analysis is follows detection. Our model requires that each event will include the relevant data of the event and be associated with knowledge about the detailed data that will be needed to analyze the event. It is processed by the detailed mechanism and a manager can directly access the corresponding first order data from the detected event. This reduces the effort needed for analysis. Figure 1 is the sketch of our event-based monitoring model.

2.3 Sharing network management information in the large

In general, an administrator is interested in his/her network only. But in many cases it is important and interesting to share event related information. For example an administrator may want to know whether his/her network is a singular target of an attack or whether several networks are under attack. This requires sharing of event related data.

Sharing all monitored data is impractical because of the volume and security issues. Further, in the absence of an accepted common format, sharing data in meaningful way is impossible.

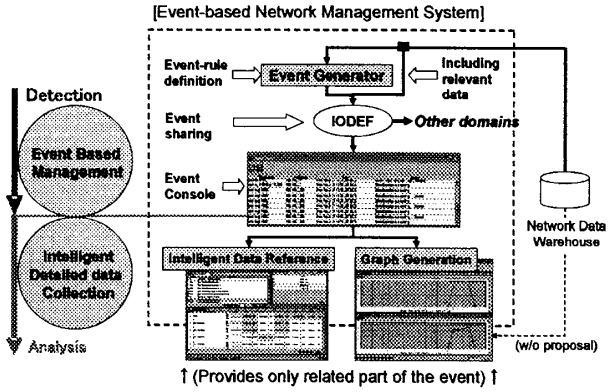

Figure 2: Outline: Event-based network management system.

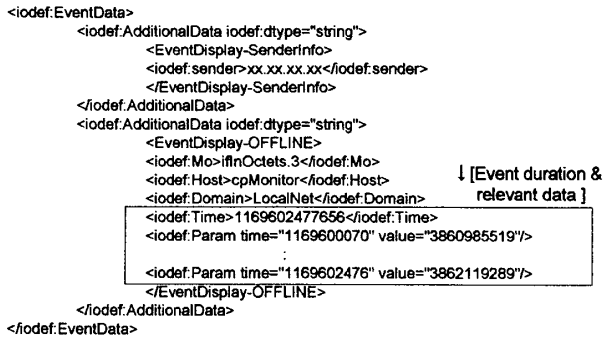

Figure 3: Structure of event.

Our design addresses these issues. Event information itself is quite smaller than first/second order data and suitable for the exchange. As the common format for representing event information, we propose to use IODEF (Incident Object Definition Format) for representing network management events. This design aspect extends the scope of network event diagnosis to large distributed networks, and the Internet itself.

2.4 Related works

It can be said that the concept of *event* is the extension of a generic alarming mechanism like SNMP notifications[1] message. There are many systems that use SNMP notifications. [2] proposes an event-driven management architecture. But the definition of *event* in [2] is based on SNMP notifications, and the issue of event-based analysis has not been addressed.

Our goal is not event detection itself, but analysis of the detail for effective network monitoring. Manual analysis is needed and effective for network management. We can define events by various detection methods without concern for cause and harm of the detected event. In this point, our scheme is deferent from signature detection system[3] and anomaly detection system[4].

It is important to standardize the data format for sharing management information between autonomous management domains. [5] is the de facto standard for various log format. These are first order data and unsuitable for sharing management information between autonomous management domains. We have attempted to address the issue of sharing event information, and have proposed the use of IODEF(the Incident Object Description Exchange Format)[6] format for describing event information

3. Event-based Network Monitoring system

We are developing a system that enables event-based network management. Now the system mainly focuses on traffic monitoring. Figure 2 shows the merit of our system. Event detection, presentation and analysis mechanism enables to reduce the effort of monitoring daily traffic, and analysis of relevant data.

3.1 Event definition

One of the core parts of our system is the "event generator". It has a simple user interface and allows a definition of a rich variety of "event-rules" based on change in any type of time series data. An event-rule has two components, the first component detects the change by comparing the statistic against a specified threshold or another statistic, the second component detects the persistence of the change by comparing the duration or frequency of the change against some pre-specified threshold. If the threshold is breached the "event generator" will generate an "event-notification" in accordance to the defined event-rules.

Information about more detailed related data is defined in each event-rule. If an event would be detected, pre-defined related data were automatically collected or provided for a network manager. This will help network managers to analyze detected events.

Our system can provide protocol-wise (IP, UDP, TCP, ICMP), IP address-wise and port-wise traffic usage[7]. These can be used as respective time series data for event detection. For analyzing the cause of the event, these can be combined and provide a fine-grained traffic usage, e.g. counters for all destination ports for all source addresses. This is the first order data with fine granularity.

3.2 Structure of event

Figure 3 shows the structure of event object itself. The event object is generated in the form of XML-based text that extends the IODEF. IODEF is the standard format for exchanging operative data related to computer security incident between different CSIRT (Computer Security Incident Response Team).

There are mainly two reasons for using IODEF-based format. First, IODEF is on the standardization track. Naturally, it will be easier to develop event-handling system using IODEF-based format. Second, the IODEF format is extensible and thus can accommodate the requirements of generic events with relative ease.

3.3 Event console and detail information viewer

The "event console" provides a comprehensive view of detected events. It provides a list of events. Events may be selected by the type of events. The most important function is stepwise refinement of the cause of the event based on the detailed data.

Figure 4 shows the screenshot of our implemented system. The upper middle of the figure shows event list. The left bottom shows IP address-wise traffic composition, and the right bottom shows port-wise traffic composition. These are available for each event.

4. Information-sharing in Wide-area Network Management

4.1 Usefulness of wide area information in network management

Some types of network incidents, virus transmissions etc. affect many networks over a wide area. The changes in time series traf-

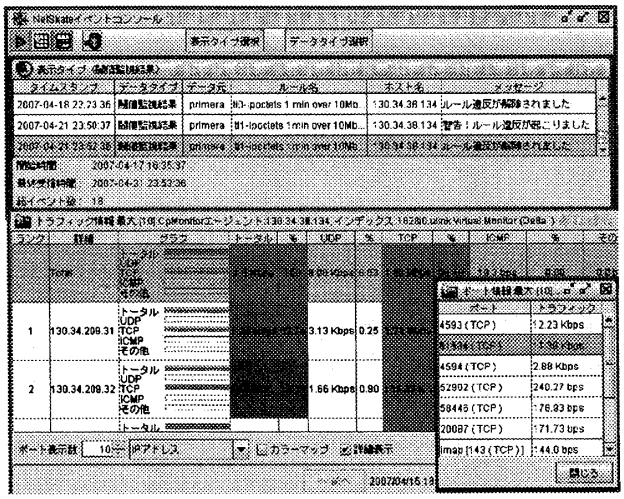

Figure 4: Screen shot of our implemented system.

fic data will be detected as an event in multiple networks in such cases.

When analyzing the cause of these events, correlation with the monitored results of other networks will be useful. Wide area network information is useful for network management. But it is almost impossible to share network information, especially first-order data that may include personal information, between organizations. Sharing second-order data (time series data) is possible in some cases. But there are problems in how to share the data. It is not practical to share all data as the volume will be very large.

4.2 Event-based information-sharing

Sharing information has two aspects. 1) How to query the server of a domain for detailed information about an event. 2) How to share the detailed information for analyzing event. Towards this end we define (1) an "event query" message using IODEF for solving 1), and (2) extend the IODEF format to include event relevant data itself.

"Event query" only has the start/end time stamp (duration) of the event and the name of the event (it is pre-defined and shared between organizations).

When the event query is received, the system checks the validity and integrity of the query. After that, if the queried event is found locally, it generates an event object corresponding to the event-query and adds the event related data. The event object is easily extended to include many types of data. The generated event object will be sent back to the query sender.

4.3 Implementation and application

We implemented the IODEF server that generates an event query, and the IODEF agent that can generate an event object corresponding to the event query for our event-based network monitoring system to exchange event information and relevant data between different domains.

IODEF is transport independent. We have used e-mail (SMTP and POP) in our implementation. In general, firewalls will not affect e-mail transportation. E-mail already has secure transport, privacy and sender authentication mechanism (PGP, etc.). It can be used without difficult configuration. Our system has an e-mail address that is used for exchanging event queries/objects. Each system has a list of peer e-mail addresses, these peers will be queried for relevant information. The system receives the event query and sends back the event object to the e-mail address of the query originator.

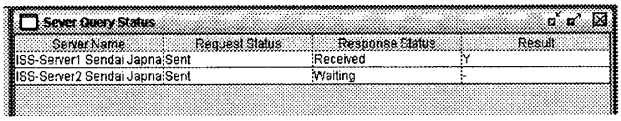

Figure 5: Query broadcasting.

The generated event query is broadcasted to the peers. Figure 5 shows the status of each of the query to each peer. After event objects are collected, the administrator can access the relevant data that is included in the event object from other domains.

5. Evaluation

In this section we demonstrate that how our developed system can collect event related detailed data effectively.

There are two focuses on evaluation. One is a capability of our system in a real data analyzing situation. And another is an effectiveness of our system.

As shown in Figure 1, our system has three main functionality, event definition, detection, and analysis. As events are user-definable, we use the existing event definition and detection method for objective evaluation. We use the method of [4] that uses information entropy for detecting, summarizing and classifying traffic anomalies.

Detected anomalies are considered as *event*. We evaluate a capability and an effectiveness of our system through analyzing these detected events.

5.1 Outline of Experiment

First we detected several events by the method of [4]. The method has the ability to inspect the type of event by variation of Src./Dst. IP addresses and Src./Dst. port observed in a certain monitoring time-bin.

On the other hand, for each detected event, we analyzed first order data for more detailed data, and validate the results using our system. First we check whether our system can provide good insight of the event or not. This will show a capability of our system. Second we estimate the amount of first order data that is used for analyses. If the amount of the detailed data that needed for analyses is smaller, it shows an effectiveness of our system.

5.2 Data for Experiment

We made the experiment by using two types of traffic data. Data-A is collected from the activity of fixed-point observation for the Internet in Japan. The data is corrected from 02 Oct. 2006 to 15 Mar. 2007. Time-bin of event detection method equals 1 hour. Data-B is collected from access links of our laboratory in Tohoku University. The data is corrected from 20 Jan. 2007 to 28 Jan. 2007. Time-bin equals 5 minutes.

We need to explain more detail about Data-A. It is from the activity of monitoring Internet dynamics. Monitoring system has some servers that have fixed global IP address and they never initiate communication from them. They only receive incoming packets from the Internet and simply send Ack / reply to them, so causes of packets coming are speculated to be misuses or attacks. Events

Table 1: The list of detected events and analysis result.

ID	timestamp	Estimation	Main Fact
A-1	2006-11-04 09:00	portscan	portscan(small)
A-2	2006-11-09 14:00	portscan	portscan(small)
A-3	2006-12-14 03:00	DoS	backscatter
A-4	2006-12-15 14:00	large flows	port 135 access
A-5	2007-03-04 17:00	large flows	port 135 access
A-6	2007-03-12 22:00	DoS	backscatter
B-1	2007-01-20 14:00	flash crowds	port 22,80 access
B-2	2007-01-20 14:05	flash crowds	port 22,80 access
B-3	2007-01-21 06:45	flash crowds	port 22,80 access
B-4	2007-01-25 04:50	flash crowds	port 22,80 access
B-5	2007-01-25 04:55	flash crowds	port 22,80 access
B-6	2007-01-25 05:00	flash crowds	port 22,80 access
B-7	2007-01-28 18:05	flash crowds	port 22,80,2907 access

detected from these data will imply "abnormal misuse or attack", so it needs careful analysis. Data-B is a typical data of communication pattern from/to small or middle-size university / enterprise network. We analyze it for objective comparison.

5.3 Evaluation-1: Capability

Table 1 shows the list of detected events and analysis result. Estimation is the prediction by the detection method[4]. Main fact is the cause of the event figured out by manual analysis using our system.

Especially the results of Data-A shows the capability of our system. A-3 and A-6 are predicted as DoS attacks. But from the analysis, these events can be thought to be ICMP backscatters of an attack to "68.xxx.227-228/24" network that spoofed its source address. Our system shows that there are many src hosts that have "68.xxx.227-228/24" address, and they generate only 1 ICMP packet. And also, monitoring host sent no packet to "68.xxx.227-228/24" address.

Other results of Data-A show that the actual cause of event was figured out by analysis. A-1 and A-2 were narrow range (1026-1033) portscans. A-4 and A-5 were port 135 accesses. Actual related service and host can appears easily.

All events of Data-B are inferred as flash crowds, that is, concentrated access to the local network services. Our system revealed that the services are mainly port 22, and port 80. But it also shows that D-7 includes access to port 2907. This shows that event D-7 is different from others.

These results show that in actual environment detailed analyses of event is important, and our system is available to the purpose.

5.4 Evaluation-2: Effectiveness

Table 2 shows that how the amount of detailed data that our system used for analyses to obtain results of above section is.

If there is no event-based management system, to realize detailed analyses network manager has to correct and store all first order data. As shown in Table 2, the amount of data is quite large. And it is difficult to share these data to other domains. By using the event-based management system, it needs to correct only first order data that is related to events. The amount of data is smaller and suitable for sharing.

Table 2: Reduction of analyzed detailed data.

	data-A	data-B
first-order data	231 MB	2.1 GB
analyzed data	264 KB	1.9 MB

Actually in this case the number of event is 6-8. If the number of events is larger, the amount of detailed data should be larger. How to reduce the number of event itself is an open problem.

6. Conclusion

We proposed the event-based network management for more effective network management, especially traffic monitoring and analysis. We also have designed the event-based network management system including event information sharing scheme, and evaluated the effectiveness through the analysis of real operational data.

In the next step, we intend to carry out experiments over a wide area and establish the efficacy of the system.

Acknowledgement

This work is partially supported by Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research (A), 19200005 and SCOPE project (071502003).

References

[1] J.Case, M.Fedor, M.Schoffstall and J.Davin, "Simple Network Management Protocol (SNMP)", RFC 1157, May 1990.

[2] J. Yang, J. Wu and Y.You, "A web-based, event-driven management architecture", In *APCC/OECC '99 Fifth Asia-Pacific Conference on Communications*, pages 1214–1221, 1999.

[3] snort, http://www.snort.org/.

[4] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions", In *ACM SIGCOMM*, Philadelphia, August 2005.

[5] C. Lonvick, "The BSD syslog Protocol", RFC 3164, Aug. 2001.

[6] R.Danyliw, J.Meijer, and Y.Demchenko, "The incident object description exchange format data model and xml implementation", May 2006.
[ONLINE]. Available: draft-ietf-inch-iodef-06.txt.

[7] M.Nagao, G.M.Keeni, T.Suganuma, K.Koide and N.Shiratori, "Detecting and Diagnosing Events from Monitored Data in a Wide Area Network", Proceedings of the 2006 IEICE Society Conference, pages S25–S26, Sep 2006.