第 V 部

ネットワーク管理とセキュリティ

第5部
ネットワーク管理とセキュリティ

第1章 Introduction

Network monitoring is an essential aspect of network management. The WIDE Netman Working Group has been working over years to make network monitoring effective. During 2006 our main focus of the WIDE Netman Working Group has been on effective monitoring over a large distributed network. We have also worked on support for monitoring mobile devices and networks.

There are two practical problems in effective network monitoring. In practical situations, network monitoring yields a massive volume of data which the network operator/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. We propose to enable effective monitoring by saving the operator/administrator from browsing through all the monitored data. Only the "interesting" data will be presented to the operator. We call an interesting occurrence in the network an "event". The interesting occurrence will manifest itself in the monitored data. The operator will be notified of the interesting events and will be expected to examine the relevant data.

The second problem arises when the operator/administrator attempts to examine the relevant data to diagnose the event. Detailed examination requires detailed data. It is not practical to collect detailed data during routine monitoring as that would mean heavy load on the monitoring system, the monitored devices, the data storage and the network itself. So for all practical purposes, detailed examination of events is not possible. To make the system more effective for diagnosis, we propose a model in which an event will be associated with knowledge about the detailed data that will be needed to analyze the cause of the event. When the event is detected, the relevant detailed data will be collected and stored for later examination.

The above two steps takes network monitoring a major leap forward by making monitoring simpler and more effective.

In general an operator is interested in his/her network only. But in many cases it is of importance and interest to share event related information. For example an operator 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 information. Sharing of network event information is technically hindered due to the lack of a standard format for representing event information. We have proposed to use IODEF (Incident Object Definition Format) for representing network management events. By sharing information in this format, the scope of diagnosis for network events is extended to the Internet itself.

第2章 Sharing network management information in the large

2.1 Introduction

Network managers get massive data from network monitoring applications. They have to understand this data and notice significant "events" in the network. This is a difficult task. For example, in one scenario, hundreds of traffic statistics may show large fluctuations. The network manager has to notice these changes from amongst the very large number, potentially

thousands, of traffic statistics. Then he will have to analyze large amount of data to understand the cause of the changes. The cause may be a DoS attack, a network mis-configuration, or, a normal network occurrence.

It is a challenge to extract interesting/important part, and to effectively extract the cause from the massive swamp of data. We call an interesting occurrence in the network an "event". The interesting occurrence will manifest itself in the monitored data. The challenge is to construct algorithms to "notice" these events from the data. We are constructing 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. The system must be simple, effective, accurate and must not load the monitoring system or the network.

Most of the existing anomaly detection methods based on traffic monitoring are built on sophisticated algorithms and focus on reducing the number of false positive "event" reports. These methods often have quite limited scope in terms of applicability, and involve complex computations. On the other hand, our system only detects "interesting data" which is potentially a manifestation of an "event" of interest. Our approach focuses on how to simplify the operation of "network monitoring".

2.2 Information Monitoring for Network Management

2.2.1 Information for network management

There are many types of information that are useful for network management. Some of them are automatically generated just like logs of servers or network equipments. Others are collected by some monitoring equipments/functions. An example is packet trace or flow trace. Packet traces are collected by passive monitors. *NetFlow* or *sFlow* will generate flow traces or sampled packet traces.

This type of data that is collected from equipments/servers without any conversion will be referred to as "first order" data or, "raw" data. In general this data is not suitable for daily monitoring and reporting as the volume of data is large and/or the format is not user-friendly. In general this data is stored and referred to only when needed.

Generally network managers monitor many types of statistics daily. These statistics are commonly generated from "first order" data by continuous aggregation/transformation. Often managers will monitor statistics that are obtained by SNMP polling of network devices. In this case the SNMP agent aggregates the traffic data. This data is called "second order" data. Counters and gauges are examples of "second order" data. This data is suitable for daily monitoring, but most of the detailed information is lost. So a network manager will monitor the statistics, pick up interesting parts and then refer to the relevant stored "first order" data for analysis and diagnosis.

2.2.2 Monitoring large networks

As the size of the network increases, the size of the monitored statistics increases. Tracking and storing the "second order" data itself becomes a challenge. One solution is to use a full-automated monitoring system, just like intrusion-detection system whereby suspicious packets and transactions are detected and logged. But these will not solve all problems.

First reason is that the scope of these solutions is often narrow. For example, a DoS-detection system will ONLY detect DoS attacks, but will not detect other types of attacks. It is not practical to introduce a detection system for all types of attacks/anomalies.

Second reason is that the cause of an anomaly may not always be clear. The network down may cause a trouble of the protocol/application that has not been seen before. Further, new types of attacks are invented every day. There are new/unknown attacks that need to be provisioned for.

Anomaly detection systems have their merits. They can detect anomalies that human managers cannot detect by eye observation. But it is

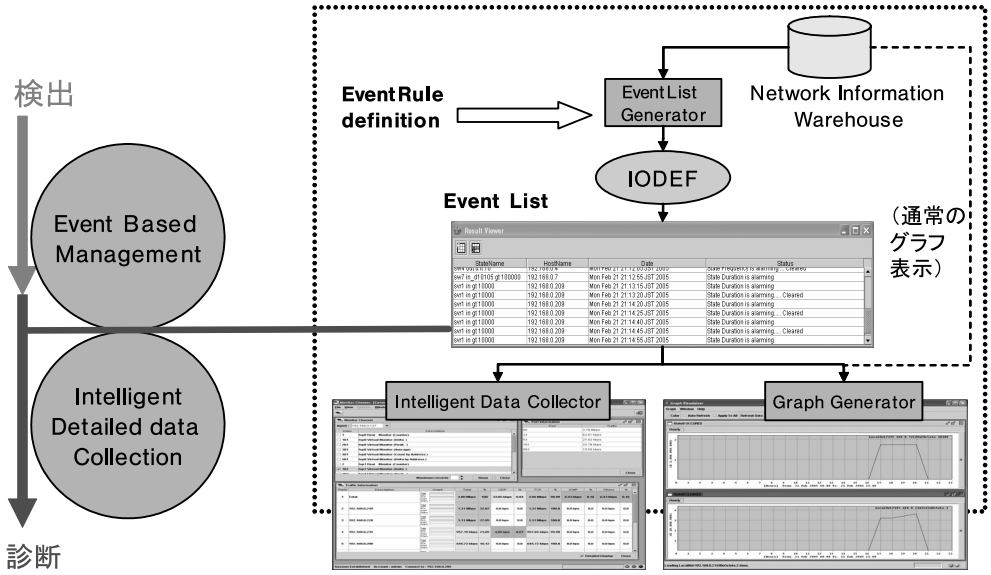

Fig. 2.1. Event-driven information monitoring

impossible to replace daily monitoring in network management with these systems.

2.2.3 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" trigger. "Event" is defined and detected as "the changing pattern of statistics that the manager is interested in". Network manager needs to monitor detected events instead of monitoring several hundred statistics of second order data. This simplifies the task of network monitoring.

In our model, an event will be associated with knowledge about what detailed data will be needed to analyze the cause of the event. It is processed by the detailed mechanism and a manager can directly access the detailed information from the detected event. This reduces the effort needed for analysis.

2.3 Implementation: Event-based Network Monitoring system

Our system focuses on traffic monitoring in the network management.

2.3.1 CpMonitor

CpMonitor is the key component that enables our concept. CpMonitor has a back-end module that monitors traffic. A front-end module is an SNMP agent that provides second order data.

The most important characteristic of a CpMonitor is that it can generate management information in the form of a MIB (CpMonitor-MIB) that includes Counter-type MOs (Managed Objects) corresponding to almost all fields of an IP packet header.

CpMonitor can provide statistics for daily monitoring and detection of events. It can also provide detailed information that is referred to when analyzing the cause of the event. It generates both first-order and second-order traffic information. It can also be accessed by the standard management protocol (SNMP). This makes our implementation simple and elegant.

The concept of CpMonitor is similar to that of an RMON device. It is a lightweight software and can work from small PC box. By distributing many small CpMonitor boxes, a scalable monitoring system for a large size network can be constructed.

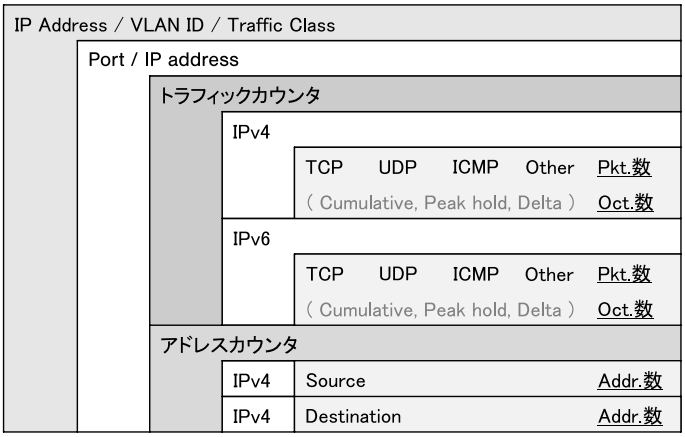

Fig. 2.2. Information categories generated by CpMonitor

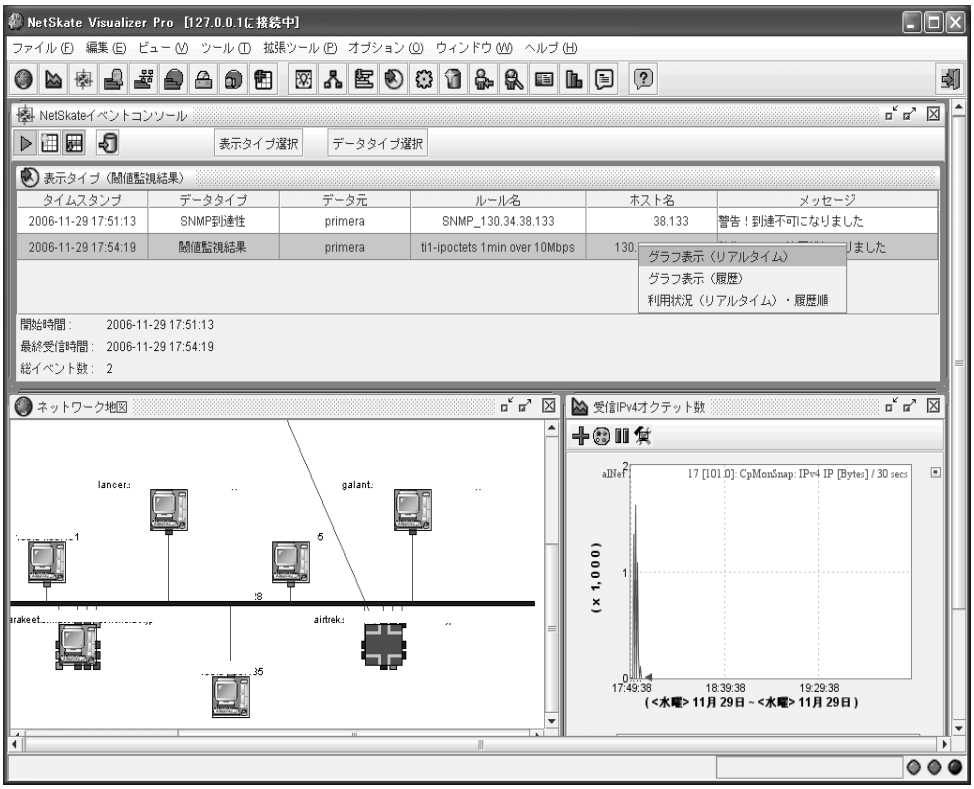

Fig. 2.3. Screenshot: Event console

2.3.2 Event console and detail information viewer

One of the core parts of our system is the “event generator”. Network manager defines events in terms of changes in one or more statistics. It has a simple user interface and almost any type of change in any type of statistics can be defined as “events”.

An “event console” provides a comprehensive view of detected events. It has a function that allows event selection by type of events. The most important function is stepwise refinement of the cause of the event. If an event is due to a change of traffic volume, with a simple operation we can access the composition of changed traffic. By only one action it provides per-IPaddr traffic composition in the first step, and in the second step it shows per-Port traffic composition.

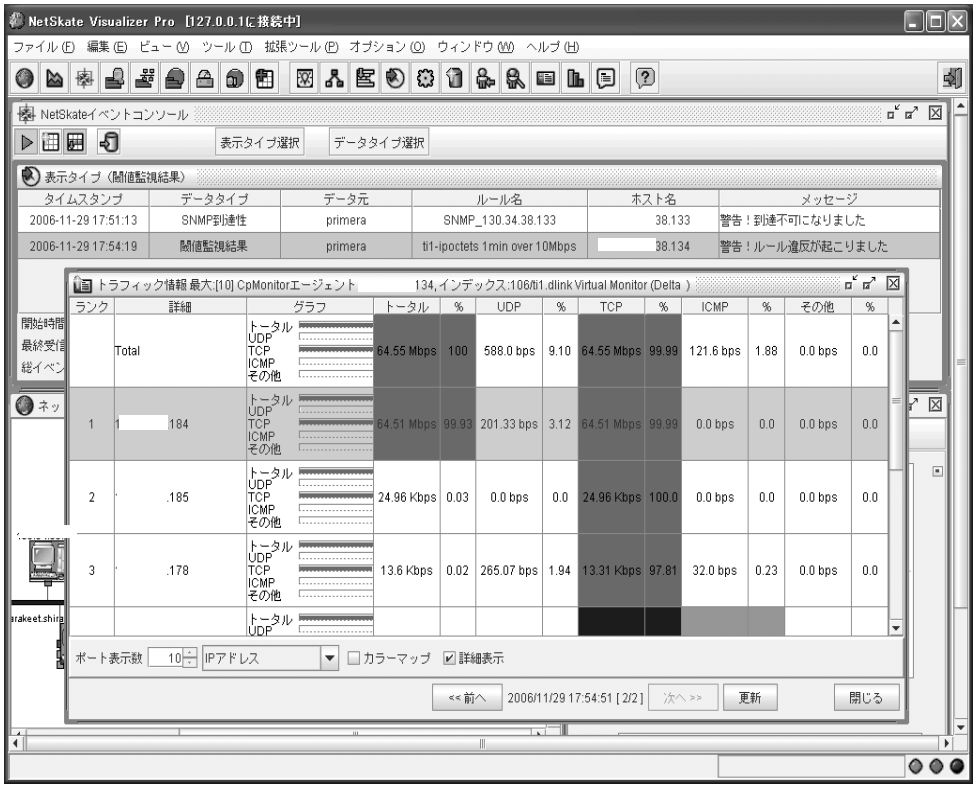

Fig. 2.4. Screenshot: Detailed information per IP address

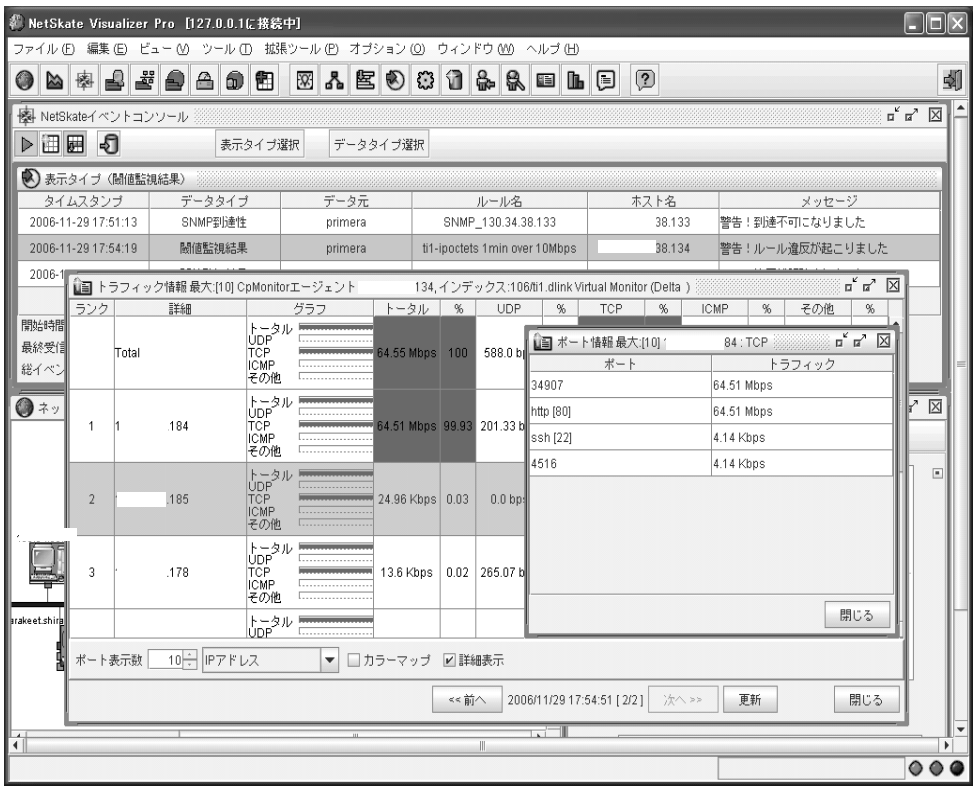

Fig. 2.5. Screenshot: Detailed information per Port

2.4 Information-sharing in Wide-area Network Management

2.4.1 Usefulness of wide area information in network management

Some types of incidents, virus transmissions etc. affect many networks over a wide area. The changes in traffic statistics will be detected as an event in many networks in these cases.

When analyzing the cause of these events, monitored results of other networks will be useful to understand the incidents correctly. Wide area network information is sometimes useful for network management. But it is almost impossible to share network information, especially first-order traffic data that may include personal information, between different organizations with different policy. Sharing second-order traffic data (statistics) 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.

2.4.2 Event-based information-sharing: using IODEF

The concept of "event" will solve this problem. Sending information on detected events to other organizations, (if the organization operates the event-based monitoring system) the manager can carve out only the data related to the event. It has a low load to share only a part of data that will be needed.

Event is defined in each organization individually. To transmit the content of the event correctly to other organization, XML-based description format will be suitable. IODEF is a good candidate for a standard format. When a network management system receives the IODEF message from the organization that detect events, it will check the content of the event, extract the detailed data that is related to the event, and can send them to the sender. XML (IODEF) is also useful to send data because it can describe any type of data.

2.4.3 Implementation and application

We implemented the IODEF agent/server functions for our event-based network monitoring system to transmit event information. As the transport method of IODEF, we use e-mail (SMTP and POP) in our implementation. E-mail itself is a connection-less communication method. So it can be used to make a query about a detected event over a wide area using some mail-broadcast mechanism. E-mail already has secure transport, privacy and sender authentication mechanism (PGP, etc.). It can be used without difficult configuration.

2.5 Conclusion

We developed an event-based network monitoring, and data-sharing system. The data sharing mechanism uses the IODEF format. This makes wide area network management more scalable and effective. In the next step we want to provide the platform to deploy the system over existing monitoring systems in many networks. In the future the platform will work as a middleware that integrates many network monitoring activities and appears to them as a large-scale network monitoring system.

第 3 章　Conclusion

During 2006 we have attempted to address the issue of effective network monitoring over a large distributed network. We have proposed mechanisms to focus the information presented to the operator, on the events of interest. Further we have put in place mechanisms wherein detailed information required to analyze and diagnose the event is collected when the event occurs. We have also proposed and used the IODEF format for representing events. This allows sharing of event information and expands the scope of diagnosis for network events to the entire Internet. A prototype

is prepared and experiments are being carried out. We intend to carry out experiments over a wide area and establish the efficacy of the system.