

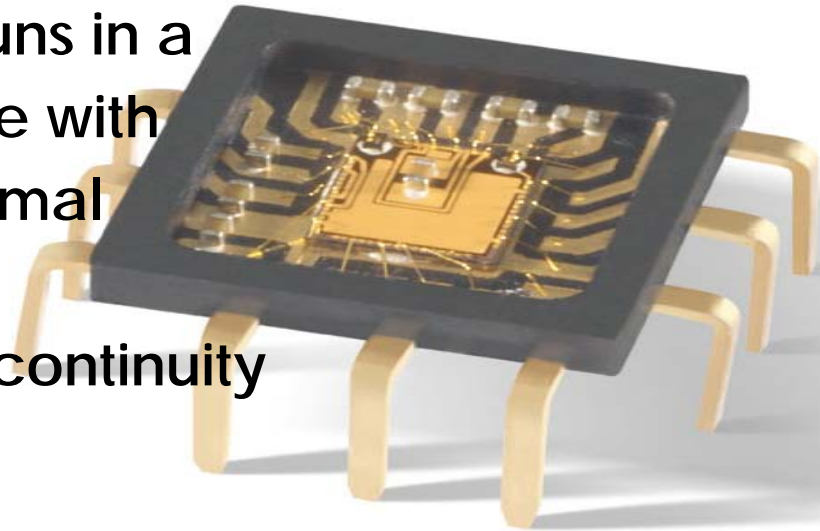


jNetmon:

jEnterprise Suite

Technical Whitepaper, February 2007

Ensure your organization runs in a peak network performance with robust protection with minimal disruptions and downtime. Uncompromised business continuity in a volatile environment.



KL Office:

1010 & 1011, Tingkat 10 Blok D,
Dataran Usahawan Kelana,
17, Jalan SS 7/26, Kelana Jaya,
47301 Petaling Jaya, Selangor
Darul Ehsan

Tel: 603-7880 0411
Fax: 603-7803 5062

Penang Office:

Suite 242, Kompleks Eureka,
Universiti Sains Malaysia,
11800 Minden,
Penang, Malaysia.

Tel: (604) 659 3590 / 659 0809
Fax: (604) 659 3591



1.0 INTRODUCTION

1.1 EXECUTIVE SUMMARY

Network administrators need to understand the data that is actually being sent over networks. Network monitoring is critical in network optimization and security. How much data was sent? When? What was sent? Who sent it? What is causing the network to crawl? And the list goes on. Current tools base their analysis primarily on the IP and TCP headers, which can be misleading or intentionally falsified. This leaves network administrators manually sifting through raw network packet dumps, piecing fragments of information together to understand problem in hand. This is tremendously time-consuming and – since networks deal with one packet at a time, while humans ask applications to perform tasks that might open a dozen simultaneous connections – ultimately not very useful to someone needing a big picture view of an employee’s suspected network abuse, or all the details of an intrusion attempt.

This white paper is designed to brief marketing professionals and systems administrators on network monitoring and introduce the main applications of the jNetmon suite of monitoring products. Network Monitoring has been around as long as there have been networks. Most routers, switches, and intelligent hubs collect some level of network traffic statistics. This information is important to network administrators who are responsible for keeping small networks at peak performance. Without network monitoring systems, it would be difficult to identify and resolve network problems. To ensure 100% system availability, IT department not only need to reduce the problem resolution time, but also needs to prevent problems from occurring. With these kind of tools, network and application administrators are alerted when there are network problems. They can regularly assess the health of their network system and take proactive measures before facing any fatal crashes.

1.2 INTENDED AUDIENCE

The intended audience for this paper is administrators, network operators, or information technology (IT) professionals. It appeals to anyone involved in managing an organization’s network be it small or large. The primary focus of this whitepaper is to help network administrators and technicians monitor and analyze network performance with jEnterprise. Often it is not easy to determine

what causes network performance problems. The suite of tools presented can be used in the production network to help identify the cause of network performance issues and in turn provide multi-level protection. This whitepaper explains how to install and configure each tool, and describes how to use the tool in the production network.

1.3 SCOPE AND PURPOSE

The job of the network administrator is complicated. If you are a network administrator, not only are you responsible for installing and maintaining the wires, hubs, switches, and possibly routers and firewalls, but you must also ensure that they all work efficiently together. The purpose of this whitepaper is to show jEnterprise, a powerful suite of network monitoring and protection tools that can be used to help monitor and troubleshoot a network, providing you with a toolkit of programs to use when problems arise.

The network administrator is often the first line of defense whenever anything behaves oddly on the network. Even when there is no clear-cut culprit, the network administrator has to prove that the network is not at fault. Network performance is often a difficult thing to measure. What is fast performance for one application can often be slow for another. It is your job to ensure that the network and the network applications are performing properly for your environment. jEnterprise is intended to help you with this task.



2.0 DEFINING NETWORK MONITORING

2.1 NETWORK MONITORING OVERVIEW

The term network monitoring describes the use of a system that constantly monitors a computer network for slow or failing systems and that notifies the network administrator in case of outages via email, pager or other alarms. It is a subset of the functions involved in network management.

While an intrusion detection system monitors a network for threats from the outside, a network monitoring system monitors the network for problems due to overloaded and/or crashed servers, network connections or other devices. For example, to determine the status of a web server, monitoring software may periodically send an HTTP request to fetch a page; for email servers, a test message might be sent through SMTP and retrieved by IMAP or POP3.

Status request failures, such as when a connection cannot be established, it times-out, or the document or message cannot be retrieved, usually produce an action from the monitoring system. These actions vary: an alarm may be sent out to the resident (SMS, email, etc.) network/system administrator, automatic failover systems may be activated to remove the troubled server from duty until it can be repaired, etc.

Most intelligent network devices offer analysis of layer 1 traffic. At this level, the analysis typically focuses on physical network problems such as link status, CRC errors, and framing errors. Dedicated monitoring equipment is often used to analyze layer 2-traffic. Layer 2 and 3 monitoring systems are commonly referred to as "protocol analyzers" because those higher level networking layers rely on special protocols to control the transmission of data. The latest generation of network monitoring products is designed to support very specific applications. For example, some monitoring products are designed to help network administrators identify security threats; some are designed to provide law enforcement officials with tools for real-time surveillance; some are designed to analyze the performance of specific applications; and some are designed to collect raw data for intensive out-of-band analysis. Each of these specializations can yield a focused solution that is designed to address the specific requirements of a vertical market.

Although the proper installation and certification of the structured cabling environment can provide a solid foundation for a maintainable network, administrators also need easy-to-use, cost-effective solutions for the ongoing

monitoring of actual traffic conditions. Thus, the need for a proper network monitoring tool is essential to both network administrators and curious users for that matter. Network administrators need tools to simulate worst case or stressed network performance scenarios to assist in network capacity planning. Network administrators generally need more in-depth troubleshooting, traffic monitoring and analysis solutions that can see across the whole spectrum of network activity under actual or simulated operating conditions. Without a sophisticated monitoring solution, network applications and devices can fail without so much as a peep. If that crash occurs over a holiday or weekend, you could be out of business for hours, or possibly days. Worse yet, you might find out from an irate customer.

Much work has been devoted to the attempt to define network performance exactly. Network performance is a complex issue, with lots of independent variables that affect how clients access servers across a network. However, most of the elements involved in the performance of networks can be boiled down to a few simple network principles that can be measured, monitored, and controlled by the network administrator with simple, intuitively designed suite of software.

Most network performance tools use a combination of five separate elements to measure network performance:

- Availability
- Response time
- Network utilization
- Network throughput
- Network bandwidth capacity

2.2 NETWORK PERFORMANCE MEASUREMENT

2.2.1 AVAILABILITY

The first step in measuring network performance is to determine if the network is even working. If traffic cannot traverse the network, you have bigger problems than just network performance issues. The simplest test for network availability is the ping program. By attempting to ping remote servers from a client device on the network, you can easily determine the state of your network.

Just about all Windows and Unix implementations include the ping program to query remote hosts for availability. The ping program sends an Internet Control Message Protocol (ICMP) echo request packet to the destination host. When the echo request packet is received, the remote host immediately returns an echo reply packet to the sending device.

While most network administrators know what the ping program is, few know that there are lots of fancy options that can be used to perform advanced testing using the ping program. The format of the ping command is:

```
ping [-dfnqrvR] [-c count] [-i wait] [-l preload] [-p pattern] [-s packetsize]
```

You can use different combinations of options and parameters to create the ping test that best suits your network environment. Often, just using the default options and parameters provides enough information about a network link to satisfy availability questions.

2.2.2 RESPONSE TIME

As seen in the ping example, while network availability is one element of network performance, it cannot accurately reflect the overall performance of the network. The network customers' perception of the network is not limited to whether or not they can get to an individual server. It also includes how long it takes to process data with the server.

To obtain a more accurate picture of the network performance, you must observe how long it takes packets to traverse the network. The time that it takes a packet to travel between two points on the network is called the response time.

The response time affects how quickly network applications appear to be working. Slow response times are often magnified by network applications that need to send and receive lots of information across the network, or applications that produce immediate results from a customer entry. Applications such as TELNET, which require the customer to wait for a keystroke to be echoed from the remote host, are extremely vulnerable to slow network response times.

While network response time is often obvious to customers, trying to measure the response time between two separate hosts can be a difficult thing to do. Determining the time it takes for a packet to leave one network device and arrive at a remote network device is not easy. There must be some mechanism to time the leaving and arriving events, independent of the two systems on the network.

In large networks, there are many factors that can affect response times between a client and a server. As the network administrator, you can control some of these factors, but others are completely out of your control. These factors can include:

- Overloaded network segments
- Network errors

- Faulty network wiring
- Broadcast storms
- Faulty network devices
- Overloaded network hosts

Any one or combination of these factors can contribute to slow network response time. Measuring the individual factors can be difficult, but the network monitoring application presented in this whitepaper can measure the overall effect each factor has on network response times by sending known network traffic samples and determining how the data traverses the network.

2.2.3 NETWORK UTILIZATION

A major factor in network performance is the utilization of each network segment along the path between two endpoints. The network utilization represents the percent of time that the network is in use over a given period. By definition, individual Ethernet segments can only carry one packet at a time. For any given moment, the Ethernet segment is either at 100-percent utilization (carrying a packet), or at 0-percent utilization (idle). The network utilization percentage shows the percentage of time the network is in use over a set period.

Calculating the network utilization requires you to find out how many bytes of network traffic are being handled by the network over a set period. This value depends on the type of network interface that is being monitored. Half-duplex devices can only carry data in one direction at a time, and therefore calculating their network utilization involves totaling the input and output byte counts for a set period, and dividing by the total capacity of the device interface for that period. To determine the total number of bits received on the interfaces, each of the packet byte rates is multiplied by 8. This value is divided by the total interface capacity multiplied by the time interval of the sample (in seconds):

$$\%utilization = ((datasent+datarecv) * 8) / (intspeed * sampletime) * 100$$

For example, a 10-MB half-duplex network interface that over a 5-second period sends 700,000 bytes of data and receives 175,000 bytes would have a network utilization of:

$$\%utilization = (((700,000+175,000) * 8) / (10,000,000 * 5)) * 100 = 14\%$$

The 14-percent utilization represents the network utilization only for that 5-second period. It is not uncommon to see high network utilization for a short period of time, given that Ethernet traffic is often bursty in nature. You have a problem when you take the same calculation for a longer period of time, such as a 5- or 30-minute interval, and still get high network utilization.

Although calculating network utilization on an individual network segment can be easy, determining the network utilization between two separate endpoints on the network can be complex. You must calculate the network utilization for each segment traversed along the network path, and determine how each segment's utilization affects the overall response time of the packet.

Due to the complexity of this, most network performance tools utilize different elements—the network throughput and the network bandwidth capacity—to determine network performance between two remote network endpoints.

2.2.4 NETWORK THROUGHPUT

Network throughput is similar in concept to network utilization. The throughput of a network represents the amount of network bandwidth available for a network application at any given moment, across the network links. As network applications use network bandwidth, the amount of bandwidth left over for other applications is decreased. The amount of bandwidth left over is considered the network throughput.

Determining network throughput allows the network administrator to find network bottlenecks that slow down performance over a given network link between clients and servers. Often a novice network administrator places a group of clients on a high-speed network device, and the application server on another high-speed network device, to increase application performance. However, what the administrator forgets is that the two high-speed devices may be connected via a slow-speed link. Figure 1.0 demonstrates an example of this.

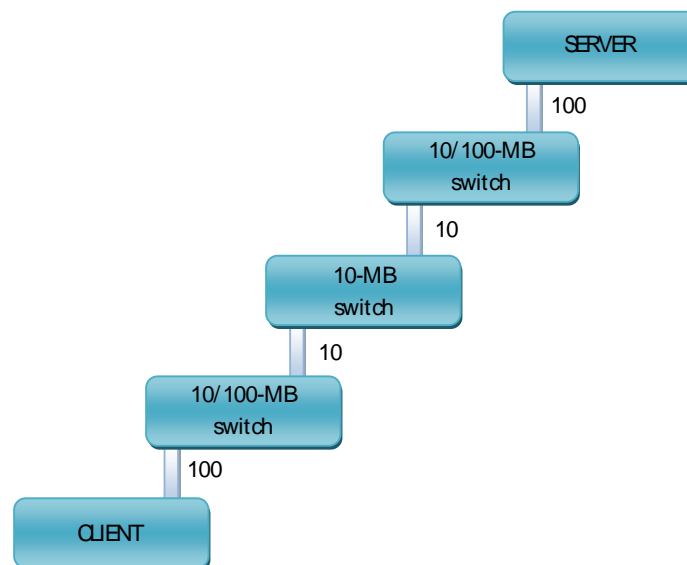


Figure 1.0. Finding the throughput bottleneck.

While the networks that contain the client and server devices are high-speed and have good network performance, it is the interconnecting network device that is causing performance problems. First off, the intermediate network link is limiting the overall speed of the end-to-end link to only 10 MB, no matter how fast the clients and server are connected to the network. Second, since the intermediate network device is a shared hub, it may contain other clients and application servers, which puts additional traffic load on the slow-speed link. Usually, finding the network bottleneck is not this simple. On complex networks, there can be several network devices within the path of clients and servers. The hardest part of determining the network throughput is calculating the effect that each intermediate link has on the overall end-to-end network connection.

Calculating network throughput is a mathematical process that is best left to the mathematical geniuses. It involves sending periodic streams of packets, and determining the rate at which the server receives the streams. Each stream sample produces data elements used to determine the amount of bandwidth remaining on the network link. The streams are increased until the maximum bandwidth is observed, then quickly backed off so as not to affect the network performance.

2.2.5 BANDWIDTH CAPACITY

Bandwidth capacity is another factor in the determination of network throughput. The total amount of bandwidth available between two network endpoints can greatly affect the performance of a network. Devices directly connected on a 100-MB network switch should have significantly better performance than devices that are remotely connected via a slower T1 circuit.

The ability to quantify this performance difference requires complex network theory to be built into the network performance tool. The network performance tool must be able to determine the possible end-to-end network bandwidth available on networks with varying link speeds. Each link that a packet must traverse must be included in the overall network performance of an application.

In an ideal network scenario, a constant data rate should be maintained between a client and a server as packets are sent back and forth. The constant data rate represents the network speed at which the two endpoints are linked together. By observing the time it takes for a packet to traverse the network, you can determine the maximum speed of the network link. As we all know, there is no such thing as an ideal network scenario. In production networks, traffic is constantly traveling between network devices, affecting the perceived speed of a network link. In order to determine the maximum bandwidth capacity of a network link, the network performance tool must do some math tricks.

2.3 WORM: A SILENT NETWORK INTRUDER

A computer today is only as useful as the network it is attached to. In the end, information technology is most valuable when it is used to aggregate data from multiple sources, perform some really interesting task with that data, and then share it with someone else. The infrastructure that makes this all happen is the network. Several years ago, Microsoft launched a marketing campaign themed around the “Digital Nervous System.” The digital nervous system was the network. It sounds corny to those of us who do not spend all day thinking about how to sell something, but it does make some sense. The network is what allows data to flow from the place where it is stored to the place where it has some impact. In the end, it is all about data; data that you convert into information and then share in such a way that you get maximum benefit from it.

Network protection is about ensuring that the infrastructure where all this happens is available, that data and information does not leak into the wrong hands, and that the data and information arrives at its destination intact.

Network security as an end state is a pipe dream, an impossible reality that we cannot attain. We constantly get asked how to make a network secure, “Security” is defined as “freedom from risk or danger; safety.” It is obvious that “security” in computers can never attain this lofty goal. Computer security is more “management of risk” as will be discussed in Section 4.0. Network security is a process, a task description, not an end state. Therefore, we like to talk about network protection as the goal, and network security as a task description. The task (as shown in Figure 2.0) is to detect problems and, preferably before someone else does, respond to those problems in a way that prevents them from becoming security vulnerabilities. At that point, the process repeats, and we look for more problems to prevent.

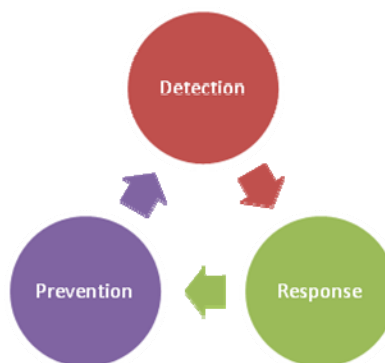


Figure 2.0. The security Process.

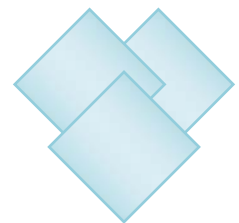
Essentially, network attacks can be distinguished on two dimensions: passive versus active and automated versus manual. A passive attack is one that uses network tools, such as a sniffer to capture network traffic, that simply listen on the network. These tools may capture traffic that contains sensitive information.

Active attacks, by contrast, are where the attacker is actively going after the protected resource and trying to get access to it, possibly by modifying or injecting traffic into the network.

On the other dimension, we have automated attacks. The vast majority of the attacks we hear about today are automated attacks, where the attacker creates some tool that attacks a network all by itself. The tool may have some intelligence built in, but fundamentally, if the network is not configured the same way as the one the tool was written for, the tool fails.

Worms are methods of automated attack. In most cases, automated attacks are based on a known vulnerability in a system. The best method of defense against an automated attack is simply to keep the system fully patched at all times and monitor your network for suspicious events or messages.

Worm is a form of **active-automated** attack. At first, it appears these attacks do not exist. Most network worms fall into this category. For instance, a worm that searches for machines that are missing a particular patch, exploits it, and then uses the compromised machine to find additional targets falls into this category. Another example of this is an attack that uses thousands of hosts to target a single network to cause a denial-of service condition. Tools now exist that can exploit hundreds, maybe even thousands, of systems at the click of a button and return information to the attacker about exactly which attacks succeeded. These attacks are very disturbing, but they are usually also very noisy. In addition, they usually rely on exploiting unpatched vulnerabilities. When doing this, the risk of crashing systems is pretty high, and that would be very noticeable.



3.0 CURRENT APPROACHES

After determining which network performance elements to monitor, the network performance tool must be able to access data for the elements. Three different methods are used to obtain data from the network:

- Querying network devices for stored information
- Watching existing traffic on the network for signs of network performance issues
- Generating test traffic to send on the network to test network performance

Part of the network administrators job when evaluating a network performance tool is to determine how the tool extracts data about the network's performance, and if that method is appropriate for your particular network environment. Network monitoring tool collects various network data and generate decision-quality information at the network administrator's disposal. It can be considered that network monitoring is part of network management, but it has more to offer compared to the well known standards of network management described above. Basically, it provides real time network traffic monitoring. It sniffs every packets going through the network, decodes the packets and analyzes it. This operation does not alter the packets, which will continue on to their destinations. Its primary purpose is to provide tools to diagnose, prevent, and treat network problems.

The key elements of the network monitoring tool are the packet sniffer, protocol analyzer, fault monitor and thin agent. Packet sniffer gathers captured network traffic in real time without causing overhead to the network traffic. Protocol analyzer decodes the packet and provides detail break down of the composition of the packet. Protocol analyzer monitors header information on each series of data, tracks data from its starting point to its ending point, determines bandwidth utilization and find out the cause of the problem. Fault monitoring detects possible network error from decoded packets provided by previous component. Finally, thin agent is installed on a different network segment to analyze the data passing through a switch.

Network monitor or probe can be placed locally (within the same location) or remotely (gathers and analyzes information locally and then transmits it to the network management station placed in other place). Network monitoring tool uses several passive monitoring methods, namely RMON Extension and SNMP Extension. RMON Extension gives access to the information gathered internally by the network devices. SNMP Extensions give access to SNMP alerts or traps and the current status

of the network devices. However, most vendors add their own proprietary extensions and add additional functionalities. Both RMON and SNMP are active monitoring tools, which pool the network devices for information.

Network monitoring tools falls into two categories

- Software based network monitor
- Hardware based network monitor.

Hardware-based probes are specific to administrators that require fast, full-duplex gigabit capture, analysis, trending and statistics. While software-based probes are appropriate for network speeds up to 100 MB (Ethernet, Token Ring, FDDI and 802.11b Wireless) and are a good solution for monitoring low utilization gigabit networks via a tap port on a switch.

The major difference that distinguish software based network monitoring from its hardware based counter part is pricing and hardware requirement. Generally, software based network monitor are cheaper and easier to implement. It does not need specialized computer, or any specific hardware to run. It can be installed on any workstation, computer or any notebook computer that connected to the network, while hardware-based network monitors are expensive and usually costs around US\$10,000.

Another difference with hardware based network monitoring is the inability of the network monitor to be used immediately on different network segment. Normally, software-based network monitor are used to manage networks in one location only. While hardware-based network monitor, network administrator can plug into the network and retrieve information without need to configure hardware and network connection. In a nutshell, hardware-based network monitors are more “plug-n-play” compared to its software counterpart.

4.0 ENTERPRISE RISK MANAGEMENT & NETWORK PERFORMANCE

What does risk management have to do with network monitoring and intrusion detection? Every organization either consciously or subconsciously makes decisions about risk. Obviously, we decide how much risk we are willing to accept ourselves. The distributed denial-of-service attacks that became widely known in February 2000 and Code Red attacks in 2001 demonstrate clearly that we also decide how much risk we are willing to accept on others' behalf. The security of my site depends, at least in part, on the security of your site. The highest and best purpose of a network intrusion-detection system is to identify the attacks being directed against our perimeter defenses so that we can ensure our systems are hardened to withstand these attacks whilst network monitoring system keeps an eye on the network performance of an enterprise.

In other words, network monitoring and intrusion detection must serve as instrumentation that enables us to define the metrics we need to manage risk intelligently.

Table 1.0. Top Twenty Vulnerabilities within an Enterprise

Operating Systems
W1. Internet Explorer
W2. Windows Libraries
W3. Microsoft Office
W4. Windows Services
W5. Windows Configuration Weaknesses
M1. Mac OS X
U1. UNIX Configuration Weaknesses
Cross-Platform Applications
C1. Web Applications
C2. Database Software
C3. P2P File Sharing Applications
C4. Instant Messaging
C5. Media Players
C6. DNS Servers
C7. Backup Software
C8. Security, Enterprise, and Directory Management Servers
Network Devices
N1. VoIP Servers and Phones
N2. Network and Other Devices Common Configuration Weaknesses
Security Policy and Personnel

Others

H1. Excessive User Rights and Unauthorized Devices

H2. Users (Phishing/Spear Phishing)

Z1. Zero Day Attacks and Prevention Strategies

4.1 ENTERPRISE SECURITY MODEL

To manage risk, we need a model, a way of describing the problem and what needs to be done from a process standpoint so that we can get our arms around the problem. A simple example of a model is the Top Twenty list as shown in Table 1.0. It lists the top twenty vulnerabilities that attackers exploit and how to fix them. Every major vulnerability scanner looks for evidence of these. This is a simple model, listing the twenty vulnerabilities most often exploited. The enterprise has to make sure there are tools to find these vulnerabilities, and describe the fixes so that all users can repair their systems. If a significant number of people do this, attackers will have a much harder time compromising systems, and everyone's risk is reduced.

4.2 MONITORING ENTERPRISE NETWORK ELEMENTS

To keep the network at peak performance, some of the network elements need continuous monitoring, such as:

Email Servers:

Organizations, nowadays depends on emails for almost all external connections specially with customers, so if the email server is down/fails, users are disconnected from the external world and key functions such as customer support takes a hit.

WAN links:

IT administrators should cautiously monitor the throughput, committed information rate (CIR) and burst rate with congestion, response time, and discards to optimize the link utilization. IT administrators should also find out who's using the most bandwidth to take the proper action or who the top users are. Apart from bandwidth monitoring, routers need to be monitored for availability and performance from time to time. If a router fails it halts the entire LAN and hence IT administrators should set thresholds on various parameters on routers and solve problems immediately.

Business Applications (Figure 3):

- Servers & Services: since Servers run the critical applications/services a careful monitoring for CPU, memory, disc space, services running on them

(FTP, DNS, ECHO, IMAP, LDAP, TELNET, HTTP, POP, etc.) and their response time, the traffic utilization trends of these servers should be taken into consideration from time to time.

- Applications, Databases, & Websites: to ensure the smooth running if any businesses all the mission critical applications, websites, and databases needs to be monitored periodically. Applications can be monitored for availability, response time etc.
- LAN Infrastructure: Your LAN infrastructure devices such as switches, printers & wireless devices.

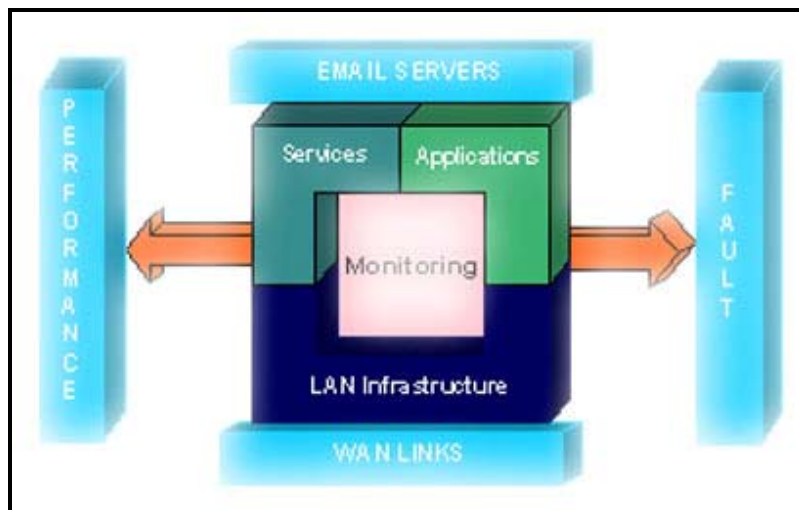


Figure 3.0. Network Elements/Services to be monitored



5.0 OUR SOLUTION: JENTERPRISE SUITE

As networking environment, especially those in large enterprise and organizations, gets more and more complex, locating and isolating the source of any network problems without proper tool is equivalent to finding a needle in the haystack. Ensuring the computing resources and the networking platform hosting them are protected is also becoming a war against the impossible because new worms/viruses/attacks crop up everyday and it is difficult to detect and remove these malicious attacks.

Network and system administrators have turned to network monitoring and protection tools to ensure smooth operation of networks within and without an enterprise. Most organization has about 50 to 1000 computers and the numbers are growing. Imagine one of the computers is flooding the network; the system administrator would then have to investigate each machine to find out which one is causing the problem by making desk-side visits. Such process may is cumbersome, inaccurate and in some cases, take days (even weeks) to pinpoint and solve the problem. While the problem is being investigated, network performance drops and in turn affects the business continuity, causing the organization thousands, even hundreds of thousands of dollars lost due to downtime. Allowing such conditions to occur and prevail may eventually jeopardize the position of the system or network administrator, for not having the foresight to install the proper prevention and troubleshooting tools to ensure that such problems do not occur.

jNetmon boasts a confluence of innovative technologies coupled with many years of research undertaken in the area of network monitoring. It is aimed to take the burden of network administrators.

Without proper tools that can interpret, analyze and display network traffic and related problems, a network administrator is limited to the time-consuming trial and error method trying to identify problems. The tool should also be able to confide to the universal belief that *"Prevention is better than cure"*. It should be able to forecast/predict any imminent network issues before it actually happens. This entails the use of convergence of various intelligent techniques.

As part of the jNetmon Product Line, jEnterprise is a suite of applications and tools that successfully addresses the needs of an organization in combating and preventing network problems. Its goal is to provide a robust, non-pervasive, real time network monitoring capabilities. The intuitively designed user interface allows a short learning curve for network administrators to get acclimatized with the applications.

We believe jEnterprise will be able to provide global access to remotely monitor any network on any corner of the world tunneling via the Internet. Total security is ensured as the remote sites determine the level of monitoring permission. It offers complete monitoring from Physical Layer to Application Layer, obtains comprehensive details of the network, and last but not least, it is easy to use and deploy. jEnterprise ensures your business productivity unperturbed by maintaining a healthy and efficient network with uncompromised performance. "Prevention is better than cure" is definitely a universal belief and certainly applies to network problems. If they can be detected earlier and the necessary actions are taken to minimize them, averts potential loses due to a productivity loss. jEnterprise extends the capabilities of a network administrator from just troubleshooting the network into developing and designing better network topologies for the company. It presents and visualizes network statistics in real time in various intuitively designed charts and reports. jEnterprise improves systems service level in terms of monitoring and troubleshooting.

5.1 METHODOLOGY

jEnterprise suite assists network administrators in finding and locating problems quickly and easily. Armed with j-Enterprise Suite, you can troubleshoot, maintain and diagnose the user's network before a problem escalates to a point where it becomes detrimental to the networks performance. The uniqueness of jEnterprise Suite as compared to other tools is that it not only provides a centralized visualization and analysis engine, jCMC (Centralized Monitoring Console), it also provides remote agents (jRemote) to monitor and protect each remote network segment within the organization and jServer to collect network data from jRemote agents and perform intelligent analysis. This interesting and unique distributed architecture creates a complete protected environment for the organizations servers, PC and network. jNetmon's j-Enterprise Suite is also the world's only cross platform analysis and troubleshooting tool. This means even if your organization uses a mix of Windows and Linux based computers and servers, Enterprise Suite would still be able to fully monitor and protect your network.

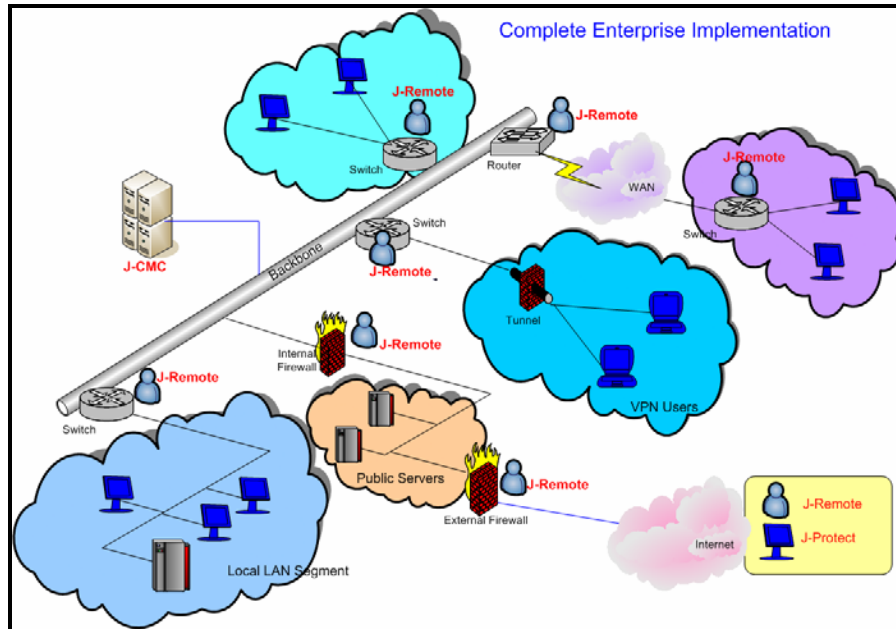


Figure 4.0. j-Enterprise suite as troubleshooting system

Figure 4 shows an example iNetmon’s j-Enterprise Suite deployment architecture. With such architecture in place, this j-Enterprise Suite deployment is able to address the following enterprise issues:

- Real-time, around-the-clock network monitoring: Continuous monitoring to prevent potential performance degradation or downtime.
- “Prevention is better than cure”: Predict potential problems and take precautionary actions before it actually happens.
- Intelligent Troubleshooting: Intuitively supplement network administrators with instructions and steps on problem fixes including those in remote segments as well as on preventing potential problems.
- Support for New Protocols: Monitoring covers next generation

5.2 IMPLEMENTATION

The traditional enterprise monitoring concept involves sending information packets from the agent to the console on a periodic basis which is known as streaming technique. This technique would drain the bandwidth and thus deny bandwidth to other more critical applications. Network infrastructure have to cope with high bandwidth applications and at the same time needs a tool to keep an eye and scrutinize network health. As we know, conventional monitoring approaches would only aggravate and cause the network to perform poorly. We need a tool that robustly monitors and at the same time does not affect the network performance while performing its tasks. In view of this, we have introduced a radical concept in addressing the bandwidth issue. One of the goals of our products is to use technology or techniques that consume minimal

bandwidth and yet provide real-time and up to date information about your network. jEnterprise suite uses non-streaming techniques to provide centralized monitoring.

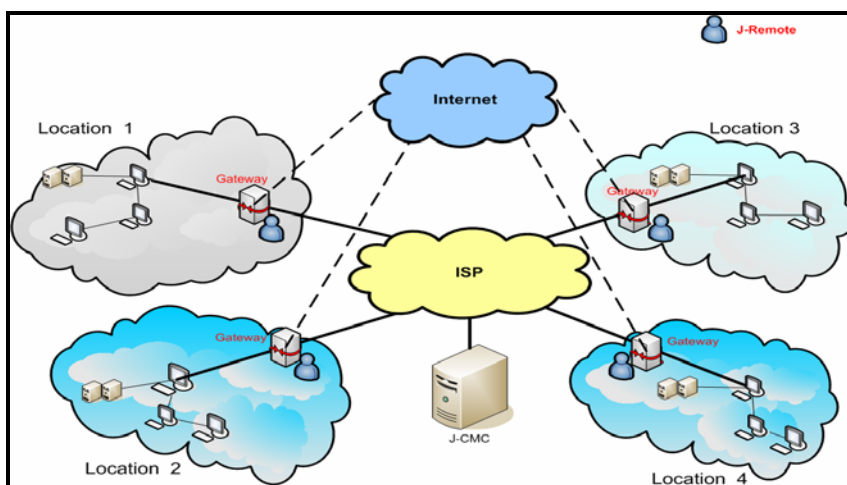


Figure 5.0. jEnterprise Suite as Real Time Network Monitoring

The remote intelligent agent (jRemote) would gather network information and store it in the local segment in the form of a simple text file. As and when where administrator needs to probe into that particular segment, the Centralized Monitoring Console (jCMC) would request for the file from the remote agent. Immediate past incident scenario can be recreated within jCMC to investigate the cause of the network problem. This would be synonymous to the playback of videotapes of CCTV to investigate criminal acts. This concept does not drain bandwidth and at the same time provides an opportunity for the network administrator to recreate the problem and find a solution. Figure 6 illustrates the communication between jCMC with jRemote agents located in different segments of the enterprise.

Via jCMC, jEnterprise provides global access services to remotely monitor your network from any part of the world via an IP connection through the Internet. The monitoring console could also be your mobile PDA using EDGE, 3G or Wifi technologies. Total security is ensured as the remote sites use encryption techniques for data transfer.

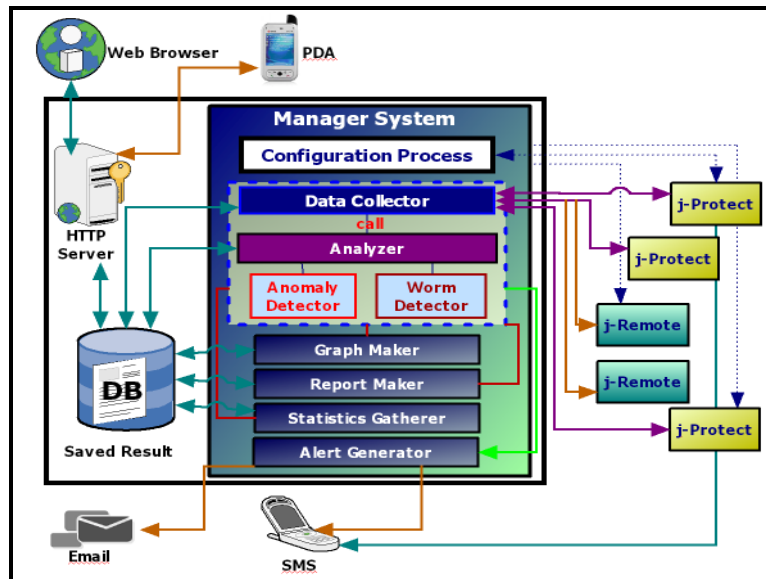


Figure 6.0. jCMC Architecture Overview

5.3 CAPABILITIES & COMPONENTS

5.3.1 JREMOTE

JRemote is an autonomous and intelligent agent that sits in every network segments, collecting various focal network data while providing full protection to the network segment against worm and related attacks. We put forward the components of jRemote:

Network Analyzer

This component is a packet Capture and Decode function that allows you to view captured traffic packet-by-packet. By using filters, you can view only the pertinent packets. It can decode all major protocols and sub protocols, and you can view both the raw data and the decoded data. For example, if you have been unable to print to a network printer, this function helps you to see whether the station is transmitting or the server is responding. If you want to ensure a mail server is sending/receiving emails as it should, you can view the packets captured from the mail server.

It is also a real time packet dissector where each single packet is captured and decoded to give meaningful information to the reader. The network analyzer can analyze the packet on multiple granular levels. The network analyzer supports packet filtering to ease and narrow down the number of packet displayed. There are three steps need to be taken by the network administrator before monitoring can proceed:

1. Choose the network interface to be monitored (if the machine has multiple network interfaces) as shown in Figure 7.

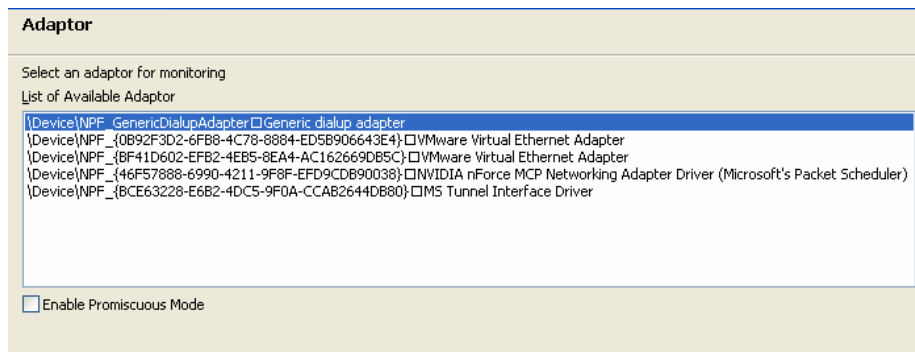


Figure 7.0. Manage Network Interface to Monitor.

2. Specify the central server where all the data will be uploaded to Figure 8.

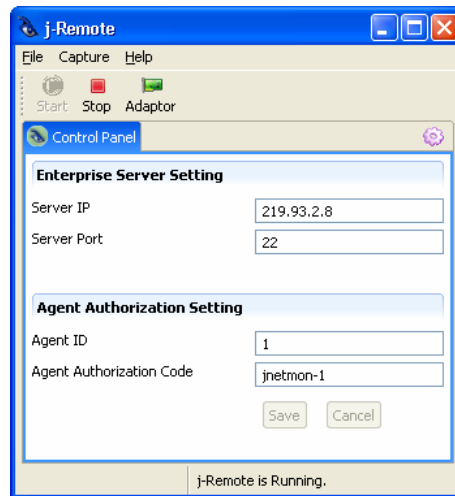


Figure 8.0. Server Setup

3. Specify the link speed, i.e. 10Mbps, 100Mbps, 1000Mbps, etc. as shown in Figure 9.

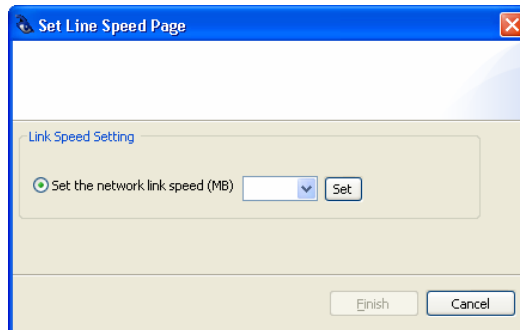


Figure 9.0. Setting the Network Link Speed.

Network analyzer can operate on real time or offline manner where a saving functionality is added to save the displayed packet. It implements a multi threaded circular buffer designed to work in multi-threaded fashion to ensure no packet loss.

When it starts the packet capturing session, two threads will be created. The first one captures packet and stores in a buffer while the other monitors and reads from the buffer. Both threads are assigned two different priority level. The capturing thread is assigned a Norm_Priority - 1 (the Norm_Priority is usually automatically assigned to the main program thread) and the packet reader thread is assigned a Norm_Priority - 2. In another words, the packet capturing process is always given a higher priority to capture all the packets and avoid packet loss. Once the application is ready to receive the next packet, it will read the buffer. The circular buffer itself is implemented using java queue concurrent implementation to enable a thread-safe way for both threads (packet capturing and packet reader) in accessing the buffer. Figure 10 visualizes the design of Network Analyzer's multi-threaded circular buffer.

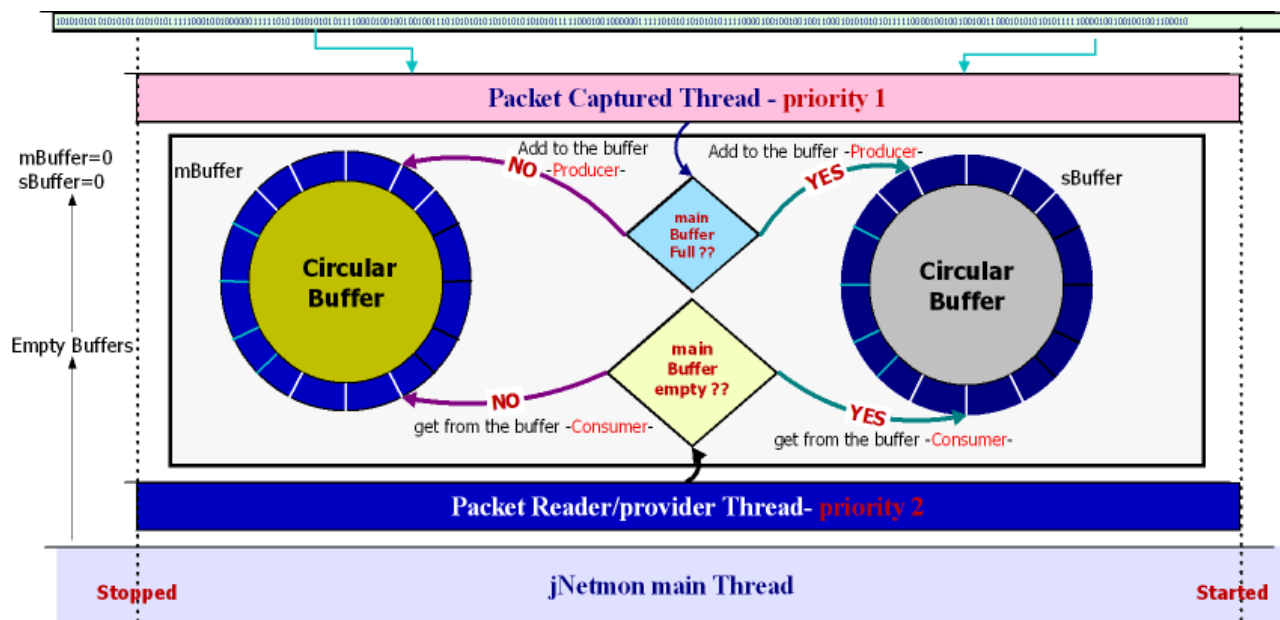


Figure 10.0. Network Analyzer's Buffering Mechanism Overview

Network Address Book

The Address Book allows network administrators to keep and supervise all nodes on the network. Useful information is kept and detected silently for further use and analysis. The Address Book will automatically discover all users on the network. It captures all network packets, analyses them to get the source or destination MAC address and then matches it with the source or destination IP and NetBios address to generate a complete network station or device address book. The address book consists of:

- Capturing and decoding

- Information Database.

Worm Protection Engine

The Engine provides RVDS packet capturing mechanism. This provides RVDS the real-time capability in capturing packets and later detecting worm attacks. All packets are captured and decoded by j-protect Engine into readable format. Next this output will be forwarded to Worm Matching Engine, to detect worm attacks on it. It is basically made up of:

- **Worm Parser:** All worm signatures in the database are parsed into memory. Worm signatures are loaded into appropriate data structure for matching purposes. Worm Parser is executed first during the start-up of jRemote.
- **Worm Matching:** After formatting the packet into readable form, the content is sent to Worm Matching Engine. Here, the packet would be matched against worm signatures that are loaded into memory by Worm Parser. This matching methodology is significantly faster compared to matching worm pattern directly from the worm signature database. In order to accommodate the speed of incoming packet, this component would be implemented using threads or multiple processes, which can utilize Symmetric Multiprocessing (SMP) system. This will further improve the available Real-Time capability of RVDS.

5.3.2 JCMC

JCMC gets network data from jRemote agents and displayed as various, time-sensitive statistics in the form of charts and tables. The charts provide information on the network activity. The packet per second (pps) and bits per second (bps) charts show the amount of packets moving on the network in every second. This can help to indicate peak hours of the network. User can monitor the flow of packets on the network. The network utilization graph shows the percentage of bandwidth utilized. This module can be used in different network bandwidth.

Traffic Analyzer enables you to see the total traffic and the bandwidth on your segment in real time. It also graphs the information against a percentage of bandwidth utilization to give you some knowledge into the total load on the segment. It is displayed in histogram format, which is useful in determining optimal network configuration and solving global traffic issues before they become global problems. Network Utilization is logged and stored in a History function in Traffic Analyzer. By comparing a current utilization to previous recorded utilizations, a network administrator can quickly determine network trends or judge the effect of a change to the network in the case of devices upgrading or topology changes.

In addition to network traffic and bandwidth utilization, jRemote also tracks all the protocols in each interval(specified in number of minutes) of monitoring activity and jCMC visualizes this myriad of network data as listed below. The parameters visualized are as follows:

- Application Protocols
- Network Protocols
- IPv4 Protocols
- IPv6 Protocols
- IPv4 versus IPv6 Protocols
- Top Usage (10 top IP addresses with high bandwidth usage)
- Application Monitoring (10 top application with high bandwidth usage)

jCMC also gathers data on worm attacks on alerts generated by the Worm Engine within jRemote and alerts generated by the built-in intelligence engine in jServer. Figure 11 show jCMC in action.

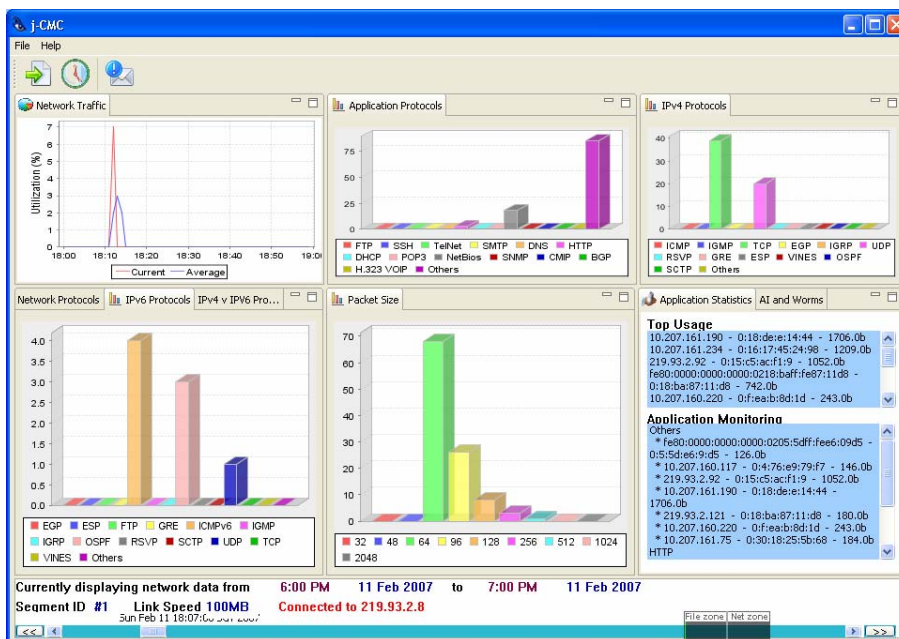


Figure 11.0. jCMC in action

The dialog box contains the following fields and controls:

- Username:** cmc-1
- Password:** *****
- Segment:** [Empty field]
- Start Date:** 11 Feb 2007, **Time:** 06:00 PM
- End Date:** 11 Feb 2007, **Time:** 06:00 PM
- Buttons:** Save & Hide, Cancel, OK
- Server IP:** 219.93.2.8
- Server Port:** 22
- Status:** Invalid server settings.

Figure 12.0. Config and Login Dialog to jCMC

Since the network data is of security concern, the network administrator would have to provide valid login credentials coupled with segment date range, server ip/port details. jCMC would then communicate with the server and retrieve the required data to be visualized. This is shown in Figure 12.

Armed with fluid and intuitive user interface design, jCMC eases the navigation of time-sensitive data with the ability to move windows around to fit the comfort of user.

5.3.2 JSERVER

jServer is a central component of jEnterprise that collects data and performs clustering and trend analysis on the data to see if there is any pattern and avert any potential disasters. It plays three crucial roles:

1. Data acquisition from jRemote agent installed in various segments.
2. Data provider to jCMC for network data visualization and playback.
3. Intelligent analysis of the data for pattern recognition, prediction and clustering.

Data Acquisition & Storage

jServer collects network data from various jRemote agents on ad-hoc basis. Each connection is monitored and logged for security purposes. Figure 13 shows jServer receiving connection from one of the jRemote agents. Network administrators can manage the database server via an easy-to-use console as shown in Figure 14. Sometimes, the server may be in storage shortage, thus it would be a better to capture data in a longer intervals. Once the storage issue has been resolved, the interval can be shortened. jServer provides the flexibility to modify the intervals (Figure 15).

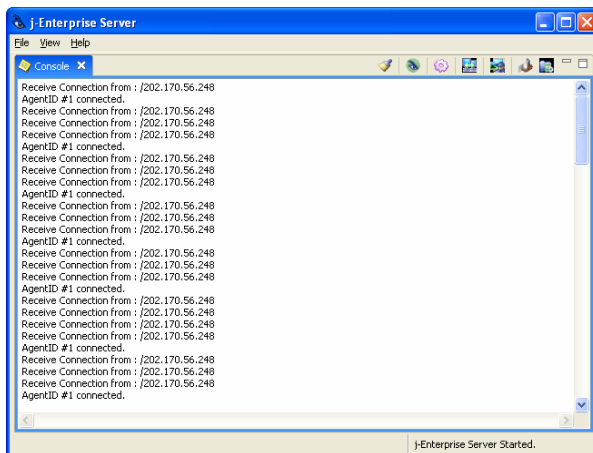


Figure 13.0. JServer-jRemote communication

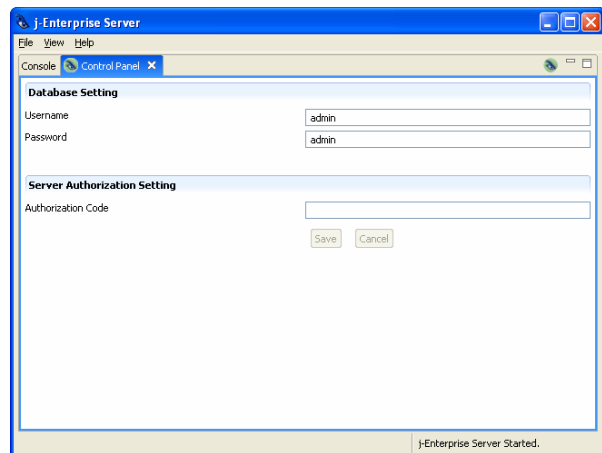


Figure 14.0. Database Management

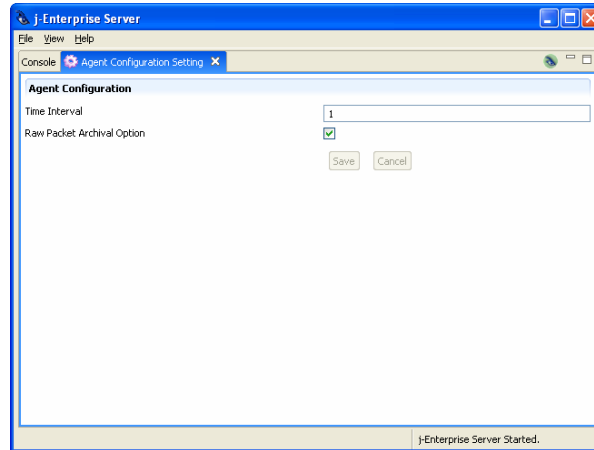


Figure 15.0. Agent Configuration

We have discussed about jRemote’s capabilities and advantages. It monitor computers in a network and the type of applications the computers are running. The application/services basically can be detected using the port numbers that being used by a particular machine and the protocol as well. As we all know, the numbers of application are increasing rapidly, that’s why the user of this module is given accessibility to add new application detail into a database through a friendly user interface. The user will gain information needed about the computers detail, e.g.: IP Address, MAC Address, the number of bytes being used (shown in “B” / “KB” / “MB”). We also associate this view with “Address Book” module, so that we can see in more details to whom a particular computer belongs to and when it was recorded. In order to provide a centralized management and configuration of distributed components, details for application monitoring is done at jServer. Figure 16 and 17 shows this.

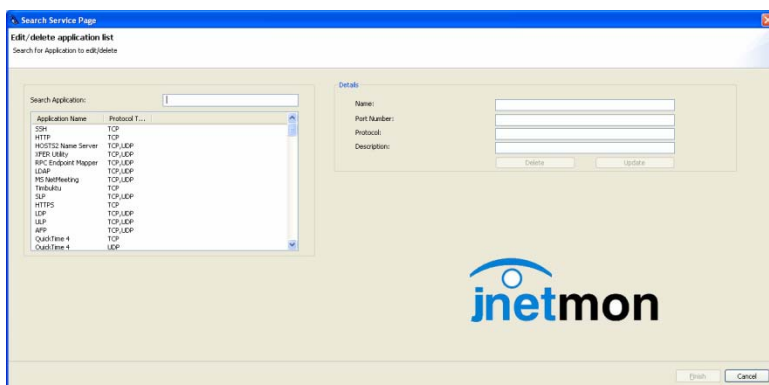


Figure 16.0. Current list of applications being monitored

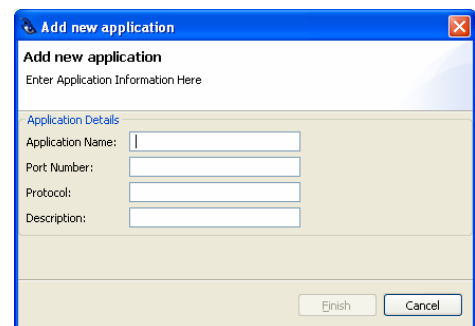


Figure 17.0. Adding new applications

Intelligent monitoring Engine

Intelligent monitoring engine (IME) is one of jServer’s main modules intended to intelligently monitor network traffic and alerts user if an anomaly is detected in the

network. The main idea is that the engine will model the normal traffic condition of a one particular network segment where jRemote was installed/tapped-in. Hence, it will only start detecting anomalies on the second week running since it will use the whole first week to train and model the normal condition. IME will only focus on detecting anomalies of network traffic flow. However, some constraint must be well considered when training the engine. To ensure the integrity and quality of the trained engine, network administrator needs to ensure that, upon starting the engine, the network state is considerably normal in a sense that the number of normal instances vastly outnumbered anomalous instances. Figure 18 illustrates the IME architecture.

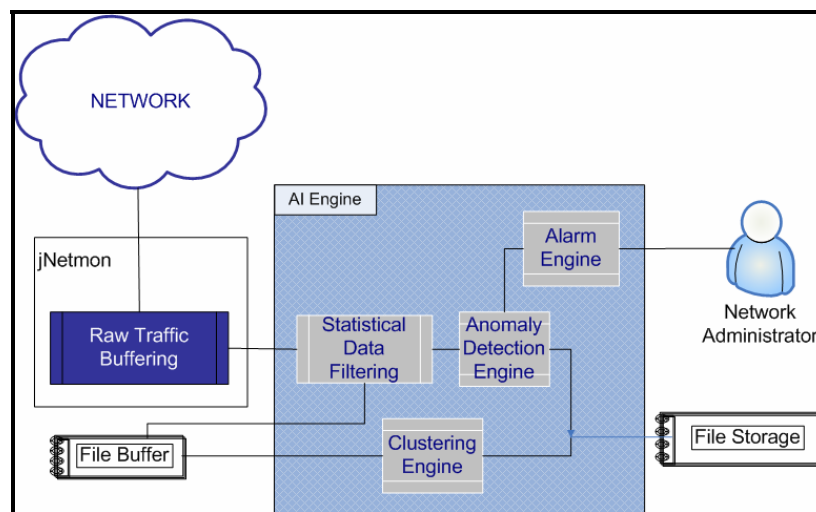


Figure 18.0. IME Architecture

IME applies the general framework of Evolving Connectionist System (ECOS) and thus is able to evolve overtime. ECOS are multi-modular connectionist architectures that facilitate modeling of evolving processes and knowledge discovery. It is a collection of artificial neural networks that resemble the human cognitive information processing models, The ECOS framework has been successfully applied to numerous challenging real world problems. Some of it involves modeling complex and dynamic process which is very difficult to model with other techniques such as gene profiling problems, adaptive speech recognition in noisy environments, image pattern recognition problems, the prediction of renal function from serum creating, simulation of central bank's decisions and forecast of short term currency rate, etc. IME utilizes an enhanced ECOS branches for online clustering specifically designed for network traffic clustering called ECMm. and uses a new type of fuzzy inference system based on ECOS framework called EFIS to detect network traffic flow anomalies.

The clustering engine module was developed to implement ECMm methodology. ECMm is an enhanced evolving clustering method for the purpose of clustering network traffic data streams. The algorithm itself is basically a combination of ECM algorithm

and its extension ECMc so that it can use the number of cluster created in previous process and optimize its cluster center in online mode. The clustering process will not affect the system performance on detecting anomalies, as once new day started, IME will use the appropriate profiles and it leaves one full day to cluster the previous day traffic data and evolve its profile if necessary. Figure 19 visualize the process of network traffic clustering using ECMm.

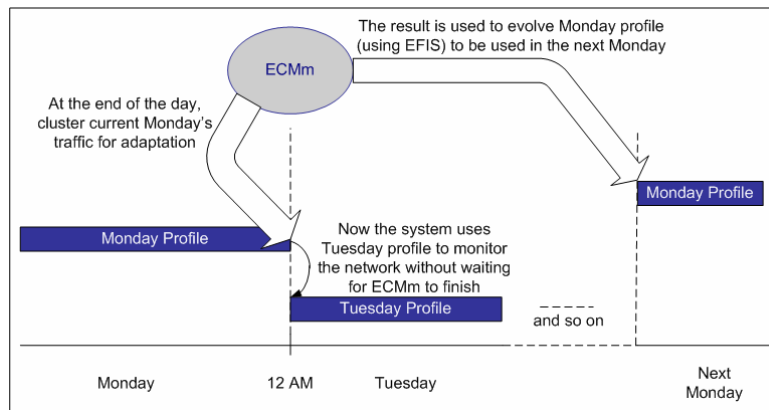


Figure 19.0. ECMm work flow scheme

If traffic anomalies detected, the alarm engine module will trigger an alarm. The way the alarm is presented is in a non-focus mode, hence there will be no interruptions sent to jNetmon main application. A balloon will be popped-up from system tray as a notification of an alarm, and will automatically disappear in 5 seconds. If the alarm table view is opened, the alarm will be automatically presented in the table along with the balloon notification. The alarm engine module is also designed to log every alarmed event into a log file. The format of the log file is specifically designed to be readable for the alarm engine module and also readable if manually opened by the user.

The IME module is purely user-driven application, hence user is required to specify the archive file resulted by the data filtering module as the 'datasets'. User is also able to modify or customize the parameter used by ECMm to cluster the data. The purpose of ECMm GUI version is for informational only. Any archives re-clustered using ECMm GUI will not affect the current profiles and yet it is purely separate from the main IME. But network administrator can modify the profile manually if he/she finds that something is in need of an adjustment after studying the clustering results visually. Figure 20 shows the clustering rendered by the clustering engine.

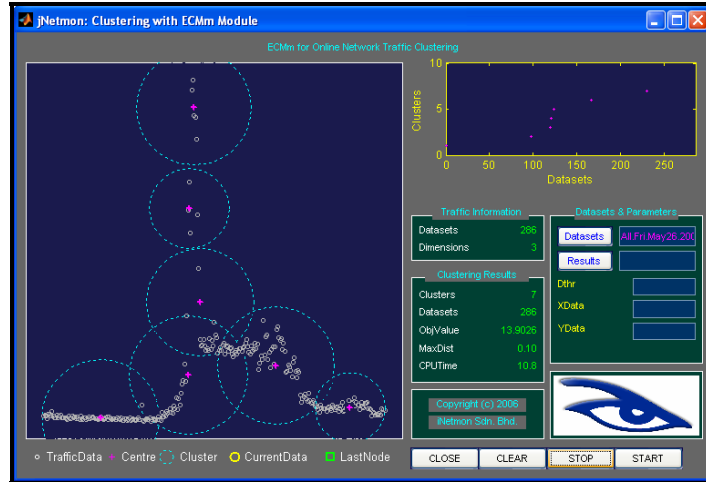


Figure 20.0. ECMm method GUI

The current implementation not only supports filtering HTTP, FTP, ARP, ICMP, and NetBIOS packets, but it is also designed and implemented to be easy to modify and upgrade to enable user defined filtering for the purpose of IME. Table 2 shows the naming convention resulting by filtering to be used by the clustering engine module.

Table 2.0. Filtering file naming convention

Day	Type	Filename
Monday	All	All.Mon.<date>.<year>.txt
Tuesday	HTTP	HTTP.Tue.<date>.<year>.txt
Wednesday	ARP	ARP.Wed.<date>.<year>.txt
Thursday	ICMP	ICMP.Thu.<date>.<year>.txt
Friday	NetBIOS	NETBIOS.Fri.<date>.<year>.txt
Saturday	FTP	FTP.Sat. <date>.<year>.txt
Sunday	<type>	<type>.Sun. <date>.<year>.txt

5.4 DEPLOYMENT

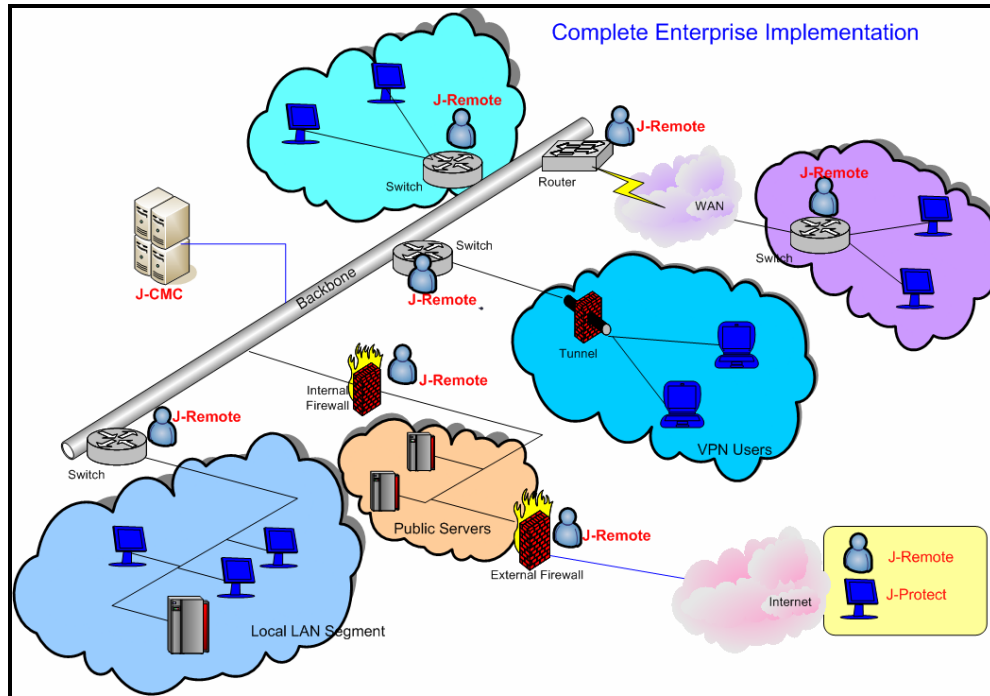


Figure 21.0. jEnterprise Architecture

jRemote can be employed over computer network at the proposed location in the Figure 21. The above diagram tries to cover common network implementation of most companies.

For constant monitoring, it is recommended to have the following elements Figure 22):

- A dedicated monitoring machine installed with jRemote.
- Dedicated hub / mirrored switch for monitoring,

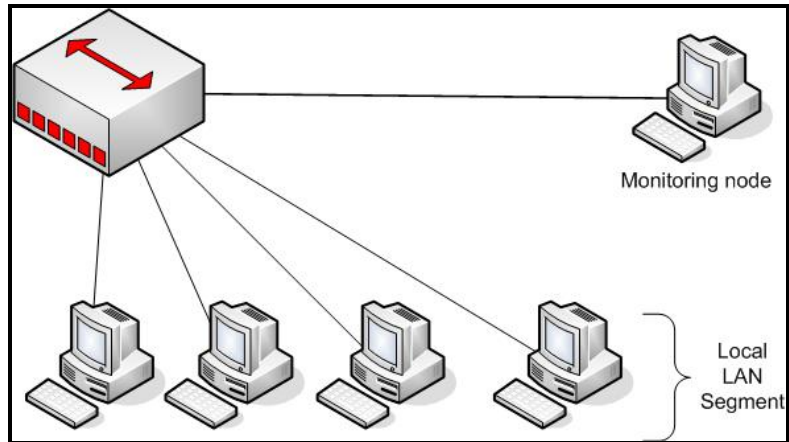


Figure 22.0. Switch Function

Figure 23 shows the correct placement of j-Remote.

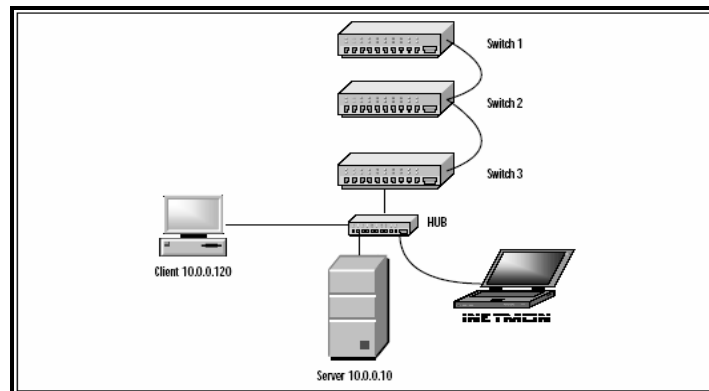


Figure 23.0. jnetmon suite placement

Or it can be used as a technician's tool kit for troubleshooting:

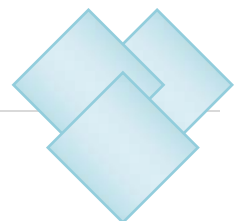
- a laptop with j-Portable
- Some straight-through and cross-over cables
- a mini-hub

6.0 CONCLUSION

The field of network monitoring and performance measurement was explored in the opening sections of this whitepaper, with particular emphasis on the shortfalls of current monitoring techniques within an enterprise or organization. A solution to the problem of a robust network monitoring tool was proposed; using a multi-approach, multi-level packet capture, analyzer and reporting toolkit.

Many current network monitoring tools make use of simplistic, symptomatic methods of reporting faults they discover. The problems associated with these methods were discussed and an alternative was proposed. By using expert systems to perform some of the routine and tedious diagnosis tasks normally associated with tracing network faults, more accurate and useful fault reports can be generated. Particular attention was paid to keeping these reports concise. An approach to intelligently gathering data to produce these reports was also investigated. The wide variety of protocol implementations was presented as an obstacle to the traditional testing of services, and it was suggested that this may be an area where neural networks could be usefully employed.

We have outlined the architecture of a jEnterprise's network monitoring service and architecture, also addressing related security and scalability issues. The basic building block of our architecture is the monitoring and protection(jRemote), visualization (jCMC) and data acquisition and intelligent analysis (jServer). The monitoring activity is performed using passive monitoring, virtually without network overhead.



6.0 ABOUT THE INNOVATOR: INETMON SDN. BHD.

iNetmon Sdn Bhd, a homegrown MSC statused R&D company was founded in 2003 as a direct result of the growing recognition of its network monitoring tool, iNetmon. Working closely together with the R&D arm of the University Science Malaysia, the Network Research Group (NRG) – iNetmon provides the much needed expertise to launch this locally developed product into the demanding IT market.

Since its first release in 1994, innovative research has been the driving force behind the success of this product today. With continuous improvement and expansions, iNetmon is expected to be the most preferred network monitoring tool in the years to come.

It has exceeded the expectations of the industry requirements and has been awarded with several grants from both government as well as private sectors for various projects. Due to these successful ventures, iNetmon has managed to clinch quite a number of accolades to its name too.

iNetmon has strong R&D capabilities, first-rate customer support and exceptional networking products which will be the crucial keypoints that make this company stand out from the crowd. The long terms vision of the company is to be part of the effective global player in the fields of network management.

