



# Choosing an IP Monitor for z/OS

By Tony Amies

---

In this article, we offer techniques to collect and analyze IP monitoring data and address the key questions one should ask when evaluating IP monitoring products.

**IP** now plays a major role in most z/OS installations. This article discusses the techniques available to collect and analyze IP monitoring data and the key points to consider when evaluating IP monitoring products.

IP is mature enough on z/OS that its implementation and subsequent operation is relatively problem-free, but as its use grows and more complex features and applications are enabled, ensuring that everything is working as designed becomes more difficult using only the few tools supplied with IP itself. However, IP is now sufficiently critical to many organizations that sophisticated monitoring tools are required to monitor the IP stack, alert potential problems and aid problem diagnosis.

Even if there are no obvious network faults, an IP monitoring tool can highlight problems waiting to happen, such as resources reaching their operational limit, or problems, which are already happening, but due to the fault tolerance and recovery capabilities of IP are not seen or reported by end users. Such problems could be of a minor nature, such as retransmissions or fragmentation, or more major problems such as router outages or routing errors. The final justification for an IP monitor is for security. It is commonly quoted that many security breaches occur from inside an organization, and firewall blocking external access to z/OS systems is not necessarily sufficient to fully secure the system. An IP monitoring tool can provide audit logs of activity, including failed and rejected connections

and alert when more sinister events occur such as port scans/attacks.

The typical high-level requirements for IP monitoring are easy to define and are not radically different from those for other system components.

## AVAILABILITY MONITORING

Availability monitoring is important to ensure that the resources associated with providing the service are available and operational. The resources typically monitored are the IP stack itself, interfaces such as OSA express, ports, gateways and specific host systems such as key routers or servers. Availability monitoring can often involve more than just checking whether the resource is “up” or “down” but also looking at the activity of the resource. Although a resource may be reported as “up”, unusual activity such as a low connection count, or an abnormally low (or perhaps high) packet rate could indicate problems within the resource itself or perhaps another resource within the network.

## PERFORMANCE MONITORING AND CAPACITY PLANNING

Performance monitoring and capacity planning both tend to involve the same process, the key difference being the sampling interval. Performance monitoring is typically real-time or close to real-time where capacity planning

tends to involve similar input data, but sampled over a much longer period, often weeks or months. The definition of performance for IP is not as straightforward as for SNA. Historically, VTAM buffers, line utilization and end-user response times have been used as performance indicators.

IP networks do not generally use a point-to-point hierarchy, and packets flowing between two network resources do not necessarily follow the same path. TCP is a peer-to-peer, full duplex protocol meaning there are no restrictions on when a connection partner can send or receive data. When either side of a connection can send at any time, the concept of measuring the time delay between a request and a response does not exist in many cases. Application logic, rather than the TCP protocol, tends to drive the ability to measure end-user response times. Response times can only be calculated when the application level protocol forces every request to have a response. This is typical with TN3270, and as this protocol generally involves real end-users, tends to be the one service for which response time can be monitored if the right tools are available. For other services that do not fall into the single request/response structure (and often do not have a human end-user), IP performance involves measurement of throughput for individual network resources or for a complete network path. Analysis of packet rates and connections can also be good indicators of IP performance and are critical for providing

longer-term capacity planning data. You can monitor performance of the IP stack in terms of its resource usage, such as CPU utilization and the allocation of storage including CSM and ECSA. On z/OS, the utilization of critical Unix Systems Services (USS) is equally important. Shortage of processes, userids and memory can lead to degradation of services and system outage.

## SECURITY

The monitoring of network security is generally not the role of an IP monitor, as the functions required are usually provided by purely security-oriented tools such as firewalls and RACF. An IP monitor can offer some additional security benefits such as providing an audit trail of network activity, especially for IP addresses both inside and outside the so-called secure network. Performance monitoring can also have some security benefits such as alerting on unusually high activity typically seen during denial-of-service attacks such as port-scans and PC-based viruses generating high volumes of network data.

## OTHER CONSIDERATIONS

TCP/IP is not a single protocol and does not necessarily have the rigid connection-oriented structure seen in SNA networking. On many z/OS systems, TCP/IP encompasses TCP, UDP, ICMP and OSPF traffic for which only TCP traffic is connection-oriented. Although UDP traditionally sees less use than the more common TCP protocol, UDP usage is now growing, especially for sites using Enterprise Extender. The connection-less oriented nature of UDP, ICMP and OSPF traffic makes monitoring more difficult when compared to the connection-based TCP protocol.

z/OS also provides some unique availability and scalability features such as a Virtual IP addressing (VIPA), Dynamic VIPA, Workload Manager links to DNS (WLM/DNS) and Sysplex Distributor.

Availability monitoring for IP is further complicated by the integration of TCP/IP with other system components such as VTAM to provide TN3270 and Enterprise Extender support.

## IP MONITORING TECHNIQUES

In order to monitor IP on z/OS, tools must be able to gain access to low-level system

data, analyze that data and present the data in an acceptable form.

Despite its maturity, IP does not provide a wide range of tools or facilities to make monitoring easy. Vendors confront having to use many different and often complex techniques to extract what is often simple data. There is no doubt that most of the data required to monitor IP is available, the problem tends to lie with the ability to access the data in a way which does not impact system performance. The monitoring technique must be able to cope with the very high volumes of data, connections and end-users possible on large z/OS systems.

## POLLED VS. EVENT-DRIVEN

Before looking at methods of extracting IP monitoring data, you should consider the concept of event-driven or polled data extraction. Event-driven is where the system automatically supplies data to the monitoring tool. The monitor simply has to wait and listen for data and take the appropriate action when it arrives. On the other hand, when the monitoring tool has to ask the system for data (normally at specific times or intervals), you have polled data.

Event-driven techniques have the advantage of being able to monitor in real-time, enabling presentation and analysis of information as it happens. The main disadvantage is the need to efficiently handle the monitoring data during times of high system activity.

Poll-based techniques cannot truly offer real-time monitoring unless the data is polled very frequently, which can lead to performance overheads. The advantage of polling is easier data sampling and extraction of specific, detailed information on request rather than collected regardless.

## IS REAL TIME MONITORING IMPORTANT?

Real-time monitoring potentially means that problems can be identified and reported much quicker, but this also means that some problems can be detected that would otherwise be missed using polling. With real time, problems can be highlighted quick enough for the system engineers to view the problem as it is occurring, which aids problem resolution. For example, real-time monitoring could highlight TN3270 response time degradation, possibly before end-users start contacting the help desk. Performance problems can often be caused by fragmentation or retransmissions and a true real-time based monitor would be

able to highlight these problems as they occur, allowing the engineer to investigate the problem while it is happening.

A non real-time monitor, typically based on polling data, may only be able to report on the poor response time at the next sampling interval, at which time the cause of the problem may have already gone away and cannot be diagnosed.

## NETSTAT

NetStat (or onetstat under USS) command is perhaps the most common way of extracting IP-related data from a z/OS system. The command can be used with a fairly complex set of parameters to display a wide variety of IP-related information, much of which is very useful to IP monitoring tools.

Despite providing much of the data required for IP monitoring, the NetStat command has several shortfalls. There is no application programming interface (API) for the NetStat command, so any monitoring tool reliant on NetStat commands must invoke the NetStat command and screen scrape the resultant output messages for useful data. There are several possibilities such as using an EMCS console to issue NetStat as an operator command, using REXX to stack the command output or using USS APIs to drive onetstat and process the output. All of these techniques allow the potentially high volumes of data returned by NetStat to be extracted, but suffer from the need to issue the NetStat command very frequently, with the associated overheads, to get even close to real-time monitoring.

NetStat cannot always be used to detect connection-related network problems. The command can be used to show active connections and show connections in a pending state, but will not show connections being rejected due to the server being unavailable or unable to accept new connections. Other than indicating active ports, NetStat provides little information on non-TCP activity, such as UDP, ICMP and OSPF, mainly due to the connection-less nature of these protocols.

## SMF EXITS

Another popular technique employed by IP monitors is to use SMF exits to access IP-related data written to SMF and move this data to another repository for subsequent analysis. This technique does benefit from being event-driven, as the exits are driven by TCP/IP when an event occurs, such as a new connection established or a connection ended. SMF

Record Types 118 and 119 can optionally be written by IP and an exit supplied to trap these records and extract the data. Type 118 records can contain TCP connection start and stop information, specific telnet and FTP-related data and basic statistical information on TCP and IP activity. Type 119 records duplicate much of the data in the 118 records, but can contain additional statistical information on UDP, ports and interfaces. As with NetStat, little information is available for connection-oriented problems and connection-less protocols such as UDP, ICMP and OSPF.

SMF exit-based monitoring forces the collection of SMF records from TCP/IP whether they are required or not with some system overhead. The exit does have the ability to suppress the writing of the physical SMF records, but you should consider the fact that multiple exits for the same record type can exist, perhaps developed in-house or supplied by software vendors.

## SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

Simple Network Management Protocol (SNMP) is widely regarded as the industry standard way to access IP-related data for monitoring purposes. Systems supporting SNMP maintain a large database of information called a Management Information Base (MIB). Monitors wishing to access the data held in the MIB use the UDP-based SNMP protocol to request data held in the MIB. Although SNMP looks like the ideal way to monitor IP on z/OS, some key issues deserve consideration.

By default, TCP/IP on z/OS does not collect SNMP data. This service requires configuration and activation (usually using the OSNMP and SNMPQE address spaces). SNMP on z/OS also uses distributed agents, which can lead to additional overhead. The SNMP component handling the request for data may have to relay that request using the DPI protocol to a subagent for processing.

The SNMP protocol, as its name suggests, is rather primitive. Every data item in a MIB has a unique address, which the monitor must supply in the GET request for that single value. To improve usability, each address can be mapped to a textual name, but due to the vast volumes of data potentially available in MIBs, these names can also become long and complex. Although a GETBULK request does exist, it is still often necessary to issue many GET requests to extract

the data necessary for a specific resource or series of resources. Each GET or GETBULK request requires the application to send a UDP datagram to the SNMP port (usually port 161) on the z/OS system. The response (which may have been generated by a subagent) is returned by SNMP on z/OS to the requestor using another UDP datagram sent by z/OS.

There is no doubt that the SNMP protocol can be used to provide very detailed IP information on z/OS, but this is still a polled protocol (the monitor has to ask SNMP for data), and in order to do even close to real-time monitoring, the SNMP MIBs must be

polled very frequently. Taking into account multiple GET requests are likely to be required each time, and the fact that at least two UDP packets flow through the TCP/IP stack for every request, SNMP can introduce a significant overhead when monitoring busy systems which support a large network.

## PACKET TRACING INTERFACE

A Packet Tracing Interface allows a monitor product to "see" all IP packets flowing through the TCP/IP stack, including TCP, UDP, ICMP and OSPF packets. This interface is very much

# The Network is the Heart of the Enterprise

Monitoring Your Network's Heartbeat...



... Will Protect Your Corporate Health

Let *implex* monitor your network for you

- ♥ *implex* works 'out of the box' with-out configuration
- ♥ *implex* uses a direct interface to the IP stack – no more reliance on NetStat output or SMF exits
- ♥ *implex* automatically monitors all IP network activity on z/OS and OS/390 including TCP, UDP, ICMP, OSPF, EE, OE and X.25 (XOT) traffic
- ♥ *implex* offers true real-time monitoring and historical data views
- ♥ *implex* recognizes and reports changes to the IP network dynamically
- ♥ *implex* minimizes system overhead, maximizes performance and is highly scalable

*implex* – Real-time IP performance, availability & problem management for z/OS and OS/390. The heart of your network.

Download *implex* today from [www.willdata.com](http://www.willdata.com)

William Data Systems LLC.  
99 Canal Center Plaza  
Suite G10, Alexandria  
Virginia 22314, USA

T +1 703-299-0008  
F +1 703-299-9776  
USA toll free: 877-723-0008

[www.willdata.com](http://www.willdata.com)

**WILLIAM**  
DATA SYSTEMS

event driven, invoked every time a packet is passed through IP and therefore lends itself to true real-time monitoring. Analysis of the packet headers, together with the maintenance of statistics based on packet counts and lengths, allows the monitor to build a complete picture of all IP-related activity, in true real-time.

As with all apparently perfect solutions, there are disadvantages. The key current handicap to a Packet Tracing Interface is that there is no published interface standard. For monitors that do have access to the interface, the main challenges are too much available real-time data, the need to handle this data without affecting the performance of TCP/IP, and deciding how much of the data is relevant for monitoring purposes.

## EVALUATING IP MONITORING PRODUCTS

One of the key things to take into consideration is how the product(s) on trial get the raw data they require. Many vendors do not advertise their techniques, but it is very likely that most products will use a combination of NetStat and SNMP, perhaps with SMF exits, and in one case the Packet Trace Interface. Of these, the products are likely to use one technique to provide the bulk of the monitoring features that deserve review.

Of the techniques discussed, only SMF exits and the packet-tracing interface provide event-driven real-time monitoring. NetStat and SNMP-based products can extract and display large volumes of information, but suffer from the need to poll for information to monitor efficiently.

## QUESTIONS TO CONSIDER

### **What are the products dependent on pre/post installation?**

Products that require OSNMP/SNMPQE to be active on S/390 will be using SNMP polling techniques to extract their data. Products that require TCP/IP or system SMF exits to be installed will be extracting limited data from IBM's SMF records (and forces high volumes of SMF data to be collected whether it's required by the customer or not).

### **Do the products display or monitor?**

Look for the ability to set thresholds or view alerts for key IP issues. Look at the range of alerts that can be generated by the product. If errors can only be detected when a user displays a screen, the product is not a monitor.

### **What frequency does the product collect its monitoring data?**

SNMP and/or NetStat based products must have a built-in or user-defined frequency at which they poll for data. If the frequency is in minutes, then the validity and real-time nature of the product should be questioned. If the frequency is in seconds, the resource consumption and network overhead (SNMP) should be questioned.

### **How does the product handle common IP problems, such as connect rejects, fragmentation?**

Connection rejects can occur even if an application still appears to be listening. Therefore NetStat/SNMP-based data collection cannot detect connection rejects; they can only (after a period of time) report that the application is no longer listening. Therefore, a key application could be down, without the NetStat/SNMP based monitor knowing.

### **What does the product do with ICMP?**

Collection of ICMP data can be valuable in IP monitoring. ICMP data can contain indications of performance and availability problems. Real-time monitors can see and react to these error notifications as soon as they occur. Statistical-based systems can report on ICMP counts, but not necessarily react to the errors within.

### **How does the product monitor performance?**

IP performance is a real-time subject. Performance monitoring over longer intervals is capacity planning. To identify potential performance problems, the product must be looking at several key indicators across multiple resources very frequently. These indicators are typically connection counts, packet rates, byte rates and fragmentation and re-transmission rates. A monitor based on polling can only take a "snap shot" of the current counters (which may require a lot of polling in itself), then wait for a fixed period, take another snap shot and compare the difference. This only gives the average performance over the snap shot period and can miss the highs/lows that may be important. As before, the longer the period, the less valuable the information; the shorter the period, the more network overhead (SNMP) would be used, possibly adding to the performance problems the product is trying to monitor.

### **How does the product do response-time monitoring?**

Many monitoring products claim to monitor response times. Typically, this means using a

PING at regular intervals to check the resource being monitored is still there and to measure the round trip time (RTT). This is NOT response-time monitoring; it is simply RTT monitoring. RTT monitoring does have its uses as a basic check on the state of the IP network and whether a resource is available, but it cannot be viewed as a response-time monitor for several reasons.

Application transactions are typically TCP protocols, but PING uses ICMP protocols. Intermediate routers can handle the priority of TCP and ICMP differently. Application transactions are typically a variable number of bytes, often up to 1K or more and usually with a different inbound/outbound byte count. PING sends/receives a fixed number of bytes. PING only measures the RTT between the stack and the resource. True response time is between the application and the resource and can only be measured by analyzing the packet data flowing end-to-end, ideally at the end-user end of the connection, but typically at the z/OS end and collecting the data centrally.

## CONCLUSION

This is only a brief overview of the requirements and techniques used for monitoring IP on z/OS systems. An article of this size and nature cannot cover all the aspects of IP monitoring nor discuss specific products and how each product works internally. I hope that it can provide some insights. 🌐



Tony Amies is a product architect with William Data Systems. He has been working with IP on mainframes for over 10 years, specializing in TCP/IP management, monitoring and integration. Tony has worked in IT since 1978. He can be reached at [t.amies@willdata.com](mailto:t.amies@willdata.com)

©2003 Technical Enterprises, Inc. Reprinted with permission from **Technical Support** magazine. For subscription information go to [www.naspa.com](http://www.naspa.com), email [mbrship@naspa.com](mailto:mbrship@naspa.com) or call 414-768-8000, Ext. 115.

**William Data Systems, LLC**  
99 Canal Center Plaza, Suite G-10  
Alexandria, VA 22314  
(703) 299-0008  
[Toll Free] (877) 299-0008  
[www.willdata.com](http://www.willdata.com)  
[info@willdata.com](mailto:info@willdata.com)