



September 2000

white paper

Network Server Monitoring: Where, What, Why and How

REAL WORLD EXPERIENCE

Certainty Solutions, Inc.
999 Main Street
Redwood City
CA 94063
650.569.4600
650.569.4697 Fax

certainty
solutions.

Introduction

System and network monitoring is an important part of any Web presence, as a tool for reliability, planning, and analysis. By “monitoring” we mean an automated mechanism to test, track, and report on the availability and condition of the systems, services, and networks that make up a Web presence. In this paper, we review the why, where, what, and how of monitoring as it applies to today’s Web services.

Why Monitor?

In the past, businesses could get by with little or no monitoring. Fewer servers, fewer services, local users, local machines, and less need for 24/7 availability meant that fewer things were mission critical and that most failures would be easily noticed. In today’s new Internet economy, system and network monitoring is a critical element in the support of any network presence. There are several reasons why this is the case:

24/7 Availability: Web sites are expected to be available at all hours of the day, and even short periods of unavailability are often considered unacceptable. Proper monitoring can identify (and sometimes correct) problems and potential problems as effectively as possible, even without someone actively on duty.

Proliferation of Servers: In past years, most services were implemented on medium- to large-scale servers, sometimes with a dedicated support staff for each machine. Servers were often multi-purpose machines, used for time-sharing by a community of users, including those responsible for keeping the system running. System administrators would often quickly notice failures because they were also actively using the server. These days, services are often implemented across “arrays” or “farms” of smaller, commodity servers, and those servers are often single purpose servers, used solely for running a single component of a Web site or service. Today, it is far less likely that a system administrator will notice a given failure of a system or service soon after it happens.

Proliferation of Services: The growth of the Web has meant that there are far more externally visible services than ever before. As customers and business partners come to rely on those services, proper monitoring becomes more and more important. The sheer number of different sites and services means that manual monitoring and problem identification are impractical.

Remote Locations: With today’s extensive use of hosting and co-location services, system administrators need to rely on remote monitoring systems to keep track of their systems and

services. It's no longer possible to rely on watching the blinking lights through the display window to make sure the systems are working.

Remote Users: When all the users of a service are in the same location as the system administrators, it's often easy to monitor the availability of a server or service by listening to what the users are saying — many outages have come to the attention of the system administrators when users start asking each other if they are having problems. Remote users make this “word-of-mouth” notification process much less effective.

In addition, a proper monitoring system provides three types of information: exceptions, trends, and history.

Exceptions are those events or situations that indicate a problem or issue needs attention. These include service outages, network inaccessibility, or resource exhaustion (disk, CPU, memory, etc.).

Trends provide information on how usage and activity are changing with time, and are most often used to plan the timing and extent of upgrades and expansions.

History data is used to track and report on outages, failures, and activity levels for such things as service level reporting, problem tracking, and resource or activity charging.

Where to Monitor?

Several choices can be made as to “where” monitoring should take place.

Self-initiated vs. remote probe

Some monitoring software can be run on the computer servers themselves, typically either as a daemon or background process, or invoked periodically by a scheduling service. While these monitors are sometimes easier to customize and potentially have access to more details of a server, they do have some difficulties in practice:

- The need to install, configure, and maintain additional software on each server
- Different implementations and interfaces on different platforms
- Reliance on a machine to diagnose itself; some externally visible problems may not be easily detected from the server itself
- Difficulty of implementation on different platforms and devices (e.g., unusual operating systems or networking hardware)

A more practical approach is to make use of a relatively simple “agent” on each server or device that can answer queries from a remote management/monitoring system. The Simple Network Management Protocol (SNMP) can be enabled or implemented virtually everywhere, and typically provides all or most of the information required by a monitoring system. In typical use, a monitoring system periodically probes each device, retrieving standard and device-specific information for logging and reporting. (SNMP agents and managers can also communicate by means of “traps,” which are typically used to alert the management system of an error, exception, or state change.) The use of SNMP allows for a more uniform and scalable monitoring system than with self-initiated, host-based monitoring.

There are some things you can’t directly monitor with SNMP. A few specialized probes — whether or not SMTP connections accepted or Web pages be retrieved, etc. — are usually sufficient to monitor these items. Other “nonstandard” monitoring can often be implemented using an extensible SNMP agent and adding “private” or customized SNMP data points.

Local collectors vs. centralized servers

Most monitoring is implemented with a single centralized server monitoring systems and devices in one or more local or remote location. This model, however, does not typically scale very well, as it relies on the ability to probe all devices within each time-limited monitoring cycle (e.g., a system might probe each device every five minutes). As networks get larger, round-trip times increase and failures and interruptions become more common, which can lead to unacceptably long monitoring cycle times. Once a certain “critical mass” of monitored devices is reached, it starts to make sense to de-centralize the collection of data and put probe systems closer to the devices being monitored.

For example, an organization monitoring systems at multiple remote data centers from a central NOC (Network Operations Center) would probably be best served by locating a “slave” probe system at each data center, and aggregating the monitoring data on a central server. This configuration makes it possible for the remote probes to dispatch problem reports, even if the central NOC is currently unreachable, and would also provide better accessibility data with reduced bandwidth required for monitoring.

In-house vs. service provider

A proper, high-quality monitoring system is often expensive and difficult to install, configure, and maintain. Unless an organization has a very large network (or is itself a service provider), it usually makes sense to rely on a third party to monitor an organization’s external network services.

Service providers — offering networking, facilities, or operations management — can often provide high-quality and cost-effective monitoring services, as it is usually a vital component of their business. Regardless of who gets paged or dispatched to deal with any problems that may arise, leaving the day-to-day monitoring to the professionals is almost always the best choice.

What Can (and Should) Be Monitored?

The simple answer: Monitor everything. The slightly less simple answer: Monitor everything you think you might ever want to know. The practical answer: Monitor the things you need to know, as well as the things that will probably help you fix the problems that you're most likely to run into.

Exception identification, trend analysis, and historical reporting are the three reasons for monitoring. Some of the key indicators to monitor for each category are as follows:

Exception Identification

Network availability: Typically a "ping" or connectivity test.

Service availability: Web page retrieval, mail service availability, database connectivity, etc.

Server health: System uptime, device availability, and performance (e.g., for RAID arrays), etc.

Network health: Link state changes, routing table updates, etc.

Security related information: Failed logins, repeated operations, etc.

Trend Analysis

Bandwidth utilization: Network link use and headroom

Server disk utilization and capacity

Processor, memory, and I/O utilization on servers

Activity counts: Web page hits, mail messages processed, etc.

Historical Reporting

Utilization and activity: User session records, files retrieved, etc.

Availability: For service level reporting

For each area of interest, the monitoring system (or systems) should be configured to provide an appropriate balance between the information that you need or want to have, and the difficulty and cost of acquiring, recording, and aggregating the information.

How to Monitor?

The “how” of monitoring is largely dictated by the monitoring software package (or packages) in use, but there are certain elements that are likely to be found in most non-trivial monitoring systems.

Polling

Monitoring systems are typically configured to “poll” every device, value, and service every few minutes to ensure availability, to identify errors or exceptions, and to collect data points to be logged. In any but the most trivial situations, it is necessary to have multiple “pollers” running in parallel, both for reasons of monitoring capacity and for the ability to deal with failures in the monitoring system itself. Pollers typically report their data to a central database and dispatch system.

Traps and alerts

Monitoring systems typically provide “trap handlers” to catch and handle asynchronously generated events and traps, outside the normal polling process.

Hierarchy of monitored elements

The configuration and reporting components of sophisticated monitoring systems implement hierarchies and dependencies in the devices and elements being monitored. This is important so that a failure in network connectivity or a single device does not generate a huge flurry of “device unreachable” messages for all the devices and elements behind the failed device, thereby obscuring the real fault.

Aggregation and de-duplication of data

Monitoring systems will often have more than one probe or poller monitoring each device to guard against certain failures in the monitoring system or its connectivity. This can result in more than one copy of each data point being reported to the central monitoring server(s) — any such duplicates need to be eliminated before being recorded or dispatched in the central systems.

As historical data is accumulated in the central database, there needs to be some mechanism for aggregating and summarizing data to retain the trends while lowering the total amount of data kept online.

Notification and reporting

Monitoring systems usually provide a number of standard notification mechanisms for use in the event of a failure or exception. These typically include sending messages to pagers, e-mail, fax, and some form of on-screen display for use in a NOC, along with some API or other interface to allow the use of custom notification mechanisms. For trend and historical reporting, there should be graphical and other interfaces to query and summarize the data.

Summary

Proper system and network monitoring is an essential element of any network- or Web-based service. Implementing and maintaining an effective monitoring system is an often complicated and expensive undertaking, most often best left to the professionals.

And finally, regardless of the quality and effectiveness of the monitoring system that you rely on, remember that without a prompt and professional response to any alerts or exceptions that get generated by the monitoring system, much of the effort and value expended will be for naught.

About Certainty Solutions, Inc.

Certainty Solutions is the leading integrated managed services provider (IMSP) for companies with sophisticated Internet businesses. The company provides customers with both pre-packaged and customized site architecture solutions, standard and a-la-carte managed services offerings, implementation services and hosting/bandwidth solutions.

For more information about this topic, or to speak to a Certainty Solutions representative, please call 650.569.4600 or email info@certaintysolutions.com

www.certaintysolutions.com