White Paper

What's Behind Network Downtime?

Proactive Steps to Reduce Human Error and Improve Availability of Networks

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408.745.2000
1.888 JUNIPER
www.juniper.net

Part Number: 200249-004 May 2008

Table of Contents

Executive Summary

Increasing network availability is an important business issue, with tolerance for network downtime steadily decreasing. According to an earlier Infonetics Research study, large businesses lose an average of 3.6 percent in annual revenue due to network downtime each year. Outages attributable to network equipment generally fall into one of three categories: planned maintenance events, system errors, or human factors. Networking vendors typically focus on the first two, but the last one, human factors, is the biggest contributor—responsible for 50 to 80 percent of network device outages.

Looking for someone to blame is a natural response to outages caused by human error, but does little or nothing to improve network availability. In complex systems like modern computer networks, human error is more a symptom of complexity than a root cause. Juniper Networks is addressing this challenge with the continuous systems approach of JUNOS™ software, and the introduction of automated scripting functionality that can capture expert knowledge, provide early warning of potential problems, and even automate troubleshooting solutions.

This paper provides information supported by recent studies on the potential impact of human factors for enterprise and service provider network managers and decision-makers. This is followed by an overview of the JUNOS software design and features to reduce downtime, including automation scripts, and the opportunity these bring to reduce errors and improve network availability.

Introduction

An operational network is a strategic business resource. It carries everyday messages and mission-critical data, and makes communications possible between people and business processes. To many people in the company it is probably invisible, an assumed utility like water or electricity. And like those utilities, it becomes most visible when it's not there.

What happens when the network is not available? A network outage can have a significant impact on a corporation's image and its customers. Employees can't get access to email, phones or critical business applications. Business processes aren't updated. And customers may look elsewhere for the information they seek, or to place an order. On average, 3.6 percent of annual revenue is lost to downtime in large businesses, according to an earlier study by Infonetics Research.

Of course, improving network availability is just one of the jobs of IT and network organizations. As the network moves to the heart of business strategies, priorities are shifting to the delivery of new services, increasing agility and supporting innovation. Keeping the network running is vital but is expected to consume less of the budget. To successfully manage this contradiction requires a continuous systems approach.

Cause and Effect of Network Downtime

A recent study of technology decision makers shows how important it is to investigate the factors that affect network downtime. The Strategy Group conducted a survey in July, 2007 that included 173 respondents from Ziff Davis Enterprise database. These were all manager level or higher working in organizations with more than 100 employees. This group demonstrated the increasing lack of tolerance for network downtime. Almost 1/3 (32 percent) said that they had zero tolerance, and the average response of the group was just 1.8 hours. This is easy to understand when the average cost of an impaired network was estimated at $3 million per day, with 10 percent giving an estimate of more than $10 million in damages and lost revenue per day.

The negative consequences of a network outage are more than financial. Decline in corporate image was the biggest concern of the respondents (69 percent), with loss of customers close behind (47 percent). Given these potential consequences, it is not surprising that the average IT budget is oriented 70 percent to keeping things running and only 30 percent to strategic and innovative activities. Overall, the group wanted to see this change over the next 12-18 months, moving towards a 60/40 split.

Almost half of the group (46 percent) had a reactive approach to network monitoring and problem solving. It is interesting to note that companies with a more proactive or strategic approach spend a smaller percentage of their budget just keeping things running (60 to 65 percent), compared to those with a reactive or chaotic approach (75 to 80 percent). The lower rate may indicate a virtuous cycle of benefit, as the companies working proactively continue to innovate and improve their IT operations and outperform their reactive competitors.

Operations teams face many challenges to increasing network availability. Network equipment downtime can come from planned maintenance activities, unplanned hardware or software failures, or human error. It is usually a deeper story, and designing systems that can maximize availability requires looking at underlying factors in more detail.

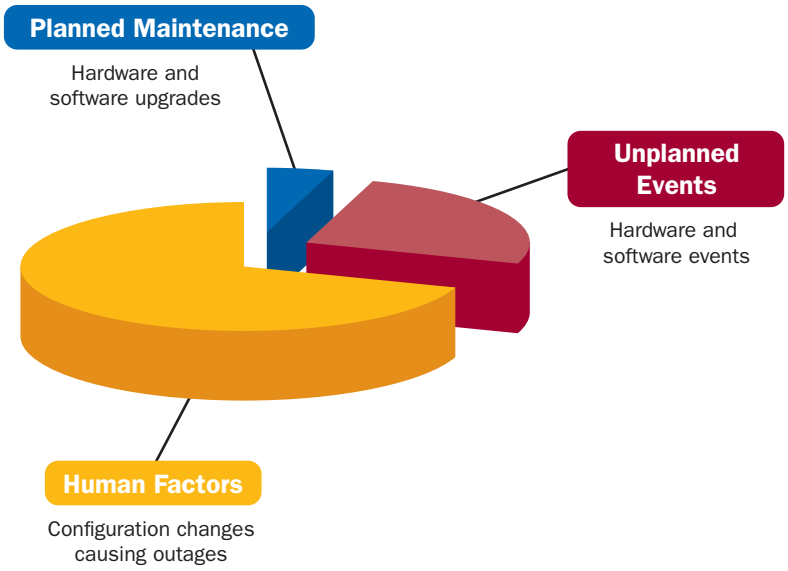

Figure 1: Sources of Network Device Downtime

Maintenance Events

Vendors have devoted a lot of resources to reducing the maintenance windows required for their products. As a result, maintenance events now make up the smallest portion of network device downtime, a reasonable estimate of about 5 to 10 percent. Hot swappable line cards and power supplies are now regular features. Redundancy, fault-tolerant software and non-stop hardware are becoming more widely available. The latest trend is in-service upgrades, which allow software modules to be added or upgraded without impacting what's already running.

This collective focus in the industry has led to a great improvement in network availability. While sustaining performance remains an important focus for networking vendors looking forward, it is not going to provide the biggest improvement to network availability. For example, assuming the 5 to 10 percent approximation, a further 20 percent reduction in maintenance downtime translates to only a 1 to 2 percent improvement in network device downtime.

System Errors

Vendors have also put a significant emphasis on reducing hardware and software errors, with the result that they are now responsible for about 25 percent of network outages, according to many Juniper customers. However, here there are two distinct approaches: reactive and proactive.

Reactive Approach

The reactive approach makes for great marketing. It promises fast response times to critical problems. All necessary resources will be committed to resolving the situation. Software patches will be implemented to resolve problems if necessary.

Unfortunately, this approach quickly leads to complex, fragmented software trains. Users have to be careful to select the correct software version, one that does not abandon an important feature or hardware support. Sometimes previously patched problems can reappear in a different software version. Customers looking for network-wide features have to carefully evaluate the documentation to ensure that a version is available for each different hardware platform. This can involve waiting for up to a year or longer for all of the various versions to be coded, tested and released. Many times these upgrades can then create other problems that result in the need to downgrade, and then upgrade once again when fixes are available. All of these steps to upgrade, downgrade and upgrade again add to network downtime, as the operations team navigates through a constantly churning and complex set of software versions.

System errors may be responsible for about 25 percent of network outages, but the reactive approach from a vendor forces its customers into a reactive mode as well. This requires a much greater resource commitment from the customers, sapping resources from other areas. The more time spent on evaluating software versions and working on patches obviously means less time and resources for innovation. It also means a longer timetable to roll out new network services, and greater risk of human error. For companies experiencing 25 percent of their outages from system errors, a 20 percent improvement translates into a 5 percent reduction in overall network device downtime. But achieving this result may require significant resource investment by customers or substantial delays in new software features and new hardware introductions.

Proactive Approach

The proactive approach sounds simpler, but requires a much greater degree of engineering discipline. Since some problems are inevitably going to occur, this methodology focuses on anticipating and addressing potential problems early. On the customer side, proactive notification of diagnostic indicators can reduce the length of, or even eliminate, some network outages. Earlier notification means that troubleshooting starts sooner, leaving more options open for a quicker remedy.

At the vendor, the proactive approach concentrates resources on a single software version with regular release dates. A sophisticated set of regression test scripts accumulates over time to ensure that previously working features continue to operate as expected. By avoiding software patches and an increasing number of software versions, customers have more time to spend using network features and new platforms, and less time evaluating and testing potential upgrades. Vendors relying on a reactive approach to problem solving can generate lots of positive sounding activity, but wouldn't it be better not to have the problem in the first place?

Human Factors

Depending on the study, human error is blamed for 50 to 80 percent of network outages. But in this era of complex systems, the cause is probably not incompetence. System complexity with multiple components and many types of interactions creates an environment where the relationship between actions and outcomes is not always obvious.

Human error should be seen not as the direct cause of the problem but as a symptom of complexity. This leads us to the conclusion that reducing and managing network complexity will have the biggest potential impact on network downtime. A 20 percent reduction in human error can result in a 10 to 16 percent reduction in overall downtime. That's two to three times the potential from reducing system errors, and 8 to 15 times the potential gains from shortening maintenance events.

Indignation or Experience?

Historical approaches to human error tend to focus on blame and punishment. In this scenario, the focus is on who was at fault and what the consequences should be. The overall expectation is that errors are discrete events caused by incompetence, poor judgment or bad decisions and without humans, the system would be safe. Rooted more in frustration and indignation, this tactic tends to encourage people to hide their mistakes rather than learning from them.

The modern approach is oriented more towards error detection, correction and prevention. In this scenario, the focus is on what happened, how it can be prevented in the future, and (most important) what aspects of the system made the error possible in the first place. The overall expectation is that complex systems are a continuous balancing act, humans are critical to maintaining that balance, and errors are often the result of a cumulative set of actions. With an attitude of continuous improvement, this approach encourages people to identify errors and potential errors, building an experience set for the organization.

Manual entry of complex configuration commands is a common source of human error in networks. It is too easy for even an experienced engineer to put a firewall across the wrong interface (like the one they are using to communicate with the router), mistype an IP address on a filter list, or misconfigure a service with a syntax error or missing argument. Detailed procedure manuals and double-checking can alleviate some of these issues but at the expense of slowing down response times. During an emergency situation, pressure and frequent interruptions can significantly increase the likelihood of an error.

Networking vendors have historically left human error issues to their customers, offering only basic training and knowledge bases to help them manage when something goes wrong. Juniper Networks has maintained a long standing focus on the human factors aspects of operations in its JUNOS software by simplifying and automating key processes that can be prone to human error. Recent innovations now bring the power of customization in a rich set of scripting tools to further address this area of network device downtime.

Continuous Systems and Automated Operations

The engineering foundations of continuous systems are rooted in the earliest design stages and development philosophies; this is not a feature that can be retrofitted. Modular software, open interfaces, independent processes and protected resources are some of the necessary inputs. Juniper Networks began with a commitment to this approach. Now with over nine years of continuous improvement and advancement, they are extending their development innovations to allow customized approaches for reducing the number, severity and duration of network problems.

Continuous Systems

Networking platforms and JUNOS software from Juniper Networks have a well-deserved reputation for continuous performance and operational stability. This reputation is based on a single operating system with a disciplined, single-release development process, and a modular software architecture that completely separates each process.

JUNOS—Disciplined Development

The JUNOS development process is highly disciplined, using a single release train which has been implemented from the very beginning. A new release cannot omit features that were working previously, and must achieve zero critical regression errors. Juniper Networks has not missed a planned release date since the first JUNOS release over nine years ago.

The benefits of this approach are a more stable code base and one implementation of each feature that can not only reduce the number of unplanned system events but also the time and trouble of planned maintenance and upgrades. The JUNOS release testing process is more comprehensive, able to focus all resources and historical experience on a single code base. JUNOS customers can plan network upgrades with confidence, requiring fewer resources. Of course in the event of a problem, all necessary resources are committed until resolution, but the renowned stability of JUNOS software means that this is less likely to happen.

JUNOS software is available as a single release train, with four regularly scheduled releases per year. To upgrade, customers simply choose and qualify a higher release number for all JUNOS-based platforms. Adding new services to the network is simply a matter of enabling the desired feature, as all assigned platform features are available in each release.

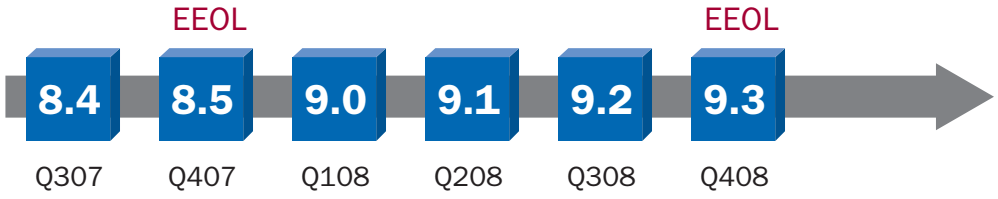

Figure 2: Single Release Train of JUNOS Software

JUNOS—Modular Architecture

The processes of JUNOS software run in separate, protected memory spaces, and process modules can be stopped or restarted without affecting the rest of the system. Memory overflows, a common problem with monolithic software architectures and frequent cause of device crashes, are prevented as one module cannot accidentally overwrite another.

The core networking functions of routing and packet forwarding enjoy additional protections. Each runs with its own dedicated resources, independent of the other. If auxiliary processes should consume too much CPU, they will be throttled back, but the core operations will continue to function. Likewise, in the event of something like a Distributed Denial of Service (DDoS) attack, the control plane and command-line interface (CLI) remain operational and available to troubleshoot and deal with the problem.

A modular architecture and single release train reduce the potential for human errors by protecting basic functions and reducing the number of times that operators need to touch a device. Further improvements to human factors can be accomplished by simplifying and automating operations.

Automated Operations

The complexity and breadth of networking devices means that even organizations with a single vendor strategy still commonly have equipment from a variety of vendors in their network. Even devices from the same company, but from across a broad product line, often have different command-line interfaces and operating systems, due to diverging development philosophies or acquisitions. This situation makes it increasingly difficult to simplify and automate operations across a complex network.

Juniper Networks has assertively promoted and adopted open standards and interfaces on their networking platforms to make it easier to manage and operate these multi-vendor networks. One example of this is the use of XML (eXtensible Markup Language) as an interface to device configuration and state information. This structured document format provides a consistent interface across different devices and makes it much easier to develop applications that can interact with them.

There are several important benefits of XML in this context. XML is an open and well adopted standard that provides the potential to interface to many different types of applications. The format makes it easy to compare configuration files and scripts across diverse devices. Juniper's broad adoption of XML formats and interfaces has enabled further automation with new JUNOScript Automation functionality.

Catching Errors Before They Happen

The increasing complexity of networking systems means that configuration errors can be a significant cause of network device downtime. With networks growing rapidly in bandwidth and scope, the impact of repeated mistakes is even greater.

The best time to eliminate human errors is before they happen. Configuration commands on JUNOS platforms use a two-stage process. Changes are first made to a candidate configuration, not the live one. This enables configuration changes to be aggregated over time and then committed when they are complete. Administrators can review the proposed changes, validate the syntax, and modify or discard any undesirable changes without affecting the running software. Additionally, automated checks within JUNOS software verify the syntax and check for conflicts, informing users of potential issues.

The most frustrating of human errors are ones that have happened before because they are repeating known mistakes that operations teams could ideally prevent. Commit scripts are among recent additions to the JUNOS toolkit for minimizing downtime. They enhance through customization the tools to catch configuration issues before the configuration is active. With commit scripts, the configuration file is parsed by a commit script to check for errors and omissions before being activated. A library of scripts that is developed and maintained by a company's experienced network engineers can ensure that configurations are compliant with business and network policies.

Catching Errors with Commit Scripts

Some examples of potential errors that can be caught by JUNOS commit scripts:

Basic sanity test: ensures that the [edit interfaces] and [edit protocols] hierarchies have not been accidentally deleted.

Consistency check: ensures that every T1 interface configured at the [edit interfaces] hierarchy level is also configured at the [edit protocols rip] hierarchy level.

Interface density: ensures that there are not too many channels configured on a channelized interface.

Link scaling: ensures that SONET/SDH interfaces never have an MTU size less than, for example, 4 kilobytes.

Import policy check: ensures that an IGP does not use an import policy that imports the full routing table.

Cross-protocol checks: ensures that all LDP-enabled interfaces are configured for an interior gateway protocol (IGP), or ensures that all IGP-enabled interfaces are configured for LDP.

In addition to warning messages or rejecting the commit action, scripts can also modify or extend the configuration. A basic set of required variables can be extended to a full, complex configuration, ensuring consistency across multiple devices. All of this functionality uses XML file and command formats, making them open and extensible with customer and third-party applications.

After the scripts are run but before the verified configuration is activated, JUNOS software makes a copy of the running configuration, and keeps it in an archive of 50 previous configurations. JUNOS software also provides an optional confirmation step. When enabled, the router will require a confirm command within a specific timeframe after activating the new configuration. If the router does not receive a confirmation, it automatically reactivates the previous configuration. Administrators can also manually reactivate any of the archived configurations with a rollback command, rapidly restoring a known working state.

Looking for Minor Errors Before They Become Major Ones

One of the characteristics of complex systems is the cascade effect of errors. Small problems can rapidly escalate into major ones. Instead of waiting for an outage that is significant enough to trip alarms and notify network operators, JUNOS operation scripts allow network engineers to automate early warning systems that not only detect emerging problems, but can also take immediate steps to avert further issues and outages and restore normal operations.

Operation scripts use the same software mechanisms as the commit scripts, but are triggered by system log events instead of configuration commits. They can also run periodically, checking status indicators, network connections and other health indicators.

When a script detects a potential problem such as high CPU usage or a dropped virtual private network (VPN) connection, it can take a range of actions—from sending notification messages, to checking other status indicators, to making changes such as shutting down low-priority

processes. It can even make changes to the router configuration, should that be the desired action. Operation scripts can also populate specific MIB variables, allowing them to work in conjunction with SNMP management systems. This enables more granular monitoring of specific devices instead of relying on generic thresholds across the entire system.

JUNOS scripts can encompass a wide range of potential conditions that are driven by different event policies. An if-then-else construct allows them to do more than just a simple action-reaction. Once initiated by a specific condition, scripts can evaluate other status indicators and variables, and provide advance notification to operators or even take appropriate actions. These early warnings give network operators greater latitude to diagnose and address an emerging problem, and they help reduce the duration of network outages.

Maximum Uptime

The flexible scripting of JUNOScript Automation allows customers to improve network operations with customized configuration validation, troubleshooting, and automated responses to specific situations. This enables a continuous improvement capability, as each network outage gets diagnosed and scripted so that it does not occur again or the next iteration has a shorter duration.

How much will these new parts of the JUNOS toolkit affect network downtime? A significant part of the improvement will come over time, as scripting libraries are built and shared within organizations and even perhaps between them. In the meantime, a recent study by Lake Partners Strategy Consultants gives some insight into the potential reduction in network downtime.

Lake Partners interviewed 122 customers about the types of devices deployed in their network, the operating systems in use, and details of their network operations behaviors. They discovered that routers consume a significant percentage of overall network operations time, but that there are substantial differences depending on the operating system.

Let's examine the details of these operational tasks explored in the Lake Partner study. Monitoring network status and device parameters consumed 20 percent of the time, on average. Survey participants reported that they currently spend an average of 25 percent less time monitoring JUNOS network devices than those running other software. Building operations scripts enables further reductions in monitoring activities, as repetitive ones get automated.

According to participants, troubleshooting took up an average of 21 percent of core routing operations. From the survey, the consistency and modularity of the JUNOS software reduced the time spent for troubleshooting and unplanned events by an average of 54 percent. Early-warning scripts and automation of consistent responses to known events (such as a CPU spike or VPN failure) can improve this even further.

Survey participants also reported that they experienced an average of 24 percent fewer unplanned events, and the events were on average 30 percent shorter than for network devices running other software. The commit process for configuration changes and easy rollback to previous configurations are key contributors to this difference. The enhanced functionality of JUNOS scripts enables further improvements to this baseline. Commit scripts that can validate configuration changes against network policies and consistently expand a key set of inputs to complex configuration commands across multiple devices can greatly reduce the number of unplanned events due to human errors.

Table 1: JUNOS Software Impact on Frequency and Duration of Unplanned Events (Lake Partners, 2007)

Network Operations Tasks	Average Reduction with JUNOS Software
Frequency of Unplanned Events	24 percent
Duration of Unplanned Events	30 percent

"The modularity of JUNOS really helps…Juniper allows the system to keep running regardless of what you are doing."

– IT Director, Educational Institution, How Operating Systems Create Network Efficiency, Lake Partners, 2007.

Conclusion

Networks are growing in reach and increasing in speed, multiplying the potential impact of an outage. Since human factors are the leading cause of network device downtime, learning from human error and enhancing the ability to manage network complexity can have the greatest positive effect on network availability.

The modular design and disciplined development of JUNOS software provides the foundation for a continuous systems approach that considers the human factors issues in complex networks. The recent innovation of JUNOScript Automation functionality builds on this base. Enterprises and service providers can capture knowledge from their most experienced personnel and from root cause analysis of previous outages to develop scripts that reduce the likelihood of a recurrence. Early warning scripts can catch minor issues before they become major ones, giving network operators a wider range of choices and more time to respond, effectively reducing the frequency and duration of network outages. These experiences can accumulate over time in a library of scripts, leading to not only automated detection but also automated resolution. Reducing the time spent on monitoring and troubleshooting networks leaves more time for strategic and innovative activities that are essential to organizational competitiveness.

As network experience gets encoded in scripts, the opportunity arises to trade this knowledge, or accumulate libraries of general-purpose and specialized scripts from selected sources. Network operators will no longer be limited to experience from their own organization, but can benefit from lessons learned in the broader community. Networks based on JUNOS platforms will experience a continuous improvement cycle for network uptime, helping teams to deliver the high-performance operations demanded to run high-performance businesses.

About Juniper Networks

Juniper Networks, Inc. is the leader in high-performance networking. Juniper offers a high-performance network infrastructure that creates a responsive and trusted environment for accelerating the deployment of services and applications over a single network. This fuels high-performance businesses. Additional information can be found at www.juniper.net.

CORPORATE HEADQUARTERS
AND SALES HEADQUARTERS FOR
NORTH AND SOUTH AMERICA
Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or 408.745.2000
Fax: 408.745.2100
www.juniper.net

EUROPE, MIDDLE EAST, AFRICA
REGIONAL SALES HEADQUARTERS
Juniper Networks (UK) Limited
Building 1
Aviator Park
Station Road
Addlestone
Surrey, KT15 2PG, U.K.
Phone: 44.(0).1372.385500
Fax: 44.(0).1372.385501

EAST COAST OFFICE
Juniper Networks, Inc.
10 Technology Park Drive
Westford, MA 01886-3146 USA
Phone: 978.589.5800
Fax: 978.589.0800

ASIA PACIFIC REGIONAL SALES HEADQUARTERS
Juniper Networks (Hong Kong) Ltd.
26/F, Cityplaza One
1111 King's Road
Taikoo Shing, Hong Kong
Phone: 852.2332.3636
Fax: 852.2574.7803

To purchase Juniper Networks solutions, please contact your Juniper Networks sales representative at 1-866-298-6428 or authorized reseller.