

Realtime
publishers

"Leading the Conversation"

The Shortcut Guide[™] To



Automating Network Management and Compliance

sponsored by



i n v e n t

Don Jones

Chapter 3: Automating Network Operations and Maximizing Availability.....	44
Business Continuity vs. Disaster Recovery	44
Disaster Recovery Means It Is too Late.....	44
Business Continuity: Continuous Operations Even Through a Disaster	45
Challenges for Continuous Network Operations	45
Lack of Centralized Configuration Repositories	47
Difficulty in Restoring Failed Devices	47
Complexity in Maintaining Accurate Inventory	48
Problems Caused by Direct Device Management	48
Lack of Consistency and Standardization.....	48
Too Much Work: Inefficiency in Network Administration.....	49
Lack of Flexibility to Respond to Business Needs	51
The Business Process for Maximizing Availability.....	51
Creating Standards and Consistency.....	51
Learning from the Information Technology Infrastructure Library Framework	52
Adopting a Change Assessment Process	54
Creating a Complete Change Management Process	55
Using a Process to Improve Network Operations.....	56
Using a Process to Support High Availability	56
Using a Process to Improve Efficiency.....	57
Using a Process to Improve Flexibility and Support New Business Needs	57
Technologies that Support Automation and Maximum Availability.....	57
RADIUS/TACACS+	57
TFTP	58
Telnet/SSH.....	59
SNMP.....	60
Syslog.....	61
All in One: Configuration Management Solutions	61
Enable Disaster Recovery	62
Improve Efficiency	62
Improve Consistency	62
Enable Business Flexibility.....	63

Business Continuity Scenarios and Solutions.....64

 Single-Device Misconfiguration.....64

 Single-Device Failure64

 Multiple-Device Misconfiguration65

 Multiple-Device Failure.....65

 Partial or Total Facility Failure.....66

 En Masse Device Management.....66

 Reconfiguring the Network to Support New Business Needs66

Building a Plan for Automating Network Operations and Maximizing Availability.....67

Checklist: Tools and Technologies You Need.....68

Summary68

Copyright Statement

© 2005 Realtimepublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimepublishers.com, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtimepublishers.com, Inc or its web site sponsors. In no event shall Realtimepublishers.com, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtimepublishers.com and the Realtimepublishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtimepublishers.com, please contact us via e-mail at info@realtimepublishers.com.

[**Editor's Note:** This eBook was downloaded from Content Central. To download other eBooks on this topic, please visit <http://www.realtimepublishers.com/contentcentral/>.]

Chapter 3: Automating Network Operations and Maximizing Availability

The previous chapter discussed how automated network management and operations can help make network security and compliance not only easier but actually practical. A manually operated network is nearly impossible to keep completely secure and compliant. In discussing compliance, the chapter touched on the fact that many rules and laws that may apply to your business actually have an availability requirement as well as security requirements, making business continuity an important part of keeping your network compliant. Of course, there are many business reasons that go beyond compliance for keeping your network up and running: As the backbone for your organization, the network plays an important role in your business' day-to-day operations and profitability. Lose the network, and you lose money. Automation can also play an important role in keeping your network—and your business—operating at all times, and can help minimize downtime if it occurs for any reason.

Business Continuity vs. Disaster Recovery

One of the most important concepts to touch on first is that of *business continuity* versus *disaster recovery*. Many companies have disaster recovery plans—which is good considering that disasters can—and too often do—happen to any business. Having a plan in place to deal with them is essential. But a disaster recovery plan doesn't provide the same level of stability and protection that a business continuity plan offers.

Disaster Recovery Means It Is too Late

The entire point of a disaster recovery plan is—as the name clearly states—to recover from a disaster. Thus, for your plan to be useful, a disaster has to occur, and your network is at least partially unavailable. In other words, in order for a disaster recovery plan to be useful, things have already gone wrong, and you're in a reactionary mode trying to deal with the problem as quickly as possible. Nobody likes to be simply reacting to a problem—once the problem has occurred, you're already losing money. All your disaster recovery plan can do is help reduce the amount of downtime, damage, or financial loss; it can't prevent these things because it's just a *recovery* plan.

Disaster recovery plans can be pretty extensive, and many companies spend a lot of money developing them. For example, companies might plan to deal with a network disaster through some of the following ways:

- Having spare equipment on hand in case one piece of equipment fails
- Having a backup location in case power or other utilities become unavailable at your main location

- Having backups of device configurations available

These are all common disaster recovery provisions, but none of them is useful until a problem has already occurred and you're already losing money. A better practice is to create a plan for continuous business operations *even if a disaster occurs*. This is referred to as *business continuity*, and it's definitely a step up from disaster recovery.

Business Continuity: Continuous Operations Even Through a Disaster

A key element of business continuity is prevention: Preventing disasters from occurring removes the need for disaster recovery, mitigates the damage and financial loss network downtime creates, and reduces the chances that network failures qualify as “disasters.” Certainly, disasters will still occur no matter how much planning and prevention you do. A meteor striking your data center isn't something you can prevent, for example, and plenty of other more prosaic problems can occur. But many disasters can be averted simply through planning.

A very simple illustration of this idea relates to utility power. Some companies might choose to have a backup generator available, perhaps running on diesel or gas, to provide power to critical infrastructure components when utility power fails. When the power goes out, the disaster recovery plan calls for the generator to be started up. Of course, at that point, the power is *already out*, and you're simply reacting to the disaster. A more proactive, business continuity approach, would be to have continuous uninterruptible power supplies (UPSs) that immediately begin providing power when the utility power fails. These UPSs might not have an infinite amount of power available, but they can certainly last long enough for other backup measures—such as the generator—to be called into play. UPSs are such a common component of most business' information technology (IT) infrastructure that most people don't even think about their role in business continuity. However, they help prevent a very specific type of disaster—a power loss taking the network down—from ever occurring. With UPSs in place, you're not simply reacting to a problem: You've helped to prevent the problem from affecting your business, thus ensuring business continuity.

In the network management world, business continuity is often referred to more specifically as *continuous network operations*, meaning the network continues operating no matter what. It is relatively easy to achieve with the right level of automation, but in a manually operated network, continuous network operations can be extremely difficult to achieve, if not outright impossible.

Challenges for Continuous Network Operations

So what stands in the way of continuous network operations? Why don't we all just implement continuous network operations right now? Understanding the hurdles and challenges to a continually operating network is important; by understanding the challenges, you can develop—or find—solutions to them more easily. You should also understand that most of these challenges are inherent to the way networks and network devices are designed and built; they're not a particular shortcoming that your organization may have. Network devices have simply never been designed to be highly manageable, leaving companies to struggle with ways to manage them more effectively. Thus, most companies satisfy themselves with simple disaster recovery

plans. The reason is that creating a continuously available network runs contrary, in many ways, to how network devices are designed.

Simple Devices

Most of the challenges involved in creating a continuously operating network stem from the fact that network devices are essentially simple, primitive pieces of hardware, running relatively simplistic (compared to a computer, that is) software. Certainly, they're powerful, and their simplicity is part of what makes them so reliable—when was the last time your router crashed with a “blue screen of death?” But that simplicity also works against them, particularly in the configuration and management arena.

For example, although network devices are usually built with the capability to defer authentication to an outside server (TACACS+ or RADIUS, usually), they have no similar capability for configuration, management, granular authorization, backup and restore, and so forth. Instead they operate more or less autonomously. Until network device manufacturers create interoperable standards for centralized management, network devices will always present management challenges requiring third-party solutions to make management easier and more efficient.

Lack of Centralized Configuration Repositories

One problem is the lack of any kind of centralized configuration repository. This shortcoming has actually been a long-standing problem in a number of areas of IT. Operating system (OS) vendors, such as Microsoft, have only begun in the past few years to attempt to deal with this problem. Microsoft's Active Directory (AD) is a good example of an attempt at a centralized configuration repository: Configuration settings can be stored and managed centrally within AD, then pushed out to managed client computers. Unfortunately, nothing even remotely like this option exists in the world of network management. Each network device has its own local configuration, and network devices aren't designed to check in with some central authority for changes to that configuration. In fact, network devices have only the most rudimentary means—typically Trivial File Transfer Protocol (TFTP), which we'll explore later in this chapter—to send and receive new configuration files. This lack of centralized configuration and control makes network devices susceptible to inconsistent and incorrect configurations, which are, according to some experts, the leading causes of network downtime.

Difficulty in Restoring Failed Devices

The lack of a centralized configuration repository and the rudimentary means most network devices provide for accessing their configuration files makes restoring a failed device's configuration problematic in some situations and highly manual at the very best. Restore actually refers to two distinct scenarios.

The first is when a device's configuration is found to be incorrect or damaged, and it needs to be rolled back to a known-working configuration. This situation isn't a technically complicated task, but it is often a manual task. Locating the correct backup can be a major undertaking because many organizations aren't as good about maintaining and cataloging configuration backups as they should be.

The second scenario is when a device physically fails, referred to as a *hardware failure*. This situation usually necessitates replacing the device and configuring the replacement to function properly. Again, the lack of central configuration repositories is a sore point here because the correct backup must be located (assuming one is available—regular backups are often overlooked in many environments) and applied to the new device. The new device is often

different than its predecessor and requires a technician to decipher the meaning of the device settings on the failed device and translate them into the configuration for the replacement device.

Complexity in Maintaining Accurate Inventory

The difficulty in maintaining an accurate inventory of network devices again points back to devices' inherent lack of enterprise-level manageability. Without some kind of tools to assist you, it's nearly impossible to maintain accurate inventories of devices on a large network. The odd router will always be overlooked, for example; it's just the nature of working in a large environment in which network devices are intended to pretty much keep to themselves. In addition to the obvious security and compliance concerns of overlooked devices (which the previous chapter addressed), it's impossible to maintain continuous network operations when you don't even know what all of your network components are. If an overlooked component fails, your plans won't have a contingency available. You'll lose some level of network functionality, but won't have anything in place to step in. Any business continuity plans you have in place will be overlooking the device, too, so proactive measures that might have prevented a failure won't be working in your favor.

Problems Caused by Direct Device Management

Another consequence of not having a centralized configuration system for network devices is that the devices must—lacking any other means—be managed directly, which is to say manually. Manual configuration is simply error-prone: It's too easy to mistype something. One famous story dates back to when America Online was first connecting to the then-new (from a commercial viewpoint, at least) Internet. One of their technicians accidentally mistyped a route into a router, and that route propagated, and wound up creating a major service outage for many users. It's difficult to keep a network running continuously when you're managing devices that way. In fact, direct manual management of devices is probably the leading cause of network failure and one of the biggest reasons a typical disaster recovery plan calls for restoring a device to its last-known-good configuration file. Of course, that assumes everyone's been making configuration backups as they should.

Lack of Consistency and Standardization

Consistency and standardization in network device configurations makes them easier to troubleshoot and helps avoid problems altogether—the very point of continuous network operations. By having a consistent, known-good configuration in place on all network devices, those devices are more likely to operate without problems, which is exactly what you want. However, maintaining consistency is practically impossible using only devices' built-in management capabilities.

Even organizations that adopt stringent configuration standards often find that the actual in-use configurations on their network devices don't meet those standards. Certainly, devices may initially be configured using the correct configuration, but they don't often stay that way. Day-to-day operations, along with manual configuration by myriad administrators, combined with a lack of centralized configuration control, typically results in highly divergent configurations. I recently did a study with one client who had more than 5000 network devices in production. We

examined devices of varying ages and logged the number of configuration inconsistencies with relation to their “official” configuration standard. We then charted the inconsistencies and found that the older the device, the greater its divergence from the “standard,” as Figure 3.1 shows.

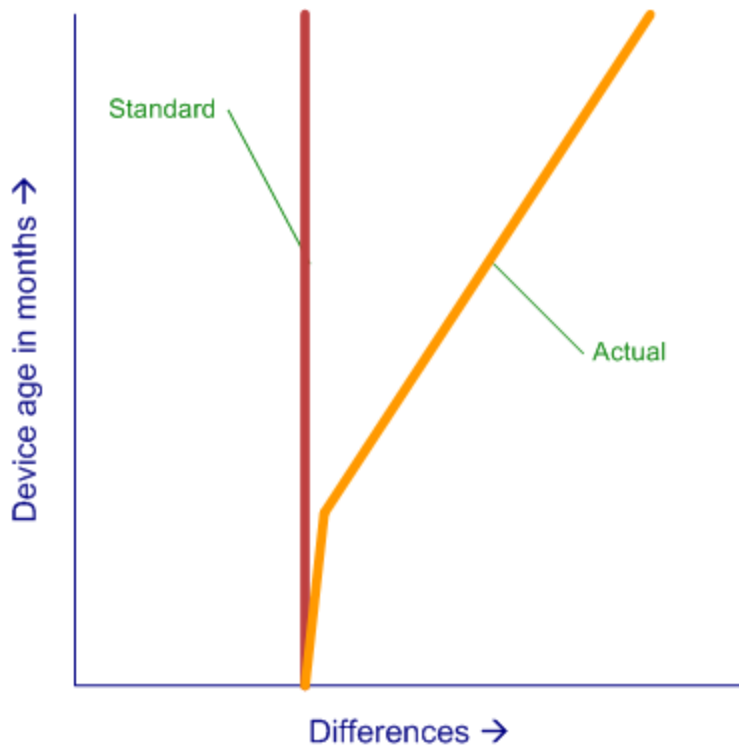



Figure 3.1: Charting configuration deviations over time.

With this many devices “off-standard,” problems were sure to crop up, and they did. We also found that device problems and downtime went up in direct proportion to the number of configuration deviations from the standard. Some problems were minor—they had a lot of difficulty, for example, keeping their Simple Network Management Protocol (SNMP) community strings consistent, which created management headaches but no end-user impact—but other problems did cause downtime.

Too Much Work: Inefficiency in Network Administration

Another major challenge to creating a continuously functional network is that network administration is often just too much work. Without tools, manually reconfiguring devices is a major task. For example, another client I worked with admitted that they hadn’t changed their device administration passwords in more than 4 years. This truth was revealed as part of a white-hat security test. One of the white hat “attackers” found an old memo floating on a file server—completely unsecured, by the way—that listed a device admin password. The attacker didn’t actually use the password for nearly 2 days; the file itself was so old he figured the password *must* have been changed in that time. It hadn’t, simply because, the client said, changing passwords on so many devices—they had about 2500—would take too long. They had once

estimated that the task would require half a month of an administrator's time, which wasn't time they could spare.

 If you're using TACACS+ or RADIUS, you might think you've got your password management handled because you can change passwords centrally. *User* passwords, that is—you still need to manage the TACACS+ or RADIUS keys, used by network devices to communicate with the server, and those can create as much management overhead as changing regular user passwords.

Thus, the attacker was able to prove that he could have completely disabled the network, simply because, in the end, managing the network properly required too much overhead. A network that encounters problems can't be said to be a continuously operating network, and problems that could have been prevented simply through more efficient management techniques are among the easiest to prevent—or would be, if network devices weren't so inefficient when it comes to management.

Lack of Flexibility to Respond to Business Needs

What if you needed to reconfigure 3000 network devices to use a new TACACS+ server for authentication? What would you expect that reconfiguration to take in terms of administrative time? Perhaps 40 hours total? Probably a lot longer: figure the reconfiguration would take an administrator 5 minutes or so and you get about a month of 8-hour workdays for a single administrator. Of course, you would probably script the change, so maybe you could do it in 2 minutes per device, but that's still twelve 8-hour workdays. That is just for *one change*. Given that kind of time requirement, you might just choose—as in my prior example about administrative passwords—not to make the change.

What if the change was more complex? What if you needed to reconfigure 50 devices to provide new connectivity to an important business partner? A more complex change might take 10 minutes to make manually, which would be a full day of work for one poor administrator. That work might easily be put off or interrupted, thus limiting the network's ability to adapt to changing business requirements. That is not a hallmark of a continuously operating network. Although the network isn't *down* because you didn't make the change, it certainly isn't fully meeting the business' requirements. It all comes down to the extreme difficulty in accurately managing network devices without the right tools.

The Business Process for Maximizing Availability

Before you can begin making your network a continuously operating one, you need to create business processes for maximizing network availability. These business processes can safely ignore the relative difficulty of managing network devices; instead, simply focus on having the right standards and ideas in place, assuming that tools for implementing them will follow (and they will, later in this chapter). You simply need to get the business thinking along the lines of maximizing availability—which is to say, keeping things working at all times, not just responding to failures.

Creating Standards and Consistency

Creating configuration standards is the first step toward creating configuration consistency, which improves not only security and compliance efforts but also uptime, the ease of network administration, and much more. Obviously, this guide hasn't yet discussed ways to *enforce* these standards, but there is no point in trying to enforce anything until you have standards in place.

Once you do create standards, they should exist in your production configurations *at all times*. Changes should be made not to devices but rather to the standards, then devices reconfigured to meet the new standard. Standards should reflect not only your business needs but also your

security needs, industry best practices, and so forth. Version your standards—router configuration v1.2, for example—and roll out new standards to all appropriate devices at once.

Learning from the Information Technology Infrastructure Library Framework

The Information Technology Infrastructure Library (ITIL—<http://www.itil.co.uk/>) provides guidance for creating management frameworks (see Figure 3.2). Among these are frameworks for configuration and change management, which are intended to help reduce downtime by creating processes that help to reduce erroneous changes and to ensure proper configuration backups and other safeguards. Figure 3.2 shows a sample ITIL-inspired process, which includes key elements such as change categorization, change review and approval, and built-in provisions for backing up devices prior to applying changes. Of course, simply having this process in place doesn't mean it will be used—process enforcement will be discussed later in this chapter—but you do need to start with the process so that the business is aligned and committed to doing things this way.

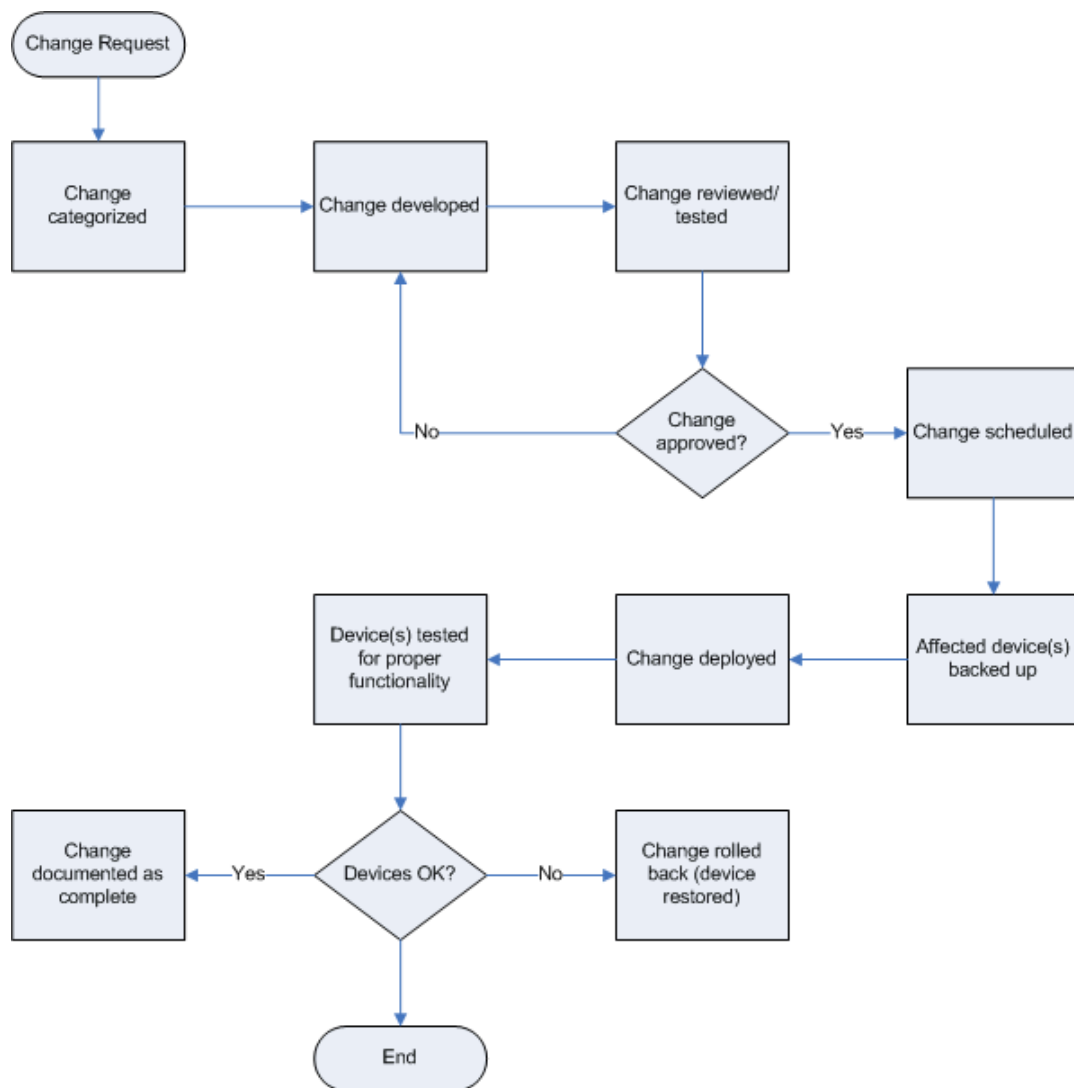


Figure 3.2: Example ITIL-inspired configuration management process for networks.

Adopting a Change Assessment Process

A big part of the ITIL framework is *change assessment*, the part of the process that examines a potential change and ranks it in terms of both priority and risk. Higher-priority changes can thus be given more prompt attention, while riskier changes might receive additional reviews to help minimize any negative impact they might have the potential to create.

As you move to a more automated network management process, having change assessment in place is one of the most important human inputs that you can have into your system. Automation makes it almost frighteningly easy to move changes from development into production; a change assessment process can control what changes *enter* your development process, thereby controlling the pace at which change is introduced to the environment and ensuring that only changes that benefit the business are introduced.

ITIL recommends a Change Advisory Board (CAB), which meets regularly—perhaps monthly—to discuss and assess change requests. Riskier changes are identified and treated accordingly, and changes are grouped into *releases*, batches of changes that will be applied to the organization's configuration standards. For higher-priority changes that come along between CAB meetings, an Executive Action Committee (EAC), consisting of front-line network management and senior administrators, exists to review changes more quickly and get them into the development process—or defer them for the next CAB meeting, if appropriate. Figure 3.3 shows an example change assessment process.

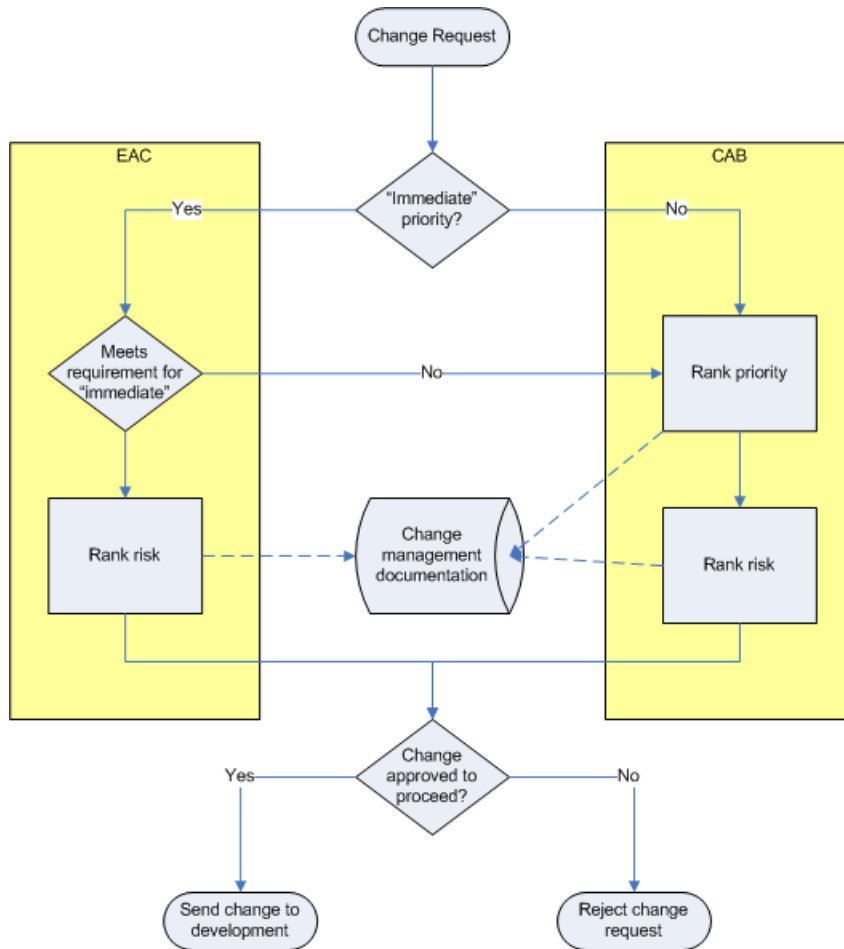


Figure 3.3: Change assessment process.

Creating a Complete Change Management Process

Once changes are assessed and approved for development, your real change and configuration management process begins. As mentioned earlier, ITIL provides suggestions for creating such a process. Typical elements include:

- A formal development process where a change is created.
- A formal review process, where the change receives a technical review. The length and level of detail often depends on the risk level assigned to the change by the initial CAB or EAC review.
- A scheduling process for change deployment. This process often includes conflict-resolution so that changes that impact other pending changes result in some kind of notification or conflict-resolution action.
- Notification to appropriate personnel of pending changes.
- A backup of all affected devices prior to change deployment.
- Testing of deployed changes to ensure they do what they were supposed to do.

- Updated documentation and backups of approved configurations after changes have been deployed and tested.

Such a process helps to ensure that changes present the least possible opportunity for either short- or long-term negative impact. Of course, as stated earlier, a process is just words on paper and can be difficult to enforce; later, this chapter will discuss tools that can help enforce the process. But *the process needs to exist* in order for any kind of enforcement to be effective.

Using a Process to Improve Network Operations

How does a configuration management process support improved network operations? In addition to adding security and compliance controls to network device management, an effective process can also improve availability, make network management more efficient, and even make the network more flexible.

Using a Process to Support High Availability

A configuration management process helps to reduce the likelihood of improper changes being deployed to the production network. Most network downtime is caused by misconfiguration; a configuration management process helps to keep that from happening. In addition, the misconfigurations that *do* occur can be recovered from quickly because your process includes a pre-change backup, giving you an easy “last known good” configuration to roll back to.

Using a Process to Improve Efficiency

An effective configuration management process also goes a long way toward making network administration easier. You'll be managing *standards*, not devices; that helps to consolidate change control and management activities. In addition, by assessing changes and really moving them through a process, you can eliminate a lot of redundancy. Combining the process with the idea of standards-based management generates huge efficiency gains: You'll spend a little more time planning and assessing but a lot less time implementing. Planning a change takes the same amount of time no matter how many devices the change may affect; the task of implementation grows based on device population—thus, by investing in planning, you can eliminate a lot of unnecessary implementation and save time.

Using a Process to Improve Flexibility and Support New Business Needs

Managing to standards through a configuration management process also helps make your network more flexible. Because changing a standard doesn't require thousands of hours of effort, you're free to make whatever changes your business needs, which is exactly the way it should be. Of course, *implementing* those changes is another issue, but we'll touch on automating implementation very shortly in this chapter.

Technologies that Support Automation and Maximum Availability

Unlike some technologies that have significant prerequisites, automating network operations doesn't require your network to have much in place already—and the things it does require are things you most likely are using anyway. The next few sections will briefly describe some of these enabling technologies and help you understand the role they play in network automation.

RADIUS/TACACS+

RADIUS and TACACS+ both play nearly identical roles in the network; TACACS+ is more or less a Cisco-proprietary solution (Cisco adopted the technology years ago and has been the main force behind its development), while RADIUS is more broadly used by other vendors. Both are AAA solutions, which stands for authentication, authorization, and accounting:

- *Authentication* is the process of verifying your identity—typically called *logging in*.
- *Authorization* is the process of determining what permissions you have—once logged in, in other words, what you're allowed to do.
- *Accounting* is a logging or auditing process, which keeps track of what you've done.

These technologies play two important roles in an automated network. First, most network automation solutions are able to utilize the accounting messages sent from network devices to a TACACS+ or RADIUS server as a form of notification. Typically, network devices don't log terribly detailed accounting messages. They'll log, for example, an event when an admin logs in or out but won't usually give you much detail about what the admin did. As Figure 3.4 shows, however, that basic event notification can tell the automation solution to pull the device's

configuration and compare it with the prior version, thus determining what changes the admin made while logged in.

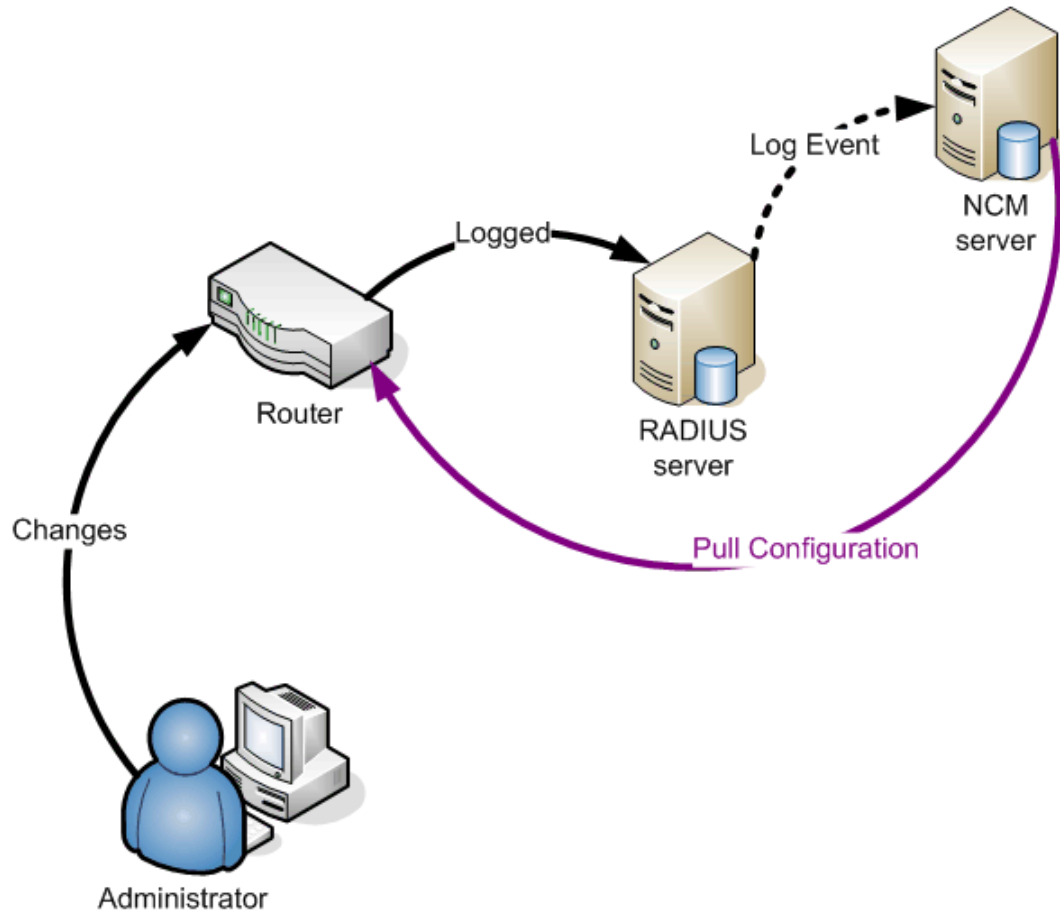


Figure 3.4: Using RADIUS for configuration change detection.

Most network configuration management solutions can also utilize TACACS+ or RADIUS for authentication, allowing you to continue using these central-authentication systems if you're already doing so. However, most solutions will not utilize RADIUS or TACACS+ for authorization, instead relying on their own internal, very granular permissions systems to determine which users have what permissions within the solution.

TFTP

Unlike regular FTP, TFTP does not require authentication and uses the User Datagram Protocol (UDP) rather than the more reliable Transmission Control Protocol (TCP) to send data—hence the name *trivial*, indicating that the protocol isn't intended for serious, large file transfers. It's actually perfect for intranet transmission of network device configuration files, and it's the primary means by which most network devices are able to send and receive entire configuration files. Typically, devices can be instructed to upload (or *dump*) their configuration data to a TFTP server, and can be told to load a new configuration file from a TFTP server.

Network configuration management solutions often act as TFTP servers, allowing them to log into network devices and have configurations sent back and forth between the devices and the solution. This setup allows the solution to act as a centralized configuration repository. Because the solution is a completely automatic one, it can do far more than a normal TFTP server, which would usually just let the configuration files sit on a hard drive. Instead, the solution can load the configuration files into a database, compare them with each other, maintain a version history, retrieve any version instantly, and push versions out to network devices to restore a specific configuration version.

Telnet/SSH

Telnet and Secure Shell (SSH) are the two primary means by which network devices are managed. Although most network devices also feature a console connection of some kind, this connection typically requires a physical, serial port connection from a terminal; Telnet and SSH both permit remote administration and are far more practical for day-to-day use.

Network configuration management solutions utilize Telnet or SSH in a few ways. First, they use it to log into and manage network devices, much as a human administrator would do. They can then run configuration scripts, work with configuration files, and so forth, managing the device automatically. In addition, network configuration management solutions can provide Telnet or SSH pass-through, or proxy, capabilities, as Figure 3.5 illustrates.

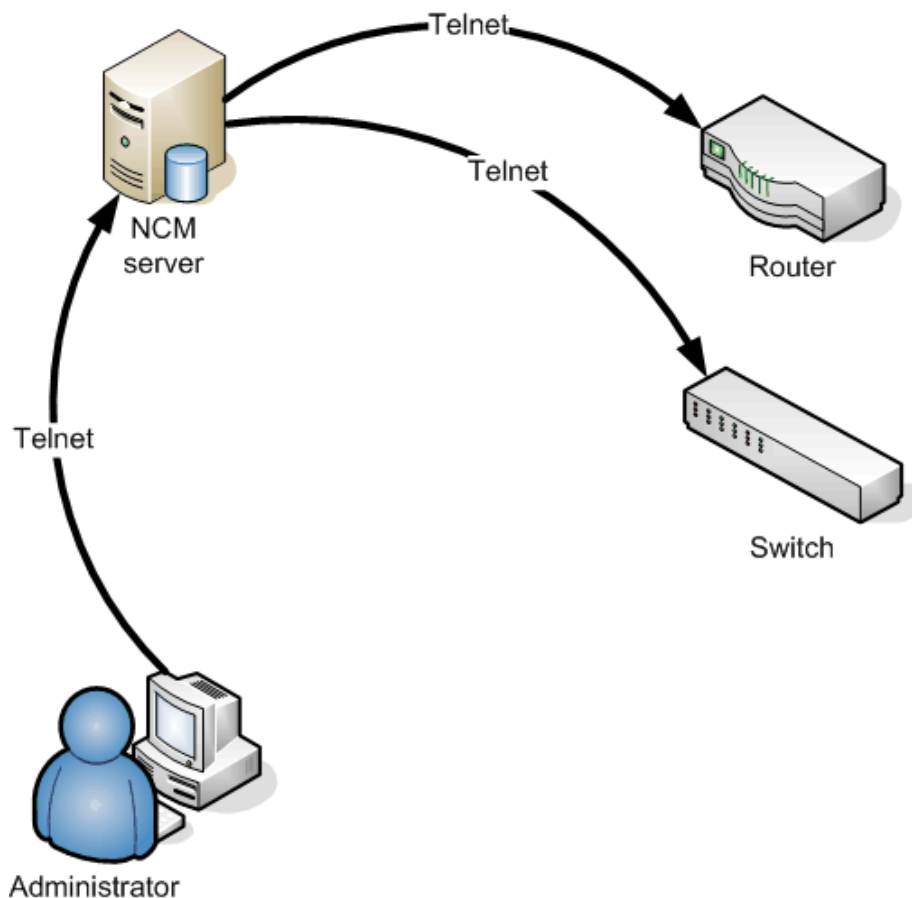


Figure 3.5: Telnet pass-through or proxy.

This proxy capability allows administrators to continue using familiar Telnet or SSH sessions to manage their network devices, allowing administrators to, when necessary, “bypass” the automated configuration management solution. Of course, because the Telnet or SSH traffic is passing *through* the solution, it isn’t really being bypassed. The solution can log the session’s keystrokes, providing a full account of what went on during the session. It also knows to examine the device’s configuration for changes after the session is complete, and it can—if designed to do so—examine changes *as they’re being made* within the Telnet or SSH session. This setup permits the solution to, for example, display alerts for improper configurations or capture the configuration commands for use as an automated configuration script. This latter capability can be especially useful, allowing an administrator to manually configure one device, then have that activity captured and automated to configure additional devices.

🔴 Depending on how the network configuration management solution is designed, it has the potential to become a single point of failure: If it fails, then Telnet or SSH access to devices might become unavailable. Ensure that whatever solution you select has provisions for multiple proxy or pass-through servers so that complete management access to network devices can be available at all times.

SNMP

SNMP is most often used by network configuration management solutions as another form of event notification, not unlike TACACS+ or RADIUS. Figure 3.6 shows how it usually works.

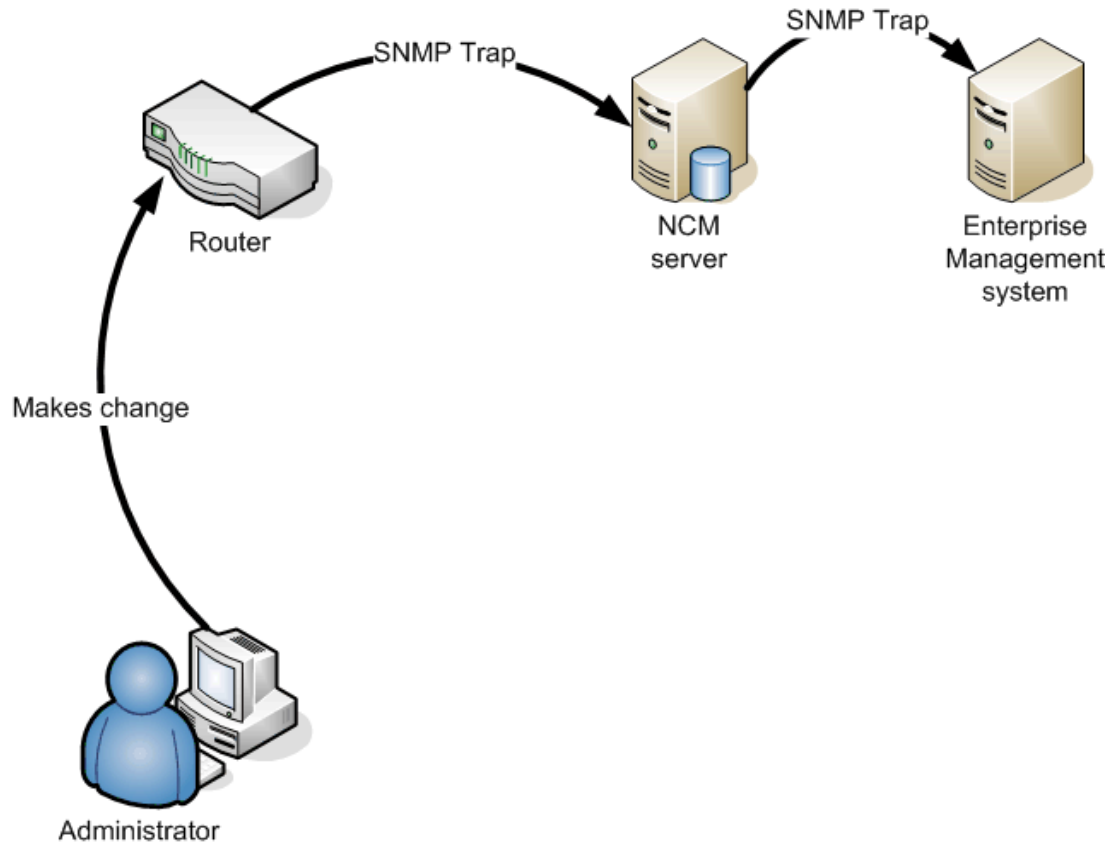


Figure 3.6: SNMP in a network configuration management solution.


For example, devices can be configured to send SNMP traps whenever they are modified or when an administrator logs in. That trap can be received by the network configuration management solution (then forwarded to an enterprise management framework, in many cases), notifying the solution that it is time to pull the device's configuration and look for changes.

Syslog

Syslog is another logging technology similar to the accounting functionality in TACACS+ or RADIUS; network configuration management solutions can use it as another means of determining when a device's configuration may have been changed. Some solutions may look at an external Syslog server, while others may act as a Syslog server themselves, including being able to pass along Syslog entries to an external Syslog server—thus allowing the solution to directly receive the notifications it needs, while continuing to function with whatever Syslog infrastructure you may already have in place.

All in One: Configuration Management Solutions

This chapter has referred to all-in-one network configuration management solutions often. The next few sections briefly touch on some of the major business—rather than technical—capabilities they bring to your environment.

 All-in-one network configuration management solutions will be discussed in more detail in Chapter 4.

Enable Disaster Recovery


A network configuration management solution can help make disaster recovery easier. You should do everything you can to *prevent* disasters through business continuity planning, but disasters *will* occur anyway. Being able to recover quickly is the critical factor when disaster strikes, and by maintaining a fully indexed version history for your devices' configurations and firmware, a network configuration management solution can help you quickly find the configuration you need, then can automatically deploy it to a failed device. This capability is also useful when replacing a device, as the old device's configuration, firmware, and hardware assets can be quickly identified and used to locate and configure the appropriate replacement device.

Improve Efficiency

A network configuration management solution offers an obvious benefit in terms of efficiency—automation. Earlier, this chapter touched on how a fairly complex change might require an administrator to spend 10 minutes reconfiguring a device. Multiply that by 500 devices, and you're looking at a couple of weeks' worth of work. If that same administrator could use a network configuration management solution to deploy the change, he or she might need 20 minutes to set up the change (depending on how the automation solution worked), but that's it—the solution would take it from there. Two weeks to 20 minutes—quite a savings. Start multiplying that time savings by the amount of money that the administrator is paid, and you start seeing some significant dollar values—just for a single configuration change. The numbers add up pretty quickly. In fact, efficiency is one of the easiest reasons by which to justify implementing a network configuration management solution, simply because of the time and money saved.

Improve Consistency

A network configuration management solution can improve consistency in a few ways. First, most have the ability to automatically discover devices on your network, thus preventing overlooked devices from becoming a liability, as discussed in the beginning of this chapter. Second, a network configuration management solution can often be configured with rules and policies regarding your network devices' configurations. By examining device configurations against these rules or policies, the solution can alert you to devices that are incorrectly or inconsistently configured. Some solutions can even have actions associated with the rules so that when a problem is found, the solution can automatically open a new ticket with the Help desk to track the issue or reconfigure the affected device to bring it into compliance.

 There are several ways that a network configuration management solution can implement rules and policies. A preferred method involves a level of abstraction, where the solution actually translates device-specific configuration settings into a generic representation. For example, the solution might simply allow you to specify the public SNMP community string you want, and would know how to check for that setting across a variety of devices from different manufacturers. Similarly, it would know how to *change* that setting across manufacturers.

The other major way of creating rules and policies is to allow you to specify actual configuration settings. This functionality is also powerful and flexible, but means you'll have to have a different set of rules for different classes of devices, or devices from different manufacturers. For example, for the same SNMP setting, you might have to have a set of rules for your Cisco routers, one for your firewalls, another set for your Nortel switches, and so forth, because each set of devices would represent that SNMP setting in slightly different ways.

When available, a solution that offers configuration abstraction requires less work on your part to set up and maintain rules and policies.

Enable Business Flexibility

A network configuration management solution that offers policy- or rule-based configuration can make your network much more flexible. For example, if your business needs change, you simply change your configuration rules or policies and allow the solution to reconfigure your devices appropriately. Even without setting up rules or policies, however, a network configuration management solution can improve flexibility simply through automation—by making it easier to make changes across hundreds or thousands of devices. You'll never again have to consider “how long will this take us?” in making a configuration decision; instead, you can focus on “what business benefit does this change offer?” which is definitely the right way to be thinking about when assessing and planning change.

Business Continuity Scenarios and Solutions

The next several sections briefly touch on several potential failure scenarios that could occur in network device management. For each, the section will provide not only a potential recovery solution using a network configuration management solution but also the ways in which the network configuration management solution can prevent, or help prevent, the problem from occurring in the first place.

Single-Device Misconfiguration

Scenario: A single network device is manually modified by an administrator using incorrect or improper configuration settings. As a result, the device either fails to operate as desired or is left in a vulnerable condition and can be compromised.

Recovery: A network configuration management solution can provide for rapid recovery by restoring the last-known-good configuration to the device. A network configuration management solution might even be configured to take this step automatically when the device was changed without authorization; not *all* changes will cause an immediate and noticeable failure, but that doesn't mean the change can be allowed to sit in production.

Prevention: A network configuration management solution can prevent this type of problem by enforcing a workflow process that requires peer approval prior to deployment, thus helping to ensure that improper configurations don't make it into production. The solution might be used to block manual changes (for example, only allowing the solution to know the devices' administrator passwords), or might be configured to automatically roll back any device configuration changes that are made outside the solution's workflow.

Single-Device Failure

Scenario: A single network device experiences a hardware failure and needs to be replaced.

Recovery: A network configuration management solution can restore the failed device's last-known-good configuration to a replacement device, thus ensuring that the replacement is able to quickly take over operations for the failed device. By automatically backing up device configurations on a regular basis, the solution can help ensure that the latest configuration is available to be loaded into the replacement device. An effective solution can also tell you exactly what hardware the failed device used—memory configuration, add-in modules, and so forth—so that a suitable replacement can be provisioned more quickly. The solution can also let you know the exact firmware level the failed device was using so that the replacement can be configured identically.

Prevention: This failure isn't a situation you can typically anticipate and prevent.

Multiple-Device Misconfiguration

Scenario: Multiple network devices are modified with an improper or incorrect configuration. Although the new configuration might not cause immediately undesirable operation, it might easily pose a security or compliance risk.

Recovery: A network configuration management solution can restore multiple devices' configurations to a prior version pretty much as easily as it can restore the configuration of one device. By keeping a repository of device configurations available, the solution can ensure that the latest or most correct configuration is always available to be sent out to devices.

Prevention: A network configuration management solution can prevent this type of problem by enforcing a workflow process requiring peer approval prior to deployment, thus helping to ensure that improper configurations don't make it into production. The solution might be used to block manual changes (for example, only allowing the solution to know the devices' administrator passwords), or might be configured to automatically roll back any device configuration changes that are made outside the solution's workflow.

👉 Blocking the ability to make manual changes may seem scary or risky, but it doesn't need to be. For example, with a network configuration management solution in place, you can set up the configuration so that device administrator passwords are known only to the solution itself, although you would obviously keep a written copy of the passwords, perhaps in a locked safe. Without the ability to log in and bypass the solution, the solution becomes the only practical way to modify device configurations, thus ensuring that its workflow, rule-checking, and other abilities are always used to enforce configuration standards and prevent erroneous changes. Network configuration management solutions that offer this type of locked-down environment must have robust, native failover capabilities to prevent the loss of access to network devices.

Multiple-Device Failure

Scenario: Multiple network devices experience a hardware failure and need to be replaced.

Recovery: A network configuration management solution can restore the failed devices' last-known-good configuration to replacement devices, thus ensuring that the replacements are able to quickly take over operations for the failed devices. Because a network configuration management solution is typically capable of automatic device discovery, you don't need to worry about a device having been overlooked and not having a configuration backup available. As with a single device failure, the ability to track hardware and firmware information can also be invaluable in properly provisioning a replacement device.

Prevention: This failure isn't a situation you can typically anticipate and prevent.

Partial or Total Facility Failure

Scenario: A portion, or all of, your facility becomes unavailable due to an environmental disaster, utility failure, or other means outside your control.

Recovery: A network configuration management solution can be used to quickly reconfigure surviving devices, thus reconfiguring the network, if possible, to work around whatever failure occurred. In the event of a total facility failure, a network configuration management solution can be used to restore configuration settings to devices at a backup facility or to reconfigure devices at other surviving facilities to accommodate the failures that have occurred. By deploying these changes quickly, the solution helps to minimize downtime.

Prevention: There is little you can do to prevent total facility failure, but providing redundancies in power and other utilities can help mitigate the effects of a facility-related disaster.

En Masse Device Management

Scenario: Multiple devices need to be reconfigured, perhaps to deploy a new configuration standard.

Solution: A network configuration management solution can typically create configuration scripts simply by “watching” an administrator manually configure one model device. That configuration script can then be automatically deployed to whatever devices are required. An effective network configuration management solution will come with scripts for common device management tasks, such as changing device passwords, providing built-in efficiencies for these common tasks.

Reconfiguring the Network to Support New Business Needs

Scenario: Multiple devices need to be reconfigured more drastically, involving multiple configuration changes, in order to support new business requirements.

Solution: A network configuration management solution can easily script the changes, no matter how complex, and deploy them to multiple devices. The solution can track which changes are successfully deployed and which are not, and an effective solution can provide detailed information on failures as well as guidance for solving the problem. This feature ensures that the network is reconfigured as desired. In the event that some devices fail, the solution can be used to automatically roll back other devices, thus ensuring that all the network devices are consistently configured. A good solution also provides deployment scheduling so that all the devices can be reconfigured at a fairly precise time to correspond with other infrastructure changes that may be required.

Building a Plan for Automating Network Operations and Maximizing Availability

So how can you start preparing your business for automated network operations and maximum availability? Start with your processes. Create processes, and do your best to get everyone to live and work by them. Find out what does and does not—process-wise—work. Get the feedback from everyone involved in the process to see what they like and don't like, and tweak the processes as a result. Before you do *anything else*, you need to have processes in place that you're comfortable with.

As you're creating your processes, note the places where they're bypassed. Try to find the reasons behind the bypasses and evolve the process to accommodate those reasons. The more people *want* to work with the processes you create, the better the processes will function for your business.

Next, begin assessing your network devices, creating as complete of an inventory as you can. A key evaluation criterion for selecting a solution is in making sure it supports all your devices. In addition, get some of the key technologies—TACACS+, RADIUS, Syslog, and so forth—in place if you don't have them already.

Finally, create configuration standards. Decide what your devices' configurations *should* look like, even if they don't right now and even if it's not feasible right now to make them look that way. Your standards should include considerations such as regular password and SNMP community string changes. Also, include standards for best practices, such as timeframes for patch deployments and other device management issues.

All of these steps give you the background that you need to put an automation solution to good use: You'll have standards that can be implemented as the solution's configuration policies and rules and a process that utilizes automated workflow and scheduling. In addition, you'll have worked out and fine-tuned the process so that the solution is *supporting* the process rather than *imposing* it on your business.

Checklist: Tools and Technologies You Need

There are several factors that your network needs to order to become more efficient and more automated:

- Appropriate logging capabilities, such as TACACS+, RADIUS, and Syslog, many of which support automation solutions' own capabilities.
- Tools that *assume things won't work*, which are often referred to as *failure resolution tools*. For example, having a solution that pushes out a new TACACS+ configuration to all your network devices is only useful if it can also identify the devices that weren't able to be properly reconfigured—allowing you to take additional steps to handle those exceptions. Ideally, failed deployments should be grouped for automatic analysis or retries.
- In keeping with the theme of assuming things won't work, tools that can have built-in intelligence for dealing with failure. For example, a change deployment script might have logic built into it that allows it to try alternative means of contacting a device, should a failure occur, helping to keep the administration as automated as possible.
- Tools that support your business processes, particularly in workflow automation. These tools should, at a minimum, include technical and managerial review capabilities, requiring signoffs before changes can be scheduled. The tool's permissions should be granular enough to allow different individuals to develop and approve changes, or to allow the same group to have both development and approval—while preventing anyone from approving changes they created themselves.
- Tools that offer scheduling conflict resolution. For example, if a change in the queue modifies SNMP configurations, and another change modifying those same settings is added earlier in the queue, the author of the original change should be notified, as his or her change may be affected by the new addition to the queue.

Some of these criteria are pretty specific and will be visited in much more detail in the next chapter, where we'll look at very real-world and very specific evaluation criteria for selecting an automation solution.

Summary

This chapter explained how network operations—the day-to-day management of the network—can be improved through the use of a carefully designed set of processes and supporting tools and technologies. Automation is the key behind many of these improvements, and the next and final chapter of this guide will look at how your organization can assemble a complete plan for automating your network operations. It will look in more detail at supporting business processes that you can develop and adapt, and help you develop evaluation criteria for network automation solutions that can help make those processes an enforceable reality. In addition, the next chapter will explore how automation affects other business plans and processes and will show you how to create plans for day-to-day administration, auditing, disasters, and more.

Content Central

[Content Central](#) is your complete source for IT learning. Whether you need the most current information for managing your Windows enterprise, implementing security measures on your network, learning about new development tools for Windows and Linux, or deploying new enterprise software solutions, [Content Central](#) offers the latest instruction on the topics that are most important to the IT professional. Browse our extensive collection of eBooks and video guides and start building your own personal IT library today!

Download Additional eBooks!

If you found this eBook to be informative, then please visit Content Central and download other eBooks on this topic. If you are not already a registered user of Content Central, please take a moment to register in order to gain free access to other great IT eBooks and video guides. Please visit: <http://www.realtimepublishers.com/contentcentral/>.