



realtimepublishers.com®

*The Definitive Guide™ To*

# Enterprise Network Configuration and Change Management

VOYENCE™

*Don Jones*

---

Chapter 3: Network Change Management and Security.....	45
The Impact of Change on Security .....	45
Real-Time Monitoring of Changes .....	46
Developing a Change Auditing Plan.....	50
Auditing Planned Changes.....	50
Periodic Audits.....	52
Providing the Means for Auditing .....	54
Security Best Practices.....	54
Role-Based Authorization for Changes .....	55
Regular Password Changes.....	57
Templates for Consistency and Compliance.....	57
Proactive Security .....	59
Developing an Incident Response Process .....	60
Supporting Corporate Governance Requirements .....	61
HIPAA .....	61
The Gramm-Leach-Bliley Act .....	63
21 CFR Part 11.....	63
Summary .....	64

## Copyright Statement

© 2004 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.

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](mailto:info@realtimepublishers.com).

## Chapter 3: Network Change Management and Security

How can change management—essentially a set of processes and procedures—improve security in your environment? Managed devices such as routers, switches, and firewalls play an obvious role in the overall security of your network. An improperly configured device can, for example, allow unwanted traffic into the private network from the Internet, transmit internal traffic to the Internet, and unexpectedly drop traffic and impact productivity. Therefore, controlling the configuration of each device on your network is critical to maintaining a high level of security. Change management provides that control.

### The Impact of Change on Security

Many organizations think of security as protecting information. If someone steals information from your network, you've had a security breach. To combat such attacks, there are many tools available, including encryption, permissions, and passwords. However, some of the most common attacks aren't attempts to steal information; their goal is to simply disrupt information access, making information unavailable to *anyone*. To truly protect the network, an organization should approach security by eliminating ways in which an outsider (or insider) could affect your environment, including stealing data, accessing data, and making parts of the network unavailable.

Well-meaning experts often claim that a router configuration change cannot result in the loss or theft of information. However, it is possible for an attacker to modify a routing table so that data is routed off the network or to a specific location *on* the network where the attacker had installed a capture utility. Certainly, such a change might be quickly detected as a result of its impact on the network—but such a scenario is within the realm of possibility.

There are additional device configuration changes that would permit data theft, particularly configuration changes on firewalls. By modifying a firewall to allow a couple of extra ports, an attacker would potentially be provided private access to the network and an opportunity to attack permission structures, user accounts, and more.


Recent well-publicized attacks focused entirely on making systems unavailable:

- The Blaster worm that attacked Windows-based computers was relatively harmless but could easily have destroyed the entire system.
- The Slammer worm that attacked Microsoft SQL Server computers made systems unavailable.
- A series of Cisco IOS bugs that, if exploited, could crash routers or force them to reload simply by sending packets containing certain data. These attacks would lead to the device being unavailable.
- The December 2003 Cisco PIX vulnerability (which also affects the firewall software on Catalyst 6500 and 7600 series switches), which causes the firewall to crash and reload when processing certain Simple Network Management Protocol (SNMP) messages and Virtual Private Network Concentrator (VPNC) traffic.

In fact, a quick review of the recent security patches issued by Cisco, Microsoft, Nortel, and other manufacturers reveals that the overwhelming majority of attacks are aimed at making devices or systems unavailable rather than attempting to obtain access to data. Many of these attacks rely on specific software configurations within a device.

For example, the Cisco PIX vulnerability affects PIX firewalls configured to receive and process SNMP messages. Although PIX firewalls don't support SNMPv3, they will receive and attempt to process the messages; this particular software bug will cause the firewall to crash. (PIX devices configured only to generate traps aren't affected.) Cisco quickly provided a patch for the problem, but this particular exploit is a good example of how simply changing a device's configuration can make it more vulnerable to attack and underscores the need to maintain tight control over changes made to network devices.

The message is simple: Unauthorized changes to network devices can result in data loss or theft. The only way to prevent such changes is to be aware of all device configuration changes. An administrator needs to be notified each and every time the slightest configuration change is made to a device. In addition, you need a response plan in place to deal with those changes when they occur.

 Is it realistic to believe someone could harm a device in these ways? Absolutely:

- A junior administrator, not knowing any better, might enable a router to receive SNMP messages, creating a vulnerability that an attacker could exploit.
- Someone might inadvertently change a device's SNMP community string to "public," making the device an easy target.
- An IT staffer might accidentally reset a device's passwords to something simplistic, leaving it an easy target.

Seemingly harmless changes, which independently create no immediate problem, can still pose a significant security risk. Thus, it is crucial to keep on top of change in your environment.

## Real-Time Monitoring of Changes

The first generation of software tools designed to aid in change management provided notification to an administrator when a device's configuration changed. Essentially, the software would poll devices' configurations—generally using Trivial File Transfer Protocol (TFTP) or other techniques—on a regular basis, perhaps once per evening. The software would compare the current configuration to the last "authorized" configuration, and compile a report of any differences. This report would be emailed to an administrator for manual review. Administrators could then review the reports to determine whether the changes were authorized. Most change-management applications would allow the administrator to reload the device with an older configuration if the changes weren't authorized.

This technique certainly works and provides decent control of your environment. But periodic polling of devices' configurations isn't adequate from a security standpoint. After all, if an unauthorized change is made on Tuesday morning, discovered by the software on Tuesday night, and reviewed by an administrator sometime Wednesday afternoon (after the morning crises are dealt with), an attacker has a pretty large window of time in which the unauthorized change can be exploited. A more real-time means of monitoring and detecting changes is necessary.

### Configuration Data Theft


One long-standing concern about the security of network devices is the fact that almost all configuration data is passed to devices in clear text. SNMP messages, for example, contain crucial information—such as community strings—that, if captured by an attacker with access to the physical network, can be used to easily modify device configurations. Entire device configuration files are often transferred in clear text via TFTP, providing eavesdroppers with detailed information about the network’s configuration, layout, and other information.

With the growing popularity of wireless networking and with the poor security provided by Wi-Fi’s Wired Equivalency Protocol (WEP), configuration data becomes easier to capture without access to the physical network. If configuration data is modified from a wireless computer, an eavesdropper within transmission distance can easily break the WEP encryption and have complete access to the configuration data—which might include SNMP community strings or device configuration passwords.

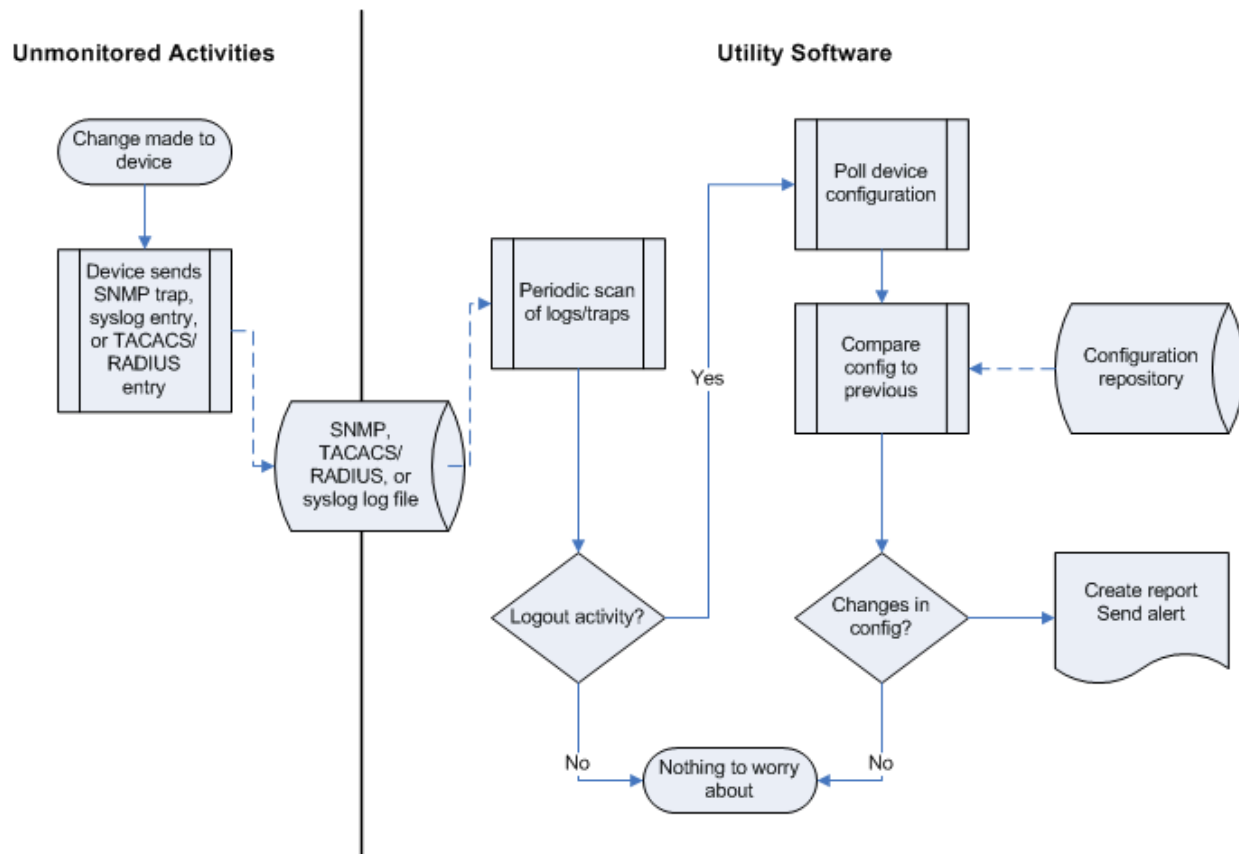
It’s nearly impossible to construct a completely secure configuration environment. Although wireless connections, and even wired ones, can be secured with more advanced encryption (such as Wi-Fi Protected Access—WPA) or 802.11x port-based security, network devices themselves simply don’t support much in the way of security for their configuration data. Configuration is performed over unencrypted Telnet sessions, unencrypted TFTP, and other inherently non-secure means.

Therefore, it is almost impossible to prevent a determined attacker from at least reading configuration data, through some means, and using it to construct a useful picture of the network or even modifying device configurations. However, by monitoring—in real-time—changes to devices, you will quickly be aware if someone makes unauthorized changes. Certainly, such reactive security is less desirable than more proactive techniques, but today’s network devices offer only limited capabilities for better, more proactive security. Ensuring that your environment contains suitable reactive measures is essential to maintaining a reliable, secure environment.

Unfortunately, network devices aren’t set up to perform real-time notifications. Routers, firewalls, switches, and other devices operate more or less in a vacuum, accepting configuration changes from anyone with the proper password. The closest they come to providing real-time notifications is syslog entries, TACACS or RADIUS accounting messages, or SNMP traps generated when certain events occur. Although such notifications are not particularly detailed—there is no log entry, for example, that says “Joe just changed the routing table”—these logs can create a trigger for more sophisticated, third-party software.

 I’ll provide more information about syslog, TACACS, RADIUS, and SNMP in Chapter 5. These are all important underlying technologies that are utilized in an enterprise-class change-management process.

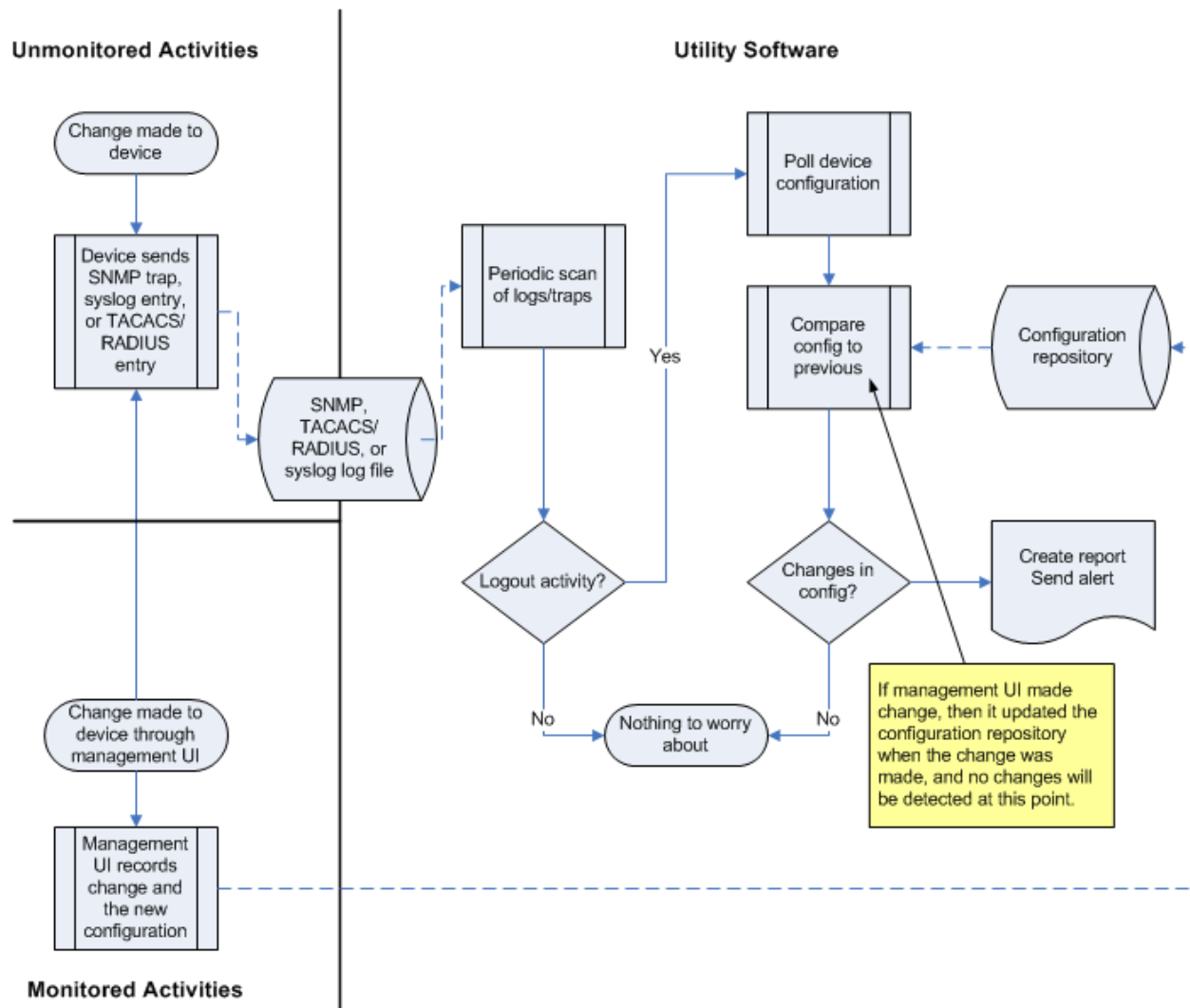
By monitoring syslog files, TACACS or RADIUS accounting logs, or SNMP traps, third-party software can detect basic events, such as an administrator logging on to or off of a device. Logging on or off is a clue that a change *might* have occurred—there is no way to change a device configuration without logging on to its administrative interface in some fashion (this includes SNMP configuration messages, which go through a sort of authentication process). An administrator might simply log on to view the current configuration, rather than to modify it, but the simple act of logging on tells a third-party utility that more scrutiny is required. Figure 3.1 shows a third-party application’s response to a logon event.



**Figure 3.1: Process for real-time device configuration monitoring.**

When the utility detects a logon or other pertinent event in the log file, it then polls the device's configuration and compares the current configuration with the most recent authorized configuration, which is stored in a separate database. If there are any differences, a report is created and an administrator is notified immediately. Because it is feasible for the utility to constantly look for new log activity—scanning the log every few minutes, for example—notification of a device configuration change can come within minutes of the change being made.

What about false positives—legitimate, authorized changes made to a device? You certainly don't want a barrage of email notifications or pager alerts just because a perfectly legitimate, planned change is being implemented. To prevent such alerts, most third-party utilities provide a user interface (UI) for making (or at least downloading) changes to devices; the utilities then ignore the changes made through the software. In other words, the software provides a central management interface, and it will alert you only to changes made *outside* that interface. Changes made within the interface can be controlled and audited by the software, closing the loop on device configuration management. Figure 3.2 illustrates this process.



**Figure 3.2: Eliminating alerts for authorized changes made through a management UI.**

Because the management interface has access to the database of stored configurations, the utility can store any new, authorized configurations. The devices will still generate an SNMP trap (or syslog entry or TACACS/RADIUS accounting entry), but the management software's periodic scan won't detect any differences between the device's now-current configuration and the latest one stored in the database. Changes made outside the interface (such as manually through a Telnet session), however, will still trigger an alert because the device will have a different configuration than the one stored by the management software.

Another advantage to third-party management software is role-based security. Later in this chapter, I'll discuss role-based security as a best practice. A device-management utility that provides an alternative, centralized management interface can implement role-based security that is much more flexible and secure than devices' own built-in security.



## Developing a Change Auditing Plan

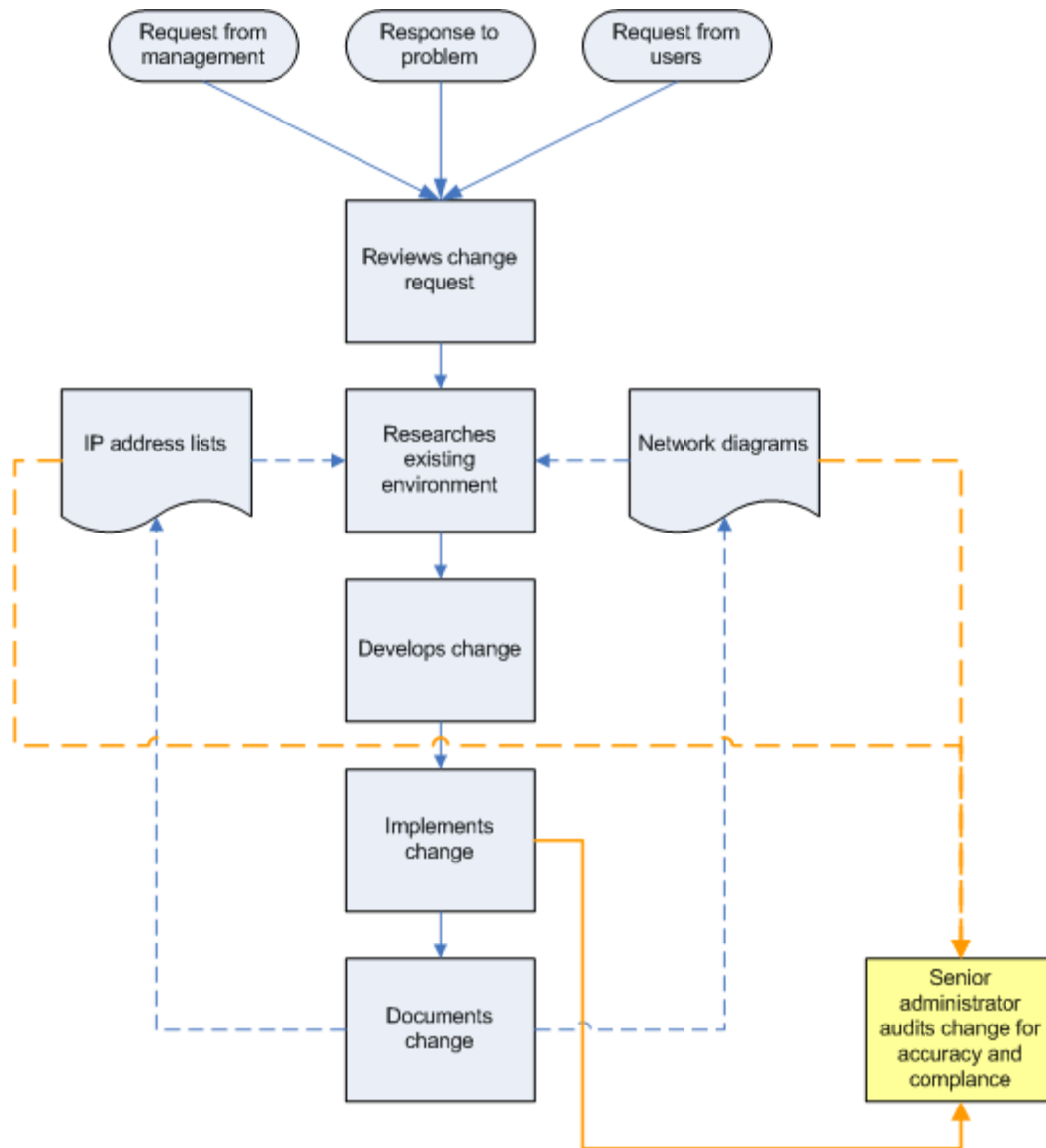
Auditing changes is a key part of an effective overall change-management process and consists of two parts: First, you need to ensure that authorized, planned changes are audited for accuracy and for compliance with corporate standards. Second, you need a plan in place to periodically audit devices for unplanned or unauthorized changes so that devices can remain under tight change management even when out-of-scope changes occur.

### *Auditing Planned Changes*

Even changes that were planned and authorized and made through your complete change-management process (which I began describing in the previous chapter), should be audited. Auditing allows a senior administrator or other technology professional to ensure that

- Changes were made according to plan
- Changes were made consistently and in compliance with corporate standards
- Nothing else has changed on the device

One way to accomplish this auditing is to simply include it in your change-management process as a final step, as Figure 3.3 shows. This last step allows an administrator to double-check not only the change but also the documentation that should have been updated after the change was complete, ensuring that the entire change-management process is being followed correctly.



**Figure 3.3: Adding auditing to your change-management process.**

This type of auditing is the easiest, technologically speaking, because it doesn't require any special tools or capabilities, provided that the auditing occurs *immediately* after the preceding steps in the process. However, it's unusual for administrators to have the time to immediately audit changes. It's also unusual for *every* change to *every* device to be audited *every* time. For planned, authorized changes, it's more likely that auditing will be done on a random basis, simply sampling the changes that are made to spot-check for accuracy and compliance. In that situation, you'll need a third-party tool, such as a change-management application, to help.

Because periodic spot-check audits will occur outside the change-management workflow, you'll need a database of previous changes. That way, if multiple sets of changes have been made to a device, you can audit each individually or simply audit the ones in which you're interested. Change-management software that regularly polls devices and stores their configurations in a repository can assist in the auditing task by providing an easy-to-access history of changes. Most change-management solutions can also highlight the changes that have been made, making it easy to determine what is going on with a device's configuration files.

### ***Periodic Audits***

Periodic audits are an important part of any security process. Regardless of whether changes have been made to a device, these audits allow you to check devices' compliance with corporate standards and ensure that a proper change-management process is being followed in your environment.

Periodic audits should begin with a simple compliance review of the current configuration, ignoring the devices' history of changes. The device's entire configuration file should be reviewed and compared with corporate standards, configuration templates, and any additional documentation and standards. The audit should ensure that the device's current configuration matches any supporting documentation, such as device lists, IP address lists, network diagrams, and so forth.

Audits should then look at the history of a device's configuration, using a third-party change-management solution that can provide access to the device's entire history of change. This portion of the audit should focus on ensuring that changes have followed proper procedure and workflow for your change-management process. Each change should be supported by documentation that describes the reason for the change, includes peer (and/or management) sign-off for each change, and so forth. This audit is designed to ensure that the organization's procedures—with regard to device management—are being followed. Figure 3.4 shows a sample checklist you might use during a periodic device audit.


## Network Device Change Management Audit

	Current config	Recent change 1	Recent change 2	Recent change 3	Recent change 4
Change in ticket tracking system	✓	✓			
Network diagrams updated	✓	✓			
Other docs updated	✓	✗			
Change was peer reviewed	✓	✓			
Change deployed on schedule	✓	✓			
Change made thru management UI	✓	✓			
Change risk assessment on file	✓	✓			
Change details on file	✓	✓			
Config meets corp standards	✓	✓			
Implemented as planned	✓	✗			

**Notes**  
Change included an additional route which was not added to the network documentation

**Notes**  
Most recent change was not implemented according to plan. A new route was included in the actual change which was not in the original change proposal.

Figure 3.4: Sample auditing checklist.


 A proper change-management process involves a “paper trail” for device configuration changes, including planning documents, reviews and approvals, and follow-up changes to supporting documentation. The main purpose of a periodic audit is to ensure that this paper trail exists and matches the device’s configuration history.

These audits might, from time to time, catch unauthorized changes, particularly if you aren't using a third-party management solution that can monitor for these changes automatically. Auditing is critical if you don't have a monitoring solution in place; without auditing, it is unlikely that you'll catch unauthorized changes before they cause a problem on your network.

### ***Providing the Means for Auditing***

Auditing can be difficult, if not impossible, without a continual record of changes made to a device's configuration. This record forms an audit trail, allowing you to later review past changes. Without this audit trail, auditing changes made to a device becomes practically impossible.

Providing the means for auditing—in the form of an audit trail or configuration database—might be a requirement. Many industries are governed by legislation such as the Health Insurance Portability and Accountability Act (HIPAA) and Title 21 Code of Federal Regulations (21 CFR) Part 11. Outside the United States, government and regulatory bodies such as the European Union have established similar rules and regulations. These regulations generally require auditing capability for companies dealing with certain types of information, such as customers' personal information. Although routers and switches don't actually store this information, these devices could provide an avenue for disclosure of that information to unintended parties. Thus, the devices' configurations are required to be audited under the regulations.

 Later in this chapter, I'll discuss these regulations as well as tips for ensuring compliance.

## **Security Best Practices**

The information security industry has defined a set of best practices for managing security. The basic elements of the best practices are:

- **Identification**—Listing critical assets and their locations. The purpose of this element is to ensure that you know where your important resources are located and that everyone else in your environment recognizes each and every resource as important. This list might include specific files as well as folders and network resources such as routers and firewalls.
- **Assessment**—A review of your hardware and software resources that provide access to, or storage of, the assets you have identified as critical. This assessment should produce a list of potential threats to each resource. The idea is to determine the most likely of your assets to be compromised. Although you won't be able to address every possibility, you will be able to mitigate—or prevent entirely—the most likely possibilities.
- **Prevention**—Measures that you take to manage and control the threats you've identified to your critical assets. This element might be as simple as applying permissions to files or creating a policy that requires SNMP community strings and device configuration passwords to be changed every 30 days.
- **Detection**—The analysis of auditing information with the intent to detect threats or evidence of a compromise. This element requires an auditing plan, which I've already discussed. You'll need the means to collect auditing information and the resources (time and personnel) to review audit records for signs of threats or compromise.

- **Response**—Procedures designed to address a compromise, once identified. This element might include immediately locking down a device to preserve evidence of a compromise.
- **Recovery**—Procedures and tools to restore functionality or information after a compromise. This element might involve rolling back a device’s configuration to a recent, authorized version of the configuration file to undo the effects of an unauthorized change. Or, if preservation of evidence is important, this element might include replacing an affected device with a hot spare so that the original device can be retained as evidence.
- **Training**—Inform your staff of proper procedures and the role they play within those procedures. Training needs can be reduced by using intuitive, highly controlled systems that don’t require much training. For example, providing a role-based security mechanism will prevent users from performing actions they’re not trained or authorized to perform, thus providing users with the access they need without having to train them on what not to do.
- **Testing**—Regular tests of your program to ensure that critical security controls and procedures are functioning according to your design. So-called *white hat testing* attempts to compromise your systems in the same fashion as an attacker—short of actually causing a negative impact on production. The purpose of this testing is to confirm that the security measures you have in place are sufficient and to highlight vulnerabilities that aren’t yet addressed. For example, you might make a minor device configuration change from outside your centralized management interface to determine whether your device-management software detects the unauthorized change and creates an appropriate alert.

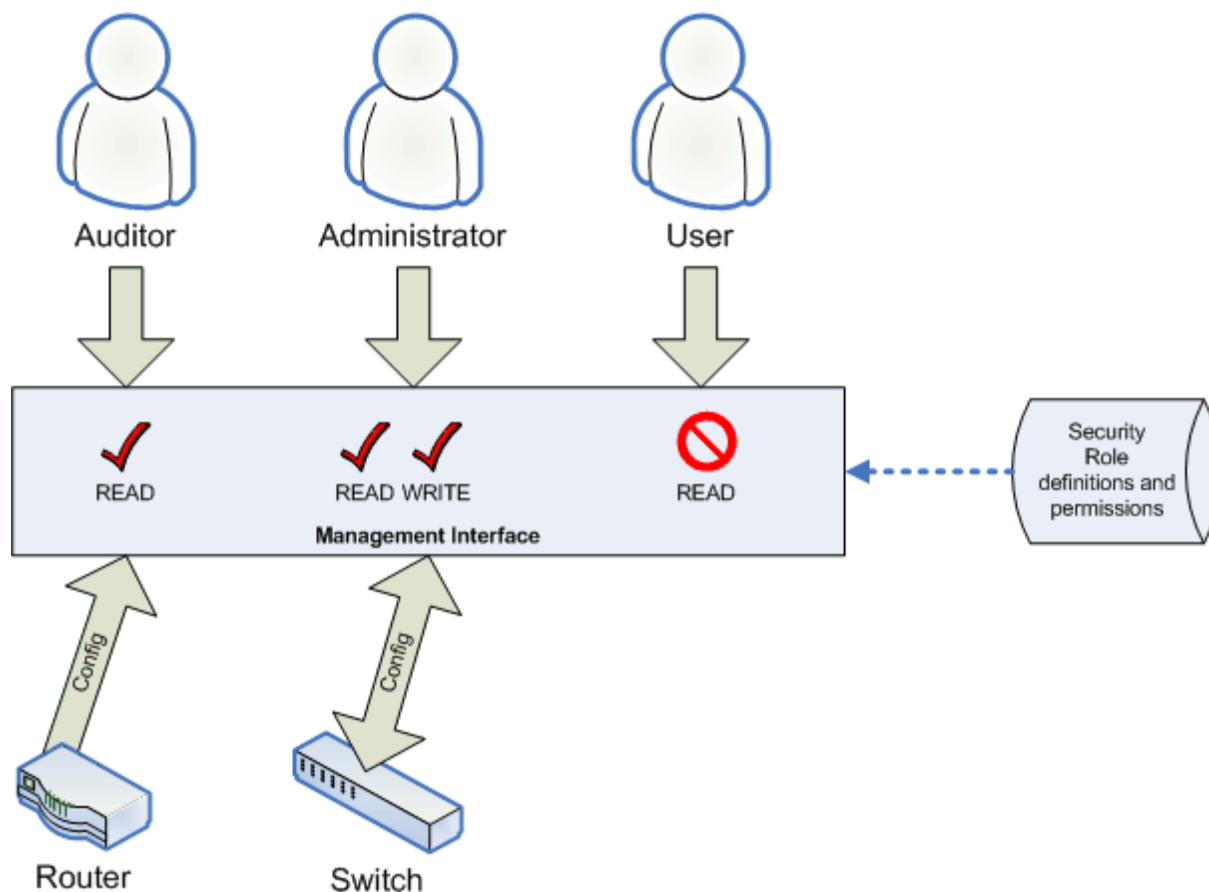
In practice, there are specific steps you can take to improve the security of your network devices. I’ll discuss these steps in the next four sections.

### **Role-Based Authorization for Changes**

Typical security assigns permissions to individual users or to groups of related users, perhaps by department. Such security can be difficult to troubleshoot and less-than-intuitive to apply. Network devices offer even less flexibility, typically providing just one password for general device access and another for full access. Unless the device is configured to use TACACS or RADIUS for authentication, every user who works with the device will use the same password, which is a very ineffective security practice.

Role-based security defines specific roles based on how users need to interact with the device. For example, an auditor role might need read-only access to device configuration files or simply read-only access only to the device’s configuration history rather than to the configuration file on the device itself. Administrators might need the ability to define configuration changes and push those changes out to devices. Regular users might not need any permissions on the device.

An effective device-management solution will eliminate the need for *anyone* to access the device directly. Instead, the solution will maintain user information for logging on and managing devices; the solution will then provide security that offers read or write access to the device’s configuration history, current configuration, and so forth. Users perform their work entirely within the management solution, and the solution accesses devices on the behalf of users as required. Figure 3.5 illustrates this concept.



**Figure 3.5: Role-based security within a device-management solution.**

In this example, the management interface determines that the Auditor role is permitted to read device configurations, so the interface retrieves a router's configuration file upon request. Similarly, an Administrator role is permitted to both read from and write to a switch's configuration, so requests to do so by members of that role are carried out by the management software.

Role-based security is an effective management tool. Other benefits include:

- When users leave your organization or their permissions change, you don't need to immediately change device passwords. You can simply remove them from the appropriate roles.
- Managing security is as easy as placing a user into the appropriate job roles. You don't need to worry about device-specific permissions.
- Because the central management interface defines roles and manages security, adding new devices to your network doesn't require complex security configurations. The elimination of complex security configurations improves consistency and reliability.

### **Regular Password Changes**

Most administrators agree that regular password changes on devices are a good security practice. Most honest administrators will admit that they don't do so. Simply put, changing the device passwords—or SNMP community strings, which serve a similar function—on dozens of devices can take too long, is too error-prone, and creates too much confusion in the support staff.

Utilities are required. Ideally, if you've already implemented a centralized device-management solution, you have the necessary tools. Most solutions can push device changes out to as many devices as necessary, ensuring that passwords are changed properly, consistently, and on schedule. Some management solutions provide a centralized management UI, removing the need to log on to the devices directly. Thus, you won't need to inform support staff of a password change; only the management solution needs to know. Coupled with role-based security, management solutions can offer more effective device security simply because they make it more feasible to observe long-standing industry best practices.

### **Templates for Consistency and Compliance**

Consistency can improve both security and maintenance activities for network devices. For example, consider the partial configuration file from a Cisco AS2509 access server that Listing 3.1 shows.

```
!  
version 11.3  
service timestamps debug datetime msec localtime  
no service udp-small-servers  
service tcp-small-servers  
!  
hostname 2500-DialOut  
!  
enable secret 5 $1$WG3K$8Zhlh6hx4U3U2KFPyW0  
enable password abc  
!  
ip domain-name company.com  
ip name-server 10.0.0.0  
ip address-pool local  
!  
interface Ethernet0  
ip address 10.0.0.1 255.255.255.0  
no ip mroute-cache  
no ip route-cache  
no lat enabled  
no mop enabled  
!  
interface Serial0  
no ip address  
no ip mroute-cache  
no ip route-cache  
shutdown  
!  
interface Serial1  
no ip address  
no ip mroute-cache  
no ip route-cache  
shutdown
```



```

!
interface Group-Async1
ip unnumbered Ethernet0
no ip mroute-cache
encapsulation ppp
no ip route-cache
async default routing
async dynamic address
async mode interactive
peer default ip address pool local
dialer in-band
no cdp enable
ppp authentication chap
group-range 1 8
!
interface Dialer0
no ip address
no ip mroute-cache
no ip route-cache
no cdp enable

```

**Listing 3.1:** A sample of a configuration file from a Cisco AS2509 access server.

And suppose that your organization has two of these devices. The second device is configured as Listing 3.2 shows.

```

!
version 11.3
service timestamps debug datetime msec localtime
no service udp-small-servers
service tcp-small-servers
hostname 2500-DialOut2
enable secret 5 $1$WG3K$8Zh1h6hx4U3U2KFPyW0
enable password abc
ip domain-name company.com
ip name-server 10.0.0.0
ip address-pool local
interface Serial0
no ip address
no ip mroute-cache
no ip route-cache
shutdown
interface Serial1
no ip address
no ip route-cache
shutdown
interface Ethernet0
ip address 10.0.0.2 255.255.255.0
no ip mroute-cache
no ip route-cache
no lat enabled
no mop enabled
interface Group-Async1
ip unnumbered Ethernet0
no ip mroute-cache
encapsulation ppp
no ip route-cache
async default routing

```

```
async mode interactive
peer default ip address pool local
dialer in-band
no cdp enable
ppp authentication chap
group-range 1 8
interface Dialer0
no ip address
no ip mroute-cache
no ip route-cache
no cdp enable
```

**Listing 3.2:** A sample of a configuration file from another Cisco AS2509 access server in the same organization.

Do you see the differences? Of course not—these files are long and complex, and although these two files are very similar, they aren't laid out the same. Because they're inconsistently formatted, it is difficult to spot errors or differences in the configuration. A consistent configuration based on a template offers better security simply because using one makes it more difficult to make configuration mistakes. In addition, it is easier to spot mistakes when configuration settings are in a familiar format.

Based on this concept, some third-party device-management solutions provide built-in templates that almost resemble electronic forms; by completing the forms, administrators can configure devices consistently and more reliably. The templates can often be created with constraints, ensuring that only valid data is specified for various parameters, enforcing naming conventions, and so forth.

### **Proactive Security**

We've already explored how auditing can play an important role in proactive security. It helps you measure compliance and consistency and spot potential security errors before they occur. For example, you might have a corporate standard requiring that all access servers have a secret and password:

```
enable secret 5 $1$WG3K$8Zhlh6hx4U3U2KFPyW0
enable password abc
```

Regularly auditing will catch devices that are missing this configuration setting or that have an inconsistent setting, allowing you to catch the problem before it becomes an exploited security vulnerability.

One of the most important and oft-overlooked steps in creating a change-management process is to re-create every current device configuration in a consistent, compliant form. Doing so will create a "clean slate" for future management and will make it easier for you to be more proactive about device and network security.

## Developing an Incident Response Process

When a security incident does occur, you need a plan in place to quickly restore functionality while preserving evidence of the incident for later review. The balance between restoring functionality and preserving evidence—goals that are often conflicting—has a management term: *convergence*. It occurs when two management goals, such as preserving evidence and restoring functionality, are at odds. A predefined incident response plan codifies an organization’s convergence policies by laying out exactly which steps will be taken.

When a security or operational incident occurs, it is easy for administrators to take whatever action is necessary to restore functionality, often destroying evidence. To prevent the loss of evidence, an incident response plan serves as a checklist and guide, as Figure 3.6 shows. The plan is designed to minimize downtime while preserving as much evidence as possible of the incident for later review.

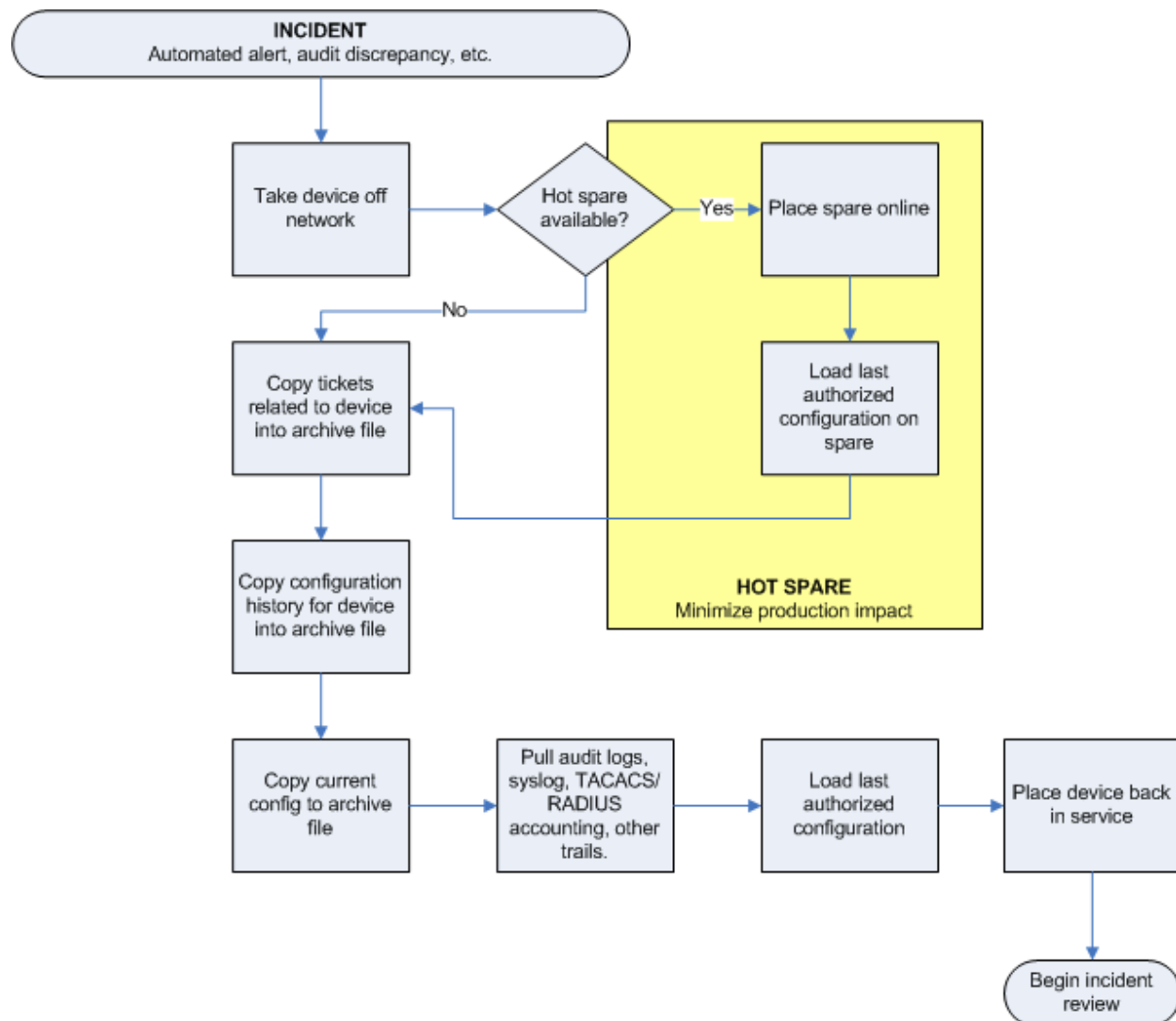


Figure 3.6: An example incident response plan.

## Supporting Corporate Governance Requirements

The days when industry could be relied upon to govern itself are gone. In light of recent scandals and an increasing focus on public welfare, governments are passing laws and regulations that define how specific industries must treat the data entrusted to them. Usually, simply following industry best practices will bring you into compliance with these regulations and requirements.

There are network security industry best practices that we explored earlier in this chapter. Broadly speaking, network devices don't meet these best practices on their own. However, a third-party change-management solution can provide everything you need:

- An inventory of device resources (identification)
- Records permitting the review of devices' configurations (assessment)
- Centralized management of device security (prevention)
- Notifications of unauthorized changes (detection)
- Records of unauthorized changes (response)
- The ability to roll back unauthorized changes by using an authorized configuration (recovery)
- Simplified role-based security that ensures that administrators have only the required access to device's configurations (training and prevention)
- The ability to support periodic auditing and testing procedures (testing)

In the next few sections, we'll explore three of the major regulations affecting industries in the United States, and how proper network configuration management can play a vital role in complying with these regulations.

### **HIPAA**

HIPAA is an enormous piece of legislation designed to improve the portability of healthcare as well as tighten controls over who has access to patients' private information. The act affects any device, system, or component that stores, handles, or transmits certain private patient information. Because devices such as routers, firewalls, and switches can transmit this information on the network, they can fall under HIPAA's regulations.

For example, one of HIPAA's requirements is that patients be able to request not only their records but also an accounting of who their records have been disclosed to. A router or firewall doesn't keep a list of transmission recipients; however, to be certain that your organization meets the HIPAA requirements, you must ensure that your devices won't transmit information to undisclosed parties. Requests for access to patient files, permissions granted to users, and any other configuration that could result in access to patient data must be constantly reviewed—including the configuration of your network devices. By showing a complete history of device configurations, you can prove that only authorized users were able to receive data from those devices; if you're unable to show an audit trail of a device's configuration, you could face allegations—which would be difficult to defend against—that the devices were modified to transmit confidential data to unauthorized persons.

### Getting Paranoid with the Government

Regulations such as HIPAA don't consider the *likelihood* of unauthorized disclosure through certain means; they consider the *possibility* in general terms. For example, the HIPAA regulation doesn't specify that network switches must be audited for configuration changes; the regulation simply states that you must be able to account for all access to patient information.

It is *possible*—however improbable—that an intruder with access to your physical network could attach a laptop computer to a switch port, then program the switch to place that port into promiscuous mode. Doing so would duplicate all traffic handled by the switch to that port, allowing the laptop to capture traffic, thus disclosing confidential information. The allegation could be made that you let this disclosure occur. By maintaining an audit trail of device configurations, you can attest that no such configuration change was made and that the switch's configuration at no time created any unusual configuration conditions that could bypass your file permissions and other security measures.

Much of HIPAA relies on written policy statements rather than pure technological solutions. However, in practice, complying with HIPAA requires every device, server, or component that comes in contact with confidential information to be regularly audited. Certain HIPAA regulations are best implemented through the security best practices that were discussed earlier. Role-based security, for example, is a best practice that makes it easier to comply with many HIPAA requirements for access to confidential information. Using a third-party device configuration management solution to centrally manage device passwords, SNMP community strings, and other sensitive information also makes it easier to comply with HIPAA requirements for access control and accountability.

Figure 3.7 illustrates how a third-party device-management solution can overcome the rudimentary security built-in to most network devices and provide many of the tools needed for HIPAA compliance: an audit trail of device access and changes, role-based security for accessing device configurations, and so forth.

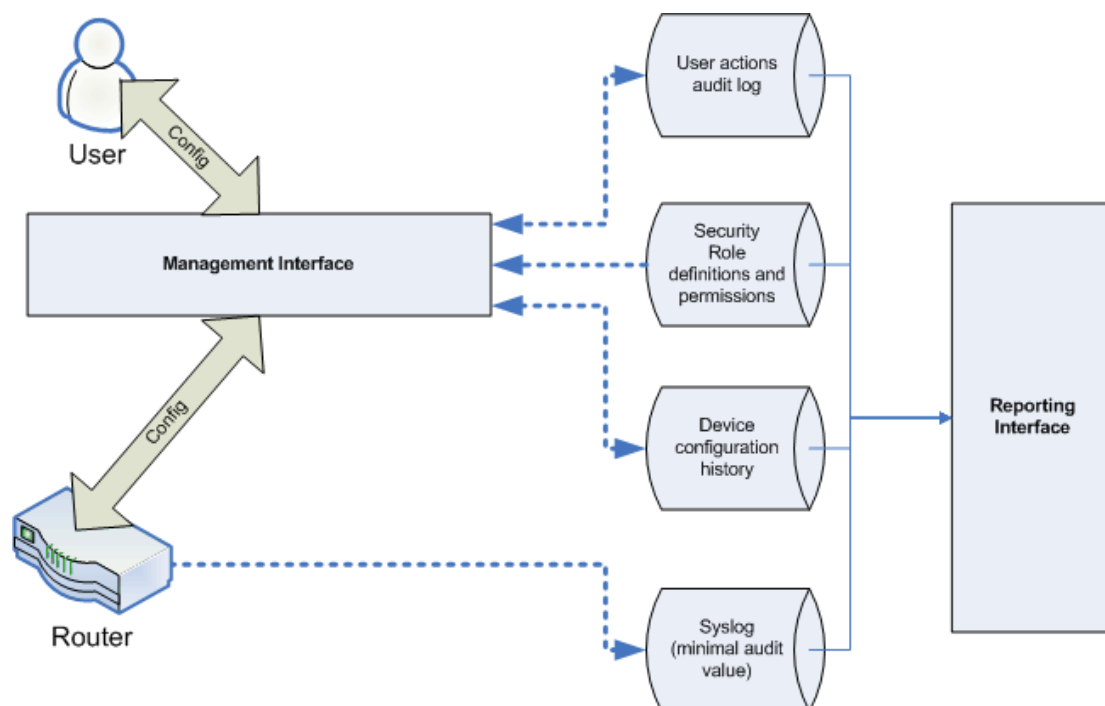


Figure 3.7: Using a third-party management solution to ensure HIPAA compliance.

### **The Gramm-Leach-Bliley Act**

While the healthcare industry deals with HIPAA, the financial services industry is working under the strict Gramm-Leach-Bliley Act (GLBA). Passed in 1999, the act contains seven titles and 740 sections, making it a large and comprehensive piece of legislation. Title V, Section 502, “Obligations with respect to disclosures of personal information” is an important section of the act that deals with information privacy and security. This section is having a major impact on IT operations.

As with HIPAA, financial institutions must not only protect customers’ confidential information but also provide an accounting for all disclosures of that information. And, as with HIPAA, while your network devices aren’t strictly responsible for such disclosure, they can cause it. Thus, you need to implement measures—auditing, role-based security, and centralized management, for example—that provide you with controls and auditing tools to ensure that your devices don’t become an unintended source of disclosure.

The GLBA has three broad requirements with regard to information security. Institutions must:

- Provide safeguards that ensure the security and confidentiality of customer records and information.
- Provide measures that protect against any anticipated threats or hazards to the security and integrity of such records.
- Provide protection against unauthorized access to, or use of, such records or information that would result in substantial harm or inconvenience to any customer.

In addition, you must be able to provide records that prove you have taken steps to meet these requirements and that your measures have been continuously in effect. A change-management solution can provide evidence in the form of a device change history.

### **21 CFR Part 11**

Title 21 CFR Part 11 is comprehensive legislation that specifies stringent requirements for the security of electronic records. Many government contractors and institutions are required to comply with these requirements, which include:

- Validating computer systems to ensure reliability and consistency
- Maintaining an audit trail listing changes to data
- Requiring authority checks to ensure that only authorized individuals can use the system

With regard to network devices, these requirements are most easily met through third-party management software. Although a device, such as a router, can provide rough audit trails through technologies such as TACACS or RADIUS, such logs don’t provide the level of detail required by 21 CFR Part 11. A third-party solution, however, can periodically poll devices’ configurations to create a detailed, historical accounting of how the device has changed. Solutions that provide a centralized, alternative management interface can also implement role-based security, authority checks, and detailed auditing records that indicate *who* made changes to devices’ configurations. As with HIPAA and the GLBA, this audit trail is one of the most important government requirements.

## Summary

Change management undoubtedly has an impact on the overall security of your environment. Just as network devices play a critical, often-unappreciated role in the operation of your network, they play a critical role in its security. Devices with unauthorized, inconsistent configurations can be a security vulnerability—devices under an effective change-management process are more likely to be secure.

As IT security continues to evolve and as governments and other regulatory bodies pass legislation concerning information security, tighter control of network devices and their configurations becomes an important factor of any business. Although network devices provide only basic intrinsic security features, the addition of a third-party solution to provide centralized role-based management, auditing capabilities, and incident response features can provide all the tools you need to create a more secure network environment.

In the next chapter, I'll discuss the scope of network change management, and how it applies to routers, servers, switches, firewalls, load balancers, access concentrators, and intrusion detection and prevention devices. I'll also discuss how high-end network change-management solutions can become an integrated part of an overall network-management strategy, including integration with frameworks such as Hewlett-Packard OpenView and more. I'll introduce you to the tools available to collect change management information and to create a central, secure management point for change in your environment.