

Realtime
publishers

"Leading the Conversation"

The Shortcut Guidetm To



Automating Network Management and Compliance

sponsored by



i n v e n t

Don Jones

Introduction to Realtimepublishers

by Sean Daily, Series Editor

The book you are about to enjoy represents an entirely new modality of publishing and a major first in the industry. The founding concept behind Realtimepublishers.com is the idea of providing readers with high-quality books about today's most critical technology topics—at no cost to the reader. Although this feat may sound difficult to achieve, it is made possible through the vision and generosity of a corporate sponsor who agrees to bear the book's production expenses and host the book on its Web site for the benefit of its Web site visitors.

It should be pointed out that the free nature of these publications does not in any way diminish their quality. Without reservation, I can tell you that the book that you're now reading is the equivalent of any similar printed book you might find at your local bookstore—with the notable exception that it won't cost you \$30 to \$80. The Realtimepublishers publishing model also provides other significant benefits. For example, the electronic nature of this book makes activities such as chapter updates and additions or the release of a new edition possible in a far shorter timeframe than is the case with conventional printed books. Because we publish our titles in “real-time”—that is, as chapters are written or revised by the author—you benefit from receiving the information immediately rather than having to wait months or years to receive a complete product.

Finally, I'd like to note that our books are by no means paid advertisements for the sponsor. Realtimepublishers is an independent publishing company and maintains, by written agreement with the sponsor, 100 percent editorial control over the content of our titles. It is my opinion that this system of content delivery not only is of immeasurable value to readers but also will hold a significant place in the future of publishing.

As the founder of Realtimepublishers, my *raison d'être* is to create “dream team” projects—that is, to locate and work only with the industry's leading authors and sponsors, and publish books that help readers do their everyday jobs. To that end, I encourage and welcome your feedback on this or any other book in the Realtimepublishers.com series. If you would like to submit a comment, question, or suggestion, please send an email to feedback@realtimepublishers.com, leave feedback on our Web site at <http://www.realtimepublishers.com>, or call us at 800-509-0532 ext. 110.

Thanks for reading, and enjoy!

Sean Daily
Founder & Series Editor
Realtimepublishers.com, Inc.

Introduction to Realtimepublishers..... i

Chapter 1: The Essentials of Automating Network Operations.....1

What Is Automation?1

The Price of a Non-Automated Network3

 Inconsistency.....3

 Security Breaches.....5

 Downtime.....6

 Inability to Respond to Business Demands7

 Lack of Business Reliability8

 Loss of Customer and Investor Confidence.....8

 Lack of Efficiency.....8

Your Network’s Challenges.....9

 Poor Administrative Practices.....9

 Technical Attacks.....10

 Social Engineering11

 Physical Disaster and Other Elements Outside Your Control12

 Process Adherence12

The Benefits of an Automated Network14

 Configurable14

 Consistent.....15

 Recoverable.....15

 Securable.....16

 Compliant.....18

 Auditable.....19

 Efficient.....19

 Flexible19

Traditional Management vs. Automated Management.....20

 Ad-Hoc Configuration20

 Manual Backup and Recovery22

 Scripted Administration23

Summary23

Chapter 2: Automating Network Compliance and Security24

What Does the Network Have to Do With Compliance?24

- How Legislation and Rules Affect the Network.....24
- How Network Operations Affect Compliance.....26
- How Networks Become Non-Compliant26

How Automation Affects Security and Compliance27

- How Automation Improves Daily Administration27
- How Automation Adds Accountability to Network Operations.....28
- How Automation Ensures that the Network Remains Compliant31

Traditional Security and Compliance Loopholes32

- Overlooked Managed Elements.....32
- Point-in-Time Auditing Misses Changes.....33
- Direct Device Administration Causes Inconsistency.....34
- Inability to Tie Changes to Requests Reduces Accountability.....34
- Inability to Enforce a Configuration Management Process Affects Compliance.....36
- Inability to Effectively Report Makes Compliance Audits Difficult.....36

Technologies and Tools that Close the Loopholes36

- Pass-Through Administration in a Management Solution.....36
- Automated Detection of Out-of-Process Changes.....37
- Automation of the Change Management Process.....37
- Role-Based Security in a Management Solution37
- Auditing in a Management Solution38
- Compliance Reporting38
- Automated Configuration Remediation.....38

The Need for Auditing.....38

- End-State vs. Configuration Auditing.....38
- Network Management Through Templates and Policies.....39
- Automated Configuration of Devices40

Building a Plan to Support Compliance and Security40

- Creating a Business Process That Supports Compliance and Security.....42
- Mapping Technologies to Your Business Process.....42
- Evaluating Solution Features that Support Compliance and Security44
- Evaluating Solutions’ Ease of Adoption.....44

Checklist: Tools and Technologies You Need.....	44
Summary	45
Chapter 3: Automating Network Operations and Maximizing Availability.....	46
Business Continuity vs. Disaster Recovery	46
Disaster Recovery Means It Is too Late	46
Business Continuity: Continuous Operations Even Through a Disaster	47
Challenges for Continuous Network Operations	47
Lack of Centralized Configuration Repositories	48
Difficulty in Restoring Failed Devices	48
Complexity in Maintaining Accurate Inventory	49
Problems Caused by Direct Device Management	49
Lack of Consistency and Standardization.....	49
Too Much Work: Inefficiency in Network Administration.....	50
Lack of Flexibility to Respond to Business Needs	52
The Business Process for Maximizing Availability.....	52
Creating Standards and Consistency.....	52
Learning from the Information Technology Infrastructure Library Framework	53
Adopting a Change Assessment Process	55
Creating a Complete Change Management Process	56
Using a Process to Improve Network Operations.....	57
Using a Process to Support High Availability	57
Using a Process to Improve Efficiency.....	58
Using a Process to Improve Flexibility and Support New Business Needs	58
Technologies that Support Automation and Maximum Availability.....	58
RADIUS/TACACS+	58
TFTP	59
Telnet/SSH.....	60
SNMP.....	61
Syslog.....	62
All in One: Configuration Management Solutions	62
Enable Disaster Recovery	63
Improve Efficiency	63

Improve Consistency	63
Enable Business Flexibility.....	64
Business Continuity Scenarios and Solutions.....	65
Single-Device Misconfiguration.....	65
Single-Device Failure	65
Multiple-Device Misconfiguration	66
Multiple-Device Failure.....	66
Partial or Total Facility Failure.....	67
En Masse Device Management.....	67
Reconfiguring the Network to Support New Business Needs	67
Building a Plan for Automating Network Operations and Maximizing Availability.....	68
Checklist: Tools and Technologies You Need.....	69
Summary	69
Chapter 4: Building the Plan for Automating Network Operations	64
Building the Business Process	64
Frameworks for Change and Configuration Management.....	65
Examining Your Current Processes	66
Adapting Your Processes.....	67
The ITIL Connection	69
From Process to Reality: Where Automation Fits In.....	72
Creating a Solution Evaluation Project.....	74
Determining Critical Criteria	74
Evaluating Solutions	76
Scoring Evaluations	76
Integrating Your Processes and a Solution	81
Creating Business Plans.....	83
Planning for Daily Administration.....	83
Planning for Auditing	85
Planning for Disaster.....	87
Summary	89

Copyright Statement

© 2006 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 Realtime Nexus—The Digital Library. All leading technology guides from Realtimepublishers can be found at <http://nexus.realtimepublishers.com>.]

Chapter 1: The Essentials of Automating Network Operations

Modern computer networks are an absolutely mission-critical part of almost any business. Yet because the network doesn't do anything visible—most users, that is, are more focused on services such as email or file sharing than on the network that makes these services possible—the network's critical role is often overlooked or diminished. However, the important role the network plays in the business has a very significant impact on the business' efficiency, overall operations, and the bottom line. This guide explores the ways in which a more efficient, more automated network can lend significant value to the business in a number of different areas.

What *Is* Automation?

au•to•ma•tion *n.*

1. The automation operation of control of equipment, a process, or a system.
2. The techniques and equipment used to achieve automatic operation or control.
3. The condition of being automatically controlled or operated.

- *The American Heritage Dictionary of the English Language, Fourth Edition*

Automation seems like a strange term to apply to network operations. After all, at first glance, networks don't seem to offer much in the way of manual operation. Routers already route packets automatically, for example. A technology such as Dynamic Host Configuration Protocol (DHCP) might seem to be a beneficial form of network automation, but not much else immediately comes to mind.

Network operations, however, are in fact a *tremendously* manual process in most companies. The provisioning, configuration, ongoing maintenance, disaster recovery, and other tasks involved in managing the network are typically accomplished entirely through manual effort, or (too often) inefficiently automated through cobbled-together techniques. In many companies, every change made to the network's configuration—from modifying a router's routing table to backing up the configuration of a switch—is made manually. Recovery of failed devices is performed manually. *Everything* needed to keep the network up and running, in many companies, is done manually. But by automating network operations—that is, by automating day-to-day changes, a change management workflow, disaster recovery operations, and other tasks related to network management—companies can realize significant business benefits.

To help you do so, this chapter will explore the essentials of automated network operations: The real business costs of a non-automated network, the benefits offered by automation, and a comparison of automated versus more traditional management techniques.

The next chapter will look at network compliance and security from an automation viewpoint, focusing on how compliance affects the network, how automation can affect security and compliance efforts, and how traditional—that is, largely manual—management techniques leave loopholes in security and compliance management efforts. This chapter will also look at how various technologies and tools can be used to close those loopholes to create a more secure, more compliant network. Chapter 2 will also look at auditing, and explore the differences between various auditing philosophies and how network automation can be used to make auditing easier and more efficient.

Chapter 3 will explore ways in which automation can help improve the network’s operational efficiency and increase the network’s overall availability, or uptime. This chapter will explain the important differences between commonly sought-after disaster recovery goals and the more desirable business continuity goals, and the challenges that exist to a network’s overall availability in a traditionally-managed and -operated network. Chapter 3 will help you develop a business process for maximizing network availability, and look at some of the core technologies that make a high-availability network a possibility. I’ll also help you understand how all-in-one network configuration management solutions can improve uptime through automation, and how they can respond to various business continuity scenarios and challenges with ease.

Finally, Chapter 4 will help you put together a complete plan for automating your network operations tasks. It will start with the supporting business processes, helping you adapt frameworks such as the Information Technology Infrastructure Library (ITIL), and continue with how to examine and evolve your current business processes. Because tools and technologies play such an important role in network automation, this chapter will help you develop a solution evaluation project and list of criteria that works for your business so that you can evaluate various tools and technologies and identify the ones that work best for you. The chapter will then show you how to integrate your new business processes and whatever solution you select to create a complete, “closed loop” system for automated network operations. The guide will finish by covering core business plans for daily network administration, security and compliance auditing, and disaster planning.

This guide has a lot of ground to cover—I’ll keep everything moving quickly so that you can begin implementing a more efficient, more automated network right away. Let’s start, however, by clearly identifying the real business costs of an old-fashioned, non-automated network.

The Price of a Non-Automated Network

Many network managers and engineers don't realize that a manually operated network carries a number of significant risks and business costs. Often, an automated network is seen as a convenience to the engineers and technicians who run the network; in fact, an automated network provides significant value directly to the business. Of course, that means a non-automated network detracts significant value from the business, and comes with real, tangible costs. The next several sections will discuss some of those costs and how they can negatively impact the business.

Inconsistency

Inconsistency is a major problem for any network. For example, consider Figure 1.1, which shows two router configuration files side-by-side in Windows Notepad. Both configuration files are intended for use on the same or similar routers; can you spot the differences?

```

Untitled - Notepad
File Edit Format View Help
version 10.3
!
hostname blah.test.org
!
enable secret 5 (#>30u34(#(%)_$%*#!_BIEWQBMX
enable password passwordhere
!
ip subnet-zero
!
interface Ethernet0
 ip address 5.5.5.5 255.255.255.128
 no ip directed-broadcast
!
interface Serial0
 description Gateway to Internet provider
 ip address 4.4.4.4 255.255.255.252
 no ip directed-broadcast
 encapsulation frame-relay IETF
 no ip route-cache
 bandwidth 1536
 frame-relay lmi-type ansi
!
interface Serial1
 no ip address
 shutdown
!
ip classless
 ip route 0.0.0.0 0.0.0.0 4.4.4.3
 banner login AC
 -=welcome to the main gateway=-
 AC
!
line con 0
 password passwordhere
 login
line aux 0
 transport input all
line vty 0 4
 password passwordhere
 login
!
end

Untitled - Notepad
File Edit Format View Help
version 10.3
!
hostname blah.test.org
!
enable secret 5 (#>30u34(#(%)_$%*#!_BIEWQBMX
enable password passwordhere
!
ip subnet-zero
!
interface Ethernet0
 ip address 5.5.5.5 255.255.255.128
 no ip directed-broadcast
!
interface Serial0
 description Gateway to Internet provider
 ip address 4.4.4.4 255.255.255.252
 no ip directed-broadcast
 encapsulation frame-relay IETF
 no ip route-cache
 bandwidth 1536
 frame-relay lmi-type ansi
!
interface Serial1
 no ip address
 shutdown
!
ip classless
 ip route 0.0.0.0 0.0.0.0 4.4.4.3
 banner login AC
 -=welcome to the gateway=-
 AC
!
line con 0
 password passwordhere
 login
line aux 0
 transport input all
line vty 0 4
 password passwordher
 login
!
end

```

Figure 1.1: Comparing router configurations.

These are *simple* configuration files, dozens of times shorter than the one a production router might usually contain. There are two inconsistencies here: The first is in the login banner, “Welcome to the main gateway,” which simply says “Welcome to the gateway” in the second file. There is no real harm in that type of inconsistency. The second inconsistency, however, is more serious: The password for the “line vty” interface, near the bottom of the files. Inconsistent security settings means inconsistent network access. Some devices may be more secure than others. Inconsistencies can cause one device to operate and another, similar device, to fail. It can interfere with the operation of the network and waste hours in frustrating troubleshooting. Other configuration inconsistencies can make devices behave more or less efficiently than others, and make them more or less difficult to manage. Inconsistent configurations are a hallmark of manually configured networks, simply because human error makes it nearly impossible for manual changes to be made consistently on any kind of large scale. Although devices might start out consistently configured (perhaps using a template of some kind as a starting point), every manual change made to a device increases its deviation from that initial standard and increases the amount of inconsistency in the network.

The business costs of inconsistency include:

- Lowered security—In a large device configuration—such as a firewall or perimeter router—it’s extremely easy for minor inconsistencies to go overlooked in a manual review. It’s also easy to introduce minor inconsistencies when making changes manually. However, minor inconsistencies—such as an incorrectly opened port—can result in vastly decreased overall security. In companies dealing with compliance requirements, lowered security can result in a failed compliance audit, with the associated costs. Regardless, lowered security opens the door to very real business losses.
- Less manageability and less flexibility—An inconsistently configured network is more difficult to manage. When making configuration changes, devices must be carefully reviewed by skilled (and expensive) engineers to ensure that the new change won’t negatively impact the device or the network. With each device configured differently—that is, inconsistently—this review takes longer and is more error-prone because small details are likely to be overlooked. As a result, every change to the network becomes more difficult and risky. Businesses often deal with this risk through avoidance, meaning changes to the network are discouraged. This method leads to a network that is less able to respond to changing business requirements, keeping the business from evolving and remaining competitive.
- Less recoverable devices—When devices are inconsistently configured, recovering a device after a failure becomes riskier because there is no one standard that describes how the device was configured. Thus, downtime lasts longer, as devices’ inconsistent configurations are more painstakingly recreated. Downtime equals lost money for the business.

There is no question that inconsistently configured networks present significant business risks, both short- and long-term. There is also no question that a manually configured network is more likely to contain these inconsistencies, increasing the chance that the network will detract value from the business rather than adding value.

Security Breaches

Security breaches are a major concern for any business. The risk of losing proprietary intellectual property, personal identifiable information, and other confidential data—and thus losing money as well as customer and investor confidence—is well understood. At the network level, security breaches are made possible through unpatched devices, improperly configured devices, and poorly configured devices (devices that have, for example, old or inappropriate Simple Network Management Protocol—SNMP—community strings, thus leaving themselves open to unauthorized reconfiguration). Manually configured networks exhibit some common characteristics and management practices that make security breaches more likely:

- **Unpatched**—Deploying patches can be a nightmare, especially on large networks, so overworked engineers and administrators are often well behind the curve in deploying patches to devices. However, because patches often fix security issues and other vulnerabilities, unpatched devices are a major security risk.
- **Poor practices**—Network security elements such as configuration passwords, SNMP community strings, and so forth should be changed on a regular basis, just like network user passwords. However, *rolling passwords*, as they are called, are a major undertaking in a manually configured network. Most companies simply ignore the security risk and accept poor practices as a way of life.
- **Time**—The sheer time involved in manually configuring devices often leads harried administrators to not implement deep defensive measures which, were they in use, could help stop a broader range of attacks.
- **Unauthorized changes**—Manually configured networks are open to problems with process adherence. In other words, businesses may have a thorough configuration management process in place, but a manually configured network provides a number of loopholes to these processes, allowing administrators to make “one minor change” here and there, outside the process. These changes, bypassing as they do the process review and approval stages, are more likely to open security holes in the network that go undetected for a significant period of time.

Why Are Network Devices Vulnerable?

Some network managers and engineers don't believe that their devices are vulnerable. An infrequently changed SNMP community string, they feel, doesn't present a risk because the device itself is behind a firewall, thus preventing—or so they believe—an attacker from taking advantage of any vulnerability. Likewise, unpatched devices aren't viewed as a risk because there's "no way" for an attacker to access the device and exploit any vulnerabilities.

However, that viewpoint is naïve. *Most* attacks on corporate resources come from *within* the firewall, either from internal attackers (such as disgruntled employees) or from malicious software (such as viruses) that make it into the intranet through users' removable media drives or other means. The idea that the intranet is somehow inherently safe is completely false, and creates a false, and highly inappropriate, sense of security. The intranet is *not* safe.

Therefore, best practices suggest changing SNMP community strings, device configuration passwords, and other security elements on a frequent (often every 30 to 45 days) basis. In addition, best practices—and device manufacturers—recommend applying the latest device software patches as quickly as possible, particularly for patches that repair a security vulnerability. Ignoring these practices invites attack, with all the very real costs that a successful attack can introduce to the business.

Of course, these best practices are often difficult and even risky to perform manually. For example, what if a device's SNMP community string or configuration password is input improperly? The device becomes unmanageable. These reasons highlight why network automation is crucial and why a manually configured network offers so many costs in terms of poor security.

Downtime

Most businesses are well aware of the costs of downtime, but most don't realize the degree to which a manually configured network makes downtime both more likely and longer in duration. Simply put, manual changes are error-prone. One Cisco study found an error rate of 5 percent for manually performed changes; in a network with just 100 devices, that means five of those devices will be improperly configured at some point. Because of the precise nature of network device configuration, a single error can result in partial or total network service outages. Simply planning a single minor change to every device on the network will, statistically, result in five of them being changed incorrectly and therefore introducing at risk for downtime. That risk, by the way, is why so many manually configured networks avoid deploying patches, changing device passwords and other security elements, and so forth—error, and potential downtime, is practically inevitable.

Once downtime does occur, a manually configured network is more likely to have a longer downtime. The reason is that device configuration backups must be manually made, as well, with the same error rate and likelihood of someone forgetting to make the backup. Restoring the device configuration is also a manual process, requiring the correct backup configuration to be located, manually loaded into the device (meaning someone may have to look up the proper device access passwords first, yet another delay), and so forth. An informal study with a consulting client found that the average manual recovery time for a network device was 30 minutes. In some businesses, literally millions of dollars can be lost in 30 minutes if the network isn't available.

Inability to Respond to Business Demands

With all of the problems—inconsistency, downtime, security issues, and more—that a manually configured network can present, it's no surprise that most companies adopt a strict policy regarding network changes: They perform the changes only if they are absolutely necessary. The idea is to reduce the risk of inconsistency, downtime, security problems, and so forth, and that severely limiting change will, in fact, reduce that risk. In many shops, the very word *change* is a bad one, and engineers will share horror stories of the supposedly minor change that took the company down for an entire day.

But modern business is *about* change. In order to remain competitive, businesses need the flexibility to adapt quickly to changing market conditions, and that often means changing the network as well. Companies such as Hewlett-Packard, IBM, and Microsoft are all touting frameworks and technologies that enable rapid change (Adaptive Enterprise, OnDemand, and Dynamic Systems Initiative, respectively) because smart businesses realize that flexibility is one of the most important competitive advantages they can possess. Restricting network change because of the risk of change itself artificially limits what the business can do to remain competitive; this limits the business' growth, frustrates employees, and hinders investor confidence.

Most companies don't see themselves as inflexible. However, almost any employee at any company can tell you about the workarounds they're forced to take simply because the IT staff isn't capable of providing the proper services. Some examples from a recent client include:

- Employees were sending confidential materials to business partners through free email services such as Yahoo and GMail. The corporate email system prevented these materials from being sent, and the network staff didn't want to create any kind of extranet or other connectivity options.
- Several new services needed to be implemented that required changes to the corporate firewall. However, permanent firewall changes were limited to a once-per-quarter scheduled change period, meaning the company simply had to wait 3 months for the needed changes to be implemented.
- The company failed a compliance audit because device configuration passwords hadn't been changed. Network managers had determined that the yearly cost of changing passwords would be in the neighborhood of \$5184 and require 103 hours per year; they had decided it would be cheaper not to do so. As a result of the failed compliance, the company was delayed 6 months in an initial public offering.

In almost every case, the reason for the lack of a real solution was because the network infrastructure team didn't want to make the necessary changes because changes required too much advance planning, too much effort to implement, and too frequently created additional issues. As a result, they strictly limited changes—and thus limited the business.

Lack of Business Reliability

What is a reliable business? A business upon which customers and employees can depend. If you work for a bank and network infrastructure issues take your self-service banking Web site offline for a day, customers complain and feel you're unreliable. If you work for a hospital and a network failure makes patient data unavailable for a few hours, customers could *die*. Any kind of network failure can result in public impact and make customers—and even employees, in the case of purely internal failures—view your business as unreliable. People don't like to do business with, or work for, unreliable businesses. If you're viewed as unreliable, people find someone else with which to do business, or someone else to work for. The cost of this perception and its associated losses are difficult to precisely quantify but are very real nonetheless.


A network that is consistently configured, well-secured, and highly available can help bring more reliability to your overall business. A network that's inconsistently configured, poorly secured, and that experiences downtime is a network that makes your business unreliable, and unreliable businesses don't last long.

Loss of Customer and Investor Confidence

Inconsistency, security issues, downtime, inflexibility—these are mainly internal issues that external observers won't see directly. But the results of these issues can cause a significant loss in customer and investor confidence. For example, if your company's poor network security results in a loss of intellectual property, your stock price could fall through the basement and customers—wary of entrusting you with their own personal information—could stay away in droves. Witness the public relations nightmares that financial services firms have had when credit card holder information was stolen over the Internet. In a competitive marketplace in which consumers are always a step away from going to the competition, this sort of failure can ruin a business for years, if not permanently.

Lack of Efficiency

Suppose you have a network with 800 devices and that an administrator can manually change device passwords, when the time comes, at the rate of about 1 per minute—a task the administrator performs once per quarter—which is a long time to go between password changes. This task results in about 95 hours per year. Let's also say that you spend about 300 hours per year deploying patches to those 800 devices (not unreasonable; that allows about 10 minutes per patch and just 2 to 3 patches per device for an entire year), and another 90 hours gathering configurations from devices (for backups and so forth). These tasks add up to about 485 hours per year of network management, which can run into some pretty significant monetary costs, just in terms of administrator salaries. And that is just for some very minimal administrative tasks—not even the complete scope of what an administrator would actually do all year.

 Automating these tasks reduces the numbers to about 30 hours for patch deployment, 15 minutes for configuration backups, and about an hour for password changes—*per year* (numbers based on estimates by both Cisco and Nortel). That is a lot of savings just in administrator time (and the associated salaries): Your administrators will have a lot more time to work on other projects as a result. This automation can also allow you to reduce the number of staff or contractors, and make a really significant impact on the bottom line. True, you'll spend some money on tools and technologies to achieve this automation, but the savings are significant. And keep in mind that I've only touched on the tip of the iceberg with this illustration. In reality, your network administrators will save time on almost every device-related management task, and that time will translate into some pretty significant monetary savings.

Automating network operations increases efficiency. Increased efficiency results in monetary savings. It's a pretty simple formula.

Your Network's Challenges

It's important to understand the challenges that your network faces. These are the things that make manual network operations so dangerous and can compromise a manual process and cost your business money, public confidence, and resources. Let's explore how each challenge can result in problems within a manually configured network and how automation of various kinds can help mitigate the challenge and remove the risk.

Poor Administrative Practices


Poor administrative practices are one of the biggest dangers faced by the network. Ad-hoc, poorly, or insufficiently planned changes are the ones most likely to result in downtime, insecure configurations, and so forth. The only way to eliminate poor administrative practices is to have a well thought-out configuration and change management process in place. Such a process helps to ensure changes receive the appropriate amount of planning and review to prevent problems. Of course, simply *having* a process doesn't guarantee its *use*; the problem of process adherence is a real one that I'll discuss shortly.

Another poor practice that challenges your network relates to security. Tasks such as patch management, password changes, SNMP community string changes, and so forth are all critical operations for a secure, reliable, compliant network, although many companies accept poor practices by not deploying patches in a timely fashion, failing to change passwords, and not changing SNMP community strings. The only way to improve these poor practices is to make a conscious effort to do a better job. Unfortunately, the risks associated with better practices—such as the risk of error when rolling passwords or the risk of failure when deploying a patch—can be significant. Automation can help reduce the risks by improving the consistency of changes and by providing a rapid and even automated recovery when a patch deployment goes wrong.

Technical Attacks

Your network is *always* at risk from technical attacks: Denial of Service (DoS) attacks, exploits of unpatched security vulnerabilities, and more. Best practices—such as frequent password changes, conscientious patch management practices, and so forth—can help reduce the likelihood of a successful technical attack. Automation makes these best practices easier to implement by helping make password changes and patch management more efficient, less error-prone, and more automatic.

Another way technical attacks often succeed is through improper configuration settings. For example, a new administrator might accidentally configure a router using a configuration template, similar to the one that Listing 1.1 shows. The use of such templates is normally a good idea, as they can be used to ensure consistent initial configurations of devices.

 This configuration is excerpted from a more complete one offered by Rob Thomas at <http://www.cymru.com/Documents/secure-ios-template.html>.

```

! Use TACACS+ for AAA. Ensure that the local account is
! case-sensitive, thus making brute-force attacks less
! effective.
aaa new-model
aaa authentication login default group tacacs+ local-case
aaa authentication enable default group tacacs+ enable
aaa authorization commands 15 default group tacacs+ local
aaa accounting exec default stop-only group tacacs+
aaa accounting commands 15 default stop-only group tacacs+
aaa accounting network default stop-only group tacacs+
tacacs-server host 7.7.7.5
tacacs-server key cheezit
!
! In the event that TACACS+ fails, use case-sensitive local
! authentication instead. Keeps the hackers guessing, and
! the router more secure.
username <USERNAME> secret <PASSWORD>
!
! Don't run the HTTP server.
no ip http server
no ip http server-secure
!
! Allow us to use the low subnet and go classless
ip subnet-zero
ip classless
!
! Disable noxious services
no service pad
no ip source-route
no ip finger
no ip bootp server
no ip domain-lookup

```

Listing 1.1: An example configuration template.

However, templates are static files often kept on individual administrators' workstations, making them unreliable as an authoritative configuration model. In this case, suppose the new administrator had obtained an older version of the template, not realizing that the TACACS+ server addresses had changed. Deploying this template would leave the device open to potential attack, as the old address could now possibly be used by an unauthorized TACACS+ server, thus providing a means by which an attacker could compromise the router's security.

An automation solution could help prevent this type of problem by analyzing all proposed changes against a set of centrally configured policies. When the configured TACACS+ server address doesn't match the solution's central policy, the change could be flagged or rejected outright, preventing it from being applied to the device and therefore preventing a potential attack. In fact, almost any type of configuration problem of this nature can be avoided through an automated configuration management solution that supports the use of policies as a check on new configurations.

Social Engineering

What is social engineering?

“Hi, this is the head of research. I need you to open an incoming firewall port for a project we're working on. I just need it open for a couple of hours. I'll send the authorization later. Can you do this for me now?”

Thus begins a social engineering attack: Exploiting the weak link—the people—in any process to bypass safeguards. An automated configuration management solution can, depending on the solution itself, help prevent this type of attack in a number of ways.

- **Process adherence**—By forcing administrators to make changes in accordance with a process, ad-hoc changes such as this become more difficult to make. The change can be tracked and rolled back, and the solution can be configured to require secondary authorization by a manager or peer. The greater number of people that have to be manipulated, the less likely the attack is to succeed.
- **Tie-in**—Tie-in is the idea of having a “closed-loop” configuration management system. Changes can only be deployed if they're matched to a request in a Help desk ticketing system, thus ensuring that requests such as this have to go through proper channels and thus reducing the opportunity for a socially engineered bypass to the process.
- **Policy enforcement**—If the configuration solution provides policy support, a central policy can be configured that details what changes are allowed, especially with regard to sensitive configurations such as open ports in a firewall. Changes outside the policy are rejected. A change request such as this one would have to start with a change to the policy itself, which typically wouldn't be accessible to a front-line administrator, thus defeating the social engineering attempt.

Physical Disaster and Other Elements Outside Your Control

All networks are subject to acts of nature: flood, fire, or simple failed electronics. Although no amount of automation can *prevent* these problems on your network, automation can help significantly reduce their impact. For example, in the event of a failure, an automated configuration solution can be used to quickly deploy a failed device's configuration to a backup device, restoring service much more quickly than a manual reconstruction could. In the event of a natural disaster, an automated configuration solution can be used to restore multiple devices' configurations to backup devices located at an offsite recovery facility, decreasing the time it takes to get the remainder of the business functioning at the backup facility.

Whether your network fails because of an earthquake or because of a simple, temporary failure in utility power, having an automated network configuration management solution can make the recovery process much faster. The automated solution can be configured to routinely back up device configurations, ensuring that your configuration repository *always* has the latest and greatest operational configuration—something that manual configuration management processes often fail to do consistently—and can push out configuration files to dozens or hundreds of devices simultaneously, vastly reducing the time it takes to get the network back up and running.

Process Adherence

Many of the challenges I've discussed to this point can be solved, or at least highly mitigated, though a well thought-out process. But processes only go so far; a flowchart alone can't force your network to become more secure, more efficient, and more reliable. In this arena, an all-in-one configuration management solution can be a benefit for network automation: A good solution can *enforce* a process.

For example, imagine that your network devices are configured with passwords that aren't known to your administrators and engineers. Instead, only your configuration management solution "knows" the passwords (perhaps written copies are kept in a fire safe as a form of insurance). Thus, administrators *can't* exercise poor practices by directly configuring devices; they're *required* to go through the configuration management solution because only it has the passwords needed to implement changes. Thus, your entire process is *enforced*, ensuring that appropriate change reviews and approvals occur, that any configuration policies you've created are enforced, and that scheduled maintenance windows are properly utilized. Out-of-process changes become difficult, if not impossible, ensuring that your process is able to do its job of protecting the network and, therefore, the business (see Figure 1.2).

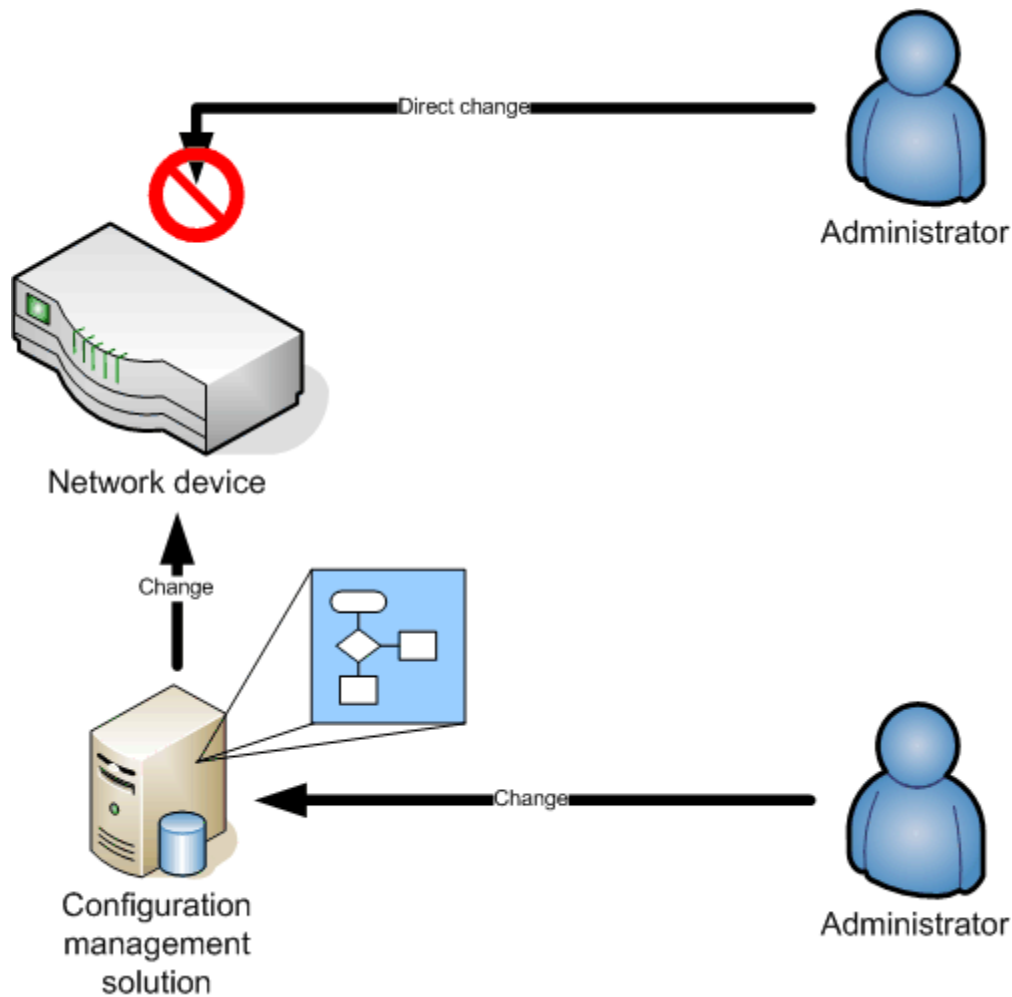



Figure 1.2: An effective solution can enforce a configuration management process.

 Think of how much easier auditing would be if you could prove that all configuration changes took place within your solution (something easy to prove if it's the only thing with the correct passwords to actually make changes on devices) and that the solution was configured to enforce required configuration settings. Auditors would simply need to verify the policies you had set up in the solution and they would be well on their way to finishing the audit.

The Benefits of an Automated Network

Thus far, we've looked at some of the costs of a non-automated network, and some of the challenges and risks that your network must face every day. Let's build on this foundation of knowledge by exploring some of the specific benefits of a fully automated network. Remember, for the most part, these benefits are ones that can *only* be achieved through automation of some kind—you won't really have them if you're running your network manually.

Configurable

Above all, an automated network is highly *configurable*, meaning it can be changed and reconfigured as quickly as your organization requires, while maintaining a safe, controlled configuration management workflow. Despite the fact that a process of any kind is often viewed as a hindrance, an effective configuration management solution can implement a process almost invisibly. In fact, by providing an application through which the workflow takes place—where changes can be submitted and scheduled, changes can be reviewed and approved, and so forth—the process is often made easier because everything a person needs to participate in the process is readily available.

Configuration management solutions—useful ones, at least—can also help prevent change *conflict*. For example, consider the timeline of changes shown in Figure 1.3.

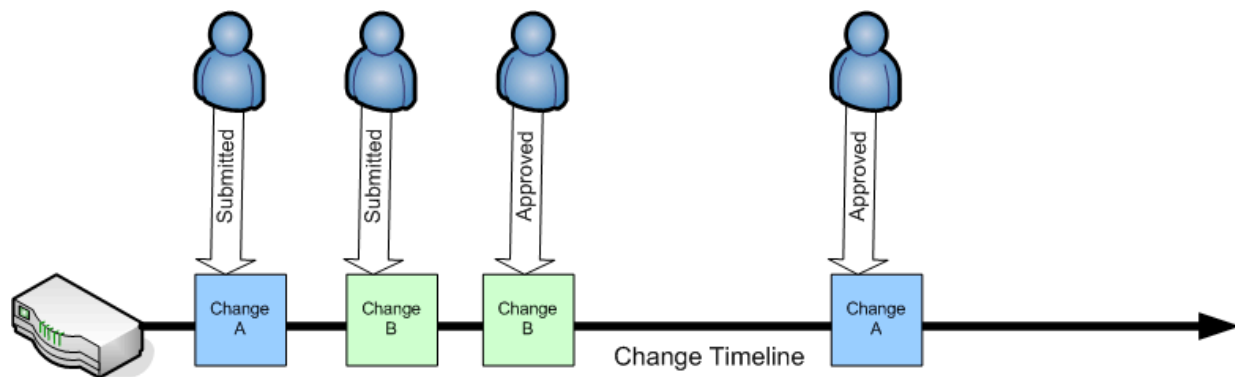


Figure 1.3: Timeline of changes to a device.

This figure illustrates two users who submit changes. Suppose that each change modifies the same portion of the router configuration. Because Change A isn't in effect at the time Change B is developed, Change B is based on the router's current configuration, just as Change A is. Change B is then approved for deployment *before* Change A (perhaps Change B is a higher-priority change). You now have two changes in the queue waiting for deployment, and they're going to conflict with one another. An effective configuration management solution can help resolve this conflict in one of two ways:

- Notify the person submitting Change B that another pending change affects the same area of the device configuration.
- Notify the person submitting or approving Change A that another change (Change B) has been approved that affects the same area of the device configuration.

Particularly in large environments, this sort of conflict resolution can help prevent mistakes and ensure that devices retain a high degree of configurability.

Consistent

Perhaps one of the most important benefits an automated network offers is consistency. When a configuration management solution is used to manage devices, configurations can be designed *once*, entered *once*, approved *once*, then deployed consistently to all appropriate devices. The key is that the manual portion of the process is performed only *once*, reducing the chance for error, and even that one manual step can be reviewed for accuracy.

An effective configuration management solution goes even a step further and helps to eliminate error-prone manual configuration. For example, a solution might allow you to configure a device in a test lab, then, when you've got the configuration working just the way you want, capture that configuration to use as a model for configuring other devices. This process creates a high level of consistency and a very low rate of error, helping to improve the network's overall reliability and security.

Consistency is the key to so much of a well-run network:

- **Security**—When devices are configured consistently, the overall network is more secure because no one device represents a potential weak point due to a poor, inconsistent configuration. For example, if you can imagine an administrator trying to manually deploy even a short access control list (ACL) change similar to the following example, you can also imagine how many errors could creep in:


```
access-list 100 remark VTY Access ACL
access-list 100 permit tcp host 7.7.7.5 host 0.0.0.0 range 22 23
log-input
access-list 100 permit tcp host 6.6.6.1 host 0.0.0.0 range 22 23
log-input
access-list 100 deny ip any any log-input
```
- **Manageability**—Consistently configured devices are easier to manage because they all follow the same conventions; administrators don't have to spend as much time examining configurations before proposing new changes because devices can be relied upon to have the same basic configuration.
- **Reliability**—Because devices with consistent configurations use a single, tested configuration, they are less likely to have configuration errors that result in network downtime.

It sounds simple, but consistency is probably the most desirable trait in any network, and an automated network is pretty easy to keep consistently configured.

Recoverable

Some disasters are beyond your control to prevent or even anticipate—loss of utility power, natural disasters, and so forth. But even smaller disasters—a failed device, a missed configuration error, and so forth—can result in network downtime. A *recoverable* network can bounce back from these problems quickly, and automation can provide that capability. Because

configuration management solutions can provide a complete configuration repository of *every* past version of *every* device's configuration, a solution can be used to restore any one of those configurations, either to the original device—in the case of a misconfiguration—or to a backup device—in the case of a device failure or natural disaster.

An effective configuration management solution grabs not only periodic scheduled backups of devices' configurations but also point-in-time backups in reaction to a change in the device's configuration. For example, a solution might monitor syslog or TACACS+ accounting traffic, or even SNMP traps, to notice when a device's configuration may have changed (a syslog entry indicating that the device was placed into configuration mode, for example, is a good clue that a change has occurred). The solution can use that cue to grab the device's configuration, essentially providing an automatic backup every time a change occurs, or could have occurred, to a device. You can then roll back that device's configuration to any point in time, which is truly the ultimate in recoverability.

Securable

A secure network is a goal for most organizations. How does automation improve security? Consistency is one way. Another way is through policy enforcement. Consider Figure 1.4.

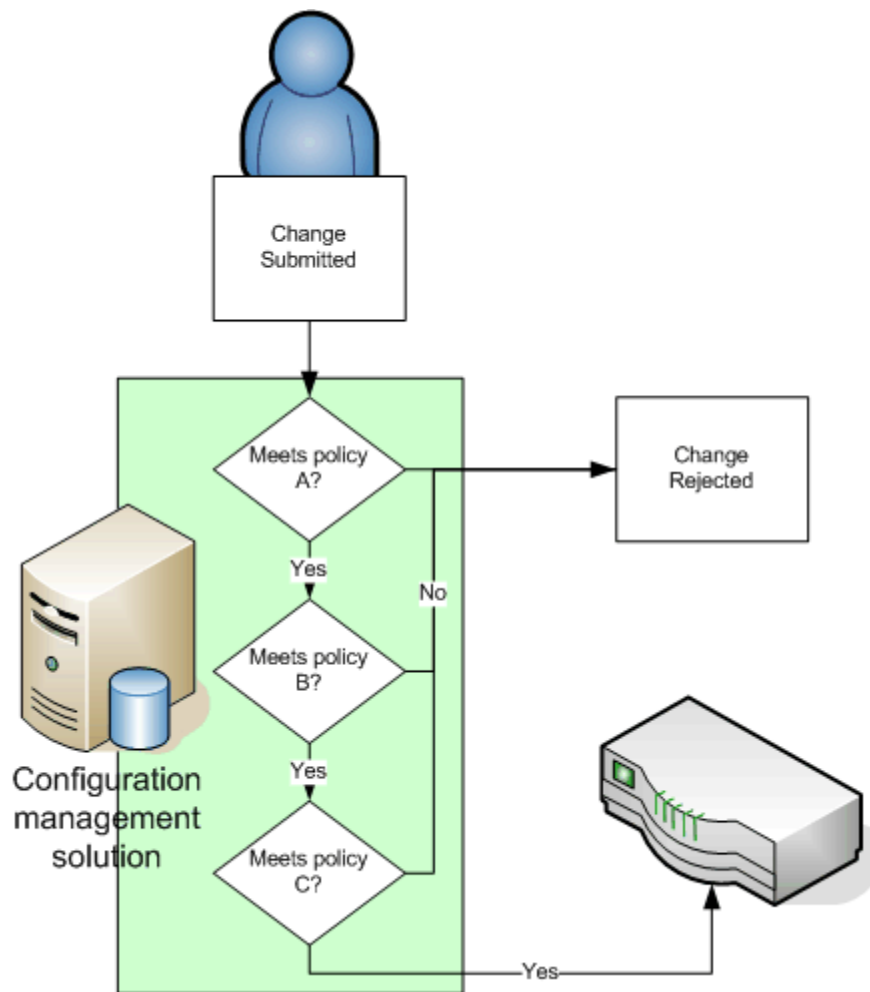


Figure 1.4: Policy-based management.

Because security-related configuration settings are so important, a configuration management solution can be configured to proactively enforce policies. In other words, in addition to scanning existing device configurations for policy compliance, a solution might also review all submitted changes. Changes that meet all configured policies are cleared for deployment; changes that don't meet all configured policies are rejected before ever making it to a device.

Configuration management solutions can also provide much more granular security than network devices can offer. For example, a configuration management solution might be able to allow entry-level technicians to perform basic operations such as restarting a device, while only more experienced technicians are allowed to submit configuration changes to the device. Auditors might be able to access devices' configurations in a read-only fashion. This role-based security helps to automate and control access to device configurations, also improving the overall security of the network.

Compliant

Security isn't *quite* the same thing as compliance. True, most compliance legislation concerns itself with security, so the two are definitely tightly intertwined, but they're not precisely identical. In a secure network, you've simply got everything configured to be however secure you want it; for a *compliant* network, you have to be able to prove it. In other words, compliance is almost a union between having a secure and auditable network.

Typically, compliance management works like this: You assemble a checklist of configuration parameters that, if implemented, result in your network complying with whatever rules and regulations you're required to comply with—the Health Insurance Portability and Accountability Act (HIPAA), the Sarbanes-Oxley Act, CISSP, and so forth. Typically, you also need to have a process that prevents unauthorized changes to these critical configuration areas. You must then be able to demonstrate your compliance—prove that the correct configurations are in place, that they *have been in place*, and that, thanks to your process, they're likely to remain in place in the future.

How can network automation help with compliance? In several ways:

- Policy-based management can be used to enforce key configuration parameters, either by automatically alerting you to unauthorized changes or, better yet, automatically remediating unauthorized changes.
- An enforced workflow ensures that unauthorized changes can't occur. If all changes have to go through a configuration management solution, and if that solution enforces a workflow, you will remain in compliance.
- Configuration management solutions that automatically grab backups of device configurations can report on any unauthorized changes. This process “closes the loop” on compliance: With policy-based management and an enforced workflow, unauthorized changes should be impossible; reports based on analysis of actual device configurations proves that unauthorized changes haven't occurred.

Auditable

Any network is auditable. The difference is that an automated network is *easily* and *continuously* auditable. In a manual network, auditing requires that you dump each device's configuration and then manually examine it. As with manual configuration, this process is error-prone. Imagine reading through a 300-line configuration file—you're going to miss a few details; multiply this example by hundreds of devices and the review quickly becomes an almost pointless exercise.

With automation, this process is much easier. An effective configuration management solution, for example, will allow you to define the specific configuration parameters that you *want* to see, then generate an exception report of all device configurations that *don't* meet your requirements. Instant audit. What could take weeks to perform manually can literally be done in seconds, because the configuration management solution doesn't need to connect to the actual network devices; it can simply run the report from its own configuration repository, which contains all the devices' current configurations.

Efficient

Need to roll out a change to 500 devices? Need to change passwords on 1000 devices? Need to deploy a patch to 100 devices? In a manual network, you would need to calculate the time it takes to perform the task once, then multiple that time by the number of devices. The numbers add up fast. In an automated network, however, you perform the task only *once*. The automation solution performs it however many hundreds of other times you might require. The savings in time is truly amazing.

In addition, a configuration management solution can perform the mundane tasks associated with a change but not directly part of it—backing up the device beforehand, backing up the device again after the change is applied, ensuring that configuration policies are not violated by the change, and so forth. A well-automated network can be managed with a fraction of the manpower a manually run network requires, freeing up staff for other projects or helping to reduce staff and outsourcing requirements.

Flexible

Finally, one of the most important—and yet often overlooked—benefits of an automated network is flexibility—the business flexibility offered by frameworks such as IBM OnDemand and Hewlett-Packard Adaptive Enterprise; the ability to quickly reconfigure your business to meet new competitive challenges or business needs. With the ability to quickly reconfigure the network in a secure, compliant, consistent fashion, and with the ability to quickly roll back changes in a moment if required, making changes to the network is no longer a great undertaking. The massive risk formerly associated with making changes to the network is highly mitigated by your automated management capabilities, meaning you can bend, flex, and reconfigure your network in whatever way best suits your current business needs. This ability is a major contribution to the long-term profitability and success of any company.

Traditional Management vs. Automated Management

To wrap up this chapter, let's take a moment to compare and contrast the three major forms of traditional network management with management in an automated network, highlighting the specific differences and benefits.

Ad-Hoc Configuration

Traditional network device management is almost entirely ad-hoc, meaning that changes are made on a per-device basis whenever changes are needed. This is a poor practice, although in many cases, it is done simply because that is how network devices are designed to be managed. The reason ad-hoc management doesn't work is that it is applied inconsistently. For example, it's a common security practice to configure routers with black hole routes rather than using TCP Intercept. In a Cisco device, you would add text similar to the content that Listing 1.2 shows to the configuration file.

```
ip route 1.0.0.0 255.0.0.0 null0
ip route 2.0.0.0 255.0.0.0 null0
ip route 5.0.0.0 255.0.0.0 null0
ip route 7.0.0.0 255.0.0.0 null0
ip route 10.0.0.0 255.0.0.0 null0
ip route 23.0.0.0 255.0.0.0 null0
ip route 27.0.0.0 255.0.0.0 null0
ip route 31.0.0.0 255.0.0.0 null0
ip route 36.0.0.0 255.0.0.0 null0
ip route 37.0.0.0 255.0.0.0 null0
ip route 39.0.0.0 255.0.0.0 null0
ip route 42.0.0.0 255.0.0.0 null0
ip route 49.0.0.0 255.0.0.0 null0
ip route 50.0.0.0 255.0.0.0 null0
ip route 77.0.0.0 255.0.0.0 null0
ip route 78.0.0.0 255.0.0.0 null0
ip route 79.0.0.0 255.0.0.0 null0
ip route 92.0.0.0 255.0.0.0 null0
ip route 93.0.0.0 255.0.0.0 null0
ip route 94.0.0.0 255.0.0.0 null0
ip route 95.0.0.0 255.0.0.0 null0
ip route 96.0.0.0 255.0.0.0 null0
ip route 97.0.0.0 255.0.0.0 null0
ip route 98.0.0.0 255.0.0.0 null0
ip route 99.0.0.0 255.0.0.0 null0
ip route 100.0.0.0 255.0.0.0 null0
ip route 101.0.0.0 255.0.0.0 null0
ip route 102.0.0.0 255.0.0.0 null0
ip route 103.0.0.0 255.0.0.0 null0
ip route 104.0.0.0 255.0.0.0 null0
ip route 105.0.0.0 255.0.0.0 null0
ip route 106.0.0.0 255.0.0.0 null0
ip route 107.0.0.0 255.0.0.0 null0
ip route 108.0.0.0 255.0.0.0 null0
ip route 109.0.0.0 255.0.0.0 null0
ip route 110.0.0.0 255.0.0.0 null0
ip route 111.0.0.0 255.0.0.0 null0
```

```
ip route 112.0.0.0 255.0.0.0 null0
ip route 113.0.0.0 255.0.0.0 null0
ip route 114.0.0.0 255.0.0.0 null0
ip route 115.0.0.0 255.0.0.0 null0
ip route 116.0.0.0 255.0.0.0 null0
ip route 117.0.0.0 255.0.0.0 null0
ip route 118.0.0.0 255.0.0.0 null0
ip route 119.0.0.0 255.0.0.0 null0
ip route 120.0.0.0 255.0.0.0 null0
ip route 121.0.0.0 255.0.0.0 null0
ip route 122.0.0.0 255.0.0.0 null0
ip route 123.0.0.0 255.0.0.0 null0
ip route 127.0.0.0 255.0.0.0 null0
ip route 169.254.0.0 255.255.0.0 null0
ip route 172.16.0.0 255.240.0.0 null0
ip route 173.0.0.0 255.0.0.0 null0
ip route 174.0.0.0 255.0.0.0 null0
ip route 175.0.0.0 255.0.0.0 null0
ip route 176.0.0.0 255.0.0.0 null0
ip route 177.0.0.0 255.0.0.0 null0
ip route 178.0.0.0 255.0.0.0 null0
ip route 179.0.0.0 255.0.0.0 null0
ip route 180.0.0.0 255.0.0.0 null0
ip route 181.0.0.0 255.0.0.0 null0
ip route 182.0.0.0 255.0.0.0 null0
ip route 183.0.0.0 255.0.0.0 null0
ip route 184.0.0.0 255.0.0.0 null0
ip route 185.0.0.0 255.0.0.0 null0
ip route 186.0.0.0 255.0.0.0 null0
ip route 187.0.0.0 255.0.0.0 null0
ip route 192.0.2.0 255.255.255.0 null0
ip route 192.168.0.0 255.255.0.0 null0
ip route 197.0.0.0 255.0.0.0 null0
ip route 223.0.0.0 255.0.0.0 null0
```

Listing 1.2: Example addition to a configuration file.

That is a long list. Although you can add this content to a template, it's easy to enter a mistake when you're configuring devices manually. In an automated network, however, you might use a configuration management solution to make these settings policies. The solution could also have a job that applies these settings, and the policy could be linked to the job for remediation. The solution would then simply scan every appropriate device to make sure these settings are in place, per the policy; if they weren't, the solution would run the job that puts these in place. You would only configure these settings once, and they would be consistently deployed to appropriate devices *automatically*. If these settings ever change, you would simply change your policy and the remediation job; the solution would then take care of reconfiguring devices to match the new policy.

Ad-hoc changes have another problem—a lack of workflow. Although your company may have a configuration management process for the network, that doesn't mean it's used each and every time a change needs to be made. With an automation solution in place, however, that workflow can be *enforced*, improving the network's reliability and accomplishing everything that your workflow was intended to provide.

☞ In organizations that must meet compliance requirements, such as the Sarbanes-Oxley Act, HIPAA, the Graham-Leach-Bliley Act, and so forth, enforcing a configuration management workflow is typically a requirement that helps you remain compliant. If auditors feel that there is a way for your configuration management process to be bypassed, they may not feel you're capable of remaining compliant. Auditors will fail companies simply because there wasn't an *enforceable* workflow in place.

Manual Backup and Recovery

Backup and recovery are two elements of network administration most often performed poorly when done manually. The very act of logging into every device and commanding it to dump its configuration—typically to a Trivial File Transfer Protocol (TFTP) server—is time-consuming and incredibly inefficient. If you have 800 devices, and it takes you just 3 minutes to back up each one, you're looking at a full 40 hours to get them all. How often are you really going to do that? Plus you're forced to manually manage a configuration repository, ensuring that past backups are kept. If you change a device's configuration, you have to remember to create a new backup—will you always do that when you're in a hurry?

Backup and recovery are often the first two things organizations automate when it comes to network device management. Numerous inexpensive solutions exist for automating basic backup and recovery; these solutions can usually be configured to automatically take a new backup on a scheduled basis and to maintain a repository of configuration backups. Of course, this step is only one tiny part of a fully automated network; all-in-one configuration management solutions provide this automation functionality along with much, much more.

Automated *recovery* is something else. This functionality isn't typically provided by low-end automated backup solutions; instead, automated recovery is usually provided as a configuration management feature. Essentially, a configuration management solution is used to detect changes to devices—often by monitoring SNMP traps or syslog output—and to automatically roll back the device's configuration (in other words, restore the configuration) to the last-known-good version when unauthorized or inappropriate changes are detected. This option is a powerful security feature. If, for example, an administrator or attacker modifies a device directly, the solution can put back the device to its authorized, configuration-managed version automatically.

Scripted Administration

Scripts are the first tool most administrators turn to for basic automation needs. For example, at <http://forums.vandyke.com/showthread.php?t=370>, you'll find a script designed to automate the backup of device configuration files. This example is a fairly complex script capable of handling different types of devices, including Cisco IOS, CatOS, and PIX devices.

Administrative scripts are a huge part of network device management, and, in fact, most high-end configuration management solutions incorporate scripting to some degree, often generating scripts based on activity so that changes can be deployed consistently to multiple devices. However, manually created and managed scripts—although they offer a basic degree of automation—come at a high price. Because network administrators aren't often trained software developers, their scripts are often difficult for someone else to use and maintain, thus creating a dependency on the person who wrote the script—a problem if that person leaves the company, goes on vacation, or moves on to other duties. As a form of administrative automation, scripts *work*, but they often don't work as well—depending, of course, on who wrote the script—as a professionally designed solution.

Summary

This chapter has focused on the benefits of an automated network as well as the specific costs your business is facing by *not* using automation in your network operations. It also looked at some of the challenges your network faces, and discussed the ways in which automation can help to mitigate or remove those risks. The next chapter will jump right into how automation works on a more technical level, and how automation can improve your network's security and compliance position.

Chapter 2: Automating Network Compliance and Security

One of the biggest drivers behind the adoption of network automation technologies is the need for companies to improve their security and compliance postures. Security and compliance have become a major new requirement for most companies—any publicly traded company in the United States, for example, must comply with the provisions of the Sarbanes-Oxley Act—and the public and in-house attention devoted to compliance and security issues has become significant.

What Does the Network Have to Do With Compliance?

Most compliance legislation deals with considerations such as corporate accountability, customer privacy, and so forth; it's not always immediately apparent how the network fits into the compliance picture. However, the network is the very basis of all information transmission in your organization—every file on a file server, every record in a database, and every Web page on a Web server is transmitted across your network. Network devices can include items such as firewalls and proxy servers, intrusion detection (or prevention) devices, and other elements that have an obvious impact on security. If a single router on your network is compromised, an attacker could conceivably route all traffic from that router to an unauthorized destination, exposing all your corporate data and creating a security—and potentially a compliance—breach. Almost every other data security measure your organization takes—securing file servers, locking down database access, securing ports on servers and client computers, and combating viruses and spam—is useless if the network becomes compromised. Because the network is used to transmit nearly all data within the organization, the network is the final point for compliance and security control.

How Legislation and Rules Affect the Network

Compliance begins with basic security. In general, legislation such as HIPAA, the Sarbanes-Oxley Act, and the Graham-Leach-Bliley Act, as well as rules like Visa's Cardholder Information Security Program (CISP) merchant requirements, focus on the security and privacy of customer data. They also focus heavily on accountability, meaning you're required to provide an audit trail of some kind so that all access to data, whether legitimate or not, is logged and can be reviewed at any time. At first glance, these broad requirements seem well-suited to a file server or database server, where data can be protected with access control lists (ACLs), data access can be audited, security logs can be aggregated and used to report on data access, and so forth. However, all of this access occurs over the network. Consider the functional diagram that Figure 2.1 shows.

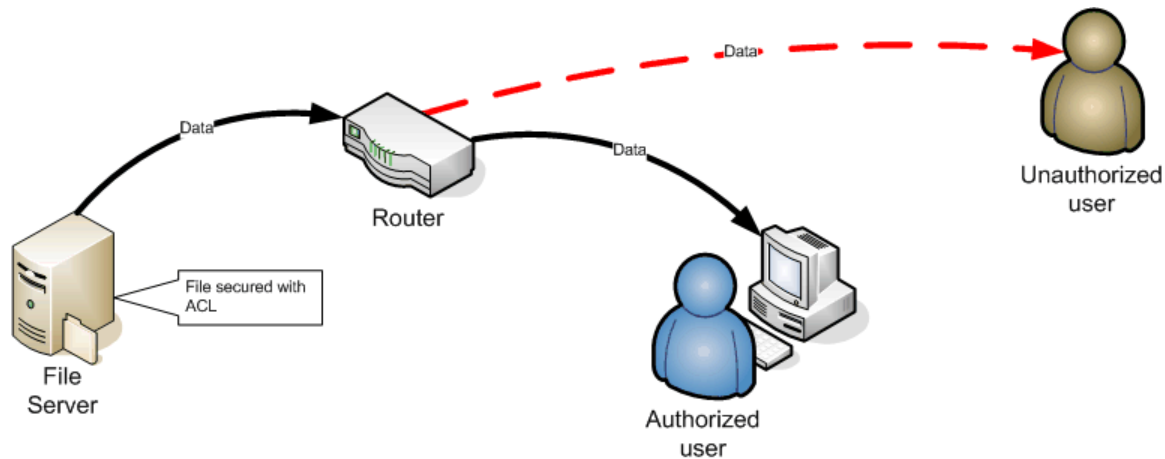


Figure 2.1: Functional diagram of data access.

In this figure, data on a file server is properly secured with an ACL. Auditing could also be configured so that data access can be tracked and reviewed. An authorized user accesses the data, which is transmitted to his or her computer across the network. However, in this example, the router's configuration has been modified by an unauthorized user, who is having the router send a copy of the information directly to him or her. The file server is unaware that this activity is occurring, so the unauthorized access isn't logged by the file server; the access, in fact, bypasses the server's security completely. In this way, the network is affected by compliance rules and legislation—the network offers a means of bypassing the more obvious security and auditing precautions you've put in place, which means the network can allow the activity that the rules and legislation are in place to stop.

Interestingly, compliance rules and legislation tend to say very little about how your environment should be configured, and instead focus on the end result. In other words, you won't find a legislative body passing laws requiring that file servers be configured to audit data access. Instead, the laws simply state (more or less) that *all* access to data must be logged. Thus, unauthorized access makes you non-compliant because you've failed to log a particular access to data. That means your network becomes a huge non-compliance liability, with all the associated financial and legal penalties. As outlined in the previous chapter, manually configured networks are highly susceptible to a number of conditions that make them more likely to be non-compliant.

How Network Operations Affect Compliance

Daily network operations—day-to-day configuration changes, for example—can have a significant impact on your network’s state of compliance. Remember, the compliance laws and rules don’t care about the actual state of your network’s configuration; they only care about the end result of that configuration. For example, suppose an administrator temporarily opens a firewall port to allow for a one-time activity—perhaps the administrator wants to play a game of networked Doom with a friend. The administrator closes the port when the game is over and puts everything back to the way it was. If nothing untoward happens during this period, you’re fine and you’re still compliant. However, if the administrator’s actions allow some private data to be disclosed without maintaining accountability, you’re non-compliant—you’ve broken the rules, and you may be subject to the consequences (which, of course, can extend far beyond mere fines or penalties; public disclosure of security breaches can cause significant market and financial harm).

In another example, suppose a new device is added to your network and configured according to a basic configuration template. However, the latest version of the device’s OS was not installed, perhaps leaving it vulnerable to attack—an attack which could compromise your compliance. The network technician logs the activity, but no mention is made of the installed OS version of the device in the report. This example highlights the fact that a manually configured and administered network is nearly impossible to properly maintain from a compliance and security point of view. Too many small changes or oversights can cause significant security harm, and those small changes and oversights are endemic to a manually configured environment.

How Networks Become Non-Compliant

In the end, nearly every action that happens or *doesn’t* happen to your network can affect your state of compliance. Making a small, improper change to a Simple Network Management Protocol (SNMP) ACL in a router, for example, can result in unauthorized, unaudited data access. Failing to apply a patch can result in the same outcome. A network in stasis—that is, one that doesn’t receive the latest patches and secure configuration changes—is sure to be a compliance problem eventually; conversely, a network in flux—one in which configuration changes are being made—is just as susceptible to becoming a security vulnerability. You can afford to neither leave the network nor make changes. *Manually*, that is: manual change of any kind is what ultimately leads to errors, inconsistencies, poor practices, and oversights—all of which lead to compliance and security problems.

Sometimes the path to non-compliance can be circuitous: Suppose an administrator deploys a new device and fails to change the configuration template’s SNMP community string settings. Because the template might be accessible to a broader range of people (after all, it’s just a text file with no special security significant in and of itself), that default SNMP community string might be well-known. That information can be used to reconfigure portions of the new device, thus compromising the device. Once compromised, the device can become a gateway for bleeding data off the network unnoticed or for introducing malicious software into the network. A seemingly innocent mistake—simply failing to change a text string consisting of a dozen characters or so—could result in an entire organization accidentally disclosing sensitive data with no accountability—a double hit on compliance.

The key, then, is to remove the manual configuration and administration from the loop. Your organization probably has, or is at least developing, business processes designed to ensure that changes don't create problems, and that the network is properly administered. Automating and enforcing that process is the way to a secure, compliant network.

How Automation Affects Security and Compliance

Automation is a concept that many IT managers and senior administrators hear constantly but don't always appreciate. They often think they know what automation is—scripts, written by administrators, to help make configuring multiple devices easier. That is certainly *one* form of automation, but frankly, it's the least-useful form. Although scripts are certainly better than manually typing configuration settings into devices, they still present many of the same problems as fully manual network management. If you must run the script manually, the process is still manual. True automation, however, goes far beyond mere scripts. And it's not just a marketing term invented by the companies who offer automation solutions; automation offers true business benefits that have a marked, positive impact on security and compliance.

How Automation Improves Daily Administration

Automation—that is, fully automated network configuration management solutions—can improve day-to-day network administration in a number of ways. Specific to security and compliance, automation can:

- Help ensure that no changes are made that violate specific compliance-sensitive portions of the network.
- Help ensure that changes are made consistently across devices, improving security.
- Help ensure that changes are made only when approved by a business process (which may include elements such as peer review), reducing errors that can compromise security or compliance.
- Help ensure that multiple changes to a device don't conflict by queuing changes and notifying change developers if the device's starting configuration is no longer the same as when a change was originally developed.
- Enable rapid deployment of changes to network devices.
- Allow illustration of ineffective or dangerous security changes before they are implemented.

For example, consider the diagram in Figure 2.2, which illustrates one way in which a configuration management solution might enforce security and compliance rules. As an example, suppose that the proposed change is modifying the SNMP configuration of a device, changing its read-write community string to Private. The solution pulls the device's current configuration, applies the change, then compares the new configuration with its database of configuration rules and policies. Such rules would often include a prohibition for the default community strings, such as Private, so the change would be rejected automatically and never applied to any device. The change to Private may have been accidental; it's possible the administrator creating the

change used a template file and just neglected to change that setting in the template. Regardless, the automation software caught the change before it was deployed, thus helping to maintain the security and compliance of the network.

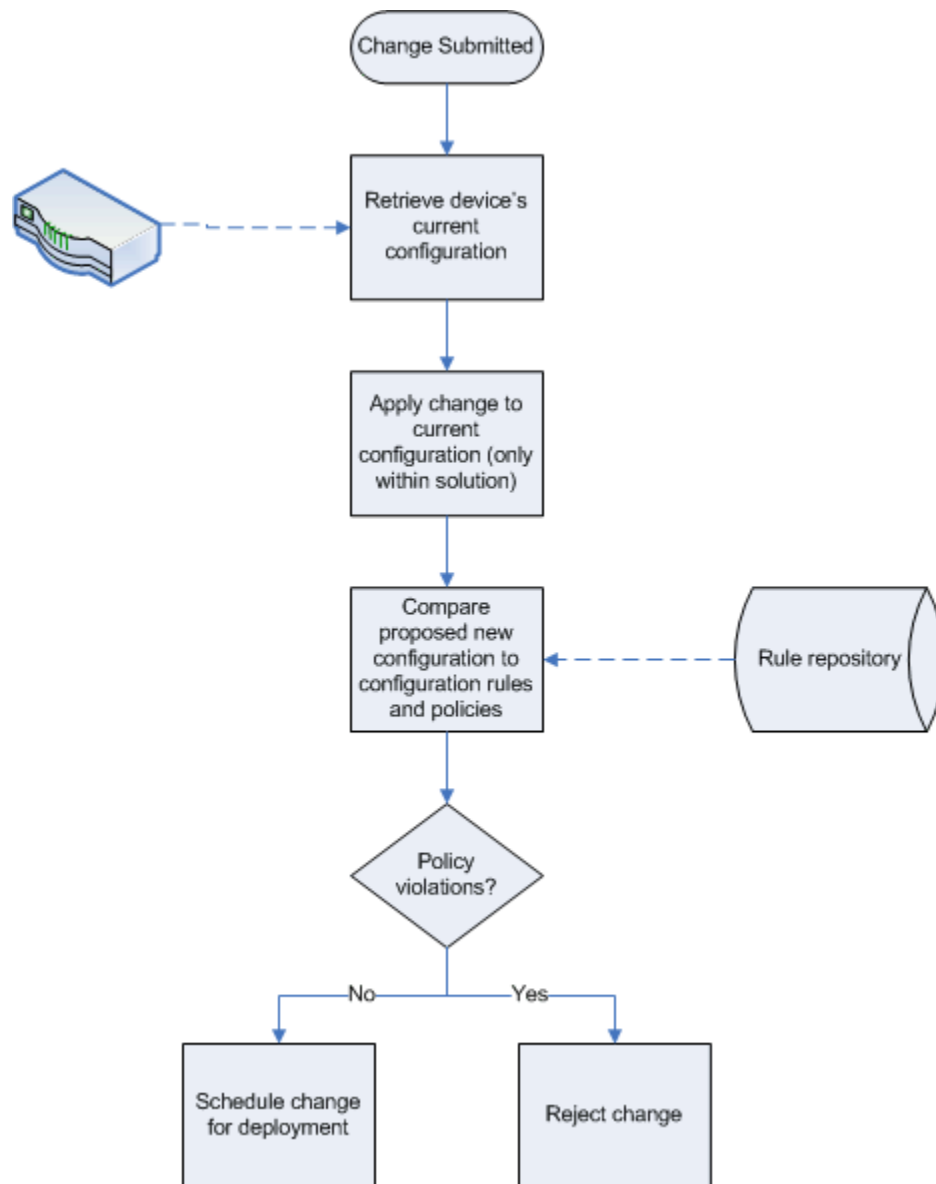


Figure 2.2: Automated change review process.

How Automation Adds Accountability to Network Operations

Accountability—being able to trace who made what change or who accessed which data—is a core concept of most compliance rules and legislation, particularly the Sarbanes-Oxley Act, the Gramm-Leach-Bliley Act, and HIPAA. Network devices, however, have notoriously poor accountability built-in. Even when configured to use a logging technology such as TACACS+, RADIUS, or Syslog, network devices don't provide a high level of granularity. They might log

an event indicating that an administrator logged in, but don't expect your Syslog database to include details about what the administrator changed while logged into a device.

A much more granular level of accountability, however, can be achieved with a network configuration management solution:

- By performing all device administration through the solution, the solution can keep a detailed database of who made what changes, and who queried what data, from devices. Solutions typically implement their own security architecture, forcing users to identify themselves and tracking, in great detail, what each user does within the solution.
- Even traditional Telnet- or SSH-style administration activities can be performed through a configuration management solution, typically through a pass through or proxy mode. This ability allows the solution to log each keystroke from each administrative session, capturing the full scope of the session's activity and relating that activity to a particular user.

Figure 2.3 shows how this fine-grained auditing capability helps provide accountability that standalone network devices simply can't offer.

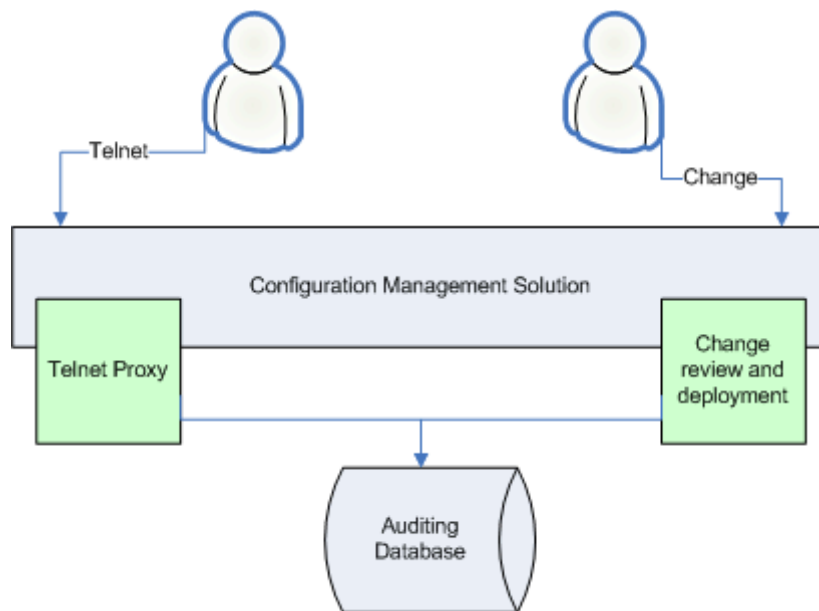


Figure 2.3: Auditing through a configuration management solution.

In fact, if your organization is subject to any accountability requirements—whether from internal governance rules or from legislation such as HIPAA, the Sarbanes-Oxley Act, and so forth—better accountability is the single best reason to implement a network configuration management solution. Without one, there is almost no way to provide fine-grained accountability and activity auditing for network devices.

👉 Imagine being able to print a single auditing report that shows every single network device management action taken within a given timeframe. If you've ever been through a security or compliance audit, you know that this is one of the first things an auditor wants to see; if you can provide it, you've just made a potentially painful audit that much easier.

How Automation Ensures that the Network Remains Compliant

As mentioned earlier, a network configuration management solution can help a network remain secure and compliant by pre-approving day-to-day configuration changes using a preconfigured set of rules and policies to spot configurations that might violate compliance-related or security standards. Some network configuration management solutions go a step further by providing automated remediation capabilities (see Figure 2.4).

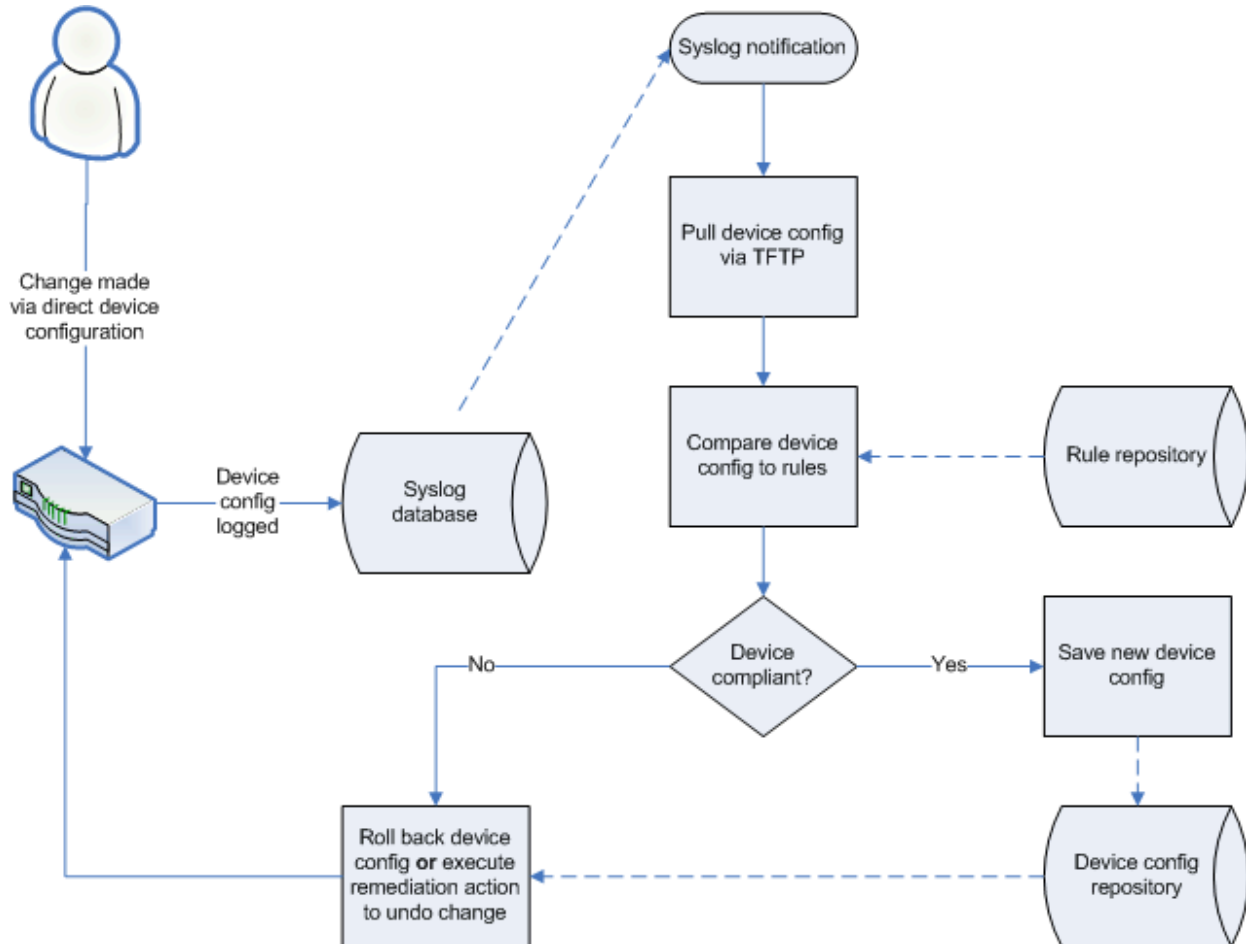



Figure 2.4: Automated remediation.

In this example, an administrator has made a configuration change to a device directly without using a network configuration management solution. That change—actually, the fact that the device was placed into configuration mode—triggers a Syslog (or TACACS+, or RADIUS, or SNMP trap) event. The network configuration management solution is configured to notice this event, and it triggers a configuration review process within the solution.

 Even without the event, the configuration management solution might be configured to pull device configurations on a periodic basis, at which time it would notice the change and perform the configuration review.

In the configuration review, the solution compares the current configuration with the solution's various rules and policies. If everything's okay, the configuration is saved to a repository as the "last known good" version of that device's configuration. If not, however, the solution can trigger an automated remediation response. Broadly, the solution might simply roll back the device's configuration, undoing all the changes and perhaps notifying someone of the action via email or SNMP alert. More specifically, the solution might have a remediation action associated with each of its configuration rules, allowing it to granularly undo just the change that violates a configuration rule or policy. Either way, the network's compliance is assured because non-compliant configurations are fixed automatically, within a few moments after the non-compliant configuration change was made.

Some network configuration management solutions come with (or offer as an add-on) preconfigured "packs" of configuration rules and policies specifically tailored to a given compliance environment: the Sarbanes-Oxley Act, HIPAA, the Gramm-Leach-Bliley Act, CISP, and so forth. These "packs" of rules can help get you up and running much more quickly, and can help you quickly determine which devices on your network are compliant.

Another way in which automation helps maintain compliance is through fast reporting. Because network configuration management solutions maintain a copy of each device's current (and past) configurations in a repository, they can examine those configurations quickly without having to physically connect to each device. This functionality makes compliance reporting much easier—you can simply run a report that lists which devices do and do not meet your configuration rules and policies. If those rules and policies define everything a compliance auditor would be looking for, your audit just became very, very easy—run the report. If it doesn't list any exceptions, you're done.

In fact, combined with automated remediation, compliance audits can often become a matter of looking at your configuration rules and policies to make sure they're comprehensive. If you can show that those rules cover everything, and that all your devices meet the rules—and thanks to automated remediation have *always* met those rules—the audit is over.

Traditional Security and Compliance Loopholes

Even with some forms of automation—such as scripted configuration changes—it's possible (and even easy) for security and compliance problems to arise. The reason is that network management technologies provide a number of loopholes through which improper configuration settings can creep in. The next several sections examine these loopholes and discuss how they can negatively affect the overall security and compliance posture of your organization. A discussion will follow with several sections about technologies and tools that can close these loopholes and keep your network more secure and completely compliant.

Overlooked Managed Elements

One of the biggest holes in most networks is managed elements—network devices—that the organization isn't aware they have. It sounds silly, but in an environment with hundreds of devices it's pretty easy to overlook one or two located in, perhaps, a remote office that doesn't have its own IT staff.

Overlooked devices represent the single biggest threat on any network simply because they're ignored—they don't get patched, they don't get the latest configuration changes, and they may not even get audited. They're completely off the radar, and as such can be compromised without you even being completely aware of where the breach came from. Once compromised, of course, you're no longer compliant or secure.

Most people don't believe ignored devices exist on their network. However, almost every organization that employs a network with more than a couple of hundred devices cannot provide a complete, written device inventory that matches what is really on the network. There is *always* one or two devices missing—perhaps a forgotten ISDN gateway that is used as backup connectivity on a remote network or an old router in a remote office that is quietly humming along, doing its job, and not attracting any attention.

Other times, the overlooked devices are ones that never should have been there in the first place. A field engineer, for example, hooking up a router for testing purposes without telling anyone—then leaving it hooked up forever. Or rogue wireless access points connected in branch offices or in departments to provide wireless coverage that the company isn't officially providing. They are often connected by well-meaning employees who just want to connect their laptops. They do not know or understand WEP keys, hiding SSIDs, or even changing the default password to configure the device. They provide access to anyone with an 802.11 device within range of the device—whether they work for your company or not. *Any* of these can quickly become a security problem; if a compliance auditor runs across a device that you don't know about, your audit will not go well no matter how well the device is configured.

Unless you're using some form of automation—and many companies are, now, for device discovery if nothing else—then you probably have forgotten devices floating around on your network.

Point-in-Time Auditing Misses Changes

Compliance is not about auditing. Compliance is about meeting the letter of the law—ensuring that data isn't disclosed to unauthorized parties, ensuring that full accountability exists, and so forth. Auditing is just a spot-check to make sure you're doing it, but be very clear in your mind that, as a means of actually enforcing compliance—that is, ensuring that you're continuously obeying whatever laws or policies apply to you—auditing is nearly useless.

For example, a major East-coast telecommunications firm had a security policy regarding the frequency with which user passwords had to be changed—a common policy. However, the environment contained a number of disconnected systems, each with their own passwords, and users weren't exactly technical experts. Thus, the password change requirements were turned off. When an audit came—and the firm always knew at least a few hours in advance—the change requirement was turned on again. When the auditor left, so did the change requirement. In effect, the firm was compliant with its policies for however long the auditor was in the building, and passed every audit. Usefulness of audit? Zero.

The problem is that auditing represents a very small point in time, and networks change too much and too quickly for that point in time to have a practical purpose. Instead, organizations need to practice *continuous enforcement*. For example, if the telecommunications environment had included a configuration management solution, the auditor would have been able to look at a simple report and see that administrators had been tweaking the password change requirement setting, and that the firm hadn't been compliant for more than a few hours each year. In fact, the configuration management solution could have easily overridden the password requirement changes, reconfiguring the password change requirements every time they were turned off. Such a system would have enforced the company's policies continuously. Of course, back then such technologies weren't available—but they are now.

Direct Device Administration Causes Inconsistency

Manual administration—meaning administration through direct interaction with network devices—is the primary cause of inconsistency and inconsistency is a major factor in security and compliance problems. Direct device administration is simply too prone to manual error. This sometimes applies even to proxied device administration, where administrators actually use a network configuration management solution to Telnet or SSH into a device to administer it. Although the solution can log keystrokes for auditing purposes, many solutions don't analyze the session's contents to determine whether a security or compliance rule is being violated. Some solutions do—they're able to review changes as they're entered into the Telnet session, determine whether the changes violate any configuration rules, and prevent the change from actually being sent to the device (or at least warn the administrator that there is a conflict). However, no solution can completely prevent direct device administration from occasionally causing problems and configuration inconsistencies.

Inability to Tie Changes to Requests Reduces Accountability

At the core of any truly secure and compliant network is a comprehensive business process, such as the one shown in Figure 2.5.

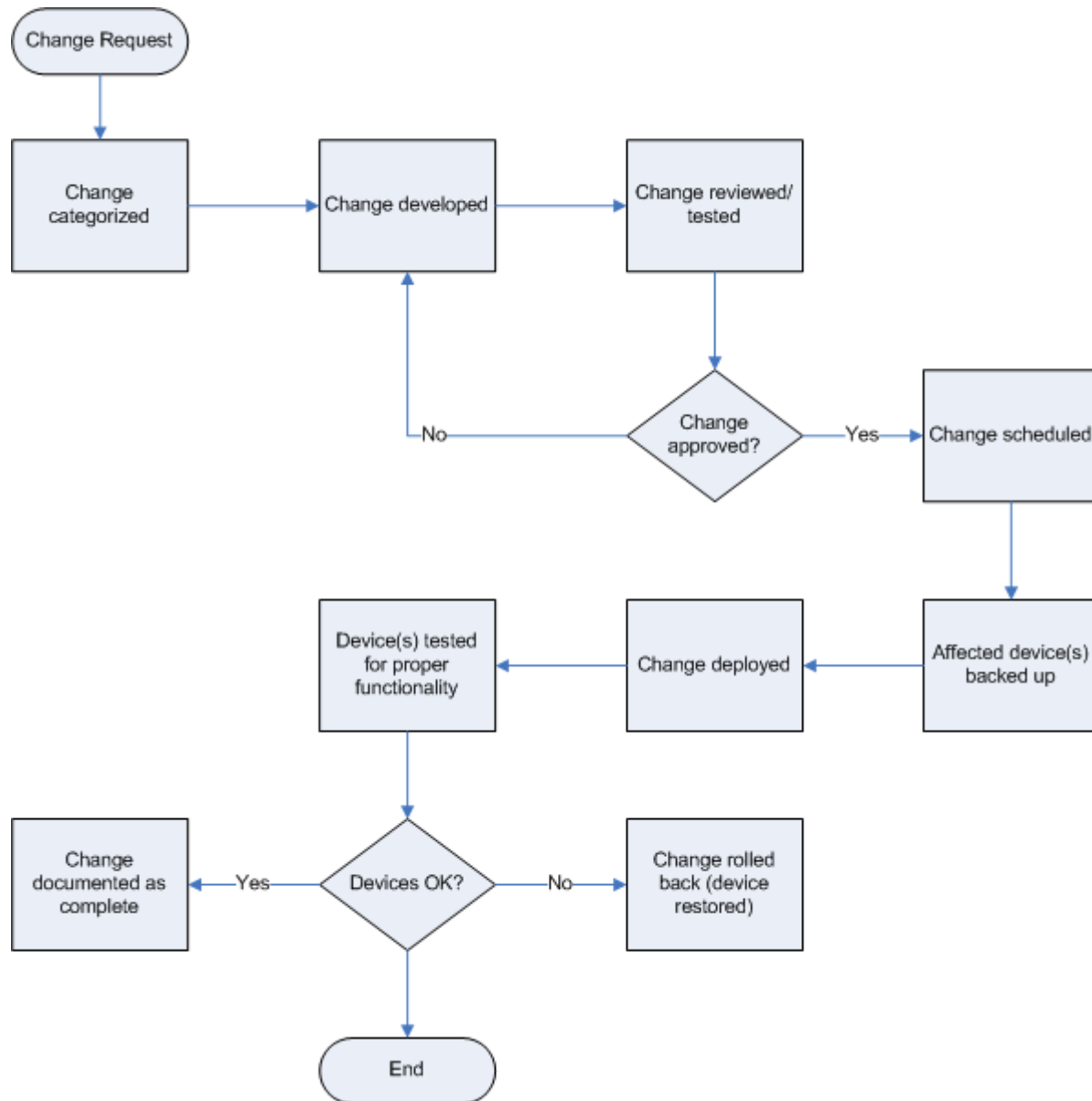


Figure 2.5: Sample configuration management business process.

Unfortunately, without some sort of automated system in place, it's too easy for changes to occur outside this process. Out-of-process changes are not inherently bad, but from a security and compliance standpoint, you want to always be able to relate any change to the original, in-process change request. That is, no change should exist without a corresponding change request, thus helping to ensure that changes have all made it through your change management process.

Inability to Enforce a Configuration Management Process Affects Compliance

Ultimately, it's *enforcement* that is the loophole. Without automation, it's too easy to bypass whatever safeguards your organization may have put in place to prevent insecure or non-compliant configurations from being put into production.

You can't effectively test or audit the end-state of a secure, compliant network; instead, you simply have to come up with a configuration that does the job and ensure that the configuration stays in place. If there is no process managing change, or if there is a process that can be bypassed, then changes that are insecure or non-compliant can occur to configurations. Simply having a process isn't sufficient; it must be enforced, and in a manually configured network that's practically impossible to do.

Inability to Effectively Report Makes Compliance Audits Difficult

Compliance audits are the low point of most administrators' and managers' work lives—auditors run around, asking for difficult-to-provide information, asking difficult-to-answer questions, and generally making everyone's lives difficult-to-live. And the auditors aren't any happier about it; they're simply there trying to do a job and ensure that the network is (and has been, and likely will be) secure and compliant, yet they have precious little information to work with. This situation is a huge loophole in the compliance (and security) world, because the audit is the last point at which compliance and security problems can be caught. Without information to work with, audits are completely ineffective. In fact, most organizations struggling with audits aren't struggling with their actual device configurations as much as they are struggling with simply trying to provide their auditors with the right information.

Technologies and Tools that Close the Loopholes

So how can you close the loops in traditional device management without removing critical functionality and capabilities that your organization and its staff require? Automated configuration management solutions can provide most of the answers.

Pass-Through Administration in a Management Solution

Pass-through administration—sometimes called *proxy* administration depending on how it's implemented—allows administrators to continue using Telnet, SSH, or other command-line administration techniques. One of the biggest fears presented by a new automated configuration management solution is that administrators will be crippled and unable to access devices in the ways they need to in order to properly maintain the environment. Such shouldn't be the case, of course; a well-designed solution can provide capabilities such as Telnet. Because the Telnet (or SSH, or whatever) traffic is passing *through* the configuration management solution, the traffic is also auditable and can be secured using the solution's own more granular, role-based security.

Automated Detection of Out-of-Process Changes

Regardless of which type of solution you have in place, however, out-of-process changes are practically a certainty in any organization. You can try to reduce these as much as possible (and you should), but they're still likely to occur, meaning they still represent a major loophole to security and compliance in your environment. However, a good configuration management solution can still deal with out-of-process changes, in two ways:

- By reading SNMP traps, TACACS+ logs, and Syslog logs, the solution can “know” when a device may have been changed, and use that knowledge to examine the device's configuration.
- On a periodic basis, the solution can simply re-read all devices' configurations to look for any changes it might have missed.

These out-of-process changes can then be examined for consistency and compliance with your configuration rules, and alerts can be generated to let someone know that devices have been reconfigured outside of the configuration management solution.

Automation of the Change Management Process

A good configuration management solution can also automate and enforce your business processes. For example, some solutions can integrate to some degree with Help desk management solutions and with enterprise management frameworks, allowing them to initiate changes based on incoming Help desk tickets. Some solutions, for example, might not allow a device configuration change to be initiated until a Help desk ticket can be linked to the change, thus ensuring that any formal review process you may have in place is followed. The solution can also enforce peer reviews by not allowing changes to be deployed until an authorized user reviews and approves them. Maintenance windows can also be enforced by preventing changes from being deployed outside the window without special approval. One of the major loopholes in security and compliance is the ability to enforce your business processes; configuration management solutions can provide that enforcement.

Role-Based Security in a Management Solution

Network devices typically don't have very granular security; it's pretty much “read” or “read/write” to the devices' configurations. A configuration management solution can provide much more granular, role-based security, allowing the right people to read and write the configurations of the appropriate devices. In fact, some solutions provide such granular security that you can designate a particular role as being able to read, read and write, or have no access to *individual items within a device's configuration*. A particular role might, for example, be able to read and write the SNMP community string from just a specific subset of your network's devices. Regardless of whether you need that level of security, having the capability provides you with a great deal of flexibility to ensure your network remains properly configured and that configuration data is available only to the people who absolutely need it.

Auditing in a Management Solution

Management solutions typically provide detailed reports that can be a real blessing when the time for an audit rolls around. Reports may simply list all unauthorized changes (an empty list would make the auditors happy), all changes made in a certain date range (for comparison to the change requests you maintain), and so forth.

Because most management solutions provide detailed auditing records, everything an auditor might need to know—including current device configurations—is immediately available. Auditors can be given permission—through the solution’s role-based security—to read (but not change) any data they might require access to, and they’ll never have to log into a single network device because all the information is maintained within the solution’s own database.

Compliance Reporting

Solutions that come with compliance-specific reports (for example, Sarbanes-Oxley-specific reports are becoming common in more high-end solutions) can make compliance audits even easier. By simply running the appropriate report, you’ll be able to provide a compliance auditor with almost everything they need to know to complete their audit. They won’t have to dig for information, you won’t have to spend days assembling the reports, and everyone’s lives will be much easier.

Automated Configuration Remediation

Finally, the ability for a network configuration management solution to quickly and automatically reconfigure devices to match your configuration rules and policies is invaluable. Everything up to this point has been about ensuring the right configuration gets to devices, but automated remediation is the last line of defense when improper configurations make it out to devices anyway. Rather than simply alerting you to a problem—which is useful but still allows the problem to exist for however long it takes *you* to deal with it—automated remediation helps keep you secure and compliant *at all times* by ensuring sensitive configuration settings are never changes for more than a few minutes without being automatically reset to your centrally configured, top-level rules.

The Need for Auditing

Auditing *is* a necessary function. However, it’s important to realize what auditing can and cannot do, and what it can and cannot tell you so that you’re auditing the right thing. IT in general, and networks in particular, don’t lend themselves well to configuration spot-checks. That doesn’t mean auditing is entirely useless, but it does mean that the auditing performed by most organizations is effectively useless. The next few sections outline effective auditing.

End-State vs. Configuration Auditing

One important concept in auditing is the idea of auditing the *end-state*. For example, if Ford claims that their cars can protect a passenger in a 20mph frontal collision, auditors don’t sit down and examine the car’s engineering drawings; instead, they crash the car into a steel wall using

test dummies to see how the dummies fare. This is the *end-state*—testing the final outcome of the claim.

However, this process is pretty much impossible in networking. If you claim, for example, that no unauthorized individual can access data, you can't *prove* it directly. For one thing, you'd have to have every man, woman, and child on Earth give it a try. For another thing, there are factors involved that are completely outside of your control. An operating system (OS) bug, for example, could result in your claim being false, despite your best efforts. Thus, in the world of IT and networking, you instead do the equivalent of looking at the engineering drawings: You examine configurations to determine whether they're likely to meet your claims. You use documented best practices and other materials to develop what you think are secure configurations, and you check to make sure they remain in place.

Thus, the first thing to understand about network auditing is that you can't audit the end-state. Instead, you can audit only the configuration. Because you're relying on the configuration, you need to ensure that it's been in place *continuously* (a concept covered earlier in this chapter). You must also recognize that individually auditing the configurations on an organization's hundreds of network devices is exceedingly time-consuming; most auditors content themselves with a spot-check of a few devices, which is pointless because just one misconfigured device is all you need to make an organization insecure and noncompliant. As you must acknowledge your inability to test the end-state and to be comfortable just auditing the configuration, there has to be a more effective way than per-device auditing.

Network Management Through Templates and Policies

Managing through templates and policies, rather than managing actual device configurations, makes much more sense when you have a need to maintain security and compliance on your network. Think about it this way: What if you could create a set of configuration rules and policies, then have a network configuration management solution automatically enforce those rules and policies? Any device on your network that didn't match up with the rules and policies would be fixed immediately. If you needed to change one of your rules, you'd simply *change the rule*. All your devices wouldn't match the new rule, so your configuration management solution would *automatically reconfigure them*. True, it takes a bit of experience and trust to come to the point where you can do that with a configuration management solution, but this style of management is incredibly flexible and really helps to implement the top-down type of management espoused by IBM in its OnDemand framework and by Hewlett-Packard in its Adaptive Enterprise framework (and by Microsoft in its Dynamic Systems Initiative, for that matter). Stop managing devices; instead, manage policies and have a solution that makes your devices comply with your policies.

In such a situation, think of how simple auditing would become—an auditor checks your policies to make sure they're right and checks to make sure your solution is enforcing those policies. Audit done. This top-down type of administration is possible without an automated network configuration management solution, of course, but it isn't efficient, and because it requires devices to be manually reconfigured to match policies, you're really still just managing individual devices.

Automated Configuration of Devices

Automated configuration is, as in so many other areas, the key to making auditing easier. Remember that network devices were, for the most part, never intended to be audited. Sure, they have some logging capability, and thanks to technologies such as Syslog, RADIUS, and TACACS+, it is possible to do a halfway decent job of tracking changes. However, network devices just weren't built to be auditable devices. Automated configuration brings the actual configuration activity out of the device and into a much more auditable and accountable environment in which very granular auditing records can be maintained. Finished configurations are pushed out to devices that still aren't very auditable, but provided the configuration solution is the *only* means through which to modify device configurations, it doesn't matter; all the auditing information you'll need is contained within the solution itself.

Building a Plan to Support Compliance and Security

An automated configuration management solution should exist primarily to support a business process, such as one that follows the Information Technology Infrastructure Library (ITIL) framework for change and configuration management. Creating an appropriate business process, then, should become your first order of business. For an example that provides a quick overview and a sample business process, consider the process in Figure 2.6, which is an expansion of Figure 2.5.

This process doesn't refer specifically to technologies (my notes, in yellow callouts, suggest technologies, but the main process steps, in blue, don't). The reason is that this is a *business* process, and it could easily apply to almost any type of configuration management activity, not just network devices. The process has a few characteristic steps:

- Changes are received and logged
- Changes are reviewed and categorized, often by a small group representing both technical and management concerns
- Changes are developed by technical resources, such as administrators
- Changes are reviewed
- Changes are scheduled for deployment inside maintenance windows
- Devices are backed up prior to being changed
- Devices are tested after the change is deployed
- Feedback into the Help desk system—closing tickets, for example—closes the loop and ensures the original request is fulfilled by the change

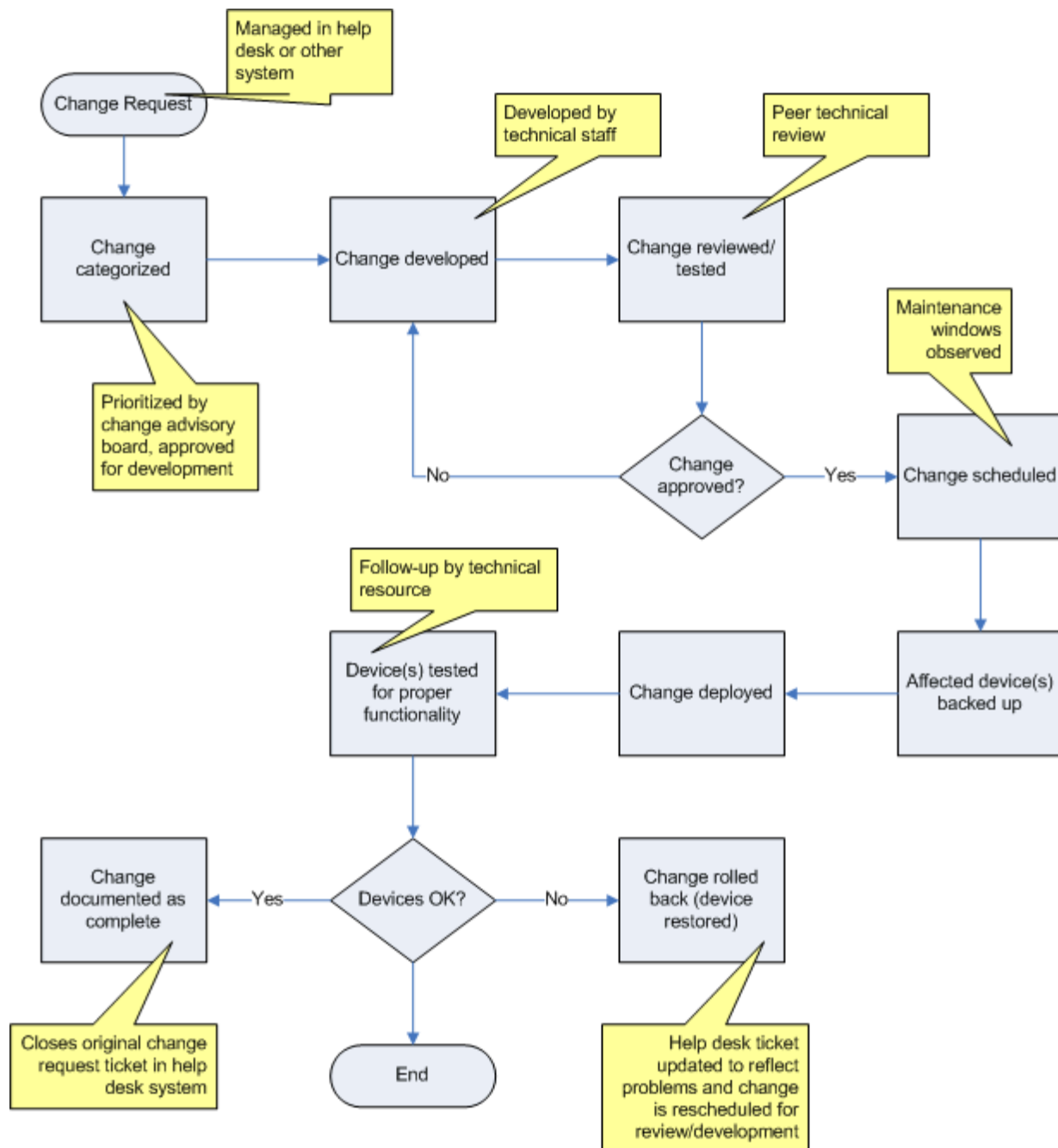


Figure 2.6: Sample configuration management business process.

Creating a Business Process That Supports Compliance and Security

Any business process can be tweaked to support compliance and security. In the example process, most of the requirements a secure and compliant organization might have are already represented:

- Changes are initiated by a logged, auditable request
- A review process ensures that changes are done correctly and that they don't violate any security or compliance rules
- Changes are tracked through to completion by being tied to a "request," such as a Help desk ticket.

The trick, of course, is in *enforcing* this process, as a process' mere existence doesn't ensure that it'll be followed.

Mapping Technologies to Your Business Process

Bringing in technology to support the business process is the next step. For example, Figure 2.7 shows the same business process, with ideas about how technology and tools can fit in to handle or support various steps of the process; the items in orange can usually be provided by a high-end configuration management solution.

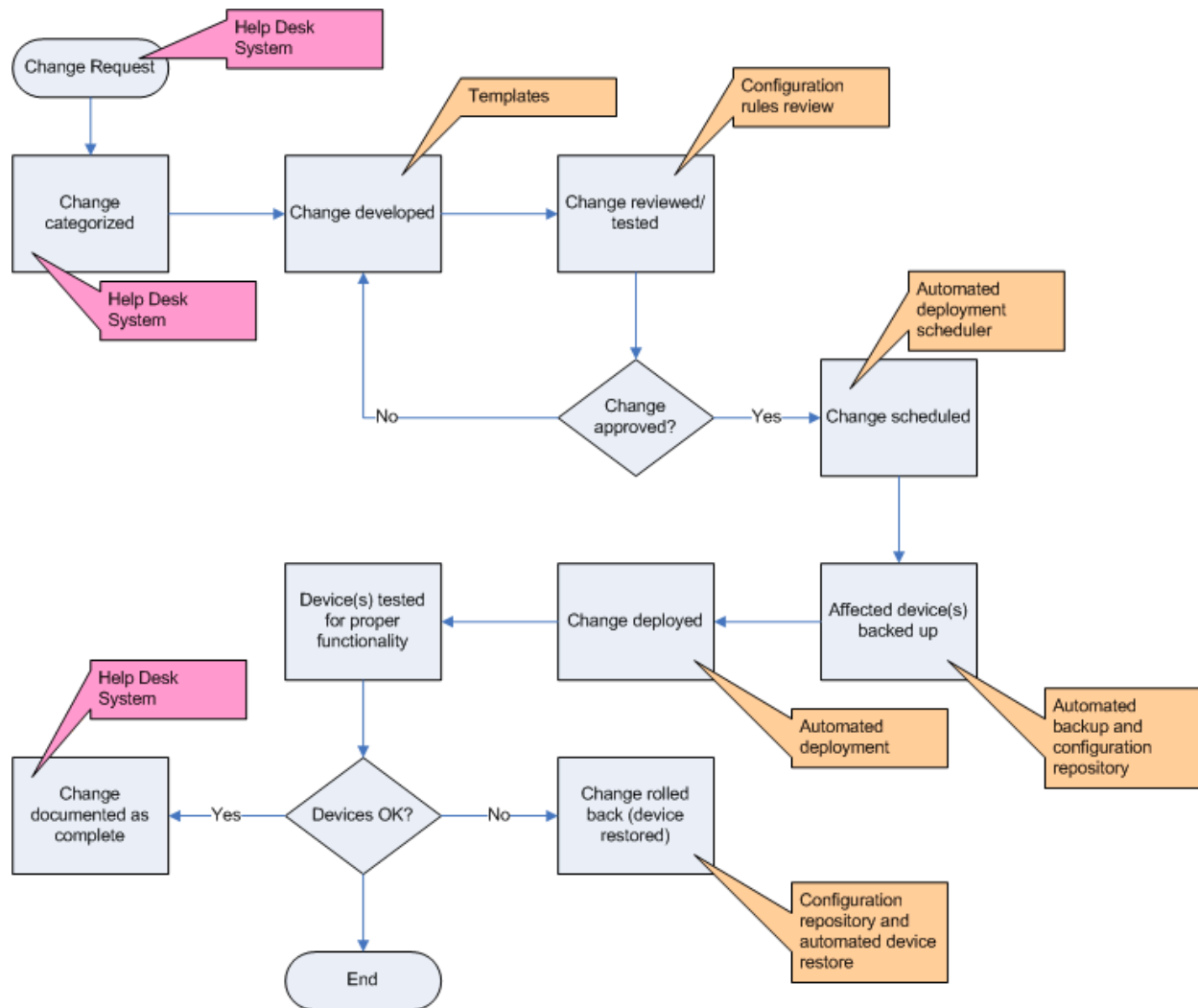


Figure 2.7: Adding supporting technologies.

Understanding that *this* is your business process and *these* are your technological needs will allow you to begin evaluating solutions to find ones that meet your technological needs. This process is different from simply looking at solutions to see what they all can do; that usually winds up in the acquisition of something that *doesn't* support your business processes, meaning your business will have to change to accommodate the technology—and that's rarely a good idea.

Evaluating Solution Features that Support Compliance and Security

Once you've identified your business and technological requirements, start looking at configuration management solutions. Ask detailed questions about *how* each solution supports each of your needs; vendors all have different approaches, and you want to find the one that seems to do the best job for your organization. Be cognizant of implementations that might require more training for your staff; this requirement is not necessarily a bad thing, but you want to be aware of that requirement up front. Construct an evaluation form that allows you to document how a solution addresses each of your needs, and ask others in your environment to review the form and generate follow-up questions. Create a scoring system that allows you to weight each solution's desirability based on how well it addresses your specific needs. As you learn more about the products you're evaluating, fine-tune your technological requirements: You may, for example, learn that some products interface with an enterprise management framework that you already have, and might choose to make that a requirement of whatever solution you eventually choose.

Evaluating Solutions' Ease of Adoption

The last thing to check out is how easily a solution can be adopted and deployed. Ask vendors for case studies and customer references; ask detailed questions about how long it takes to deploy the solution, get your configuration policies into it, and so forth. How long will it be, and how much will it cost, until the solution is fully in place and supporting your business requirements? Will your staff need special training? Is the deployment phase something the vendor (or a partner) can assist with?

Checklist: Tools and Technologies You Need

The following list provides a basic checklist of the things you'll need in your environment in order to make automation a reality:

- TACACS+, RADIUS, or Syslog, with all network devices logging to whichever one you select
- SNMP
- Automated backup of device configurations into a configuration repository
- Automated deployment of approved configuration changes
- Enforced workflow for change development, approval, and scheduling
- Ability to specify configuration rules or policies, which can be reviewed by the solution so that you can see which devices match and which don't
- Ability to *enforce* configuration rules and policies, perhaps by tying a corrective configuration action to each rule so that devices in violation of the rule can be automatically remediated
- Pass-through or proxying for Telnet or SSH so that administrators can continue using existing practices to manage devices

These are, of course, just the basics—you'll likely have additional requirements, but this list should help get you started.

Summary

Although the network might not seem to have an obvious role in compliance and overall organizational security, it is in fact the one point at which everything comes together; the bottleneck, if you will, for security and compliance. The reason is that networks transmit most organizational data, and yet networks are inherently more difficult to secure and protect than the repositories—such as file servers and databases—where data normally resides.

Creating a secure, compliant network virtually *requires* automated network management tools because a manually administered network is nearly impossible to make compliant and keep that way. Even with a compliant, secure business process for making network configuration changes, too many loopholes exist that can take the network into an insecure, noncompliant state. Automation can close these loopholes by automatically detecting devices, automatically detecting configuration changes, enforcing a configuration management workflow, and ensuring that your known-compliant configurations are in fact active throughout the network. Network automation solutions can ensure that your business processes are enforced and effective, and can provide the on-demand reporting and configuration management controls you need to maintain a fully secure, fully compliant network at all times.

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.

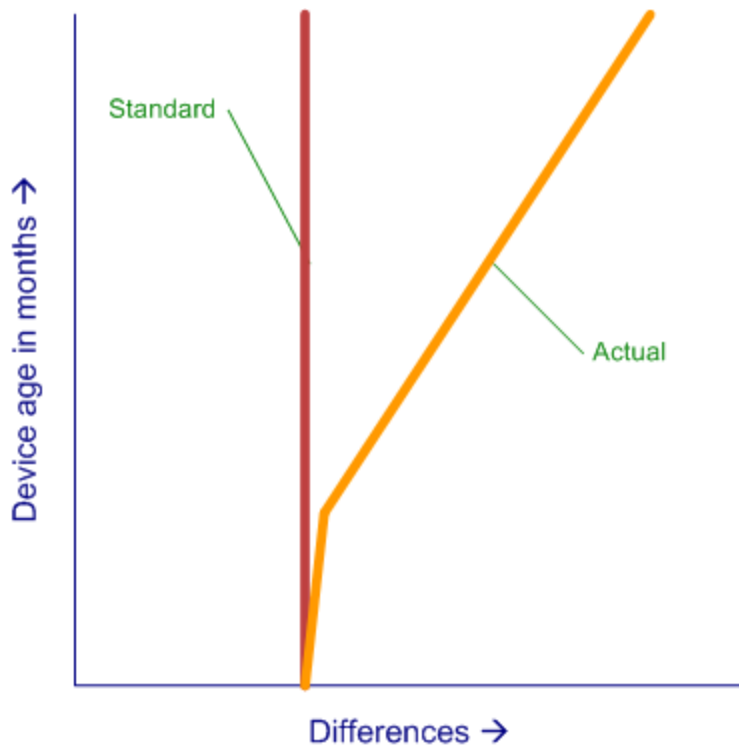



Figure 3.1: Charting configuration deviations over time.

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.

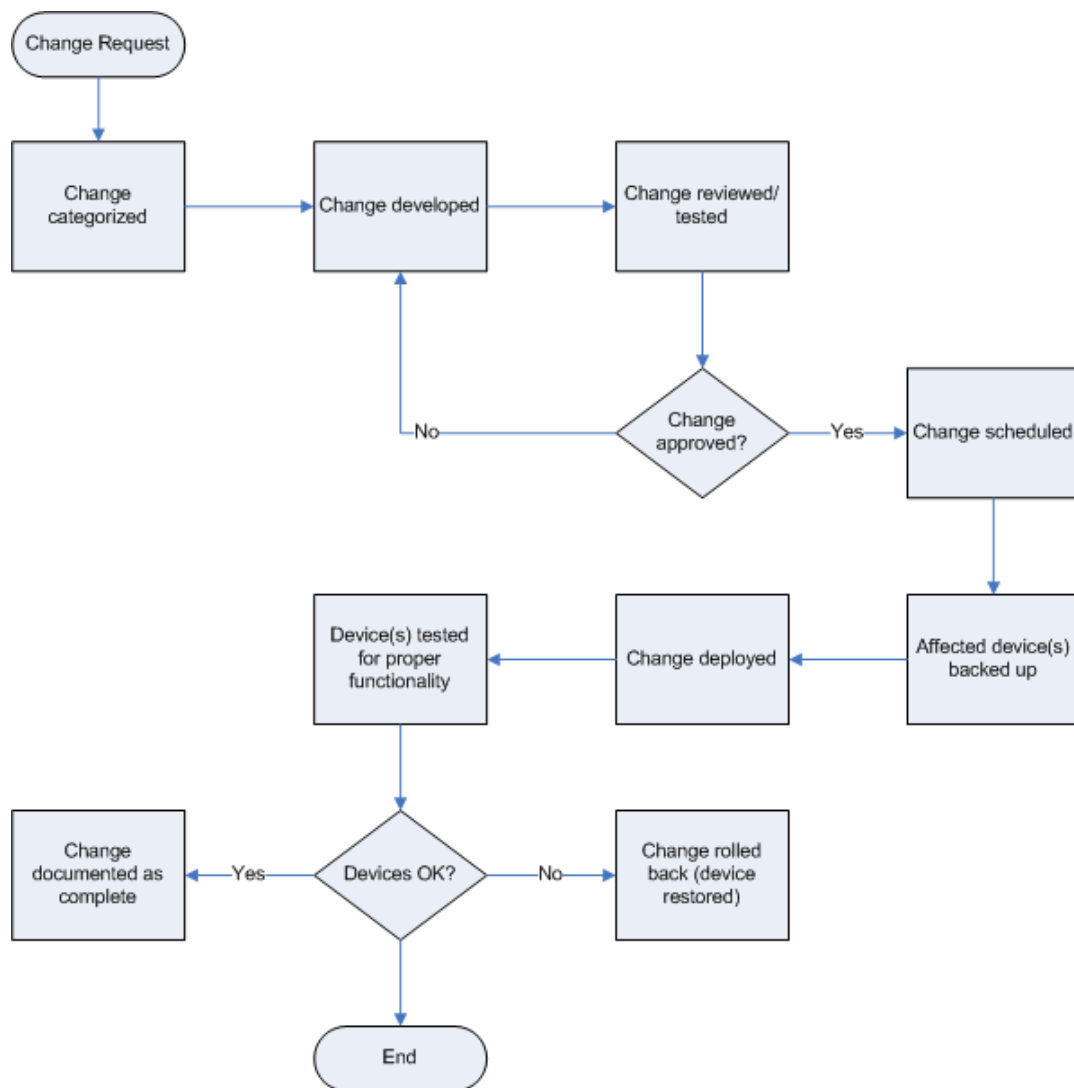


Figure 3.2: Example ITIL-inspired configuration management process for networks.

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.

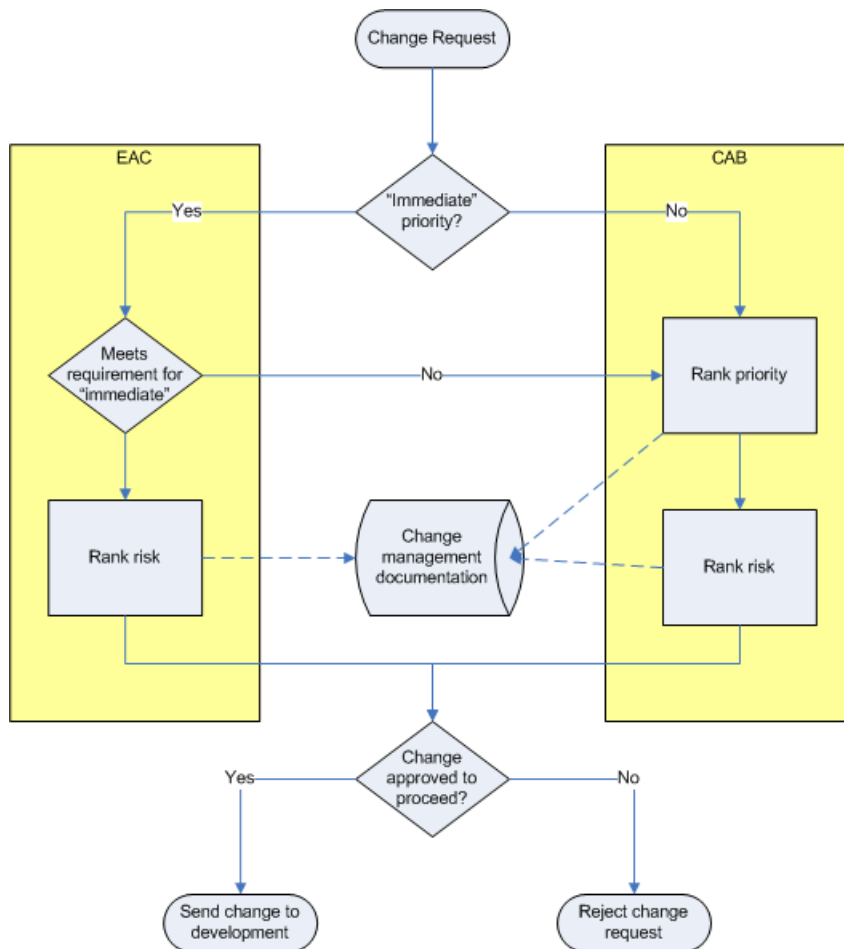


Figure 3.3: Change assessment process.

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.

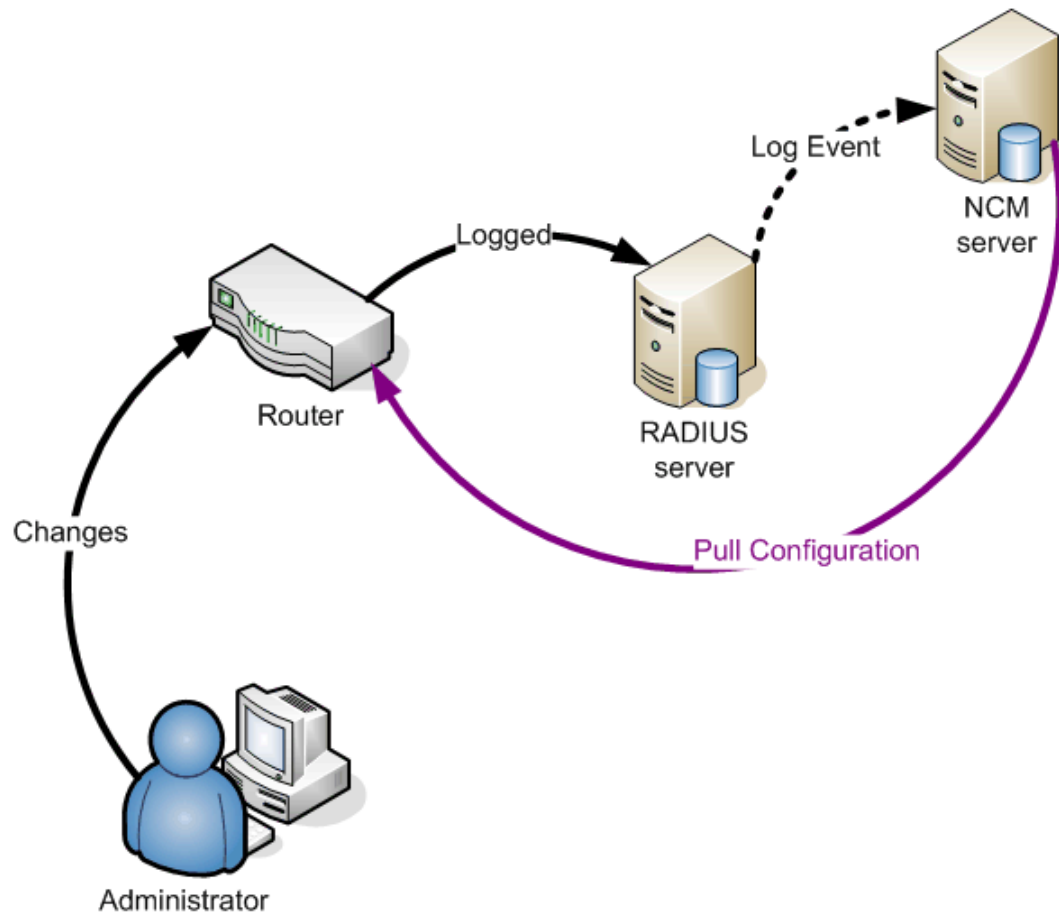


Figure 3.4: Using RADIUS for configuration change detection.

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.

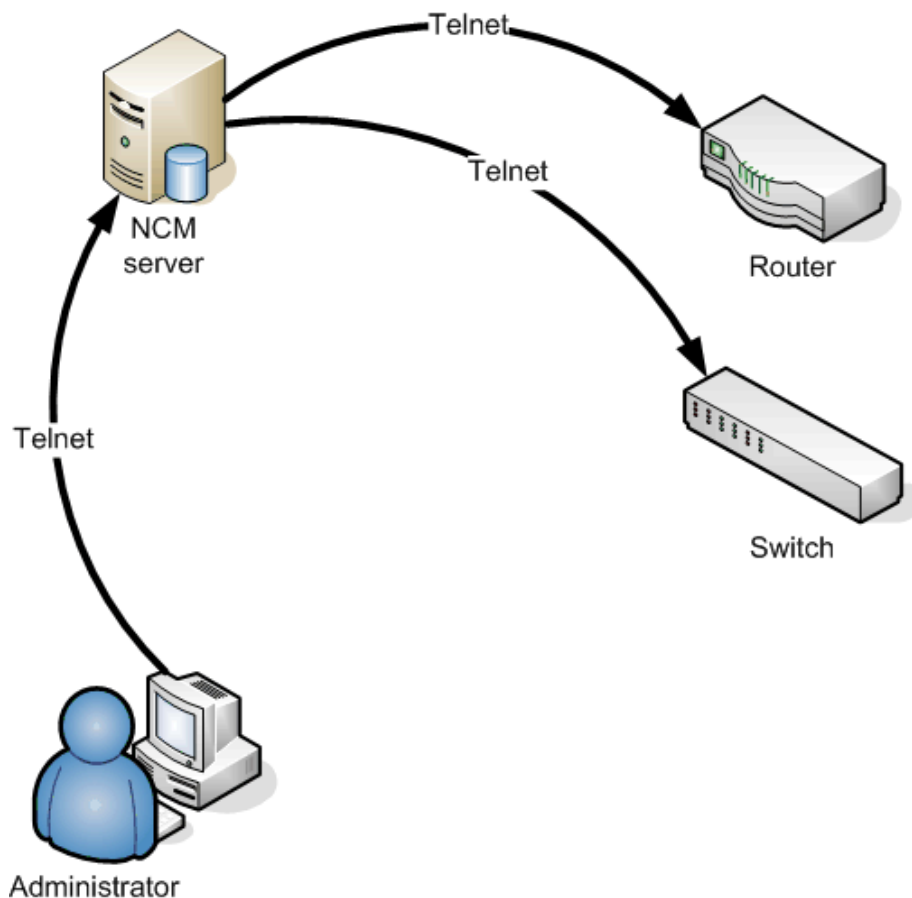


Figure 3.5: Telnet pass-through or proxy.

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.

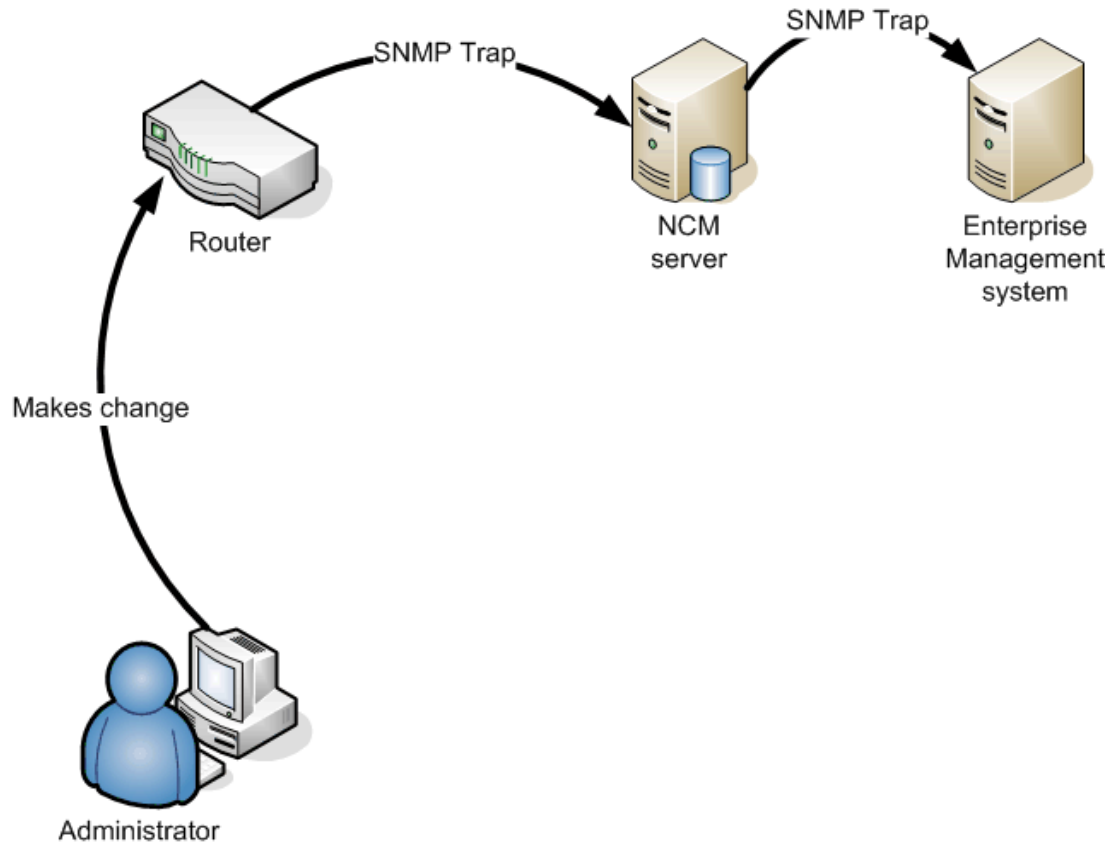


Figure 3.6: SNMP in a network configuration management solution.


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.

Chapter 4: Building the Plan for Automating Network Operations

Thus far, this guide has discussed most of the major aspects of network operations automation. Chapter 1 discussed the essentials, including the benefits offered by automation. Chapter 2 focused on compliance and security, and how they can benefit from automation in network operations. Chapter 3 covered automation from the viewpoint of maximizing availability; this chapter touched on a very important aspect of automation—business processes. In addition, it outlined how factors such as a change assessment, a complete change management process, and other processes can help drive network operations automation more effectively. This chapter expands on that theme and discusses the need for business processes for *all* network operations, for finding a solution that helps implement that process, and for integrating the solution and your processes into a cohesive whole.

The idea behind this chapter is to create processes within your company that result in your network being managed and operated in a way that meets all your business needs. From that process, you'll evaluate technological solutions that allow that process to be realized—essentially, bridging the gap between what the business requires and what you could normally deliver, on your own, in terms of implementation. Once a solution is selected, you'll need to integrate it with your processes. Finally, because coming up with business plans can often be difficult (especially if you've never done it before), the chapter will look at sample business plans that cover tasks such as daily administration, auditing, disaster recovery, and so forth.

Building the Business Process

Too often, companies purchase technology solutions or implement new technologies without first fully defining the process that those technologies and solutions will support. It's a bad idea: Without knowing what you want to do, how will you know if you're doing it correctly? Fortunately, creating a process doesn't have to be difficult. Start by thinking about what your business really needs, then utilize a framework that helps provide a structure for your new process, essentially giving you a template, or head start.


By design, networks that are up and running *stay* up and running. The only time things tend to go wrong—and therefore require management—is when changes are made (apart from unforeseeable disasters or equipment failures, of course). Disaster recovery is also a concern, but again it tends to kick in when network devices are changed: That's when you need to, for example, make a new backup of the devices' configurations. Thus, although network operations processes *support* a variety of business needs—including availability, security, compliance, and more—they tend to *focus* on configuration and change management. That is precisely why you need a process. By managing change, you ensure that the network fulfills its primary purpose of delivering reliable communications between computers.


Frameworks for Change and Configuration Management

Currently, the most respected business process frameworks in the IT industry come from the Information Technology Infrastructure Library (ITIL), a product of the United Kingdom's Office of Government Commerce. OGC's job is to work with public (that is, government) organizations to help them improve their efficiency; ITIL is a major component of how OGC accomplishes that mission. ITIL is essentially a cohesive set of best practices, drawn from public and private sectors and codified into a complete IT management framework.

 You can learn more about ITIL at <http://www.itil.co.uk>.

The portion of ITIL that applies to network operations, the Service and Support segment, is available as a complete, standalone publication. This segment covers Service Desk management, Incident Management, Problem Management, Configuration Management, Change Management, and Release Management—with the latter three being the most important for network operations. Related segments include Service Delivery (which includes IT Service Continuity Management, Availability Management, and Capacity Management).

 The complete list of ITIL publications—they're all fascinating—can be found at <http://www.itil.co.uk/publications.htm>, where you can also place orders.

 Contrary to popular belief, the ITIL materials are not public domain in the usual sense of the term. Although they are widely available, they are copyrighted by the UK government, and that copyright is recognized under international copyright laws and agreements.

Notice that ITIL defines both *configuration* and *change* management. In the ITIL world, *change management* is the practice of “ensuring all changes to Configurable Items (CIs) are carried out in a planned and authorised [sic] manner. This includes ensuring that there is a business reason behind each change, identifying the specific CIs and IT services affected by the change, planning the change, testing the change, and having a backout plan should the change result in an unexpected state of the CI.” Security and compliance, as business needs, are incorporated into the process, as part of “ensuring that there is a business reason.”

Configuration management in ITIL is a complementary practice, defined as the “implementation of a database—Configuration Management Database, or CMDB—that contains details of the organization's elements that are used in the provision[ing] and management of its IT services. This is more than just an ‘asset register,’ as it contains information that relates to the maintenance, movement, and problems experienced with CIs.”

Configuration management is much less of a *process* than change management; configuration management essentially consists of identifying managed assets for inclusion in the CMDB, controlling each asset (the change management process), recording the status of each asset in the CMDB, and periodically verifying that the CMDB is correct and up-to-date. Change management is where the real process-work comes in.

Examining Your Current Processes

First, you must understand how you *currently* do business. Take a moment to lay out your business processes for network operations *as they exist today*. This examination doesn't need to be fancy; a hand-drawn sketch is fine. Perhaps it'll look something like the sketch that Figure 4.1 shows.

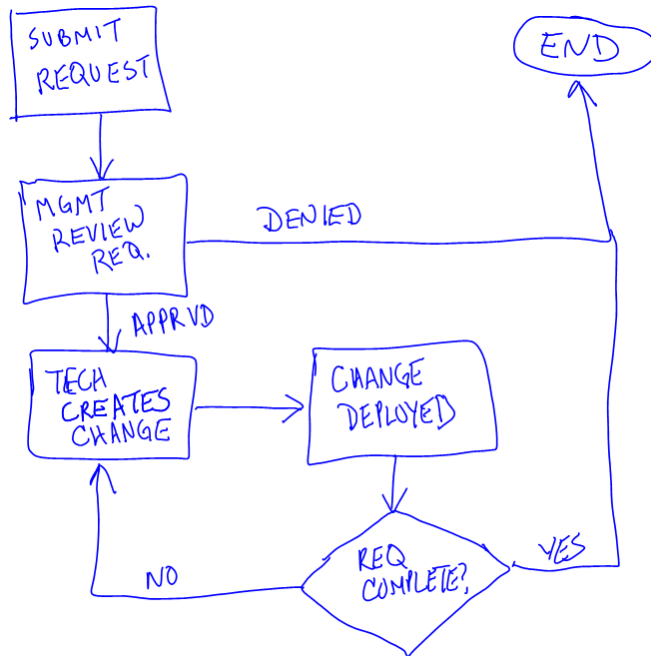


Figure 4.1: Current business process example.

This step is usually the best way to begin creating a new process because it gives you something to analyze and think about. Now you have something to start with and you can begin to identify areas where things will go wrong. For example, ask yourself whether the existing process incorporates the practices you think are needed. Using the process that Figure 4.1 show as an example, consider that following obvious weak points:

- The change isn't documented.
- There is no peer review of changes before they are implemented.
- Management has no input into the schedule of change implementation, meaning implementation may be unexpected or poorly communicated.
- Technicians can engage in an iterative process of changes until they feel they've completed the request; this process can lead to additional unnecessary downtime.
- Changes aren't archived and there isn't a process for rolling back changes that don't work out the way they are supposed to.
- Changes aren't prioritized or categorized, meaning that even important, urgent changes may be treated as longer-term changes.

- This process makes it seem as if network technicians keep most of their “documentation” in their heads rather than on paper or in a file where others can access it.

Analysis of what you actually have reveals the areas where improvement is needed. Begin adapting your business process to look the way you think it *should*, regardless of how or why you do things currently.

Adapting Your Processes

In addition to the weak points already pointed out, there are several missing elements in the existing process—you can probably spot additional things that you would like to see added, so do so. Pull out more paper or even an application such as Microsoft Visio, and start developing a process that meets your needs. As you go, make notations about where tools might be able to help. For example, if you add a step to “archive existing configuration” that happens before any changes are made, you might make a note to look for tools that can automatically archive your device configurations. For a “deploy change” step, you might look for tools that can automatically deploy changes consistently to multiple devices or even do so after hours when fewer users will be impacted. To enforce consistency, you might look for a tool that provides engineering templates.

One of the problems with many processes is that they don’t accommodate the actual players in your environment. Creating a flowchart that identifies the roles—requestors, management, technicians, and so forth—can really help identify the people who are using and participating in this process. Figure 4.2 provides an example.

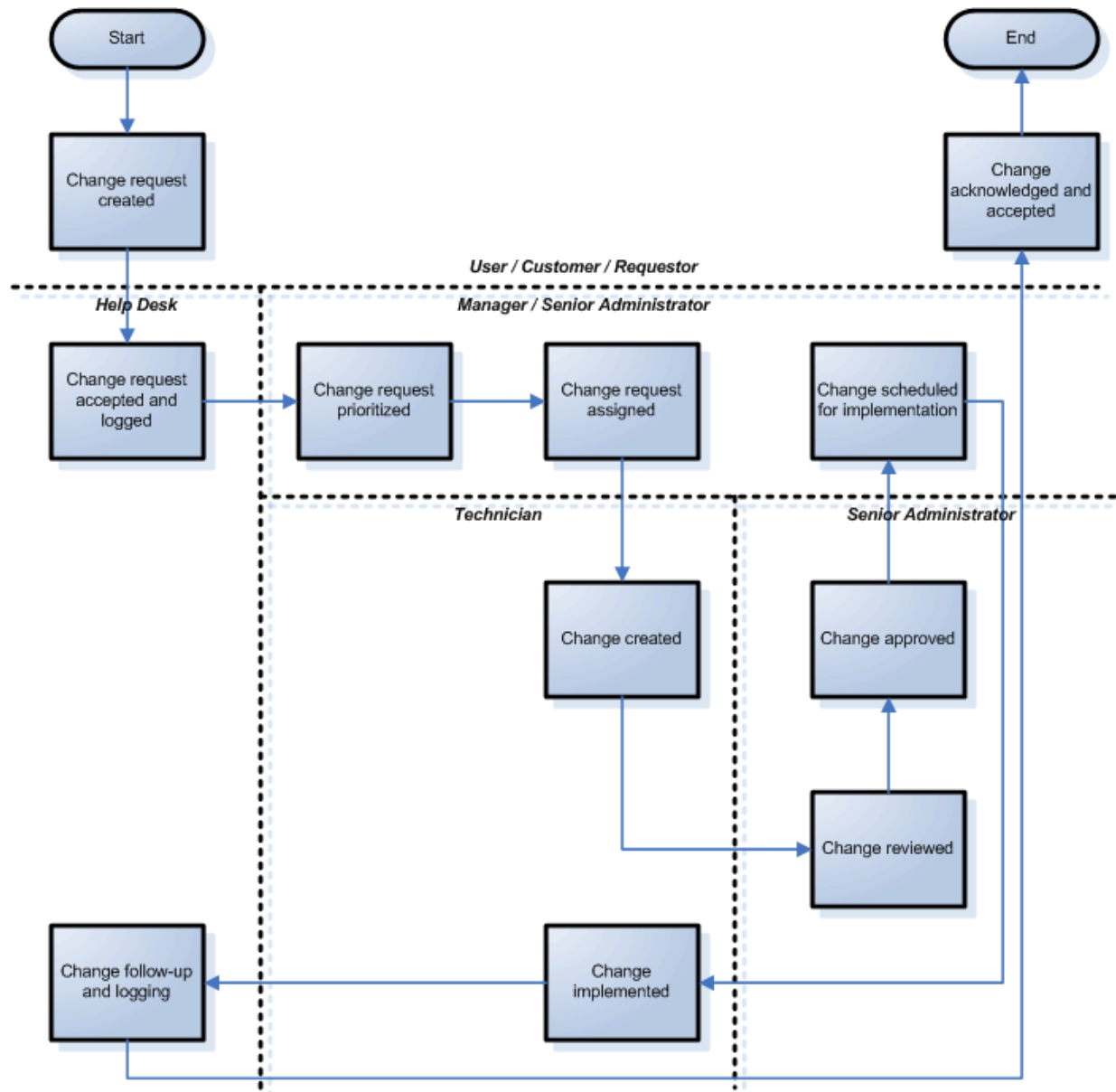



Figure 4.2: Process segmented by participant.

Although this type of flowchart might not be the most useful when it comes to ensuring a safe and reliable process, it can highlight unnecessary redundancies and bureaucracy in your project, giving you an opportunity to streamline the process if desired. For example, in the process that Figure 4.2 shows, the manager might include a tentative schedule with the request assignment. That way, when the peer review by the senior administrator is complete, the process could flow immediately back to the technician for implementation because the schedule has already been set—rather than flowing back to management for scheduling. An exception flow could be provided for changes that take too long to review and fix, making the schedule no longer feasible; such a revision would provide a more streamlined process for the majority of changes,

while providing support for changes that can't be accommodated within the streamlining. So where does ITIL fit in? Let's take a look.

 You can learn more about building processes in this fashion in *Definitive Guide to Enterprise Network Change and Configuration Management* (Realtimepublishers.com).

The ITIL Connection

ITIL is designed to help produce efficient, error-free processes. Most major professional industries have sets of best practices. Accountants, for example, follow Generally Accepted Accounting Practices (GAAP), which are a set of best practices that have evolved over time. Attorneys also have best practices, as do doctors, nurses, and many other professions. IT, however, has evolved at such a fast pace that formal best practices haven't necessarily been forthcoming, or as formally documented as something like GAAP.

ITIL is essentially the current closest thing the IT industry has to GAAP: ITIL are a set of documented best practices that come from the industry's long experience with IT management. Like GAAP, they're not laws or hard-and-fast rules but rather a set of common guidelines that have worked for a number of IT organizations over a long period of time. Using ITIL, like using GAAP, isn't guaranteed to keep you out of trouble, but you're a lot less likely to encounter problems in your IT infrastructure by implementing the practices set forth in ITIL.

An ITIL-compliant process will include some of these major elements:

- **Change logging and filtering**—In this step, Requests for Change (RFCs) are evaluated for necessity and categorized into major- and minor-impact changes. Changes are reviewed by a Change Advisory Board (CAB), and, in the case of major changes, by an executive committee (EC). The purpose of this step is to figure out which changes are important and which ones carry a great deal of risk.
- **Managing changes and the change process**—This area incorporates the actual building of the change, peer review, implementation and schedule, and so forth. A number of communications processes are necessary at this stage to keep the organization working smoothly through the change.
- **Reviewing and closing RFCs**—This functional area is a chance to look back and review changes for mistakes, documenting anything that can be used to improve future change efforts. This sort of “post-mortem” exercise is an opportunity to evaluate RFCs' original intent with the final outcome, and is an important contributor to the auditing process.
- **A backout plan**—This plan enables the ability to roll back changes that don't work as expected, to minimize negative impact on the environment.

ITIL Terminology

Reading through the ITIL documents and related information on the Web can be an exercise in jargon. The reason, in part, is the result of the many specialized terms and acronyms used in ITIL and, in part, because the original documents were developed in the UK, where the language is slightly different than American English. Some of the general terms you'll need to keep in mind include:

- Request for Change (RFC)—A formal document detailing the change that is requested.
- Document—ITIL recognizes that electronic documents—email, word processor files, and database entries—are easier to manage than paper documents in all but the smallest IT shops.
- Change Advisory Board (CAB)—Comprised of major IT department managers, senior users, and a managing change manager.
- Executive committee (EC)—A subset of the CAB that reviews urgent changes without convening the entire CAB; consists of the change manager and service level managers.
- Change manager—The executive in charge of controlling and managing all change within the organization; heads the CAB and EC.
- Configuration item (CI)—Any device, software application, or other configurable entity that falls under change and configuration management (for example, router, firewall, access control list—ACL, host file, and so on).
- Service Level Agreement (SLA)—A set of standards defining IT quality and service targets; includes maximum downtime goals, response time goals, and so forth.
- Service level managers—IT managers who manage their operations to company-defined SLAs.
- IT Planning Secretariat (ITSP) and IT Executive Committee (ITEC)—In the United States, these roles essentially correspond to “upper IT management.” If a specific group of IT managers is responsible for planning, such as a group of business process analysts, they would be the ITSP. The roles primarily come into play for reviewing and planning major-level changes that deeply impact the entire environment, such as re-architecting an entire network or deploying a major new enterprise application. The real contribution of these roles is to maintain a healthy sense of the business needs and impacts while reviewing RFCs.

The OGC maintains an ITIL glossary online at <http://www.ogc.gov.uk/index.asp?id=1000369>; this online glossary is a useful reference for any unfamiliar terms you come across.

Figure 4.3 shows a sample ITIL process that you can modify to meet your organization's specific needs. This process includes all the key elements of an ITIL-inspired process, including a CAB/EC review, change categorization, risk analysis, change development and testing, notification, implementation, auditing and documentation, and so forth. These key elements help to ensure that changes meet a business need, they cause a minimum of disruption, and they're fully tested and documented, preferably in a CMDB of some kind.

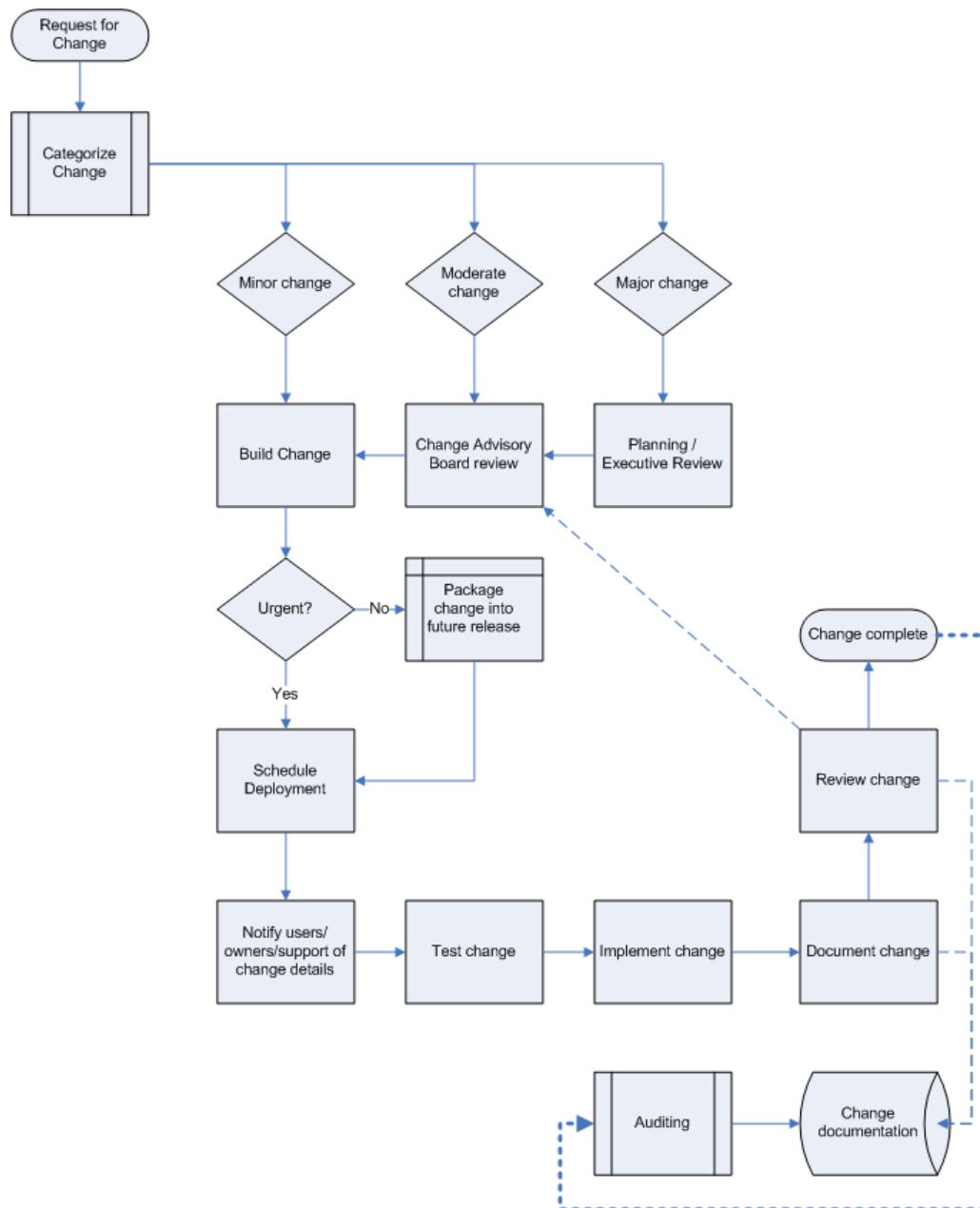



Figure 4.3: Sample ITIL-compliant process.

 This particular example is not complete: It doesn't include backups that would provide for a backout plan if a deployed change didn't work out. That's okay; this is just an example, and it isn't meant to be complete yet.

Of course, having a process such as this is just the beginning—it's useless without some kind of tool or solution that implements the process, enforcing it and handling the portions of the process that can't practically be completed manually. *That's* where a network operations automation solution comes into play.

From Process to Reality: Where Automation Fits In

The process that Figure 4.3 shows is one many companies probably wouldn't mind, except that it carries several unrealistic, or at least impractical, elements. For example, is it practical to have *every single change* reviewed by a committee? Is it realistic to *enforce* a process like this in a manner that prevents technicians from simply becoming annoyed at the process itself and bypassing it?

That's where automation fits in. In fact, you really can't have a robust, reliable operations process without some kind of automation solution, because without automation, too much of the process would require manual, impractical effort—and be too easily bypassed. For example, let's consider how automation solutions might help with the process that Figure 4.3 shows to make it more practical, more realistic, and more reliable:

- Change categorization could occur in a Help desk ticket-tracking system, allowing changes to be more easily reviewed, categorized, and assigned to a technician for development. High-priority changes would be tagged as such in the system, allowing faster review by a subset of the normal reviewers (that's what the EC is for). Help desk systems already support this kind of prioritization.
- Change building and scheduling can be handled by a network configuration management system that supports configuration templates, scheduling, and deployment. These systems might also provide built-in workflow support for change reviewing and approval, thereby encompassing the majority of the process right within one tool and enforcing the process' workflow.
- Configuration management systems are often built around a CMDB, giving you an automated documentation system that keeps track of changes to devices.

In fact, the right thing to do at this point in your plan is to tag your business process flowchart with notes, indicating where some kind of tool would be needed to help facilitate or enforce various parts of the process. That way, you'll have a clear idea of what capabilities a solution will need to suit your environment. Figure 4.4 shows a sample process with automation notes added (note that, for space considerations, this is a simplified process flowchart).

This flowchart's notes include desires for specific functionality: Enforceable change review workflow, automated backups, centralized scheduling, and so forth. These are often *technical* requirements more than business requirements; things you need the technology to do in order to make your *business* process more feasible. Those are great notes to make at this time, as they'll all coalesce into a set of firm requirements for a technology solution.

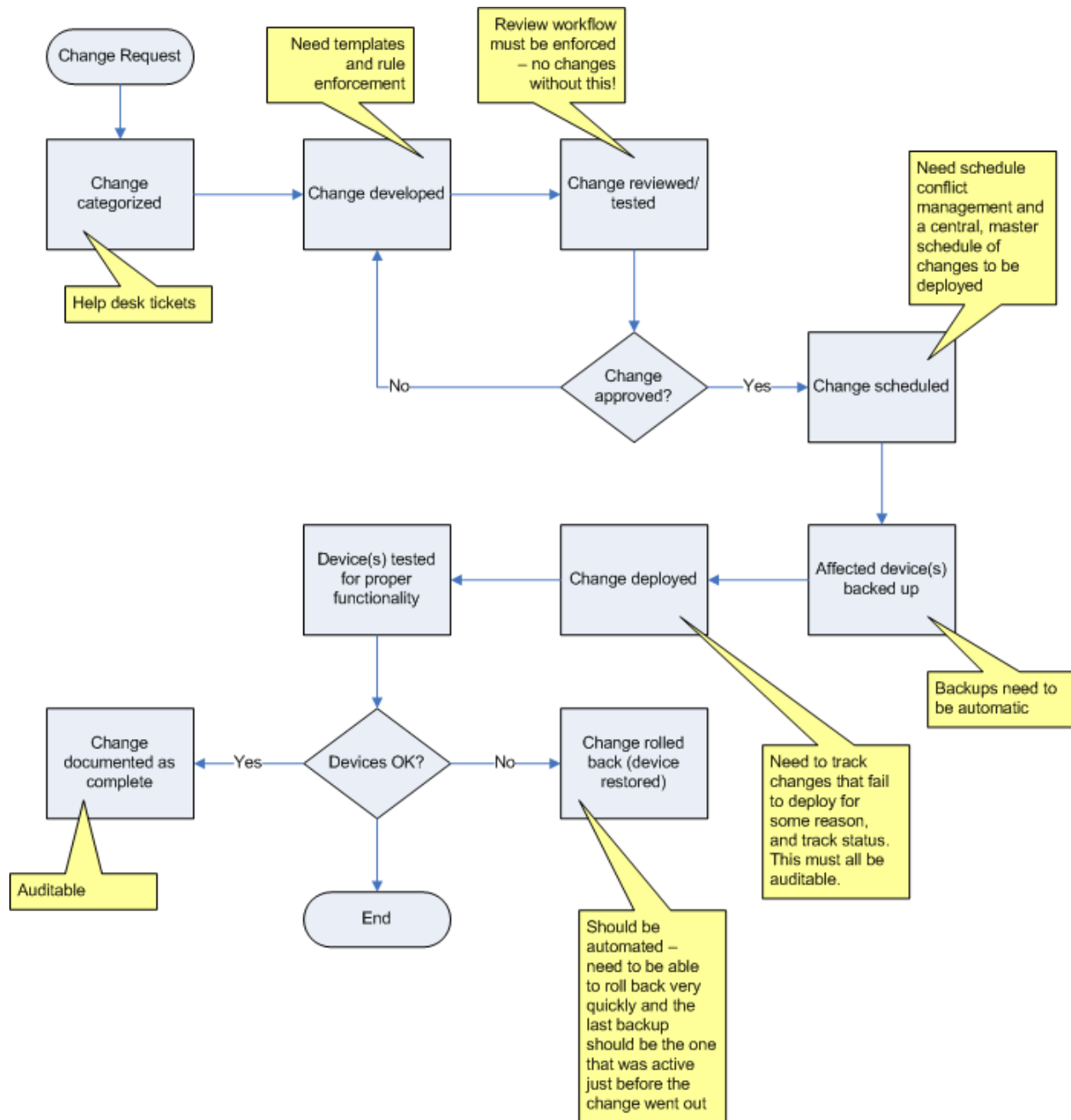


Figure 4.4: Noting automation requirements in your business process.

The bottom line is this: If you're going to automate network operations and realize all the benefits of automation, you need a plan. That plan is built from a business process, but that business process can't often be fully implemented or enforced until you've got a tool capable of doing so for you. Thus, if you want to automate your network operations, you need a tool that will do it for you. That's not a big deal, because there are dozens, if not hundreds, of tools on the market. The trick is finding the one that supports your process—because technology solutions should *always* accommodate your business, not the other way around.

Creating a Solution Evaluation Project

Evaluating software products is one of my least-favorite jobs, primarily because so many of the companies I've worked with in the past make it such a difficult, frustrating process, fraught with internal bickering and political concerns. Unfortunately, I can't help make those aspects of a solution evaluation better. However, I can help with the actual technical piece by providing some examples and suggestions that are common to almost all network operations automation needs, essentially giving you a head start in your evaluation—perhaps allowing you to move faster than your organization's naysayers and political infighters, finishing your evaluation before they've even realized it's underway.

Determining Critical Criteria

The first step is to determine what you need the solution to do. I like to start with a feature comparison chart, similar to the one that Figure 4.5 shows, in which I can list all the features and capabilities I want the solution to have.

Network Automation Solutions - Evaluation

	Product A	Product B	Product C	Product D	Product E	Product F
Auto-backup						
Approval workflow						
Customizable workflow						
Configuration rules						
Notify on broken rule						
Auto-remediate on broken rule						
Allows full Telnet/SSH access						
Secure config repository						
Built-in compliance audit reports						
Integrates with LDAP for authent.						


Figure 4.5: Example evaluation form.

Where do the features come from? From your business process. Spend some time with everyone who uses or is affected by that process and sort out exactly what kind of capabilities you want a solution to have in order to support that process. That's not saying you'll find a solution that meets every one of your requirements, but now's the time to essentially make a "wish list" of everything the perfect solution will do. The following list highlights some ideas to get you started:

- The ability to integrate or work with your Help desk ticket-tracking system. This integration doesn't need to be fancy, maybe just the ability in the automation system to enter a Help desk ticket number, thereby tying a change to a ticket.
- The ability to set up a custom workflow for change review and approval, including designating who can submit changes, approve them, and schedule them for deployment.
- Granular security, allowing you to give individuals—on a role-by-role basis—access to only the bits they need to do their job. For example, auditors may need the ability to look up, but not change, device configurations.
- The ability to schedule changes for deployment.
- Deployment tracking so that you can see which devices successfully receive a change and which don't.
- For failed deployments, per-device details about what went wrong, ideally with suggestions about how to fix it.
- The ability to define groups of devices so that deploying changes to similar or related devices as a group is easier.
- Schedule conflict tracking so that if someone submits a change that will affect the same device as another scheduled change, notify everyone involved so that they can figure out if any conflicts exist and what to do about them.
- The ability to define rules that describe how a device should be configured:
 - The ability to analyze devices for compliance with the aforementioned rules and alert you (or create reports) of problem devices.
 - The ability to automatically deploy changes in order to remediate devices that aren't complying with the aforementioned configuration rules.
- A secure configuration repository that can be easily backed up to offline storage for safekeeping.
- The ability to provide direct device access (Telnet or SSH) while maintaining an audit trail of changes made.
- No single point of failure—the solution should be built in such a way that a failed server won't prevent you from using the solution entirely.
- One-click (or at least very few clicks) rollback of a device to a previous configuration.

- Automatic backup of device configurations.
- Ability to detect possible changes by analyzing Syslog, TACACS+, RADIUS, SNMP, or other technologies and using those as alerts to log in and pull the device configuration.
- Side-by-side comparison of device configuration versions with difference highlighting.
- Support for any device on your network and an architecture that allows new device support to be added without the vendor having to release a major new version of the product.
- Built-in reports for anything you've identified a need (this is the thing you're least likely to get, since many reports are very company-specific, but it doesn't hurt to be on the lookout for it).
- The ability to create ad-hoc reports and to define your own reports.

The list goes on. If your feature comparison list isn't several pages long, then you're 'not trying hard enough: This is your only chance to ask for features, so get 'em all in there.

 Remember, adding notes to your business process can be the best way to identify specific business requirements. Those notes can then become specific features in your evaluation.

Evaluating Solutions

Engage vendors to help you evaluate their solutions. Have your criteria list in hand when you do so. If the vendor wants to point out additional features that you've not considered—they will, salespeople can't help themselves—that's great; note them down separately. If you haven't listed the feature, it might not be important to you, but it might still serve as a tiebreaker later on.

Be sure you know *exactly* what you want from each feature you've listed. If necessary, write up a short paragraph explaining what you need the feature to do, and evaluate solutions for their ability to do *what you want*. Often, these automation solutions are complicated enough that you can't simply install them yourself in a lab to check them out; you may be relying on vendor-led demonstrations, instead. There is nothing wrong with that, provided that you're driving the show. When a solution offers a feature that doesn't *quite* do what you wanted, make detailed notes about what it *does* do. It's possible that the competition will do things the same way and that you won't be able to get exactly what you want, so it's important to understand what you *can* get. It's also important to understand *how* various features are supported—ask for a demonstration, if possible, so that you can *see* features in action.

Anytime you hear an answer like “We're adding that in the next version,” get it in writing. You need to recognize that network automation is a rapidly evolving and highly competitive field; vendors frequently *do* add major functionality in new versions. But don't base your evaluation on that promise unless the vendor can make the promise in writing and really commit to it.

Scoring Evaluations

Scoring evaluations can be tricky. After all, some features will be deal-breakers for you if they aren't available, and others are just “nice to have” features that you can, ultimately, live without.

To help reflect this reality, I use a two-part scoring system. First, for each feature, I define an importance, on a scale of 1 to 3, with 1 being “nice to have” features and 3 being “absolutely required” features. Figure 4.6 shows my evaluation form modified to list these values.

Network Automation Solutions - Evaluation

	Product A	Product B	Product C	Product D	Product E	Value
Auto-backup						3
Approval workflow						3
Customizable workflow						3
Configuration rules						2
Notify on broken rule						2
Auto-remediate on broken rule						1
Allows full Telnet/SSH access						2
Secure config repository						3
Built-in compliance audit reports						2
Integrates with LDAP for authent.						1

Figure 4.6: Identifying importance values for each feature.

If a product is missing a “got-to-have” feature, it is off the list. That narrows the field. Then, when evaluating each solution, I give it a score based on how well it implements each feature I

want. I tend to make notes detailing why I gave each score so that I can better understand the differences between solutions. I usually use a scale of 0 (feature not implemented) to 3 (feature implemented exactly as I wanted), but you can use a scale of 0 to 5 or whatever you're comfortable with. Figure 4.7 shows the evaluation form with a couple of products' scores filled in.

Network Automation Solutions - Evaluation

	Product A	Product B	Product C	Product D	Product E	Value
Auto-backup	2	3				3
Approval workflow	1	3				3
Customizable workflow	0	2				3
Configuration rules	0	3				2
Notify on broken rule	0	3				2
Auto-remediate on broken rule	0	2				1
Allows full Telnet/SSH access	3	2				2
Secure config repository	2	2				3
Built-in compliance audit reports	3	2				2
Integrates with LDAP for authent.	0	1				1

Figure 4.7: Scoring evaluated products.

Then, I simply multiply my “importance” value by the product’s score, for a total weighted score. Products doing a great job with features I really need get a very high weighted score; products that do a great job with features I don’t care about as much get a lower score. The final sum of all weighted scores will help me identify the best product. Obviously, minor score differences between products will require a closer look, but this exercise is useful for eliminating products that simply don’t do a good job with the most important features. Figure 4.8 shows the completed form for two products, with weighted scores shown (the format I used is score/weighted score).

Network Automation Solutions - Evaluation

	Product A	Product B	Product C	Product D	Product E	Value
Auto-backup	2/6	3/9				3
Approval workflow	1/3	3/9				3
Customizable workflow	0	2/6				3
Configuration rules	0	3/6				2
Notify on broken rule	0	3/6				2
Auto-remediate on broken rule	0	2/2				1
Allows full Telnet/SSH access	3/6	2/4				2
Secure config repository	2/6	2/3				3
Built-in compliance audit reports	2/6	2/4				2
Integrates with LDAP for authent.	0	1/1				1

27 (50)

Figure 4.8: Completed, weighted evaluation.

Obviously, this process can be time-consuming when you're working with several pages of criteria; however, this decision is an important one for your business, and it's worth the time to understand how different products can help you solve your needs. This evaluation methodology

will help you keep your eyes on the needs of the business and help ensure that the solution that best implements the features you need the most will rise to the top of your evaluation project.

Integrating Your Processes and a Solution

One you've selected a solution, you have to start using it. That probably sounds obvious enough to be ridiculous, but most companies who implement a network automation solution don't fully utilize it. Perhaps they'll use the obvious features, such as automatic device configuration backups and configuration change deployment, but you need to look back at your business process and make sure that every bit of it that *can* be automated *is* being automated.

For example, if the solution you've selected supports the creation of configuration rules, take the time to create them. Make a rule for each and every aspect of your devices' configurations that you possible can, if for no other purpose than to allow your automation solution to show you which devices aren't configured according to the rules. Decide which configuration settings you feel comfortable turning over to automated remediation, if your solution supports it: Having SNMP community strings automatically configured, for example, is fairly low-risk and will help you gain confidence and experience with your new solution.

Look at everything your new solution is capable of doing and decide where those features might fit into your business process. Automate *as much as possible*. Remember, automation equals more efficiency, more consistency, more security, and more flexibility for your network and your business. Use that automation to the maximum degree possible.

Create and distribute process flowcharts that specifically reference your solution's functionality. Figure 4.9 provides an example: Notice that this process isn't massively different than the earlier one, but it now indicates which process steps take place within an automation solution. This is a flowchart every network administrator and technician should have readily available so that they can see which parts of their jobs are now performed within your new solution. Fully integrating the solution into your business processes in this fashion is the best way to use the solution to its fullest potential.

Obviously, how your process and solution integrate will depend on your process and the solution you select. In the figure, purple boxes indicate the portions of the process that occur within, or are handled by, the automation solution; notes provide details about how the solution handles each step. This particular solution offers templates and configuration rules to assist with change development and provides a workflow that allows changes to be reviewed and approved. Only changes that are approved can enter the solution's deployment schedule, and the solution handles deployment automatically. Affected devices are backed up automatically, and can be restored if the change doesn't go as expected.

The chart in Figure 4.9 shows a very effective way of seeing how much an automation solution can contribute to a network operation's process. Nearly every major aspect of change and configuration management is automated in some fashion, and the solution provides a means for enforcing this workflow.

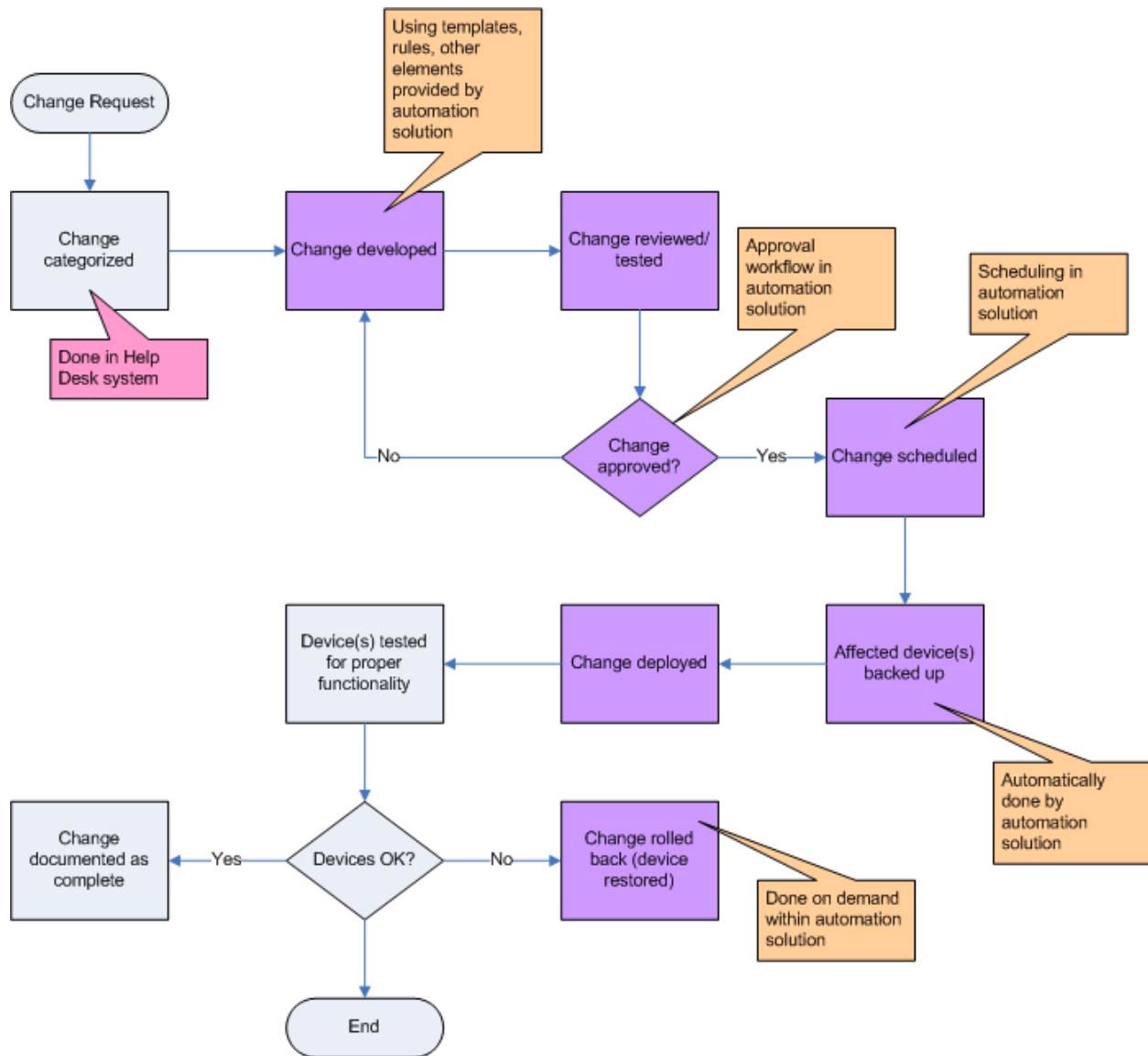


Figure 4.9: Integrating your solution into your process.

Creating Business Plans

Creating business plans can be difficult, especially for factors as prosaic as daily network device administration. The problem is that it's sometimes difficult to really pin down the steps in the process: "I log in, I make the change, I log out" might not seem like much, but you can certainly draw a flowchart for it. That flowchart—simplistic as it is—will highlight the holes in that process, and allow you to clearly see what business needs aren't being met. *That* is the big reason for drawing flowcharts in the first place: To graphically illustrate the business needs that aren't being met.

To get you started, the following sections present sample charts for various tasks, including daily administration, auditing, and disaster recovery as well as how they link in to one another. These processes are also ITIL-inspired, which means they incorporate many of the best practices documented in ITIL, although in places I may use simpler terminology (such as "change review" rather than "change review by the CAB/EC"). The goal of these examples is to simply make process illustrations of best practices, which you can then adopt and adapt as needed.

Planning for Daily Administration

Daily administration primarily deals with changes to device configurations, whether those changes are to support new business requirements or to fix problems that have occurred. Typically, organizations try to schedule changes—especially ones that enable new functionality—for scheduled maintenance periods; changes that fix a problem may be implemented on a more ad-hoc basis. Regardless, all changes should go through some kind of "daily administration" process, such as the one Figure 4.10 shows.

Looking at this process, you know that you need the automation solution to generate appropriate auditing information at certain points—when changes are developed, approved, deployed, rolled back, and so forth. By linking a change to a Help desk tracking system (for example), you can show a complete life cycle (from request to completion) for each change, proving that the change actually went through the process (process compliance is a big consideration for auditors). This auditing flowchart isn't so much a process as it is a set of requirements, helping you to select an automation solution that provides logging capabilities that comply with your auditing needs and requirements.

Planning for Disaster

An effective business process for daily administration will have built-in disaster recovery. Take a look at Figure 4.12, which highlights—in yellow—the aspects of this process that lend themselves to disaster recovery. Also notice the pink highlight, which is a part of the process that helps support business continuity.

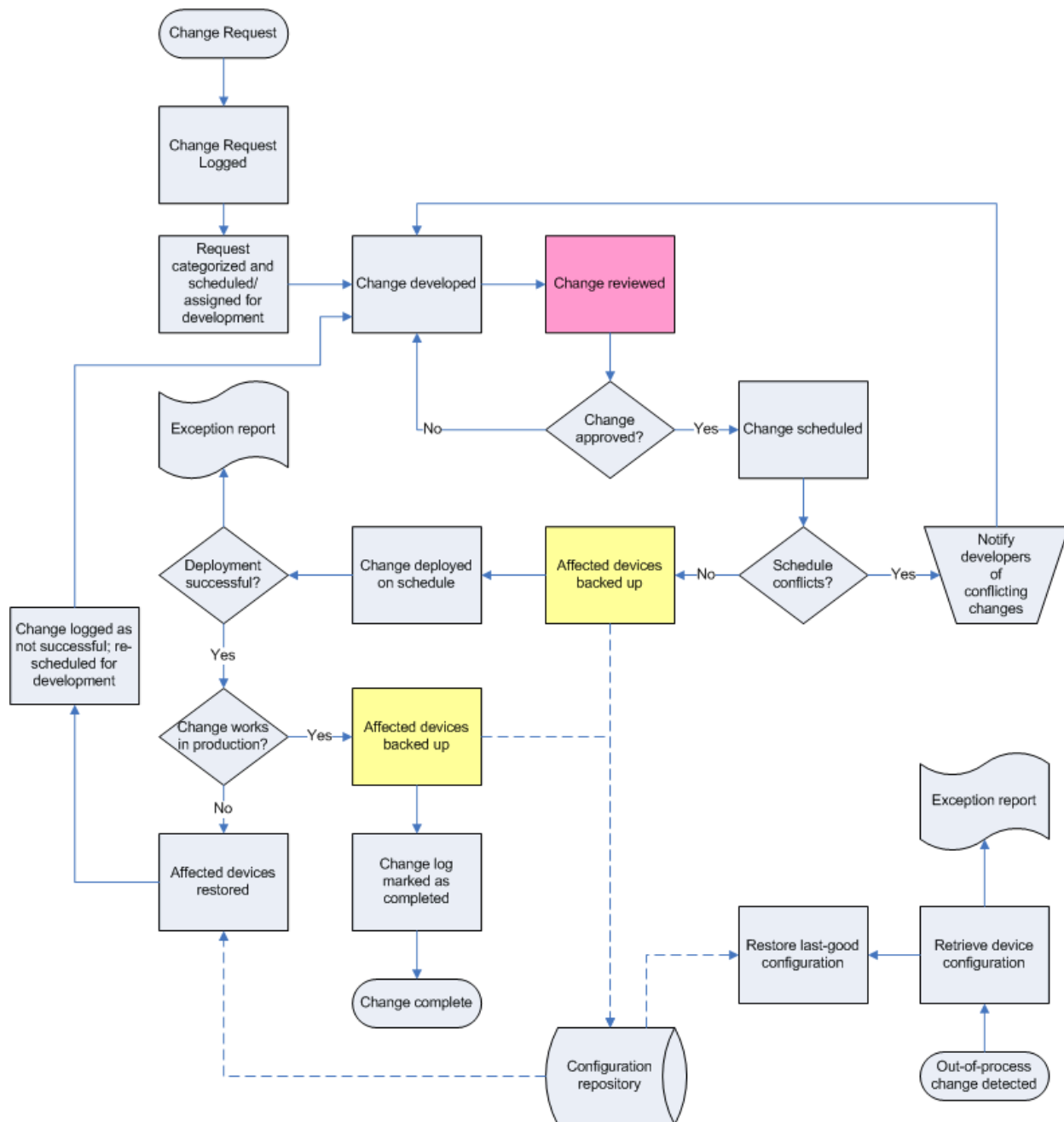


Figure 4.12: Disaster recovery and business continuity support in a robust process.

By creating a backup of device configurations before changes are deployed, as well as after changes are successfully deployed, you're assured of having the latest and greatest configuration in a repository should disaster occur and require restoration. Helping to prevent disaster (at least disaster through misconfiguration) is this process' change review and approval requirement. By reviewing changes, wrong ones are less likely to be deployed, thus providing better business continuity.

Summary

This chapter has focused on the need for a comprehensive plan for network operations automation. Simply buying tools won't work; you need to have clear business goals, which allow you to select a technology solution that will meet those goals. The way to establish your goals is simply to create a reliable business process that meets all your business needs, then select a solution that can help automate, enable, and enforce that business process. This chapter has shown you several business processes that meet various needs for daily administration, auditing, disaster recovery, and so forth; these are all common business needs that any valid business process needs to address. They're also considerations that a properly selected automation solution can provide for you.

As this is the end of this guide, it's worthwhile to briefly review the main points of the previous chapters. First, automated networks are highly desirable from a business point of view. They're easier to configure and more consistently configured, which makes them more reliable, than manually administrated networks. They're also easier to recover in the event of a disaster and, perhaps most importantly, they're easier to secure, make compliant with legislative requirements, and easier to audit. These three factors alone can almost make an automated network worth any price! Automated networks are also more efficient because you spend less time managing them and dealing with problems. Finally—and this is too often overlooked—automated networks are more flexible. They're able to evolve and change more quickly, more easily, and with less risk to meet new business requirements and keep the business competitive.

The benefits of automation are very clear when it comes to security and compliance. Automated networks can be configured to automatically detect, report, and even roll back inappropriate changes, helping to sharply reduce (if not eliminate) insecure, non-compliant configuration settings. This benefit is enormous to any business concerned about security or dealing with legislative compliance requirements because maintaining a secure and compliant environment would otherwise require tremendous manual effort—and industry experience suggests that a manually configured network is almost never fully secure and compliant. Automation can help avoid overlooked devices and can ensure that device configurations remain consistent. Automation can pick up where point-in-time auditing leaves off by *continuously* auditing devices and not allowing a misconfigured device to go unnoticed for any significant period of time. Automation can also make auditing easier by exposing auditing information quickly and clearly rather than forcing you to laboriously assemble these reports by hand.

Automation also has obvious benefits for high availability. Automation can ensure that your device configurations are always backed up and safe and can make it easy to quickly restore devices or to provision new devices as replacements to failed ones. Automation solutions provide a central CMDB that provides version-controlled storage for device configuration files, keeping them safe, accessible, and easily managed. An automation solution can help quickly recover from device failures, device misconfigurations, facility failures, and more. In addition, automation solutions can often *prevent* problems such as misconfiguration by enforcing known-good configuration templates, enforcing change review and approval workflows, and other elements that help ensure only proper changes are deployed to production.

Finally, as this chapter has explored, automation can make a “best practices” business process for network operations practical and realistic. Many best practices simply aren’t achievable without unacceptable manual labor requirements—unless you automate them. Understanding your business requirements up front, then selecting a solution that meets those requirements, will help ensure that you get a network that is fully automated, highly flexible, as secure and compliant as possible, and fully in sync with your business needs.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit <http://nexus.realtimepublishers.com>.