Chapter 4

Building Quality Into IT Security Projects

Solutions in this chapter:

- **Planning IT Security Project Quality**
- **Monitoring IT Security Project Quality**
- **Testing IT Security Project Quality**

- ☑ **Summary**
- ☑ **Solutions Fast Track**

Introduction

We often think of quality as something to check for in workmanship (e.g., whether a chair or a car is well-made). Quality in software projects is also easy to spot. A program that has few bugs and works as advertised is considered a higher quality product than one that keeps crashing or one that generates errors. When we think of Information Technology (IT) security, we typically think of secure versus vulnerable, protected versus unprotected, and safe versus at-risk. None of these really evoke thoughts of quality, but if you think about what will make a network secure, protected, and safe, it is the quality of several processes.

How well you perform your risk assessment will result in how well-protected your network ultimately is. How well you delineate the steps necessary to harden your servers or your network infrastructure ultimately leads to how secure your network is. Quality should be at the forefront of your mind as you define, organize, plan, and manage your IT security project plans. In this chapter, we look at some of the elements you should address in your IT security project plans.

Quality and security have a lot of common traits. Just as with quality, security is difficult to quantify or recognize, because the measure of its success is the absence of failure. Much like insurance, you never want to have to use it, but unlike insurance, if you have your security systems in place, the expectation is zero failure, not 98 percent success. The quality of a security project plan seems more difficult to quantify, but if you do so, adding these measurements to your project will help you maintain the highest possible security standards.

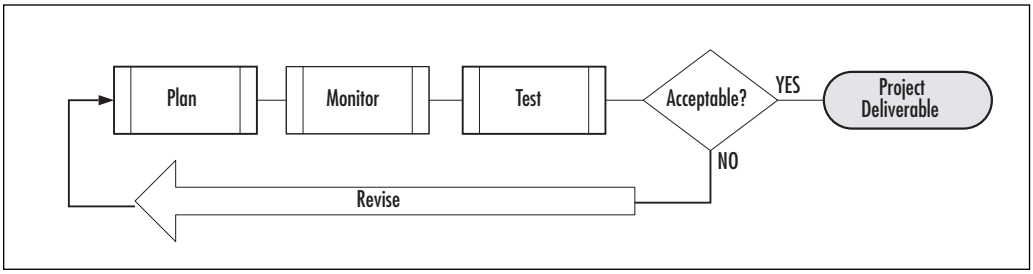
Planning IT Security Project Quality

Just as with standard project management, quality begins in the planning phases. Errors and omissions in the planning stage are amplified throughout the security project plan, and typically become very serious issues at the other end of the project lifecycle, in the implementation or maintenance phases. Planning is the first place quality is "built" into the

security project plan, but as with all project management, it's a continuous cycle (see Figure 4.1). The security project plan is created, implemented, monitored, and tested. Any changes are typically brought back through the project lifecycle as revisions through the change management process. Revisions are almost always necessary, as more detail becomes known through the planning process. As we plan, we understand the security project plan better and may revise the security project plan based on new data or a deeper understanding. In addition, we typically test our assumptions, create workflow models, and develop lab protocols. The results of these should feed back into the security project plan in a logical and thoughtful manner. This continuous loop helps create continuous improvement.

One word of caution: Don't get caught up in the loop, and never deliver final results. You will have to draw a line at some point and deliver the final security project plan. One indication that you need to stop looping and start delivering is when you find that the revisions are becoming smaller and of less importance, or that you've refined the data to about an 80 to 90 percent level of accuracy or acceptability. You'll most likely never reach 100 percent; shooting for it is one of the reasons people get locked in a feedback loop. You might also have key metrics that you define that indicate when it's time to move forward, or you might have a hard deadline that drives you to finalize your security project plan. Keep in mind that achieving higher and higher levels of quality when you're nearing 100 percent becomes more difficult. It's not a straight line increase between 95 percent and 96 percent, it's an exponential increase. The time, effort, and cost associated with perfection is rarely practical, and you'll need to assess your level of quality against your level of risk and determine when to stop revising and when to start working.

Figure 4.1 Quality Assurance Process



The following are specific areas of the planning process that directly impact quality:

- User requirements
- Functional requirements
- Technical requirements
- Acceptance criteria
- Quality metrics
- Change management procedures
- Standard operating procedures
- Federal or state laws, industry regulations, and certifications

User Requirements

If a security project plan fails to meet key *user requirements*, the project is essentially a failure. It's sometimes easy to forget the fact that IT is a service provided to the company to facilitate, enable, and enhance its ability to get the job done. Business activities are performed by various company users, so it's important to find a balance between pure IT security objectives and what users actually need. As discussed in Chapter 3, including key users into your IT planning process early in the lifecycle will help ensure that user needs are considered throughout. It's easier to negotiate

and find solutions to user objections or issues early on, than it is to go back through your security project planning steps and re-work a "perfect" IT security solution. The net result might not be the optimal one from an IT security standpoint, but if it meets corporate, IT, and user needs, you have a much better chance of successfully implementing and maintaining security.

Functional Requirements

Functional requirements are often derived directly from user requirements. Functional requirements describe how a system should perform, often from the user's perspective. The system can be described or defined as any part of your IT security solution, such as user logon requirements or firewall functional requirements. Functional requirements are typically delineated as the services, tasks, or functions required of the system. To use a more concrete example, suppose you are defining the user logon process. The functional requirements might include the requirement that the user only has to log on once to access all information, or it might state that the user be required to logon additional times as they move across domains or into more confidential data. The functional requirements would define how that logon function should work, so that when the security solution is being developed, the IT staff can create the solution that fits these requirements.

Functional requirements are often tied to federal, state, or industry regulations, as well as certifications. For example, if you are working on a security project plan that includes the student health services on a local college campus, you must be compliant with a variety of regulations including the Health Insurance Portability and Accountability Act of 1996 (HIPAA). (See Chapter 9 for more detailed information on some of the common regulations that might impact your IT security project plan.) Requirements that are part of governmental or industry regulation

typically have a precise and detailed list of specifications that your staff, processes, and technology must comply with in order to meet standards. Where available, these should be included as functional requirements. If the requirements are not clear or do not provide the set of detailed specifications necessary to meet certification or regulatory requirements, you'll need to do a bit more research. In some cases, you may need to consult with industry or regulatory experts for clarification, or you may need to seek appropriate legal advice in order to ensure your IT security project plans deliver the requirements for federal, state, industry regulation, or certification.

Business Intelligence…

Functional Requirements Can Help Reduce Complex Security Challenges

Storage Area Networks (SAN) are complex devices that are used by many network entities, from servers to users to applications, and beyond. Using functional requirements can help you reduce the complexity so that rather than trying to wrap your arms around the entire universe of users, you can look at the required functionality. In this way, you can look at the functional areas to determine how the SAN security system should work. An article that appeared on the Search Security Web site in December 2005, describes the five A's of SAN security as authentication, access, auditing, alarms, and availability. Approaching SAN security from these functional areas allows you to work through your security project plan in a methodical manner. Once your functional requirements are defined, you can move into defining the technical requirements. (For more information on the five A's article, go to http://searchsecurity.techtarget.com/tip/1,289483,sid14_gci1152494,00.html.)

Technical Requirements

The *technical requirements* are usually created after the functional requirements. Technical requirements are statements of parameters or measurements (e.g., the Transmission Control Protocol [TCP]/Internet Protocol [IP] ports that are required for various applications so that the firewall can be properly configured, the number of users a solution must support and the maximum amount of time it takes a user to login can be defined; and the minimum amount of disk space required for an Intrusion Detection System (IDS) can be configured.) These are all examples of technical requirements that should be included in your security project plan. As you can see, it usually makes sense to create your functional requirements to describe how your security systems should work, and then create your technical requirements. However, there may be times where you start with technical specifications because your system needs to meet very specific technical parameters. Keep in mind that most legal, financial, regulatory, and industry requirements begin as functional requirements, and must be translated into technical requirements in order to be incorporated into the security project plan (e.g., if you are required to protect personal information due to HIPAA, you should build those requirements into your functional and technical requirements as well as into your acceptance criteria).

Acceptance Criteria

It's always a good idea to define the criteria by which a security project plan as a whole or an individual project deliverable will be accepted. In some project management situations, acceptance criteria are the measurements by which the security project plan is accepted and paid for; therefore, these metrics can literally mean the difference between profit and loss. With internal projects, there usually isn't the issue of whether or not you'll get paid; the issue is whether or not the security project plan will be deemed completed and successful. At some point in our careers, most of us have been in a situation where it almost feels like a pair of six-year-olds quipping back and forth, "Did too!" and "Did not!" This is the kind

of situation you would always prefer to avoid rather than repair, and the best way to do that is through *acceptance criteria*.

By definition, acceptance criteria are "pre-established standards or requirements that a project must meet." When you define these at the outset of the security project plan, you can get all of the relevant stakeholders on the same page and agreeing to the same results. Referring back to the user example: Suppose the Accounting department wants users to have to login a second time to examine specific documents, but the Sales department wants one fast, easy way for its remote sales people to login to the network and retrieve accounting documents, sometimes over intermittent or slow connections. If you appease one group, you'll likely frustrate the other. In this case, you'll end up with a security project plan that is deemed a failure (or at least, less than successful) by one group or the other. To avoid this kind of push-pull, define requirements and acceptance criteria for each of your individual security area project plans, as well as for your security project plan as a whole. Make special note of any legal, financial, regulatory, or industry requirements that must be met, and build them into your acceptance criteria.

Quality Metrics

Specific *quality metrics* may be difficult to define for your security project plan because, as stated at the outset of this chapter, the definition of quality is "the complete absence of incidents." If your network is targeted by hackers and they fail to get in, you have achieved a quality result. While you and your IT staff understand the enormity of that result, it's entirely possible that those outside of the IT world see it as a non-event. In reality, it is the compilation of non-events that proves that your security solution delivered the requisite level of quality.

Still, there are ways to define quality. For instance, when implementing intrusion prevention and intrusion detection systems, you want 100 percent prevention and zero percent intrusion. Is this a reasonable and achievable metric? It would seem so since even one intrusion could cost your company millions of dollars, thousands of hours, and untold

cost to the company's reputation. However, let's remember that quality has to do with errors and defects. Even if you have zero errors or defects, a network intrusion can still occur for other reasons; therefore, you need to define the level of errors or defects that are acceptable. While most everyone would agree that zero is the best, it's usually not an achievable metric. You will have to balance your time and budget against your quality metrics to make sure you're delivering a reasonable solution. If you wait until it's perfect, you will never implement anything. Be clear about where you're drawing the line and why. Document what exposure these choices might create, and the decision-making process that you used to balance the time and cost against results, so that you have a clear record of this process that you can refer back to later, or that you can provide to management should you be asked.

As discussed previously, quality needs to be balanced against the constraints of the security project plan. For example, to meet the requirement of 100 percent prevention of intrusion, you might have to spend twice as much as is in your budget. On the other hand, you might be able to achieve 95 percent prevention within your budget. The cost also has to be counterbalanced with the amount of risk your company is willing to accept, as well as the value of the data being protected. This equation is critical to understanding and addressing the cost of security. The acceptable level of risk and the value of the data being protected should be kept in mind throughout your IT security project planning process.

In many cases, quality is not a discrete deliverable, but rather a mindset and a cultural attitude. When you focus on quality in your IT security project plan, you are sending a message to the team and the company that quality is critical to the success of the project. Defining and quantifying it in ways appropriate to your specific security project plan is a worthwhile endeavor. You might be surprised at how you and your team can define quality within the framework of an IT security project plan.

Change Management Procedures

We talk more about *change management* procedures later in the book, so we'll just touch on them in this chapter. Changes to the security project plan are made throughout the project lifecycle. While changes are inevitable, it doesn't mean that you shouldn't strive to reduce the number of required changes with thorough and thoughtful up-front planning. That said, change will always happen for a variety of reasons; therefore, having a well thought-out plan for managing change is crucial to both project quality and project success. As you know, when you change a setting in one place, it can have a ripple effect throughout the network. The same is true of any change you make to your security project plan; it can (and usually will) ripple throughout your security project plan. In addition, because you're specifically focusing on security, you also have to recognize that changes to a security project plan can have both intended and unintended results. The unintended results are those unexpected things that occur as a result of inserting change into an already thought-out security project plan.

Create, implement, and manage your change management process for your security project plan to ensure that the final result meets quality and security standards. If change is allowed to run rampant throughout your security project plans, you will deliver a patchwork project that will more than likely leave you exposed on several fronts. While change management won't prevent all errors, omissions, or gaps, it will certainly help reduce them.

Standard Operating Procedures

Standard operating procedures (SOPs) are sometimes outside the scope of a security project plan, but in the world of network security, they must tie in very tightly to the project. Once the security measures have been implemented and tested, someone has to take ownership of maintaining security throughout the organization. This is done through a number of measures including training, the defining and adherence to SOPs, and the designation and training of an incident response team (also called a

Computer Security Incident Response Team [CSIRT]). Therefore, it's advisable to include the definition of SOPs in the IT security project plan itself.

Think about it. Your security project plan team will include representatives from many key areas of your organization. What better time to develop procedures for maintaining security than when you have the key people responsible for doing so participating in the planning? While you may need to circle back to this team toward the end of the project lifecycle (since you probably cannot define SOPs until you've defined and implemented your specific security solutions), these key people should be involved in defining the SOPs that will help maintain security. If you implement a security solution that is virtually impossible to monitor and maintain, have you delivered a quality project? You may have delivered a top quality security project plan result, but users and management are not likely to perceive the project as high-quality if it leaves them completely overwhelmed or confused about how to maintain security on a day-to-day basis. Therefore, keeping an eye on SOPs, not just for the IT team, but for users, department leads, and others, will help make sure your security project plan not only delivers a quality result, but will also help ensure that the result is maintained over time.

Monitoring IT Security Project Quality

Another element of planning *IT security project quality* is defining how you'll monitor the quality of your security project plan. Suppose one of your projects is to harden network servers? Each server will be audited to determine the server role, the types of connections, the criticality of the server's data (or role), and the types of security measures that will be applied. Next, the servers will be hardened one-by-one based on the audit results and the relevant security standards.

Let's look at a brief example. Suppose Max is assigned to harden the database server. On the day that Max is changing some of the server settings, he's distracted by a personal problem he's having and he's been on the phone throughout the day trying to sort it out. He's not paying close

attention and he accidentally skips over three critical steps. He mistakenly reports that he's completed the task and everything is set. That evening, the servers are taken down for routine maintenance and the database server fails to come back online properly. Another member of the IT staff, Jill, knows that Max worked on the server earlier that day, so she calls him at home and asks him what changes he made. He says that he made the changes indicated in the security project plan for the database server. Jill thanks him and hangs up, and goes to work troubleshooting the server, without success. Frustrated, she pulls the security project plan out, looks at the list of steps to be taken, and walks through them again. She discovers that three key steps were missed. She makes the necessary changes, makes some notes, reboots the server, and it comes back online as expected.

This is the most benign scenario—several hours of Jill's time were wasted due to an error on Max's part. We might chalk it up to "these things happen," except that this was completely avoidable. In addition, while it was just a waste of a few hours' time, suppose this database was the backend on an e-commerce server and a database error exposed critical data to users over the Internet? Suppose credit card data were exposed and hackers managed to grab some of that data before the server was finally locked down. Max's error could have been very costly to the company, and all because he was simply not paying attention.

So, given that these kinds of things do happen, how can you devise a security project plan to monitor the quality of your project as you move forward? One very easy way to implement the method is to create "completion criteria," which are checklists with a bit more muscle. Completion criteria are requirements that define the successful completion of a task. (When we look at defining IT security project plan tasks later in this book, we'll revisit completion criteria and look at them in a bit more detail.) However, if Max had used a checklist that described the steps needed to successfully harden the database server, he would have been less

likely to miss three critical steps. Depending on how your company works and what its culture is like, you may choose to print out tasks with completion criteria on them and require the IT staff person (in this case, Max) to initial them after each step is completed. This gives you a paper trail to follow should things go wrong.

Completion criteria can also be used by a second person, whose primary responsibility is project quality. For instance, while all security tasks are important, you and your team may designate 10 or 20 tasks as being absolutely critical to a sound security project plan. You may have a second person check the results of these tasks as a backup (or insurance) plan. While no one likes having their work checked, most people understand that when security is at stake, the more eyes the better. Having work double-checked can help protect the company, the network, and the IT staff. Avoiding errors is always better than repairing errors. Therefore, using completion criteria as a checklist for quality assurance is a simple, easy-to-implement tool to help maintain the highest level of quality (and security) possible.

This is just one example of how you can monitor quality in your security project plan. By making quality a constant focus and part of your team's culture, you can find ways to monitor it throughout the project lifecycle that are meaningful and appropriate to your IT security project plan.

Another reminder: If your company is subject to industry or governmental regulations, you may have very specific monitoring requirements that your security project plan must conform to. Even if those regulations do not have specific monitoring requirements, you can review the regulations and develop meaningful monitoring requirements for your project.

Business Intelligence…

Stages of Security Acceptance

An article by Andrew Briney back in 2004 does a great job of helping you understand how Chief Executive Officers (CEOs) and corporate executives typically view security. Briney identifies four stages of acceptance regarding the importance of security to an organization. Even today, two years later, studies show that CEOs are still reluctant to spend money on security despite the very clear risks it poses to the organization. It probably falls back to the concept we discussed, that it's hard to quantify the cost of something that does not occur. No one likes to pay insurance, but most of us insure our cars, homes, and possessions just in case. Helping CEOs understand that security spending is one of the most important insurance policies they can have, will help you in moving your CEO up the ladder of acceptance from "security is a necessary evil" to "security is quality." To read the full article, follow this link: http://infosecuritymag.techtarget.com/ss/0,295796,sid6_iss407_art819,00.html.

Testing IT Security Project Quality

As part of your IT security project planning process, you should develop thorough testing security project plans for each of your Individual Security Area Projects (ISAPS). You can do this in any number of logical ways, depending both on the nature of the project and the culture (and sometimes budget) of the company. For example, your test security project plans should test each area secured during the project, and an overall quality security project plan should also run a more comprehensive test on the entire network. That comprehensive security project plan would be designed and developed in tandem with your ISAPs so that there are no gaps in between the various ISAPs. Since network components work together in an integrated way, it doesn't make sense to create testing silos

that don't test security as data or users moving across physical and logical boundaries. In other words, your test security project plans have to mimic real life scenarios, likely attacks, and normal (and abnormal) usage.

Based on your risk analysis, your testing should accommodate those identified risks. However, you might also consider creating tests that don't follow normal rules. Hackers are pretty creative and users can do weird things, so while specifically addressing known risks is vital, so too is addressing the potential problems you can foresee.

How much testing is enough? Again, the answer depends on how confident you want to be in your security solutions. You'll need to strike a balance, but you should err on the side of more, not less, testing, to ensure your security solutions are working as intended. If you have very serious security needs, you may choose to work with an external security consulting firm to help test your solutions to make sure you're not falling victim to "group think." While an external firm could be used for an end-to-end security project plan (from assessment to implementation to testing), it might make sense for some companies to hire a firm simply to test the results of the project. Human nature is such that if we devise the security project plan, we're also likely to test the security project plan and perhaps not test the actual security. Therefore, you might want to have a different IT group test the security than those who implemented it. This also helps ensure there is no collusion among IT staff in terms of leaving back doors open or creating security holes for their work convenience. Since most security breaches occur from within the organization, you have to build in safeguards against potential collusion, whether among IT staff or among others. In a perfect world, we could simply trust everyone, but we know that in this world, we have to take reasonable safeguards. Your testing security project plans should cover every angle, not just the ones you cover in your IT security project plan.

Summary

In this chapter, we included a section on quality for the simple reason that a high-quality security project plan is clearly preferable to a low-quality security project plan. There are numerous ways you can ensure you deliver a high-quality security project plan and result. Quality is delivered through developing a thoughtful and thorough IT security project plan. This is accomplished first through planning activities. User requirements are a great place to start, because if users' requirements are not addressed, they will circumvent security in order to get their jobs done. Technical requirements are developed after functional requirements so that you make sure your security systems have all the required functionality to deliver a strong security solution before specifying the technical elements for the project. It's easy to get distracted by defining how many failed attempts to access a resource should be allowed before an administrator is notified (technical requirements), and the failure to list administrator notification of failed attempts as a functional requirement. While they go hand-in-hand, it's a good practice to first define functional requirements so there are no gaps in your security project plan. Acceptance criteria are typically tied to user, functional, and technical requirements. These criteria should be specified and agreed to at the outset of the planning phase to ensure that all stakeholders are on the same page with regard to expected deliverables. When acceptance criteria are defined in advance, it gives everyone a stationary target to shoot for when developing and delivering a security project plan.

Your IT security project plan should also include thorough plans for monitoring and testing IT security project plan results. Remember, there are two separate entities here—the IT security project(s) itself and the ongoing IT security maintenance. We're primarily focused on the IT security project plans themselves, although you may create a security project plan that defines how ongoing security will be maintained, monitored, and tested.

Solutions Fast Track

Planning IT Security Project Quality

- ☑ Planning quality in an IT security project plan means that quality must be a mindset, not just a specific set of deliverables.
- ☑ Quality can be managed through several planning mechanisms, including well-defined user requirements, functional requirements, technical requirements, acceptance criteria, and quality metrics.
- ☑ Including user requirements helps the IT security project plan meet the end-user's needs, and will ultimately yield higher quality and better security.
- ☑ Defining functional requirements helps ensure all functionality required by the various systems is included in the security project plan.
- ☑ Defining technical requirements typically follows defining functional requirements.
- ☑ Quality metrics can sometimes be difficult to define for a security project, but looking for opportunities to quantify this data will improve quality and security.
- ☑ Change management procedures ensure that when changes are needed to the security project plan, they are evaluated, implemented, tested, and integrated in a manner that maintains or increases security.
- ☑ Standard operating procedures are used to ensure that the security solutions that are implemented are maintained on a day-to-day basis. Involving key stakeholders in defining SOPs at the appropriate time can improve the actual and perceived quality of your project.

Monitoring IT Security Project Quality

- ☑ Quality begins in the planning stages, but is implemented throughout the security project plan lifecycle. Monitoring the quality of project results is critical to a successful project.
- ☑ How you monitor quality throughout your IT security project plan will depend, in part, upon the nature of the project. Work with your team to identify ways to monitor quality as you complete project work.
- ☑ One way to manage and monitor quality is to use completion criteria for each task. This is a list of steps that must be completed before a task is considered complete.
- ☑ Completion criteria can be used by the person performing the task to ensure all steps are done in the proper sequence.
- ☑ Completion criteria can be used by someone monitoring quality to perform "spot checks." In some situations, it might be appropriate to have someone step through the most critical security tasks a second time to double-check results.

Testing IT Security Project Quality

- ☑ Testing IT security project plan results is the third element of delivering a quality IT security project plan. If you don't test the solutions you put in place, you have no idea if an attacker would be successful or not.
- ☑ Develop a test security project plan for each area of security you will be working on. Also develop an overall security project test plan so that you don't focus solely on the areas you've worked on.
- ☑ You might choose to hire an outside consulting firm to test your security to uncover blind spots you and your team may have.
- ☑ If you choose to design, implement, and test your own security solutions, make sure you get a broad representation of people together who can help think of all the things that could go wrong.

- ☑ Errors and omissions are the biggest holes in security and are the ones attackers are most likely to find and exploit.
- ☑ There is no single right answer to the question, "How much testing is enough?" You and your team will need to assess the risk of attack and the consequences of a successful intrusion or attack, and determine at what point you feel comfortable that your systems are a secure as they can be within your company's budgetary constraints.