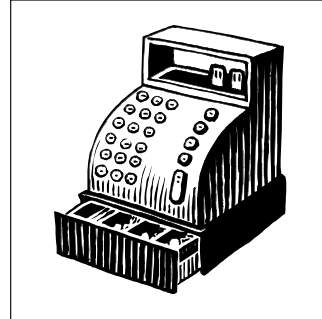


# 3 The Framework



What is a framework? Moreover, how does it apply to attacking a system? Finally, is a framework a methodology? A framework is collection of measurable tasks, whereas a methodology is a specific set of inputs, processes, and their outputs. A framework provides a hierarchy of steps, taking into consideration the relationships that can be formed when executing a task given a specific method.

For example, this book presents a framework of steps with options within each and they appear as chapters, headings, and so forth. The context within each section of this book introduces methods for performing certain tasks heeding the value represented by other points within the framework. When combined, an entire process geared towards value can be presented.

By formatting ethical hacking in a framework, as opposed to simply a collection of methods and tactics, elements can be easily removed and added to accommodate specific requirements of the test. Of course, the removal of a particular element within the framework can have repercussions when the goal of the entire framework is value.

How this applies to penetration testing is in ensuring the value of the test is realized. Given that a penetration test is part of a larger security program, one must include other characteristics of security to align the test appropriately to the demands driving it. Moreover, a framework highlights each phase, drawing relationships between them to make sure you're on track with the objectives. In addition, each step in the phase helps you take into account the nuances of performing a controlled attack. For example, there are limitations, inherent and imposed, that will have effects on each phase translating into varying degrees of value. Finally, it provides operational structure to the test. Knowing how and when to perform a task is as important as the task itself.

The mission of the framework is to explain the steps, their relation to other points within the performance of a test, and to expose the impact on value when excluding various methods within each. In the simplified Figure 3.1, we see each primary phase of the framework with points within each representing a task or value element. Some circles are larger than others, signifying more potential value. Depending on what tasks are not employed, some downstream elements may not be available simply because the required information or results from previous elements do not exist. Given that the framework is founded on related processes that span phases, the use (or omission) of a process will limit the availability or effectiveness of other processes.

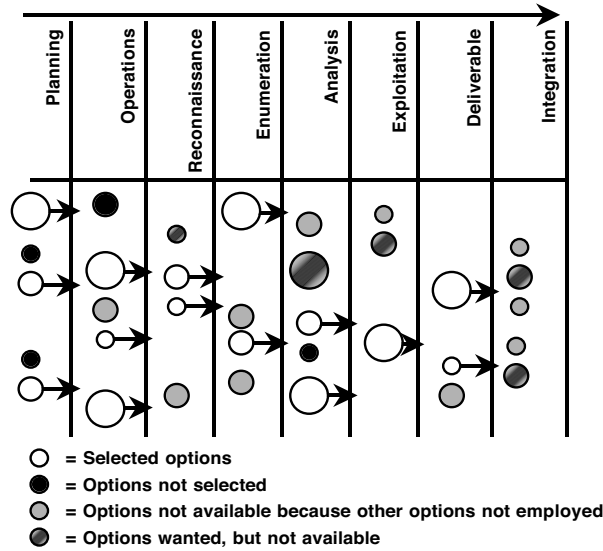


FIGURE 3.1 Determining the Impact on Value Based on Selected Options

Of course, for your specific goals of the test, the unselected or unavailable elements may prove to be of little or no value and therefore the impact is nonexistent. The important fact to evaluate is which elements are needed to meet your goals and understand there may be an inherent relationship to another point within the framework you have not considered or do not want to be exercised. The ability to gain visibility into the affiliation between one phase and another is the value a framework brings to the entire process.

While in its infancy, ethical hacking meant simply attacking a network and exploiting any vulnerability presenting itself; that was the goal—get in. And, quite frankly, this is still the M.O. for many engagements today. The tools have changed, the techniques are much more sophisticated, the knowledge of the consumers is much more comprehensive, but the essence of the test has remained much the same. Technique and tools are important and provide a strong foundation for further evolution, but with regard to security, the environment is too dynamic to base success on technique and tools alone. Racquetball is one of those sports of technique and tools: insightful volleys and a good racquet will win the match. However, the court does not change in size, the lines don't move, the back wall will always be there, and the environment is predictable.

With the absence of continuity, value rests on the shoulders of the tester and the framework that is followed. The ability to assess the situation and make quick determinations based on similar experiences is an attribute of a successful attack by today's standards.

On the other side of the equation is the recipient of these tests attempting to make value decisions based on his impression of a planned attack, an impression fed by security consultants, magazines, friends, and employees and not from extensive experience in being the target of hundreds of tests. I liken it to asking a regular



person to purchase food for a restaurant. They know what food is and have an understanding of value and use, but buying 250 pounds of meat, 10 gallons of mayonnaise, 25 pounds of cheese, and 8 boxes of detergent would challenge anyone not familiar with the process.

After performing and being involved with many penetration-testing engagements, there is a theme that begins to surface. People are not fully aware of the options available to them and how to apply those options to their environment. Many characteristics have varying degrees of intensity and requirements, such as information and limitations, that will influence other areas of the test and how they relate to the value of the test in an overall security program.

### PLANNING THE TEST

As with anything worth doing, proper planning is essential to performing a successful project. Planning provides an opportunity to evaluate existing business demands and processes, how they relate to a new business endeavor, and to make choices on which characteristics are worth doing and those in which you're not willing to accept risk.

Existing security policies, culture, laws and regulations, best practices, and industry requirements will drive many of the inputs needed to make decisions on the scope and scale of a test. Arguably, the planning phase of a penetration test will have a profound influence on how the test is performed and the information shared and collected, and will directly influence the deliverable and integration of the results into the security program.

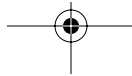
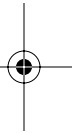
Planning describes many of the details and their role in formulating a controlled attack. Security policies, program, posture, and ultimately risk all play a part in guiding the outcome of a test. What drives a company's focus on security, its core business needs, challenges, and expectations will set the stage for the entire engagement.

### SOUND OPERATIONS

How is the test going to be supported and controlled? What are the underlying actions that must be performed regardless of the scope of the test? Who does what, when, where, how long, who is out of bounds, and what is in bounds of a test all need to be addressed. Logistics of the test will drive how information is shared and to what degree (or depth) each characteristic will be performed to achieve the desired results. Operational features will include determining what the imposed limitations of the tester are and how they are evaluated during the test.

### RECONNAISSANCE

Reconnaissance is the search for freely available information to assist in the attack. The search can be quick ping sweeps to see what IP addresses on a network will respond, scouring newsgroups on the Internet in search of misguided employees divulging useful information, or rummaging through the trash to find receipts for telecommunication services.





Reconnaissance can include theft, lying to people, tapping phones and networks, impersonations, or even leveraging falsified friendships to collect data about a target. The search for information is only limited by the extremes to which a customer and tester are willing to go.

The reconnaissance phase introduces many of the questions surrounding what actions truly provide value to the company. In this section, we examine the reconnaissance techniques, such as social engineering, and the necessary environmental characteristics that must exist to realize value from intense investigation. It is also in this section that the value of a certain type of test is questioned, which exposes the effects of poor planning or a poor understanding of limitations applied to the test.

Reconnaissance offers a plethora of options, each related to one another. However, unlike other phases within the test's framework, each option can be controlled, moderated, and measured to a surprisingly high level of granularity. Therefore, the relationship between the framework, tasks, and methods will become very clear.

## ENUMERATION

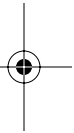
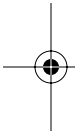
Enumeration (also known as network or vulnerability discovery) is essentially obtaining readily available (and sometimes provided) information directly from the target's systems, applications, and networks. An interesting point to make very early is that the enumeration phase represents a point within the project where the line between a passive attack and an active attack begins to blur. Without setting the appropriate expectations, this phase can have results ranging from "Oops" to "Do you swear to tell the truth and nothing but the truth?"

To build a picture of a company's environment there are several tools and techniques available to compile a list of information obtained from the systems. Most notably, port scanning is the "block and tackle" of the enumeration and NMap is today's most valuable player. The simplest explanation of a port scan is the manipulation of the basic communication setup between two networked systems using TCP/IP as a communication protocol. TCP/IP uses a basic session setup that can be used to determine what application ports a system is willing to use to establish communications.

Simply stated, port scanning is a way of detecting where a computer responds to requests to make connections. More technically, the TCP protocol has what is commonly known as the "three-way handshake" that is used to start TCP connections:

1. Computer A sends a message called a "SYN" (Synchronize) to Computer B.
2. Computer B acknowledges that message with a "SYN+ACK" (SYN with an Acknowledgement) to Computer A.
3. Computer A sends back an acknowledgement—"ACK."

Obviously, collecting information about systems is the first step in formulating an attack plan. However, information collected during the reconnaissance phase can be added to help build a picture of the target's systems and networks. It is one thing





to collect information and it is another to determine its value, and the perceived value in the hands of a hacker. On the surface, enumeration is simple: take the collected data and evaluate it collectively to establish a plan for more reconnaissance or building a matrix for the next phase, vulnerability analysis. However, this is the phase where the tester's ability to make logical deductions plays an enormous role. It is also the reason why great testers (and hackers) are not taught; they are grown.

As mentioned earlier, hacking is an art form, the ability to use rules and predictability to your advantage. Computers, if nothing else, are masters of rules and performing repeatable tasks perfectly (well, most of the time). The talent required to manipulate this rigid environment is rare. To accomplish this, a human's intellect will resolve problems by combining seemingly disparate information to formulate a hypothesis for other avenues of attack. Enumeration is inventorying all the collected information to build logical threads to circumvent the security controls of a network, system, or application.

## VULNERABILITY ANALYSIS

There is a logical and pragmatic approach to analyzing data. During the enumeration phase, we try to perform an interpretation of the information collected looking for relationships that may lead to exposures that can be exploited. The vulnerability analysis phase is a practical process of comparing the collected information with known vulnerabilities.

Most information can be collected from the Internet or other sources, such as newsgroups or mailing lists, which can be used to compare information about the target to seek options for exploitation. However, information provided by vendors and even data collected from the target can be used to formulate a successful attack.

Information collected during the reconnaissance phase from the company can provide information about vulnerabilities unique to its environment. Data obtained directly from the company can actually support the discovery of vulnerabilities that cannot be located anywhere else.

As mentioned above, information found on the Internet is very helpful. Known vulnerabilities, incidents, service packs, updates, and even available hacker tools help in identifying a point of attack. The Internet provides a plethora of insightful information that can easily be associated with the architecture of the target.

## EXPLOITATION

A great deal of planning and evaluation is being performed during the earlier phases to ensure a business-centric foundation of value is established for the test. Of course, all of this planning must lead to some form of attack. Exploiting systems and applications can be easy, such as running a tool, or intricate, with fine-tuned steps executed in a specific way to get in. No matter the level of difficulty, good testers follow a pattern during the exploitation phase of a test.

During a penetration test the details considered in the planning come into full view and affect the outcome of every action taken by the tester. A sound course of



action is needed to translate the planning into an attack to meet the objectives within the specified period and within the defined scope.

The attack process is broken up into threads and groups and each appears in sets of security. A thread is a collection of tasks that must be performed in a specific order to achieve a goal. Threads can be one step or many in a series used to gain access. Every thread is different but may have similarities that can be useful. Therefore, threads can be combined into groups to create a collection of access strategies. Groups are then reviewed and compared to support comprehensive attacks using very different threads in a structured manner.

Each test is evaluated at every point within the operation to ensure the expected outcome is met. Each divergence from the plan is appraised to make two fundamental determinations:

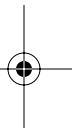
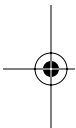
1. *Expectations.* Are the expectations of the thread or group not being met or are the test's results conflicting with the company's assumptions? The objective is to ensure each test is within the bounds of what was established and agreed upon. On the other hand, if the test begins to produce results that were not considered during the planning, enumeration, and vulnerability analysis phases, the engagement needs to be reconsidered or, at a minimum, the planning phase needs to be revisited. Meeting expectations is everything and in the world of ethical hacking it can represent a fundamental challenge when not planned properly or not executed according to the plan.
2. *Technical.* Is a system reacting in an unexpected manner, which is having an impact on the test and the engagement as a whole? Much more granular in theory than general expectations of the test, technical gaps are literally the response of a system during the test. Keeping your eyes open for unexpected responses from systems ensures you have not negatively affected the target or gone beyond the set scope of the test.

The exploitation phase is an opportunity to discuss the tactics of performing the test rather than focusing on the tactics of the exploitation itself.

## FINAL ANALYSIS

Although the attack process has many checks and validations to ensure the overall success of the engagement, a final analysis of all the collected data and exploits must be performed. Vulnerabilities associated with the test need to be categorized to determine the level of exposure and to assist in supporting a well-defined deliverable and mitigation plan. The final analysis phase provides a link between the exploitation phase and the creation of the deliverable.

The first goal of the analysis is to take a comprehensive view of the entire engagement and look for other opportunities that may exist, but are not directly observed. The idea is to build a bigger picture of the security posture of the target's environment and classify vulnerabilities to communicate the results in a clear and useful manner.





The final analysis is part interpretation and part empirical results. To define something as critical with little evidence can become problematic when presented to the recipient of the test. However, if there is enough evidence from other threads and groups that prove the vulnerability could represent a substantial risk, it becomes much more palatable and easier to communicate in terms of value and remediation.

## DELIVERABLE

Throughout the history of penetration testing there have been deliverables communicating the results of the test in numerous ways. Some are short, only listing the identified vulnerabilities and where to find the patch to fix them. Others are cookie-cutter reports from tools that simply state which port was open, the vulnerability it represents, and where to find the patch. And, there are some that detail every move made by the consultant: how she found a hole, got the etc/shadow, cracked the passwords, and took over your shipping application . . . and, of course, where to find the patch.

Are these examples of poor deliverables? In reality, no. These are simply the results of a technical test performed in conjunction with the demands of the company. Many organizations place so many controls on the test (or the lack of controls) that a comprehensive deliverable is difficult. The only avenue of the tester is simply to state the facts. In addition, ethical hacking has become so commoditized that if a deliverable doesn't drive fear into the hearts of the executives it could be considered a failure.

In contrast, I have seen reports from many companies and individuals that are, in a word, exceptional. They provide insightful commentary, step-by-step details, and rank the vulnerabilities to the best of their knowledge and understanding of the customer's business. They provide measurable levels of risk, raw results from the test, where backdoors are, how they were placed, and how to remove them. Some include status reports and all correspondence associated with the engagement. Finally, how the engagement was planned, what the drivers were, and the overall expectations, the imposed limitations, and their impacts are also included.

It is my expectation that the recipients of deliverables—good and bad—would like to know what a good deliverable should include and ultimately how to translate it into valuable security improvements. The above list contains only a few of the general characteristics of a good deliverable. In the chapter on deliverables, we take a much closer look and discuss sound practices associated with exceptional deliverables.

## INTEGRATION

Finally, how to use the test to your full advantage is directly dependent on the proposed integration process. There are several assumptions within this chapter, one of which is that the penetration test actually found something and followed many, if not all, of the previous phases. Another is that the deliverable communicates all the necessary information needed to actually support some form of integration. Of



course, the deliverable can be combined with existing materials, such as a risk analysis, security policy, previous test results, and information associated with a security program to enhance mitigation.

There are three distinguishing factors that should be considered during the integration of any test results:

1. *Mitigation.* If something were found that represented a threat to secure operations and was beyond acceptable risk, then it would need to be fixed, to put it bluntly. Of course, there are the easy things to rectify and there are very complicated solutions to seemingly simple problems. Mitigation of a vulnerability can include testing, piloting, implementing, and validating changes to systems.
2. *Defense.* How should you address the insecurities in a strategic manner? What about your networks, systems, applications, and policies that need to be addressed to ensure sound practices are employed to minimize the impact of future or undetected vulnerabilities? Defense planning is establishing a foundation of security to grow on and ensure long-term success.
3. *Incident Management.* Arguably, the core element of security—the ability to detect, respond, and recover from an attack—is an essential part of any security program. Knowing how you were attacked, the vulnerabilities exploited, and the potential impacts aids in formulating an incident response plan. The test provides an opportunity for you to learn about the various weaknesses and attractive avenues of attack. Finally, you get an understanding of critical points in the network that may need more attention than others, and this may not be the perimeter as normally assumed.

So we've covered all the bases, at least the big ones: fix what is broken, establish a plan to protect you from future mistakes and oversights, and prepare for a real assault on your company. This is what you can expect from a well-structured penetration test. Penetration tests can be a valuable component of a security program. They can provide fascinating insights to the presumed security of an organization and the actual security employed. Tests can also assist in defining acceptable levels of risk and exposure and set the foundation for future security developments.

