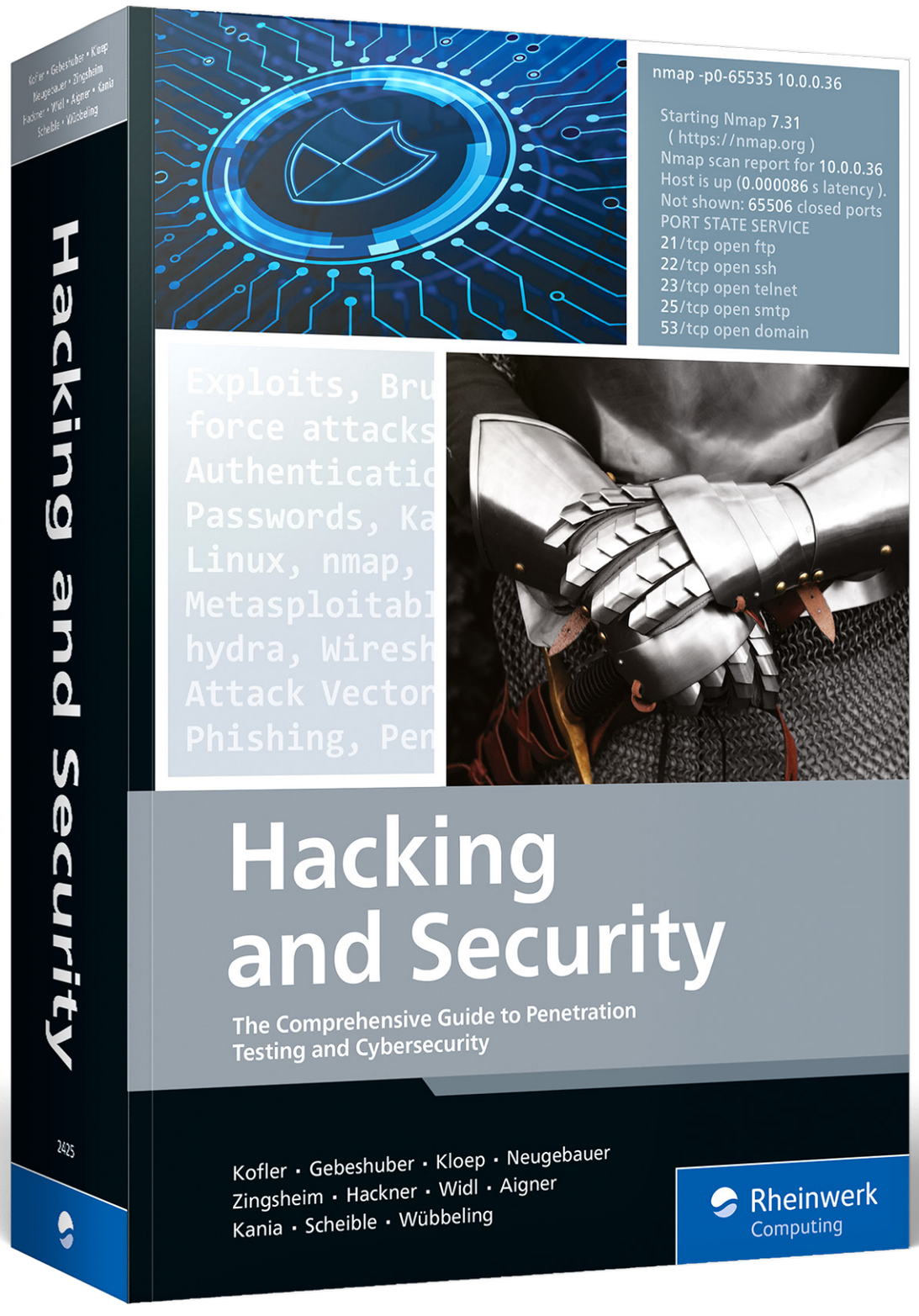


Reading Sample

In this chapter we'll cover the fundamental questions to consider when it comes to the security of websites on the internet. Professional checks or audits have become a standard measure to minimize risks; however, the services offered vary greatly in quality and scope, making it difficult to make a suitable selection without research. This chapter discusses when to ask for external help, to what extent this makes sense, and what precautions and requirements are involved.

"External Security Checks"

Contents

Index

The Authors

Michael Kofler et al.

Hacking and Security

The Comprehensive Guide to Penetration Testing and Cybersecurity

1141 pages | 07/2023 | $59.95 | ISBN 978-1-4932-2425-8

www.rheinwerk-computing.com/5696

Chapter 10
External Security Checks

When it comes to the security of websites on the internet, professional checks or audits have become a standard measure to minimize risks. However, the services offered vary greatly in quality and scope, making it difficult to make a suitable selection without research.

Fundamental questions to consider, for example, include when to ask for external help, to what extent this makes sense, and what precautions and requirements are involved. Such questions arise as soon as you make the decision to have a security check performed. As a reader of this book, you'll very likely be confronted with these questions sooner or later, so we want to support you in this process with this chapter.

10.1 Reasons for Professional Checks

The first decision you need to make is whether you want to commission a professional check. With this book, we want to support you in the best possible way to carry out checks yourself. Due to the complexity and depth of the topic, however, we can at best facilitate your entry into the various subject areas; further examination of the individual topics is indispensable in order to be able to constantly identify new and adapted weak points.

Whether you use the services of external penetration testers thus depends, among other things, on your own expertise and experience. If you have the necessary resources to deal with the subject on a daily basis and can thus gain experience yourself, this book is an ideal starting point for carrying out as many checks as possible by yourself in the future.

You may also already have your own internal department of experts who can perform checks for you. If you're only dealing with the topic in passing, then you may have gathered enough knowledge in this book to at least be able to perform automated vulnerability scans yourself (a detailed description of this type of test follows in Section 10.2). Using such scans, you can raise the basic level of your security even with few resources and help your company implement basic security procedures.

For critical systems or for very customized implementations of applications or websites, you should call in professional support. Although many tools support the process

of technical security checks, vulnerabilities in complex environments and specially developed program code can actually only be found through the experience and knowledge of professional testers. In addition to the technical checks, they can then assist with the secure architecture and perform threat analyses with their experience.

Another reason to bring in external support may be that an independent third opinion is needed—for example, if there is a risk of operational blindness because one's own team has already been entrusted with securing or programming the system under test. External testers are also needed when a third independent party is required to confirm or assess the security of the product, system, or network. This is usually the case during acceptance tests or when you purchase third-party products.

10.2 Types of Security Checks

Each check is a snapshot of the security of a given scope at a given point in time, with no guarantee of having found all vulnerabilities. When a vulnerability is identified, its existence is proven. If a vulnerability is not found, its existence cannot be ruled out.

The more time and knowledge you invest in the investigation, the more detailed the check can be, and the higher the probability of having found a larger proportion of the vulnerabilities that may exist. In order to keep the effort in relation to the value of the object to be checked or the available budget, different types of security checks are basically available.

10.2.1 Open-Source Intelligence

The term *open-source intelligence* (OSINT) originates from the military environment and describes an approach in which information is collected from publicly available data sources and its correlations are interpreted in a targeted manner. Intelligence is thus created from individual, often unrelated data points. For the area of security audits, this means that OSINT analyses use publicly available data on the internet to look for vulnerabilities, points of attack on a company, or key people in the company. Common activities include the following:

- Searching for relevant files via search engines such as Google or Bing
- Searching for information about deployed infrastructure components—for example, via advertised jobs or questions asked by employees in internet forums
- Extracting metadata—such as user names, operating system information, local paths, or software versions used—from files found on the internet
- Searching for personal interests and hobbies of selected employees who could be possible targets—for example, with the help of social media, such as Facebook, Twitter, or Runtastic

- Searching for deployed software with known vulnerabilities via infrastructure search engines, such as Shodan or Censys
- Searching databases published by hackers, so-called database leaks, on the internet

This list isn't exhaustive and will be adapted and extended accordingly depending on the objective of the analysis. An example of this is shown in Figure 10.1 where a search for "bund.de" is carried out using the Maltego OSINT tool. This tool examines published files for metadata, such as the version of Microsoft Word used to create the document.

Performing OSINT analyses can be useful, for example, when:

- internal security guidelines prescribe a certain way of handling information externally and whether these guidelines are being adhered to is to be checked,
- no security guidelines exist and an initial investigation should provide an assessment of whether internal data has already been published on the internet or has been unintentionally made accessible via the internet,
- individuals should be trained in the handling of both company and private data,
- likely targets of spear phishing attacks are to be identified and targeted phishing attacks (also called *whaling*) are to be prepared for, and
- possible points of attack on the infrastructure must be searched for without directly attacking the target company.

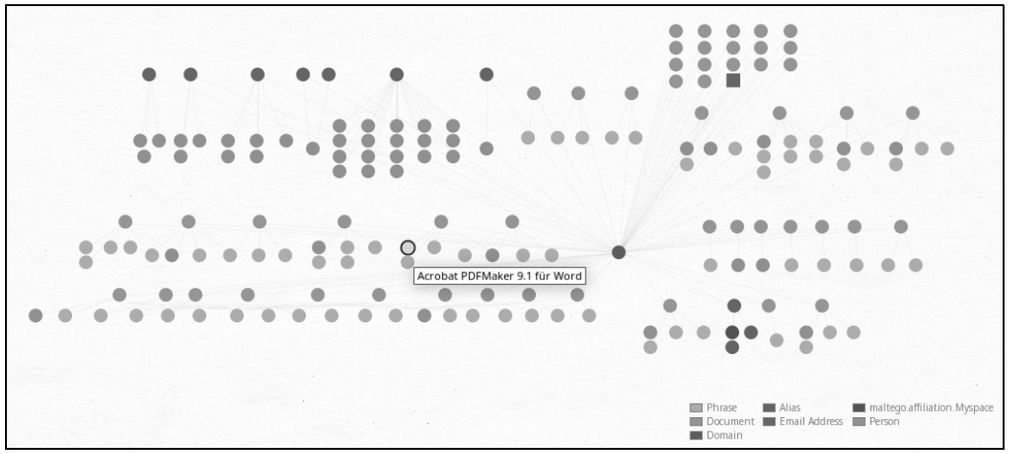


Figure 10.1 Information about Software Versions in Use Can Be Found Semiautomatically via OSINT

OSINT analyses may already be performed as part of other checks, such as red-teaming or spear-phishing assessments. However, these are usually very specific to the assessment in question. If a somewhat broader approach is also desired, this should already be included in the bidding phase.

A regular check of database leaks can be performed by companies themselves. This enables an early response and the ability to warn employees when accounts are hacked on external websites where a company email address has been used. In this way, the risk to the hacked account and the risk of the possible reuse of the password used there can be actively addressed.

10.2.2 Vulnerability Scan

The execution of a vulnerability scan is usually automated. The goal of the test is to perform a security check with as little effort as possible, identifying mainly easy-to-detect vulnerabilities, such as those that would be used by simple attackers (like script kiddies).

The main tools used are `nmap`, Nessus, Nexpose, OpenVAS, and other well-known vulnerability scanners. Figure 10.2 shows an example using the OpenVAS open-source vulnerability scanner.

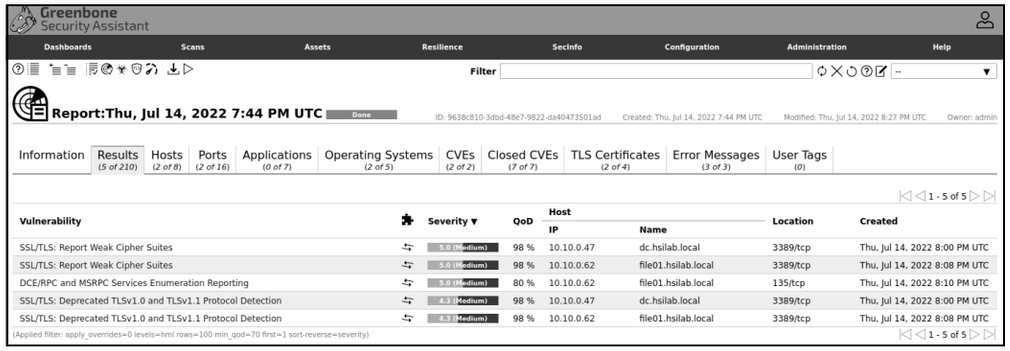


Figure 10.2 Example of Results from a Vulnerability Scan with OpenVAS

The costs of a vulnerability scan are generally lower than for targeted assessments because the proportion of manual checks is reduced to a minimum. Depending on the external provider, the client receives either direct output from a tool or a separate final report in which the tool results have already been cleaned of false results (*false positives*) and prepared in a way that is understandable for the client. Common activities include the following:

- Performing port scans to detect available services
- Performing a vulnerability scan with one or more vulnerability scanners
- Depending on the provider, the removal of incorrect results and preparation in the form of a separate report

Performing vulnerability scans may be useful, for example, when:

- no previous security check has taken place and an initial assessment of the security of standard infrastructure components and applications is to be made;

- regular, comparable scans of the infrastructure are to be performed, such as for early detection of possible security-relevant changes in the network or for performance measurement of the company's own security management; or
- compliance requirements that can be tested in an automated manner should be verified in a cost-effective way.

With knowledge of vulnerability-scanning procedures, it should be understandable that vulnerability scanners can show their strengths especially in areas where fixed, predefined behaviors or responses can be assigned to specific, known vulnerabilities. This mainly concerns the scanning of standard services, the distribution of which is correspondingly widespread, so that the scan software manufacturers also provide as many test modules as possible for these services. In the area of custom software, the use of scanning software only makes sense in cooperation with professional testers, so penetration tests are advised instead of vulnerability scans for custom solutions and application-specific tests.

If scans are performed at regular intervals and are to be compared over time, or if large areas in particular are scanned, then communicating the results and coordinating them with system managers in particular presents a challenge for many security managers.

In this case, you can be supported by so-called vulnerability-management software, which provides an overview of all vulnerabilities and allows for sorting and filtering of results. It also gives system managers selective access to results, either directly integrated or provided as an interface. Subsequently, processing of vulnerabilities can be centrally monitored via a kind of ticket system. Examples of vulnerability-management software include vendor-specific products such as Tenable.io, Tenable.sc, and Acunetix Premium, or vendor-independent software such as VULCOM (see Figure 10.3).

Figure 10.3 Vulnerability Management Allows the Monitoring and Management of Vulnerabilities Discovered in an Enterprise

10.2.3 Vulnerability Assessment

Vulnerability assessment is the most common form of security check. Due to the high proportion of manual work by professional penetration testers, even hard-to-identify and individual vulnerabilities can be found. In addition, vulnerability scanners are used to ensure that no easily identifiable vulnerabilities are overlooked.

Unlike penetration tests, this test takes care to achieve the highest possible level of coverage. In this way, an attempt is made to find as many different vulnerabilities as possible in as many different functional areas of an application as possible in order to provide the broadest yet deepest possible insight into the security of a system or application. Common activities include the following:

- Performing port scans to detect available services
- Performing an automated vulnerability scan with one or more vulnerability scanners
- Using specially developed test tools that automate simple activities
- Manually checking all automatically identified vulnerabilities for false positives
- Manually searching for new, not yet known vulnerabilities in deployed software
- Preparing the results in a separate report

The step of manually searching for new vulnerabilities is the main component and can include testing network transmissions or checking web applications, as well as reverse engineering binaries. This is fully customized each time for the customer and the system under test. Conducting vulnerability assessments can be useful when:

- software developed in-house or by partner companies is to be subjected to a security check and as complete a picture as possible of the security of the software is desired, or
- the security of an internal network is to be evaluated and the focus is on obtaining as comprehensive a picture of security as possible.

The advantage of vulnerability assessments over vulnerability scans clearly lies in the possible detection of as yet unknown vulnerabilities in software. This also applies in particular to individual software and programs that are not widely used and are not detected by automated scanners. The advantage over penetration testing is that additional testing is performed across the board so that as many different attack vectors as possible can be identified and subsequently remediated.

Well-Known/Popular Software Is Not Automatically Secure!

Experience from past security assessments shows that there is no relation between the prominence of a piece of software or its manufacturer and the security of the software. Even in programs from large, well-known manufacturers, critical vulnerabilities, such

as unauthorized access to administration functionality, are regularly identified in the course of security checks.

Vulnerability Assessment versus Penetration Tests

In common usage, there is often no distinction between the terms *vulnerability assessment* and *penetration testing*. Therefore, you should make sure that you clearly communicate your expectations about the approach during an interview.

10.2.4 Penetration Test

Penetration tests aim to achieve the worst-case scenarios defined before launch. In most cases, this means a combination of automated tools with an additional high proportion of manual work. Compared to vulnerability assessments, however, the tests do not run broadly, but focus on achieving goals such as gaining administrator access in an application or gaining domain administration rights. Common activities include the following:

- Performing port scans to detect available services, which is possibly already limited to relevant services, instead of performing a scan across all possible ports
- Using specially developed test tools that automate simple activities
- Manually searching for new, not yet known vulnerabilities in deployed software
- Preparing the results in a separate report

Performing penetration tests can be useful, for example, when:

- safeguarding measures have already been taken and you need to check whether they provide sufficient security to prevent an attacker from reaching the target; or
- a security check is intended to demonstrate that the application has serious security vulnerabilities in order to have a sufficient argument for further, more detailed tests on the basis of this finding.

Penetration testing is well suited to identify particularly serious gaps as the focus of the application is precisely on these gaps and also the main portion of the time spent testing goes into this goal.

10.2.5 Red Teaming

Red teaming is usually understood to be a form of testing in which the scope of the test is not limited to one application, but tests whether, for example, access to certain data can be obtained. The application through which the tester gains access to the data isn't specified further so that the entire security concept is tested instead of individual

applications. In most cases, the red team assessment is also regarded directly as blue team training.

In this context, the *blue team* is the team for the tested party that is responsible for protecting the systems. This also means that, compared to many penetration tests, the red team tries to hide its activities to avoid detection. Common activities include the following:

- Searching for as many access points as possible to the desired asset or information
- Quickly analyzing which of the identified access points has the least security and would mean the quickest possible success
- Manually searching for mostly new, not yet known vulnerabilities in the respective applications
- Potentially compromising multiple servers and users on the network to get to the target asset

Conducting red team assessments may be appropriate, for example, when:

- security measures have already been taken and a realistic picture of the security of the entire network against targeted attacks is to be determined;
- your company has critical corporate data that needs special protection and you want to check whether existing security measures protect it effectively enough; or
- the internal team is to be trained practically in as realistic a manner as possible and in its own environment.

Red team assessments are a special form of testing in which several specialists from different areas may work together to achieve the previously defined goal. The assessment gives you the most realistic view of an attacker's most likely attack path but will not evaluate all identified attack paths in detail unless otherwise agreed. Also, there is no detailed assessment of individual applications as the tester looks for the most promising attack opportunities across all applications.

The red team is also intended to train the response and knowledge of your blue team. In the best case scenario, a collaboration between the two teams is achieved, constantly improving your organization's security and response to attacks.

Clarify the Common Understanding of the Test Type!

Even if the client and the contractor use the same words, they might not mean the same thing by them. Because vulnerability scans are often also sold as penetration tests, this can lead to both misunderstandings during meetings and unusable project results. You should therefore clarify the intent right at the beginning of the first meeting to prevent misunderstandings.

10.2.6 Purple Teaming

The term *purple teaming* is interpreted in different ways. In general, the goal of purple teaming is to promote cooperation between the red team and blue team in order to constantly improve the blue team's capabilities and be better protected from real attacks in the future. In contrast to pure red teaming, purple teaming focuses primarily on the development of the blue team.

To make the development of the blue team as structured as possible, the first step is to define the type of attacker for which the blue team should be trained and the technical means available to the blue team. Derived from this, information from the MITRE ATT&CK framework can be used to determine the usual steps taken by these groups of offenders and, based on this, a test plan can be derived. Based on this test plan, individual attack steps are specifically recreated by the red team. A subsequent analysis then determines whether existing tools and the blue team were able to detect these activities. If not, the blue team will receive all the necessary information to detect such actions in the future. In this way, the blue team is gradually introduced to the possible attack steps over several iterations and trained to recognize them and initiate appropriate countermeasures.

Common activities include the following:

- Joint derivation of typical attacker types against which the company primarily wants to protect itself
- Researching common attack techniques from actual incidents, in many cases based on the MITRE ATT&CK framework
- Manual or automated execution of targeted test cases to test and train the blue team's detection capabilities and response in a structured manner
- Disclosure of the red team's attack techniques used to the blue team
- Examination of which attacks could have been detected and what changes are necessary to be able to detect these attacks in the future
- Regular repetition of training activities

An implementation of purple teaming can be useful, for example, if:

- the blue team is to be trained on new attacks or new offender groups;
- you want to examine in a structured way which attacks your existing security operations center (SOC) can detect, and this should be accompanied by targeted training and an improvement in detection capabilities; or
- you don't yet have an existing SOC but want to build up the know-how and tools to detect attacks in a step-by-step and targeted manner.

MITRE ATT&CK Framework

The MITRE ATT&CK framework is a knowledge base of so-called attacker tactics and techniques. The database is constantly being expanded and adapted to the findings from actual incidents and the research of IT security companies. For more information, visit *https://attack.mitre.org*.

10.2.7 Bug Bounty Programs

Compared to the techniques presented in the previous chapters, bug bounty programs provide an alternative approach for companies to check their own software or systems for security. In contrast to the forms of security audits mentioned earlier in this chapter, the ordering company doesn't pay for the time invested by the security experts; it only pays if new, previously unknown vulnerabilities are reported by them.

Which and how many security professionals can participate as bug bounty hunters depends on the type of bug bounty program chosen, which range from public programs to private events that can only be entered by invitation. For more detailed information on bug bounty programs, see Chapter 19.

10.2.8 Type of Performance

Once you've decided on an assessment type from the previous sections, you'll be faced with the decision of which *performance method* to choose for that type. What follows is a description of the three basic methods and their advantages and disadvantages:

- **Black box**
  In a *black box check*, the tester receives as little information as possible about the target system before the assessment begins. The tester should be put in the same position as an external attacker with no prior knowledge of the system. This test type thus provides a comparatively realistic view if the tester is also given a correspondingly similar preparation time.

  In most cases, however, the budget is limited, so only a limited amount of time is available for testing. This is also where the disadvantage of the black box check becomes apparent: because the attacker is not provided with any information, they must work it out themselves. For example, if several applications are available in different directories on the web server, the tester must find this out by trial and error.

  However, due to the limited time, only a limited number of options can be searched for, so the tester may not find all of them and thus not test all of them for vulnerabilities. If the word list used were sorted differently, they might find other directories in the time available. This example shows that in order to get a realistic picture, the tester must either be given more time or accept that the check can only give a first impression regarding security.

- **White box/glass box**
  A *white box check* is the counterpart to the black box assessment. In this type of test, the tester is provided with all necessary information in advance or during the assessment. This ranges from additional administrator accounts on systems, to access to the source code of applications. This assessment type allows a very detailed and, compared to the other two types, more complete security assessment. Due to the abundance of information and the necessary working time for familiarization with the existing source code, the effort can also be higher than for grey box checks (discussed in the next point).
- **Grey box/gray box**
  Grey box checks are a kind of middle ground between black and white box checks. The tester should perform the security check as freely as possible, creatively and without too much influence from existing information, in order to reflect as realistic a view as possible from an unknown attacker. However, to perform the review as efficiently as possible, information is provided that the tester would find out on his own in a reasonable amount of time anyway. This information may include, for example, hidden folders, additional domains, or access to an additional administrator account to view existing admin functions.

10.2.9 Depth of Inspection: Attacker Type

To correctly estimate the scope of a project, in addition to the information already determined so far—such as the goal and technical scope, assessment type, and method of implementation—you need to know the depth of how detailed the assessment should be. This is determined by the definition of the attacker type.

Clearly Define Goal and Scope during Preliminary Discussions

For a successful project process, you should define the scope, goals, nongoals, and worst cases right at the beginning. This ensures that you as the client receive what you expect, but it's also important for the contractor to have this information in order to provide an accurate estimate.

Concerning the scope, you should describe all systems or application parts that will be tested. You should also specify which ones should be excluded from the test. In the goal definition, you have the opportunity to communicate what you expect from the test and what your objective is. The worst cases describe scenarios that shouldn't be possible under any circumstances. Even if these are not technically possible from your point of view, you should include the scenarios. After all, it's the job of the hackers you hire to find opportunities that no one has thought of before.

The following three types of attackers are examples of rough categorizations that may be defined differently in detail from one consulting firm to another:

- **Script kiddie**
  A *script kiddie* is an individual who has learned to attack and compromise systems using existing tools and tool-related instructions on the internet. Script kiddies don't have enough basic knowledge and programming skills to create their own exploits or adapt exploits that don't work. However, due to the prevailing crime-as-a-service model, this type of attacker is becoming less important.
- **Advanced attacker**
  These attackers have a good understanding of protocols, systems, and the technical background behind the available tools. They are able to create simple exploits themselves or adapt existing ones. However, time and resources are also limited for this attacker type.
- **Expert**
  This type of attacker is most easily compared to a state attacker or an attacker that can access a significant amount of time and resources. It's assumed that an expert spares no expense or effort to install even software that is considered to be secure on his own systems, and through it tries to find new, as yet unknown vulnerabilities in order to break into the target system.

Crime as a Service

The so-called crime-as-a-service model describes the development in recent years in which advanced perpetrator groups offer their developed special software for use, often via the darknet, as a service—for example, in a subscription model. This enables even people with little IT know-how to carry out very technically advanced attacks. The best-known form of this model is the so-called ransomware as a service.

10.2.10 Prior to the Order

If you commission a security check, you should make sure the consulting firm you hire asks about all of the items described in this section before providing an estimate. If an estimate is provided earlier, then it must be assumed that the consultant has already made decisions for the customer or the assessment is based on an automated scan that finds results independent of decisions such as the attacker type.

10.3 Legal Protection

When conducting security checks, there's a strong relationship of trust between you and your contractor, as the latter gains a deep insight into your internal processes, structures and data either beforehand or during the test at the latest. You thus give one or more external persons access specifically to the data you want to protect. This makes it all the more important to be cautious when choosing the right partner. Although

trust is important, it's still necessary to make appropriate legal arrangements to ensure that reviews run smoothly. In particular, this includes a nondisclosure agreement, a liability agreement, and written permission to perform the test:

- **Nondisclosure agreement**
  A *nondisclosure agreement* (NDA) obligates both parties to keep the information discussed in the course of the project confidential. For you as the client, this document is important to oblige the contractor to keep their silence about your internal data. Penalties vary, but are usually set between $50,000 and the actual contract value. It's also customary to state that, at the request of the client, the data must be destroyed on the contractor's side, provided that this is compatible with the legal requirements for the contractor. Confidentiality can be made mandatory for a few years after the end of the project.
- **Liability agreement**
  A *liability agreement* is usually provided by the contractor, either as a separate document or integrated into the bid or permission-to-attack document. In Europe, it's usually the case that the contractor receives a release from liability in this liability agreement during the course of the project for all tests agreed upon in the offer. The client is obliged to provide the necessary information and to perform regular backups of the systems under test in order to minimize the risk. Excluded from the liability agreement are grossly negligent acts of the contractor.
- **Permission to attack**
  In the *permission-to-attack* (PTA) document, the client grants the contractor written permission to carry out attacks on specific targets, usually defined by IP address or DNS name, within the scope of the agreed upon project and the associated tests within a predefined period of time. Especially in the case of tests from the internet, you should make sure that the contractor also tells you the IP addresses from which the tests are performed so that you have the opportunity during and after tests to determine whether they were actual attacks or tests related to the security check.
- **Order processing contract**
  Depending on the objective of the assessment, the contractor may have access to personal data within the scope of the General Data Protection Regulation (GDPR). In these cases, you should be sure to conclude an order processing agreement with your service provider in order to also regulate the correct and secure processing of the data in accordance with the GDPR.

If you have an internal security team that regularly monitors internal network traffic, then you should have the test team provide you with the IP addresses they're currently using on an ongoing basis so that you can share them with your own incident or monitoring team in parallel. This prevents additional costs from being incurred due to incorrectly categorized attacks.

Legal Support

Note that the descriptions in this section can only give you an impression of the content of the documents. It's advisable to consult a lawyer for one-time preparation of the documents and to have them check all legal issues.

10.4 Objectives and Scope

As with any other project planning, to successfully complete a project, it's necessary to determine the scope, goals, and nongoals of the project prior to its start. This is usually agreed upon in the course of a joint meeting and recorded in writing in the form of a quotation or project description. When discussing the scope, you should be clear about in how much detail the respective systems or software should be tested. Your project partner will help you with this, and in some cases the response will already be codetermined based on the defined attacker type.

To provide a concrete estimate, your contractor will most likely need detailed information about the target system, such as a description of existing services, a description of the parameters of available interfaces and endpoints, a network map, or even screenshots or documentation of the application under test. If this involves internal, sensitive data, then you should establish an NDA with the potential contractor before the conversation.

In addition to defining the handling, it's advantageous for a test if you already talk through the so-called worst-case scenarios, the typical use cases, and the hot spots that you would like to have examined in more detail with your contractor in advance. All of this information will then help the penetration testers tailor the test to your needs as best as possible and spend the available time most efficiently on those aspects of the assessment that are most important to you.

Just as important as defining the scope, objectives, and worst-case scenarios is defining nonobjectives. Clearly define together which systems or applications should explicitly *not* be tested or which attack types (e.g., denial-of-service tests) should explicitly *not* be performed. For example, many security assessments explicitly exclude denial-of-service attacks.

10.4.1 Sample Objective

The goal of the black box vulnerability assessment is to determine the risk of attacks on the MySecurePortal application from the perspective of an external attacker from the internet. The assessment is intended to provide the broadest possible overview of different attack capabilities of the existing infrastructure.

The type of attacker chosen was a so-called script kiddie, who has experience in using ready-made tools, but no in-depth expertise in finding as yet unknown vulnerabilities or programming their own tools. Explicitly excluded are social engineering attacks on employees or customers; denial-of-service attacks that aim to restrict the availability of the application also are not permitted.

10.4.2 Sample Worst-Case Scenarios

The following three worst-case scenarios were defined for the project: direct access to the database or operating system, read or write access to customer data, and access to administrative functions (starting with */admin/* in the URL).

10.4.3 Sample Scope

The scope includes the entire MySecurePortal application, which can be accessed at the following address: *https://mysecureportaltest.targetcompany.com*. The infrastructure components that build on it are also part of this. Testers are provided with administrator access to the application so that they can test external access to internal functions as comprehensively as possible. However, all tests are still performed from an attacker's perspective without credentials.

The server is hosted and operated by the customer and exclusively provides the test instance of the MySecurePortal application. If other applications are found on the server under the same IP address during the test, they will also be included in the scope, as they can potentially be used as a point of attack on the application data. However, the main focus of the assessment should be on the MySecurePortal application itself.

10.5 Implementation Methods

The approach used to perform security assessments depends on the test in question, but also, and above all, on the approach chosen by the respective provider. Most providers use a combination of approaches customized to their own needs. This ensures that a minimum standard is maintained during the assessment, but the testers are deliberately given freedom in their approach. After all, security assessments are a creative activity.

The following is a brief explanation of the most popular approaches mentioned by many providers:

- **Open Web Application Security Project**
  The Open Web Application Security Project (OWASP; *https://www.owasp.org*) is a nonprofit organization founded in 2004 to make know-how for the development and the operation of secure web applications available publicly and independent of manufacturers.

OWASP operates several subprojects. Among the best known are the OWASP Top 10, a list of top vulnerabilities in web applications; and software such as the Zed Attack Proxy (ZAP) for analyzing web traffic. The *OWASP Web Security Testing Guide* briefly describes a few organizational details about web application testing and puts its main focus on detailed elaboration of tests to be performed, both in their execution and with subsequent recommendations. In comparison, the Application Security Verification Standard is used to describe requirements or tests that can be used by architects, developers, and testers. The testing guide again can be used to perform the tests.

- **NIST SP 800-115**
  NIST has issued several documents related to system security. One of the best known is the SP 800-115 standard, *Technical Guide to Information Security Testing and Assessment* (*http://s-prs.co/v569673*).

  In contrast to OWASP, this standard does not deal with technical details; instead, it describes the organizational process of security assessments.
- **Penetration Testing Execution Standard**
  Like OWASP, the Penetration Testing Execution Standard (PTES) was born out of the community and thrives on the input of security professionals. The standard is available online in the form of a wiki (*http://s-prs.co/v569674*) and covers both the organizational preparation and the process as well as the technical execution of tests. The PTES hasn't been updated for a long time. However, many forms of attack documented there are still valid. Because of its practical nature, PTES continues to be a good starting point for those interested.

10.6 Reporting

As a result of a security assessment, the vulnerabilities identified in the course of the project, including date appropriate recommendations for remediation and usually also a risk assessment, are made available to the client. When commissioning a security assessment, you should consider the best way to process the results yourself. Some companies want a completed report in PDF format, as this represents a written third-party opinion that is as unbiased as possible. Also, the results are usually documented in detail and in a way that can be understood by different groups of people.

Other companies only share the results with the internal IT team, which has been working with IT security assessment results for some time, and there is no provision for sharing them with the company management. In this case, a detailed description of the results may not be necessary, so rough documentation in the form of a table will suffice. If you've been working with your pen testing partner for a long time anyway, a digital exchange of results is also possible, which minimizes the effort on both sides. Here,

you can make your wishes known to the contractor right at the start so that you also receive the results in the form in which you can best process them.

Also, when selecting a vendor, each potential contractor should be able to present you with a demo report. This allows you to evaluate whether the target groups in your company are being addressed appropriately, whether results come directly from a vulnerability scanner or additional value has been added by the expert knowledge of the testers, and how the risk calculation of the respective findings can be integrated into your own risk management.

Every company sets its focus differently, and the type of documentation can also vary. In general, however, a report can roughly be divided into the following sections:

- **Management summary**
  This section presents the scope and key findings of the test from a business perspective on one page. There, the goal and the time period in which the assessment took place are first described in two sentences. Also, positive aspects from the assessment should be highlighted and the most critical weaknesses should be presented. The management summary should always end with a recommendation for further course of action.

  In most cases, the client knows the recipient of the management summary better than the contractor. For this reason, it can be advantageous to agree on the structure and content shortly before finalizing the report. This also gives the client the opportunity to include additional information: how much of a threat it is to the business, what points are particularly important to readers of the summary, and so on.
- **Technical summary**
  This section is intended for technical management and summarizes the actual findings in a table so that chief security officers (CSOs) and chief information security officers (CISOs), for example, can get a quick overview of the technical security status.
- **General recommendations**
  General recommendations and follow-up steps are presented either directly in the management summary together with the results or in a separate section. The goal of this section is to provide management with recommendations for subsequent steps.
- **Scope and organizational details**
  For the report to be understood as a stand-alone document even at a later date, it's necessary that all project agreements, scope info, contact persons, schedule details, and other organizational circumstances are written down.
- **Technical details**
  In this section, all results are documented in detail. This includes a general description, details of the specific security problem investigated, recommendations for remediation, and an assessment of the risk.

How the chapter is further structured depends on the results of the assessment and on the customer's wishes or how the customer can most easily process the data. This means that, for example, an outline of sections by host and then a listing of vulnerabilities by host can be made if the customer has defined responsible parties by host within the company. In other situations, it may be useful to distinguish between infrastructure and web applications. In any case, however, the same description form should be chosen for each documented vulnerability, which we'll explain in more detail ahead.

A description of a vulnerability itself is again divided into several sections, which should help different reader groups to extract the information content important for them more easily and to better understand the issue or the solution to it. The following outline is only an example and may vary from company to company:

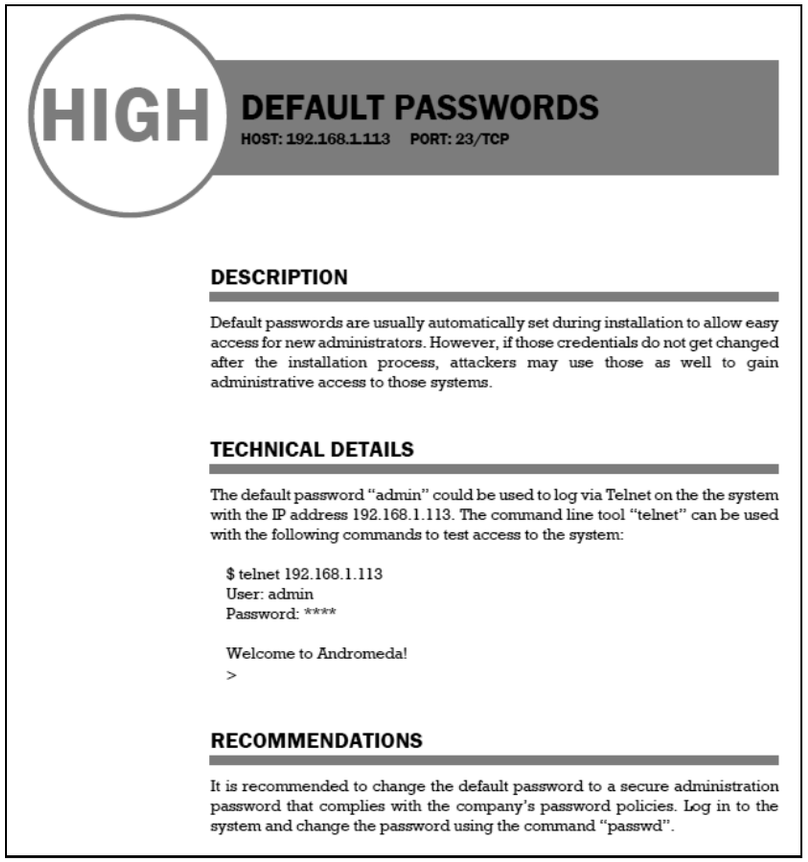
- **Description**
  The description is intended primarily for people with basic technical knowledge, but without specific security know-how. They should understand the fundamental problem of the identified vulnerability and be able to assess the associated risk without having to delve into the specific technical details.
- **Technical utilization**
  This section explains the technical details of the vulnerability. With the information contained here, it should be possible for the person who is responsible for the system or application to understand the problem so that a suitable solution can be found.

  In most cases, screenshots or code examples support the understanding. It's even better if the reader is provided with a detailed description of how to recreate the problem using open-source means. References based on results from commercial tools should be avoided, as the reader may not have the financial means to invest in paid tools.
- **Recommendations**
  The recommendations should provide the most accurate solutions possible to the specific vulnerabilities identified. If possible, dedicated configuration recommendations also should be provided, which the respective system owner can adopt directly.
- **Valuation**
  The vulnerability assessment can be based on different aspects and will vary from company to company.

  The assessment is based either on the company's own benchmarks—for example, a quality-oriented classification into high, medium, and low categories—or on well-known approaches, such as Microsoft DREAD (damage, reproducibility, exploitability, affected users, and discoverability) or the Common Vulnerability Scoring System (CVSS).

Each of these methods has its advantages and disadvantages, and there is no clear industry standard. More important than choosing the perfect assessment method is that the assessment is consistent, understandable, and correct.

Figure 10.4 shows an example of a finding for using a default password. Because this is an unencrypted Telnet service, there should be at least one more finding in the report about using Telnet.


HIGH

DEFAULT PASSWORDS

HOST: 192.168.1.113 PORT: 23/TCP

DESCRIPTION

Default passwords are usually automatically set during installation to allow easy access for new administrators. However, if those credentials do not get changed after the installation process, attackers may use those as well to gain administrative access to those systems.

TECHNICAL DETAILS

The default password "admin" could be used to log via Telnet on the the system with the IP address 192.168.1.113. The command line tool "telnet" can be used with the following commands to test access to the system:

```
$ telnet 192.168.1.113
User: admin
Password: ****

Welcome to Andromeda!
>
```

RECOMMENDATIONS

It is recommended to change the default password to a secure administration password that complies with the company's password policies. Log in to the system and change the password using the command "passwd".

Figure 10.4 Example of a Finding in a Final Report

10.7 Selecting the Right Provider

Once you've made the decision to have the security assessment performed by an external provider, the search for the most suitable provider for your purposes begins. As in other areas, each provider has its own focus, so you should choose them based on the required objective. You should ask the vendor for experience in the specific task area in which you need assistance, such as reverse engineering, for example.

The assessment type can also play a role in the decision. If a simple vulnerability scan is required, the tester's experience is less important. However, if you're looking for a highly manual assessment, such as a penetration test, then the quality of the test is very much dependent on the knowledge and experience of the tester in question. These people are specialists who have to keep their know-how up to date on a daily basis, constantly expand it, and adapt it to current technologies. This also means that price can't be the sole reason for exclusion in these forms of assessment, as the quality of the results may also depend heavily on it.

At the very least, you should consider the following points when choosing the right provider:

- **Recommendations are important**
  In addition to a technically flawless execution, the quality of a service also includes general aspects such as the organizational process, communication with the customer, understanding the customer's problems and the objective, and the preparation and communication of the results.

  In most cases, the totality of these points provides an overall impression that can best be obtained via existing customers. You should proactively ask friends, acquaintances, and companies in your sector about their experiences with penetration testing companies. This way you can obtain honest feedback most of the time as well as an initial list of interesting providers. Referrals are one of the most valuable forms of evaluation for potential providers.
- **Customer names are optional**
  Due to existing confidentiality agreements (NDA contracts), it's often not possible—or only possible with additional effort—for providers to name existing customers. Even if you're given customer names, a provider is most likely to list those customers who were satisfied with the service.
- **Description of previous experiences**
  An adequate means of better assessing experience in connection with one's own task is to ask the provider for examples of similar projects and approaches used. This gives you an impression of how often and at what level of detail the provider has already covered these topics and how confident they feel in this area.
- **Description of the procedure**
  You should get a description of the provider's approach. However, it's better to ask about the specific issues in the current project and how the provider intends to solve them than to mention known standards. The approach is interesting from both organizational and technical perspectives.

  You can also check how the vendor handles critical systems that may be in place or how they typically work with internal security teams.

If critical systems are present, the call to handle them with care should also come from the provider side. If you have someone on the team with technical security expertise, you can also ask about the tools used to get a better sense of how advanced the technical understanding is for complex tasks.

- **Check the report**
  Ask to see a demo report to assess whether it contains all the information you expect. This also allows you to react at an early stage if you still need to make appropriate adjustments so that you can process the results further in the company with as little effort as possible.

Contents

Index

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

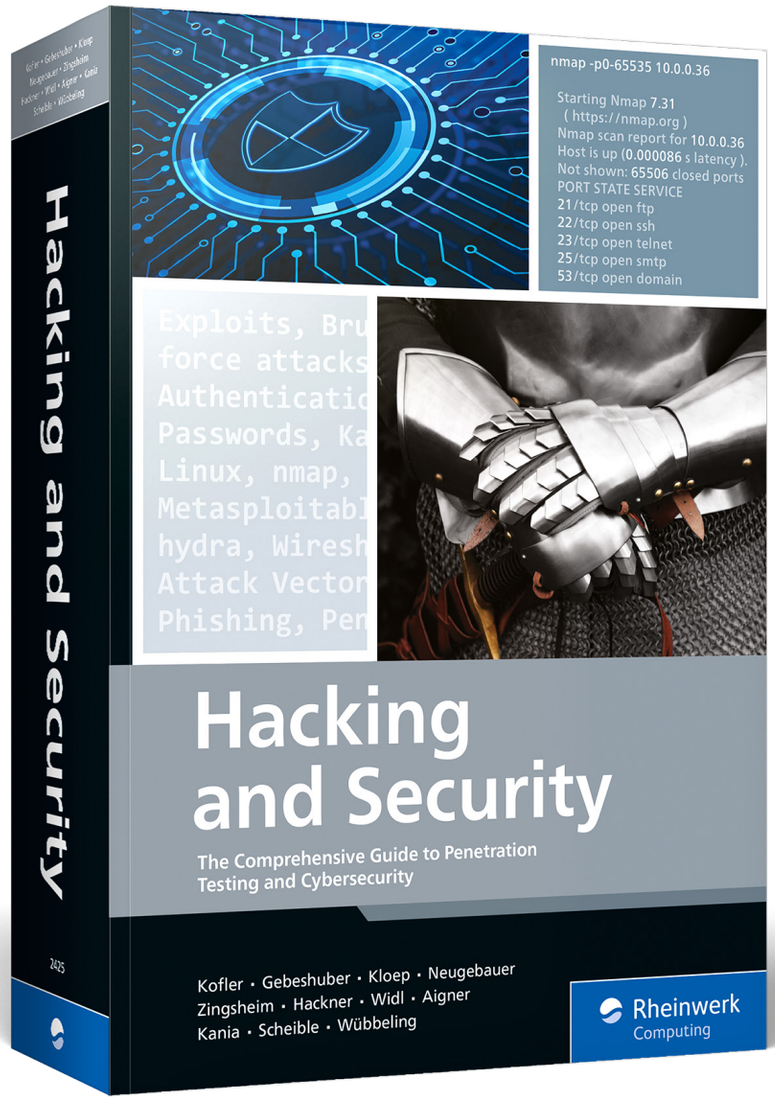
X

Y

Z

Build and deepen your coding knowledge from the top programming experts!

Rheinwerk Computing



This book was written by a team of security specialists who have decades of experience working as penetration testers, administrators, and developers. Their unique expertise has taught them how hackers operate and they use this insight to teach others how to effectively defend systems against attacks.

- **Michael Kofler**
- **Klaus Gebeshuber**
- **Peter Kloep**
- **Frank Neugebauer**
- **Andrè Zingsheim**
- **Thomas Hackner**
- **Markus Widl**
- **Roland Aigner**
- **Stefan Kania**
- **Tobias Scheible**
- **Matthias Wübbeling**

Michael Kofler et al.

Hacking and Security

The Comprehensive Guide to Penetration Testing and Cybersecurity

1141 pages | 07/2023 | $59.95 | ISBN 978-1-4932-2425-8

www.rheinwerk-computing.com/5696