CompTIA PenTest+ Master Cheat Sheet

Planning and scoping

Explain the importance of planning for an engagement.

Understanding the target audience

Target audience may be management, technical teams, etc. Usually pentest report has several groups to target, each of them having different:

- needs (operational planning, resource allocation, etc.)
- position in the organization
- knowledge of the report topic
- responsibility or authority to make decisions based on the report
- demographics (age, alliances, attitudes, etc.)

Report written without understanding of target audience is unclear and useless.

Rules of engagement

Rules describe expectations for client and limitations or rights for testers.

Testers should agree with the client on the following:

- when active portions of testing are performed? during business hours, after business hours, on the weekends, etc.
- which hosts, networks, domains & other resources are being tested, and which are not
- black box, white box or grey box
- will client's security team and/or other employees know about the engagement
- are there appliances which may impact the effectiveness of attacks, such as firewall, IPS, WAF, load balancer, etc.
- what are the acceptable social engineering pretexts
- is it allowed to DoS-test clients infrastructure and other relevant rules.

Without engagement rules, pentest

- will produce unexpected and unwanted results
- will not serve client's objectives
- may cause damage and liability
- may put pentest firm out of business or damage its reputation

Communication escalation path

Communication path and method is important for customer satisfaction, confidentiality of findings, incident handling (e.g. server has crashed after execution of a payload).

Without communication incident handling is delayed, customer does not know what the pentest team is doing and sometimes feels insecure, other problems arise.

Resources and requirements

Confidentiality of findings

Confidentiality of findings is very important since pentest unveils vulnerabilities in clients infrastructure.

In case of loss of findings confidentiality, vulnerabilities (findings) may be exploited by threat actors resulting in an incident (thus damage) for the client and bad consequences for the pentesting firm.

Known vs. unknown

Requirements & resource needs might be formalized at the beginning of engagement, but may change as the pentest progresses. It ineffective to try to foresee everything.

See Agile vs Waterfall.

Budget

Budget is important because engagement is a service provided by the pentesting firm to the client, in exchange for money.

Volume of testing, chosen methodology, quality etc. is budget-dependent. The bigger is the budget, the more time pentesting firm is able to allocate to pentest, the more qualified personnel it is able to involve in the project, etc.

Pentesting firm, from a financial perspective, is interested in minimizing expenses and maximizing revenue (compensation according to the contract), keeping quality of provided services at a level acceptable by the client.

Client in this case is interested in minimizing expenses and maximizing amount (scope, working hours)/quality of services provided by the pentesting firm.

Of course, for the client hiring a pentest firm is also an investment, which is aimed at increased revenue/trust/etc. or decreased number of vulnerabilities, risk, financial losses due to incidents, etc.

A win-win between pentesting firm and client has to exist in order for the pentest to be successful.

Impact analysis and remediation timelines

Sometimes client is not interested in low-severity (low-impact) findings. For example, CVSS might be used for impact analysis of findings.

Remediation (the process of closing holes) typically starts with high-impact vulnerabilities or sometimes with those vulnerabilities which require less time to fix. Some vulnerabilities may require a lot of resources to fix; organization might also accept the risk associated with some findings.

Disclaimers

Point-in-time assessment

Client has to understand that pentest report is not a "certificate" which guarantees security in good times and in bad, in sickness and in health. One simple configuration change one hour after pentest is over and boom, you're scr*wed. Pentests should be regular and ideally from different, qualified vendors.

Comprehensiveness

Usually clients are interested in making the scope as broad as possible while paying minimum, and they also expect results as fast as possible (something between 1 week and 1 month).

In such a case it is important to understand that the broader the scope, the bigger is the complexity of it and the time required for testing. It is most likely insane to demand the whole linux kernel project to be audited for security in a day for just 5 bucks.

Pentesting firms should aim at narrowing the scope and increasing revenue for engagement at the same time. It is also not possible to find all vulnerabilities during an engagement.

Technical constraints

Pentesters do not have superpower. Sometimes there are technical constraints which affect the effectiveness of an engagement. For example, it is hard to penetrate air gapped networks remotely.

Support resources

Documentation might help pentesters, but it is not mandatory.

WSDL/WADL

The Web Services Description Language (WSDL) is an XML-based interface definition language that is used for describing the functionality offered by a web service. WSDL 2.0 became a W3C recommendation on June 2007.

The Web Application Description Language (WADL) is a machine-readable XML description of HTTP-based web services. WADL was submitted to the World Wide Web Consortium by Sun Microsystems on 31 August 2009.

SOAP project file

SOAP (originally Simple Object Access Protocol) is a messaging protocol specification for exchanging structured information in the implementation of web services in computer networks. Version 1.2 of the specification, became a W3C recommendation on June 24, 2003.

SDK documentation

SDK usually uses API and is written for particular programming language.

It requires documentation for easier and faster understanding of available functionality, authentication, etc.

Swagger document

Swagger is a framework of API developer tools for the OpenAPI Specification(OAS).

As a modern alternative to WSDL, WADL, SOAP & other, Swagger provides possibility to document web applications.

XSD

XSD (XML Schema Definition), a recommendation of the World Wide Web Consortium (W3C), specifies how to formally describe the elements in an Extensible Markup Language (XML) document.

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="note">
 <xs:complexType>
   <xs:sequence>
     <xs:element name="to" type="xs:string"/>
     <xs:element name="from" type="xs:string"/>
     <xs:element name="heading" type="xs:string"/>
     <xs:element name="body" type="xs:string"/>
   </xs:sequence>
 </xs:complexType>
</xs:element>
</xs:schema>
```

Sample application requests

Sample application requests, as well as test code, may serve as documentation for pentesters.

Architectural diagrams

Architectural diagrams provide possibility for a pentester to get familiarized with application's architecture quicker.

Explain key legal concepts

SOW

SOW = statement of work.

It defines project-specific activities, deliverables and timelines for a vendor providing services to the client. The SOW typically also includes detailed requirements and pricing, with standard regulatory and governance terms and conditions. It is often an important accompaniment to a master service agreement (MSR) or request for proposal (RFP).

MSA

MSA = master service agreement

A master service agreement, or MSA, is a contract reached between parties, in which the parties agree to most of the terms that will govern future transactions or future

agreements. A master service agreement allows the involved parties to more quickly negotiate future transactions or agreements, because they can rely on the strong foundation of the master agreement for future business, so that the same terms need not be repetitively negotiated, and you only need to negotiate terms specific to the latest deal.

NDA

NDA = non-disclosure agreement

A non-disclosure agreement (NDA), also known as a confidentiality agreement (CA), confidential disclosure agreement (CDA), hush agreement, proprietary information agreement (PIA) or secrecy agreement (SA), is a legal contract between at least two parties that outlines confidential material, knowledge, or information that the parties wish to share with one another for certain purposes, but wish to restrict access to or by third parties. It is a contract through which the parties agree not to disclose information covered by the agreement. An NDA creates a confidential relationship between the parties to protect any type of confidential and proprietary information or trade secrets. As such, an NDA protects non-public business information.

Environmental differences

Export restrictions

In the early days of the Cold War, the U.S. and its allies developed an elaborate series of export control regulations designed to prevent a wide range of Western technology from falling into the hands of others, particularly the Eastern bloc. All export of technology classed as 'critical' required a license. CoCom was organized to coordinate Western export controls.

Currently, many countries, notably those participating in the Wassenaar Arrangement, have similar restrictions.

Local and national government restrictions

In US (national) for example, some states have their own (local) laws.

Corporate policies

Corporate policies may limit the abilities of a pentests.

Written authorization

Obtain signature from proper signing authority

Pentesting before authorization or before the contract is signed might be illegal in certain countries and is unethical.

It must also be worth verifying if signing authority has permission to authorize pentesting against testing scope. For example, ABC Inc. might not be able to authorize a pentest for XYZ Inc. website.

Third-party provider authorization when necessary

Client might use third party resources for their operation (APIs, SaaS, IaaS, PaaS, etc.). Is such cases, they should be excluded from scope if no permission has been given for pentesting from those third-parties.