Penetration Testing Report

Revision

Version 6
10/27/22 6:02 PM

SME

Lucky Munro, Process
Charles Wilson, Process
Matthew Bourdua, Report

Abstract

This document describes the process used to produce a penetration testing report.

Group / Owner

Security / Vulnerability Assessment Analyst

Motivation

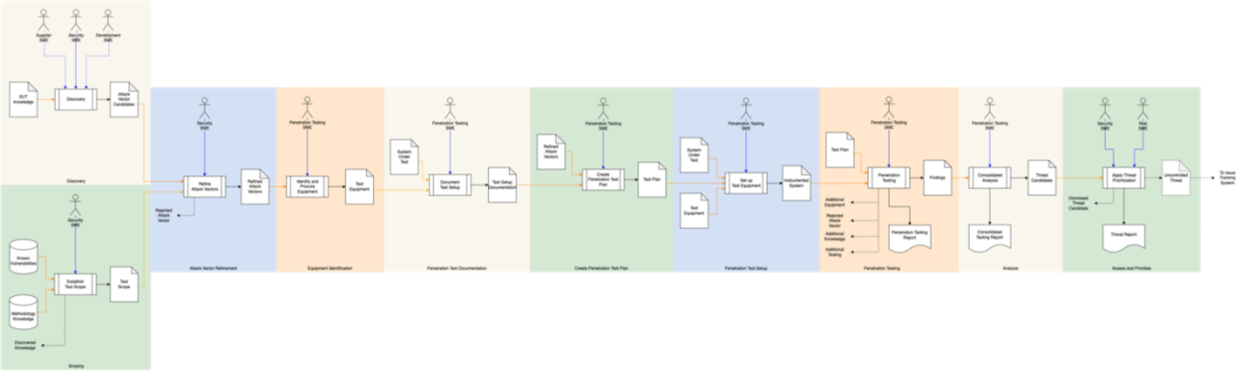
This document is motivated by the need to have whole-system, security-related feedback in the development of safety-critical, cyber-physical systems for certification of compliance to standards such as **ISO 21434** and **ISO 26262**.

License

Overview

Although the quality of feedback from the compiler, static and dynamic analysis tools, and fuzz testing provides a great deal of insight into security-related issues, they generally only expose possible security issues. Penetration testing provides analysis into possible real-world attacks on the system as a whole and exploits only achievable through taking advantage of multiple security deficiencies.

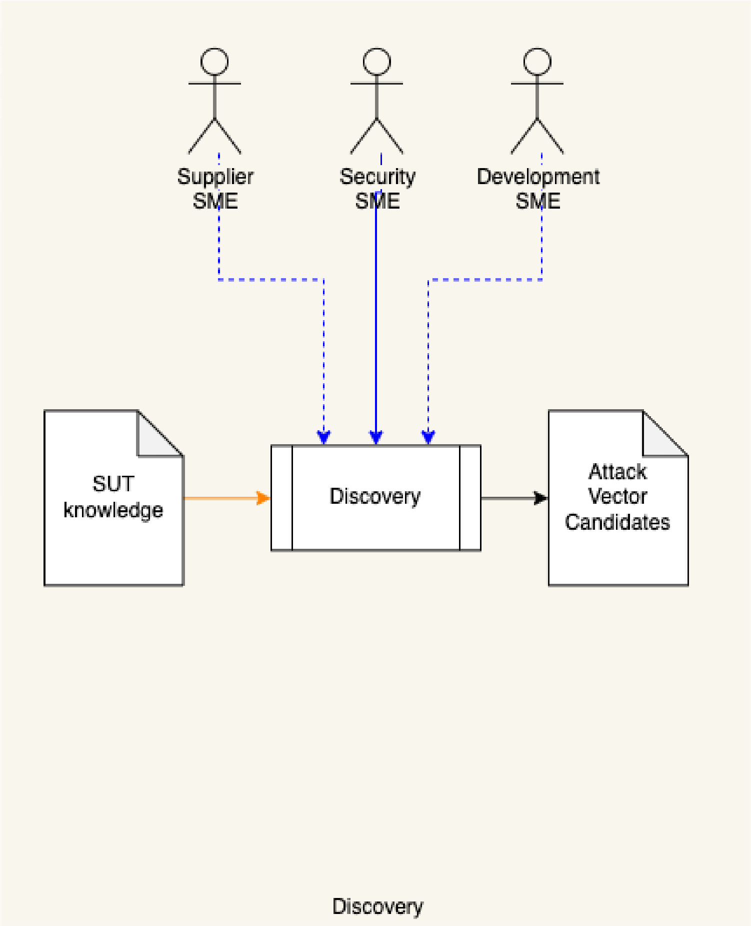
The following shows the workflow used:



Process

Discovery

Inputs	SUT knowledge
Outputs	Attack vector candidates
Participants	Security SME Supplier SME (optional) Development SME (optional)

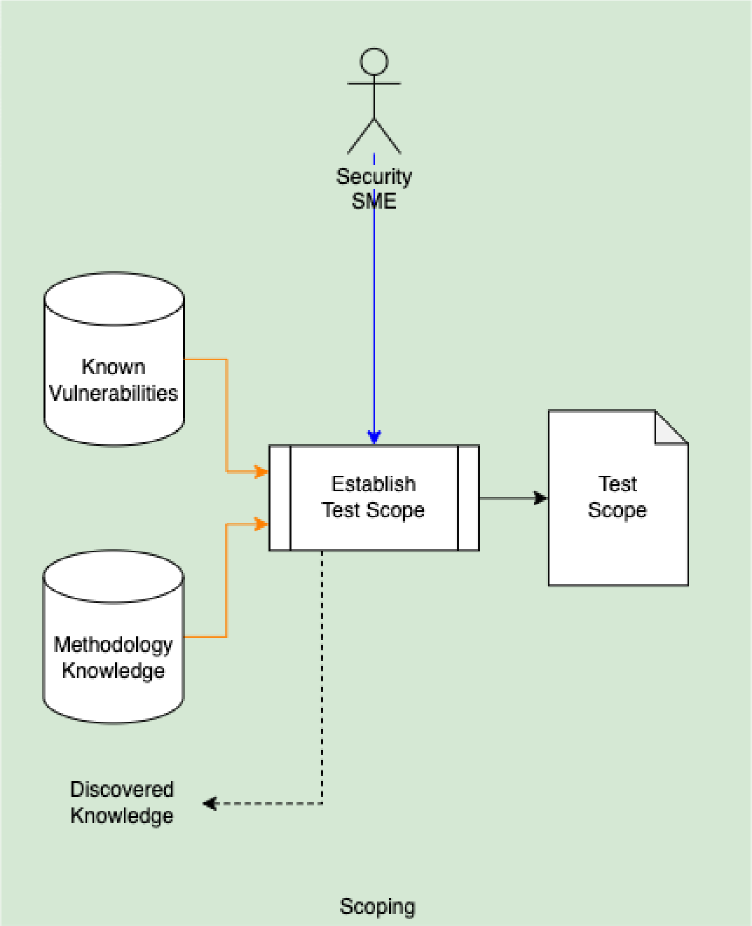


The Security SME with the possible assistance of the Supplier SME and Development SME review the **SUT knowledge** and create a list of **Attack Vector Candidates** to be used for penetration testing.

Note: This activity can be done in parallel with the **Scoping** activity.

Scoping

Inputs	Known vulnerabilities Methodology knowledge
Outputs	Test scope
Participants	Security SME



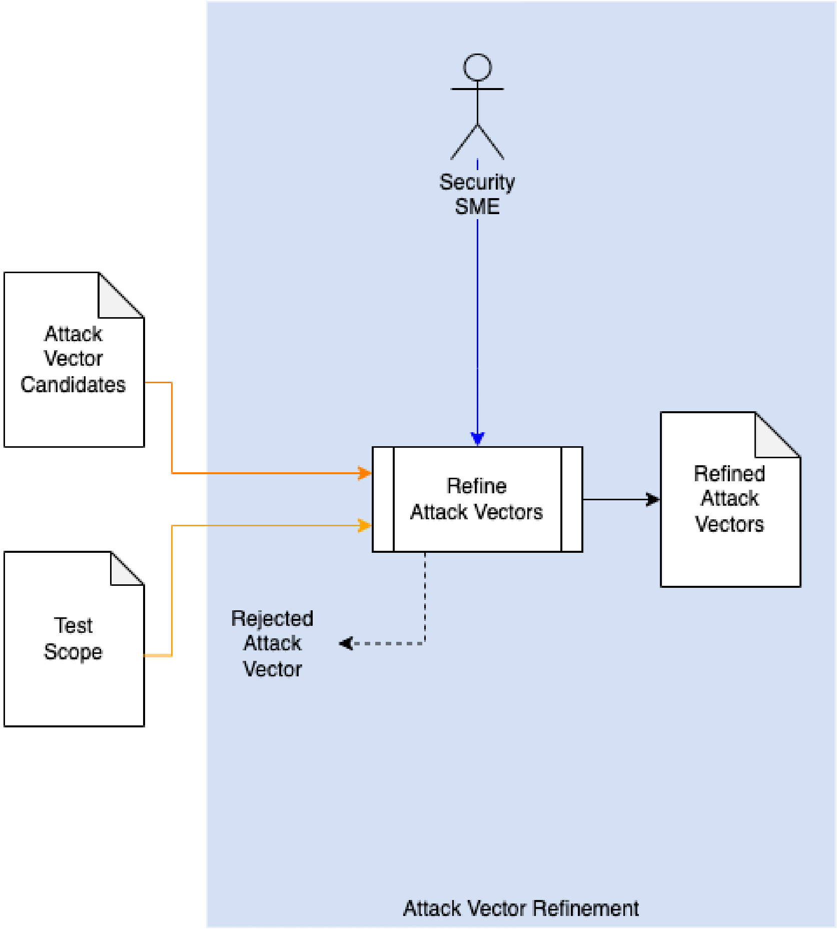
Using the **Known Vulnerabilities** of the system to be tested and penetration testing **Methodology Knowledge** [4, 5, 6, 7], the Security SME establishes a scope for the penetration testing efforts which is documented in the **Test Scope** document. If new understanding of the system to be tested is discovered, a **Discovered Knowledge** notification is generated.

Note: The **Discovered Knowledge** notification is a conceptual construct. In practice, steps will be taken to integrate this new information into the **SUT knowledge**.

Note: This activity can be done in parallel with the **Discovery** activity.

Attack Vector Refinement

Inputs	Attack vector candidates Test scope
Outputs	Refined attack vectors
Participants	Security SME

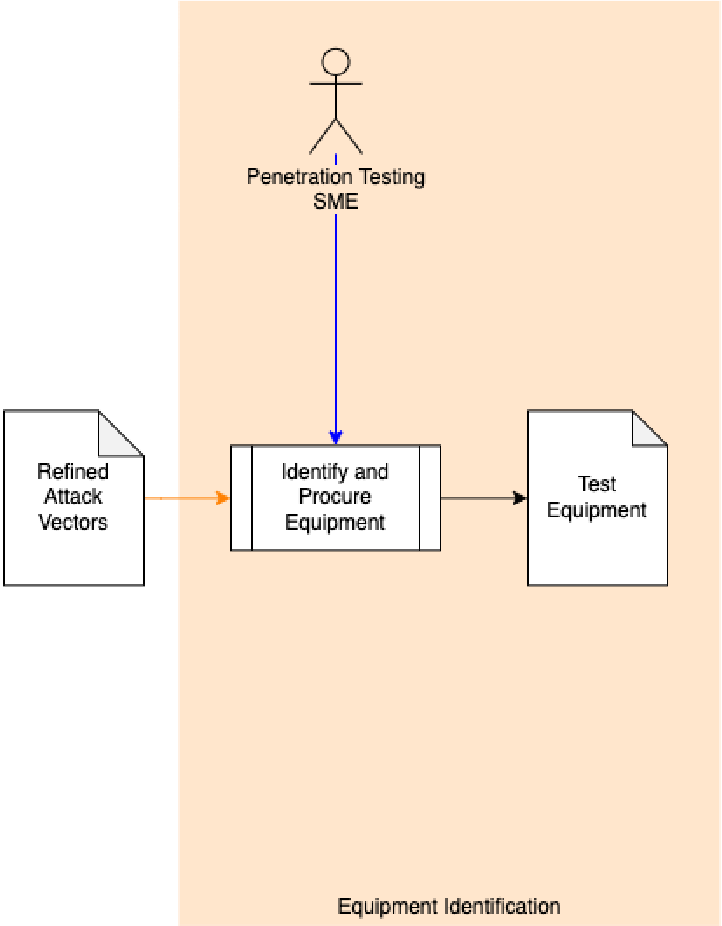


Using the **Test Scope**, the Security SME refines the **Attack Vector Candidates** to those applicable to the scope. A **Refined Attack Vectors** list is generated. If an attack vector is rejected for reasons other than scope, a **Rejected Attack Vector** notification is generated.

Note: The **Rejected Attack Vector** notification is a conceptual construct. In practice, steps will be taken to integrate this information into the **SUT knowledge**.

Equipment Identification

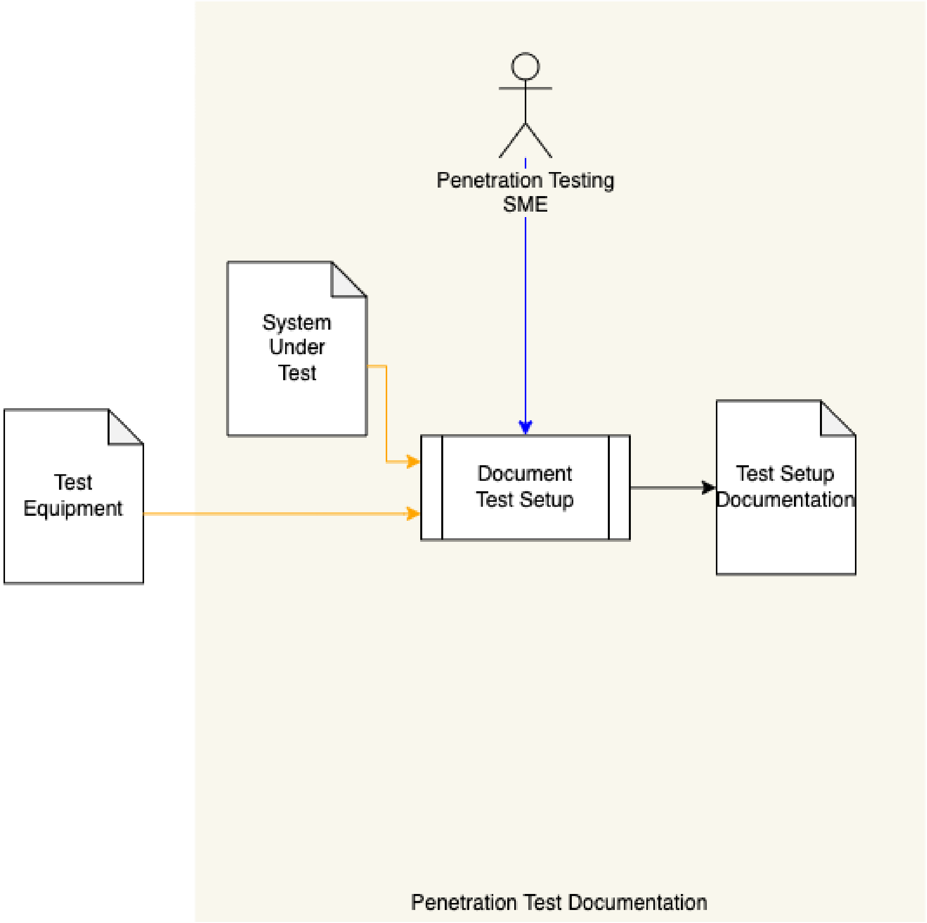
Inputs	Refined attack vectors
Outputs	Test equipment
Participants	Penetration testing SME



Using the **Refined Attack Vectors**, the Penetration Testing SME identifies and procures the equipment necessary to carry out the penetration testing. A **Test Equipment** list is generated.

Penetration Test Documentation

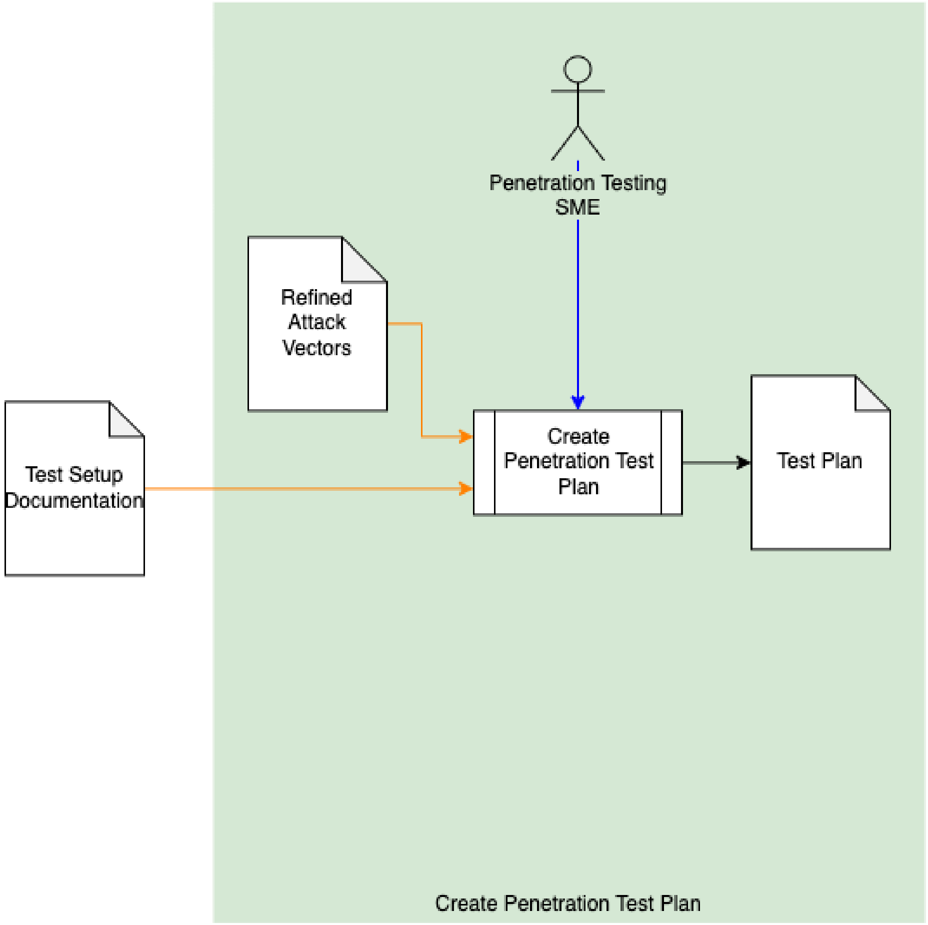
Inputs	System under test Test equipment
Outputs	Test setup documentation
Participants	Penetration testing SME



Using the **Test Equipment** list, the Penetration Testing SME reviews the **System Under Test** to determine how the system should be instrumented with the **Test Equipment**. A **Test Setup Documentation** document is generated.

Create Penetration Test Plan

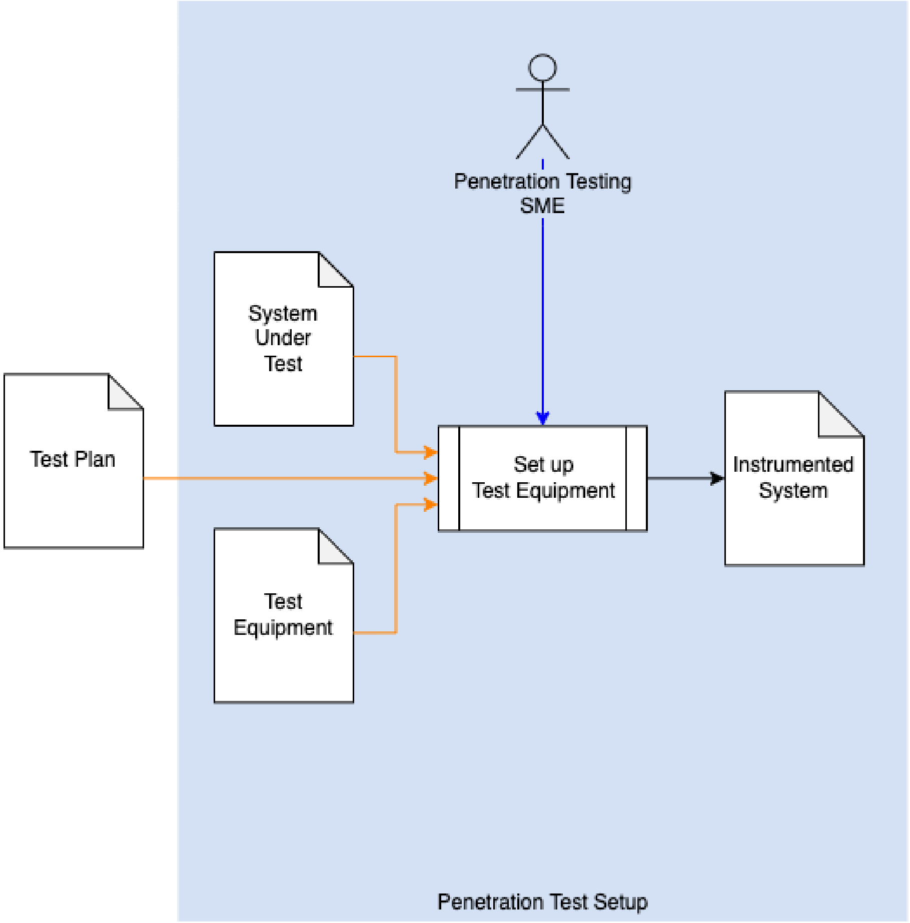
Inputs	Test setup documentation Refined attack vectors
Outputs	Test plan
Participants	Penetration testing SME



Using the **Test Setup Documentation** and **Refined Attack Vectors**, the Penetration Testing SME creates a plan to be used for penetration testing. A **Test Plan** document is generated.

Penetration Test Setup

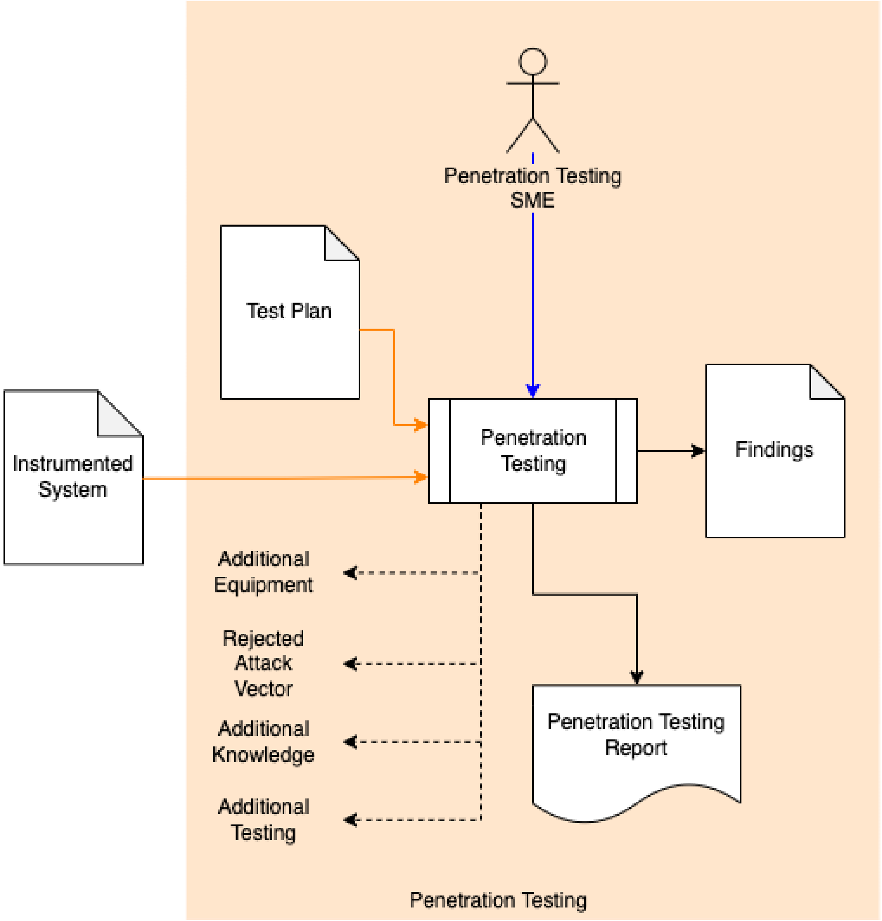
Inputs	System under test Test plan Test equipment
Outputs	Instrumented system
Participants	Penetration testing SME



Using the **Test Plan**, the Penetration Testing SME assembles the **System Under Test** and **Test Equipment**. This results in the creation of an **Instrumented System** to be used during penetration testing.

Penetration Testing

Inputs	Test plan Instrumented system
Outputs	Findings Penetration testing report
Participants	Penetration testing SME



Using the **Instrumented System**, the Penetration Testing SME follows the **Test Plan** to perform penetration testing on the SUT. A **Findings** list is generated. From the **Findings**, a **Penetration Testing Report** is generated. If additional equipment is necessary, an attack vector is rejected, additional knowledge of the system is gained, or additional testing is required, an appropriate notification is generated.

Note: The notifications generated here are a conceptual construct. In practice, steps will be taken to integrate this information into the **SUT knowledge** and **Methodology Knowledge**.

Note: The Penetration Testing SME may add issue remediation recommendations to the individual **Threat Candidates**.

Findings

The recommended form of the **Findings** artifact is a Static Analysis Results Interchange Format (**SARIF**) encoded JSON. This document assumes SARIF version 2.1.0 [3] or later.

Penetration Testing Report

The **Penetration Testing Report** is recommended to be produced from the **Findings** artifact and should detail the issues exposed by the penetration testing.

The report should be organized into a summary section and a list of one or more runs. The summary section includes:

- A use case indicator (penetration test findings)

Each run includes:

- Description of the test (copied from the test plan)
- Reference to the refined attack vectors document (typically a URI)
- Reference to the test equipment document (typically a URI)
- Reference to the test setup document (typically a URI)
- Reference to the test plan document (typically a URI)
- Reference to the instrumented system document (typically a URI)
- Timestamp, indicating when the test started (ISO 8601)
- The individual performing the test (email address)
- A list of one or more results

Each result contains:

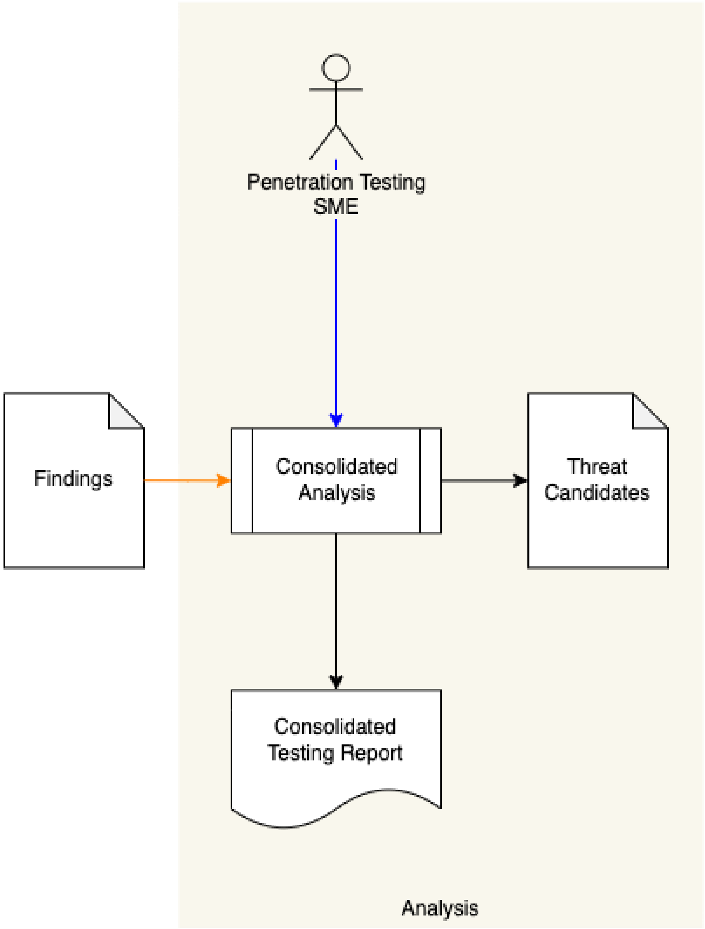
- Description of the test performed (copied from the test plan)
- Enumerated test result (typically pass or fail)
- Test ID (from the test plan)
- The expected result of the test (copied from the test plan)
- Indication of whether the test ran to completion (Boolean)
- Indication of whether the test passed (Boolean)

If the test result indicates a failure, the result shall additionally include:

- Description of the actual result
- A list of one or more findings

Analysis

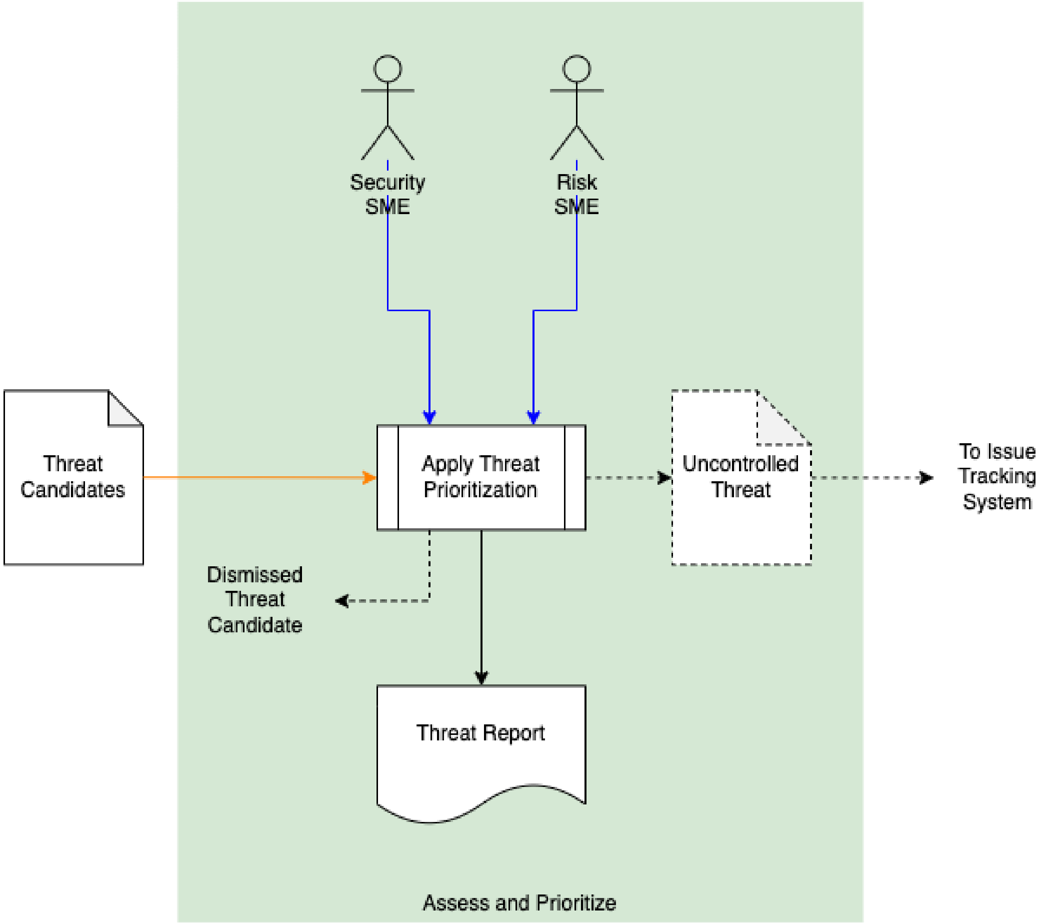
Inputs	Findings
Outputs	Threat Candidates Consolidated Testing Report
Participants	Penetration testing SME



Using the **Findings** from all the penetration testing effort, the Penetration Testing SME performs a consolidated analysis of the aggregate findings. A list of **Threat Candidates** is produced. A **Consolidated Testing Report** is generated.

Assess and Prioritize

Inputs	Threat Candidates
Outputs	Uncontrolled Threat Threat Report
Participants	Security SME Risk SME



The Security SME will take each **Threat Candidate** and apply the **Threat Prioritization Plan**. The **Threat Candidate's** rank and risk will be assigned by the Security SME and Risk SME respectively. A **Threat Report** documenting the findings will be generated. If the threat is determined to be controlled a **Dismissed Threat Candidate** notification will be generated. If the threat is determined to be uncontrolled, an **Uncontrolled Threat** will be generated.

Note: The **Dismissed Threat Candidate** notification is a conceptual construct. In practice, steps will be taken to integrate this new information into the **SUT knowledge**.

References

1. **Threat Prioritization Plan (**AVCDL secondary document)
2. **Threat Report (**AVCDL secondary document**)**
3. **Static Analysis Results Interchange Format (SARIF) Version 2.1.0**
   https://docs.oasis-open.org/sarif/sarif/v2.1.0/os/sarif-v2.1.0-os.pdf
4. **The Open Source Security Testing Methodology Manual**
   https://www.isecom.org/OSSTMM.3.pdf
5. NIST SP 800-115: **Technical Guide to Information Security Testing and Assessment**
   https://csrc.nist.gov/publications/detail/sp/800-115/final
6. **Penetration Testing Execution Standard**
   http://www.pentest-standard.org/index.php/Main_Page
7. **PCI Data Security Standard**
   https://www.pcisecuritystandards.org/documents/Penetration-Testing-Guidance-v1_1.pdf