การเจาะระบบ, การหาช่องโหว่
(ลองตรวจสอบ, โจมตี, ค้นหา ค.ปลอดภัย
เพื่อแก้ไข, ป้องกัน ค.เสี่ยง)

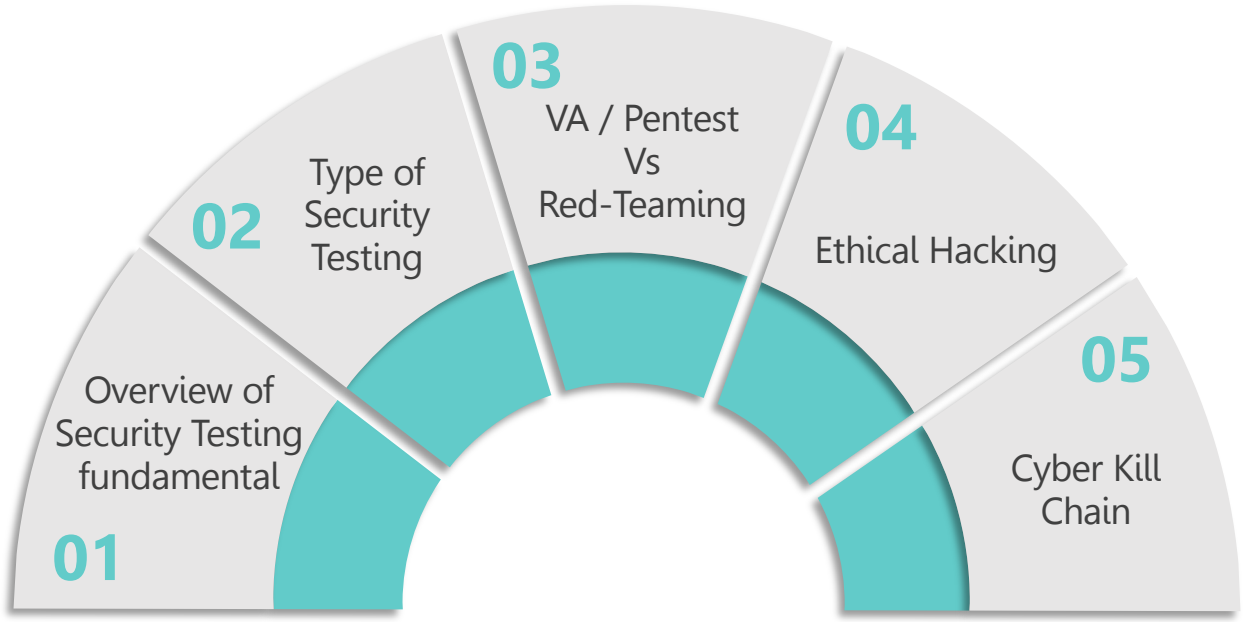
Offensive Security

Instructors : Nuttakorn T., Chananya C.

Cybersecurity Bootcamp
2024

Disclaimer

This learning material is addressed and used only for Cybersecurity Bootcamp 2024 and should not be used or relied upon for any other purposes. Our learning material is not to be disseminated to or used by any third party in whole or in part without prior consent and permission from Kasikorn Technology Group Secretariat Company Limited (KBTGSec). Accordingly, we will not accept or take any responsibility or liability for any party or any person, whether or not such material is shown, disseminated, obtained, or possessed to such party or person since such material is only for educational purposes. We reserve all of our rights, including but not limited to intellectual property rights in our learning material, such as presentations, spreadsheets, system techniques, ideas, concepts, information, forms, electronic tools, forming parts of the materials, etc.


01
Overview of Security Testing fundamental
02
Type of Security Testing
03
VA / Pentest Vs Red-Teaming
04
Ethical Hacking
05
Cyber Kill Chain

8 CHAPTERs


06
Security Testing Lifecycle
07
Mitigation & Recommendation
08
Hands-on Recommend

Key Objective

1. To understand the meaning and importance of VA/Pentest/Red-teaming
2. To understand security testing life-cycle methodology
3. To understand ethical hacking ethics
4. To understand fundamental of IT security knowledges

01

Overview of Security Testing Fundamental

01-1

Overview of Security Testing Fundamental

What is security testing?

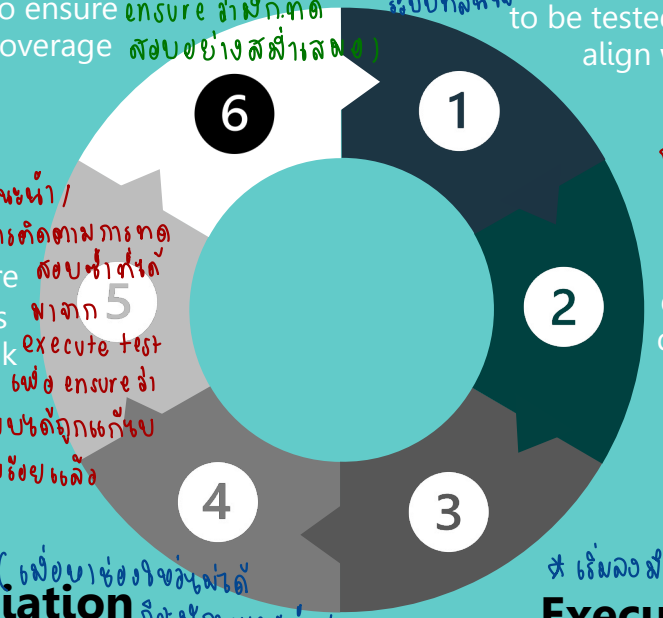
Procedure or methodology to detect weakness and vulnerability on target assets and information. To ensure the targeted has been audited properly and ensure that the discovered vulnerability had been remediated.

Importance of security testing:

- To ensure that the risk of assets have been managed and mitigated properly.
- To prevent the attacker from identifying and attacking assets vulnerabilities and system flaws.
- To gain reputation and trust from partner, related parties.
- To protect customer's sensitive data and privacy

Security Testing Lifecycle

Security testing lifecycle is a series of methods conducted over a defined period to systematically find and fix vulnerabilities on one or more assets.



Plan
Identify the asset(s) that need to be tested and time-period align with the plan

Scope
Determine testing depth method and define which assets to be in-scope testing

Execute & Test
Conduct execute test through manual by human and automated testing by tools

Remediation
Co-operate with asset owners to resolve or mitigated identified vulnerabilities

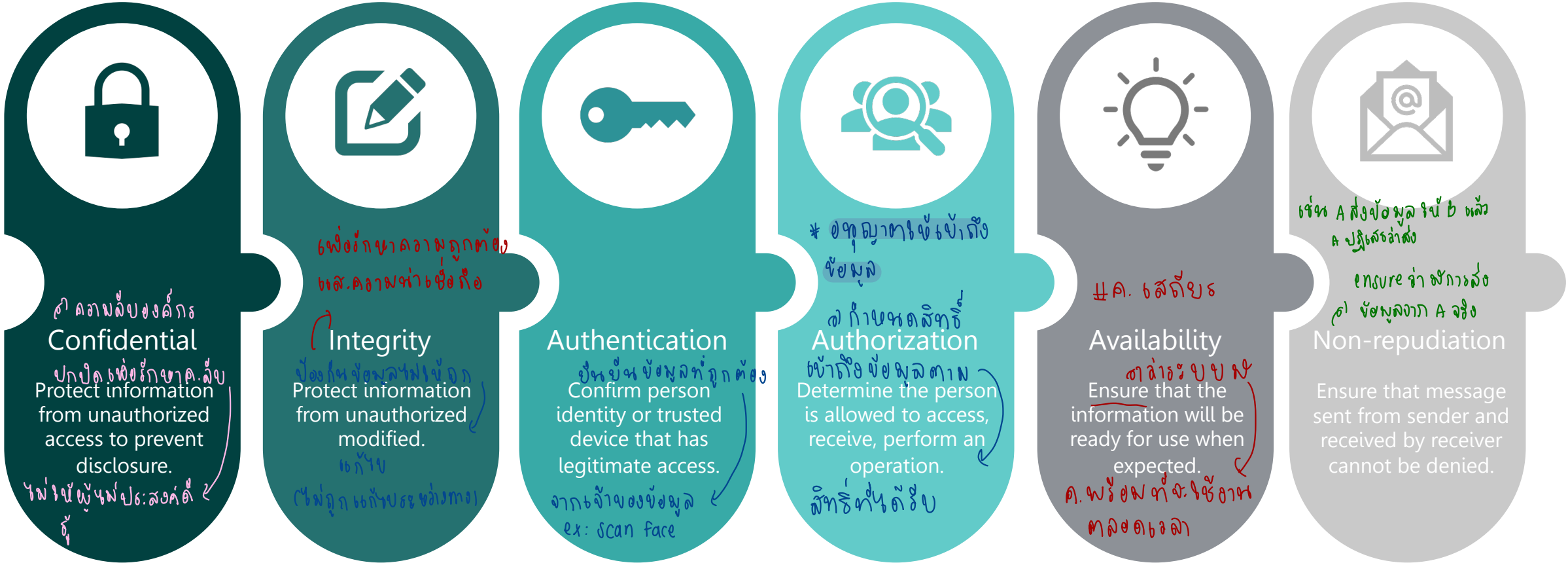
Retest
Tester check to ensure that asset owner has fixed or mitigated risk and vulnerabilities

Repeat
Define testing frequency to ensure continual coverage

Overview of Security Testing Fundamental

Security Testing Principle

Security testing principal is security requirement elements, that need to be included in security testing implementation to ensure that the assets and information are secured.



02

Type of Security Testing

Type of Security Testing

Vulnerability Assessment

- Identify potential weakness of infrastructure or network level
- Depth of assessment level is based on latest tool KB
- Usually not covered on application level
- Need to be whitelisting due to discovery all possible weakness
- Normally can be remediated by apply patch or configuration change

Penetration Testing

- Identify potential weakness of the applications
- Perform assessment by both of automation testing and manual by human
- Can be separated to many areas (network, application, mobile)
- Need to be whitelisting due to discovery all possible weakness
- Risk severity was declared by impact and likelihood of the findings
- Remediation usually fixed by code and network level.

Red Teaming

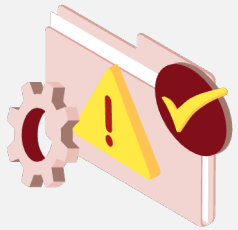
- Identify weakness of security control as real threat actor perspective
- Identify detection and response visibility of Security Operation team
- Not only focus vulnerability discovered, need to focus more on human operation
- Intelligence-Led methodology
- No whitelisting allow, to evaluate current effective security control
- Remediation and mitigation were varied on area of the finding issues.

03

VA / Pentesting vs Red Teaming

VA / Pentest vs Red-Teaming

Area	VA / Pentest	Red-Teaming
Main-Focus	To Identify and discover many of vulnerabilities as possible	- To ensure current security protection and control - To validate and enhance detection and response from Security Operation
Scope	Limited scope as agreement	Objective base-defined
Stakeholder	All stakeholders that would be involved	Exercise Working-Group Only
Intelligence-Led	No	Always being used
Social Engineering / Human Operations	No	Always being used
Execution Alignment	Technical based methodology	Align to Tactics, Techniques and Procedures (TTPs)



KBTG
KASIKORN
BUSINESS-TECHNOLOGY GROUP

04

Ethical Hacking

Ethical Hacker vs Threat Actor

KBTG KASIKORN BUSINESS-TECHNOLOGY GROUP

ฝั่งคุณธรรม

Ethical Hacker (White-hat Hacker)

#ขโมยรูกล้า Access ที่ยังไม่ได้รับอนุญาต

ค้นหาและแจ้งไปยังเจ้าของระบบ
Identify and notify to remediate
(แก้ด้วยเจตนาที่ดีต่างๆ)

รักษาข้อมูลความเป็นส่วนตัวของลูกค้า
Protect data and privacy

ไม่เปิดเผยช่องโหว่ที่ค้นพบอื่นๆ
Not disclose any finding or weakness

Threat Actor (Black-hat Hacker)

ค้นหาเพื่อทำข้อมูลเสียหาย
Identify then attack to cause damage
#ไม่แจ้งเจ้าของ

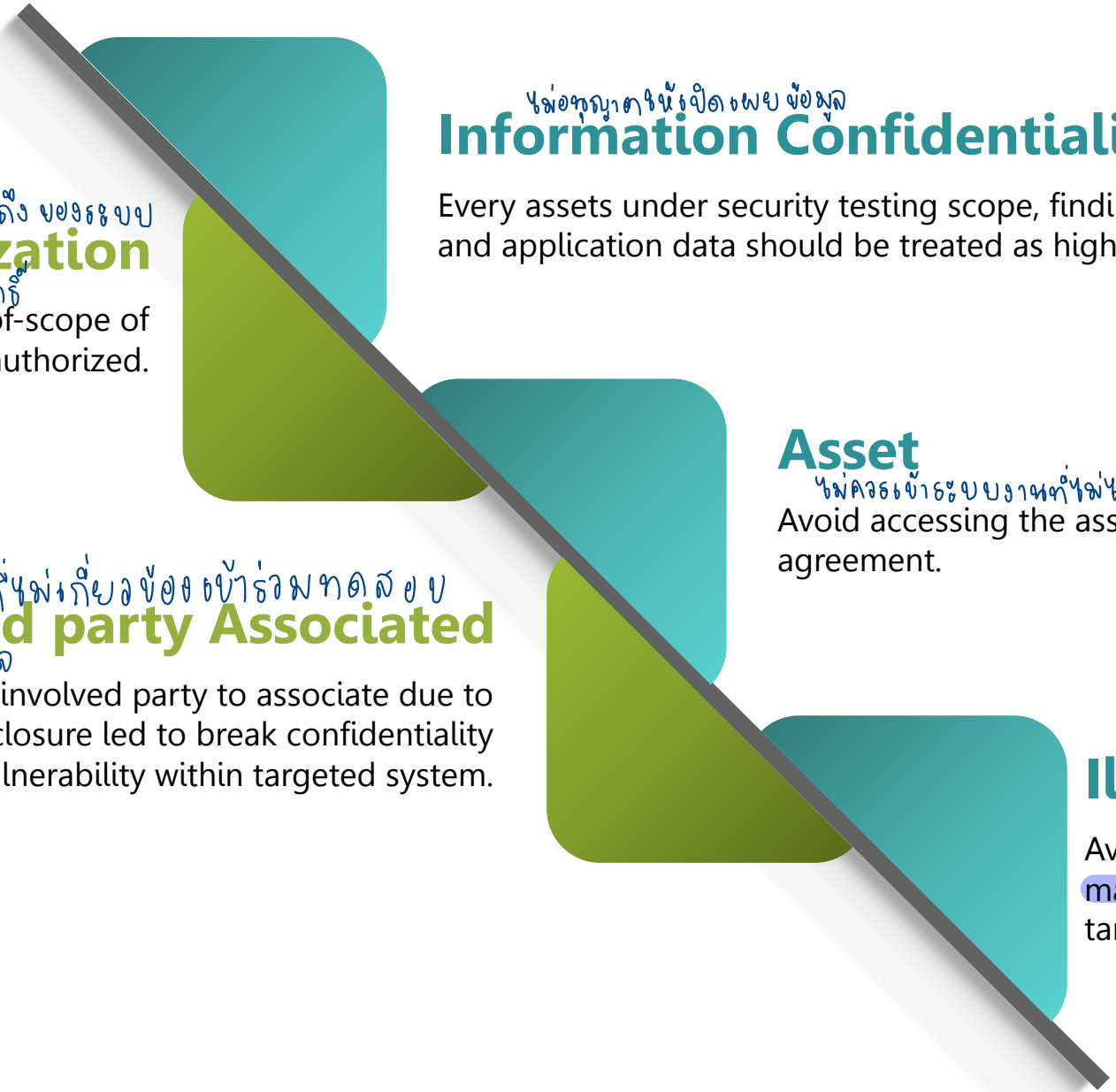
เอาข้อมูลไปทำให้เกิดความเสียหาย (เช่น ขายข้อมูล)
Espionage and sell on underground forum / dark market

ดึงเอาไปขายใน dark web
Defame by publishing to underground forum or selling data to dark market

ช่องโหว่ Vulnerabilities	
Data Confidentiality	
ความน่าเชื่อถือ Reputation	

Ethical Hacking: Code of Ethics

จริยธรรมที่ควรหลีกเลี่ยง



Information Confidentiality

ไม่อนุญาตให้เปิดเผยข้อมูล

Every assets under security testing scope, finding issues detail, and application data should be treated as highly confidential.

Authorization

กำหนดการเข้าถึงของระบบ

#ไม่ควรทดสอบในระบบที่ไม่มีสิทธิ์

Avoid accessing the out-of-scope of agreement items that we are not authorized.

Asset

ไม่ควรเข้าระบบงานที่ไม่ได้มีการตกลงกัน

Avoid accessing the assets which are out-of-scope of agreement.

Non-involved party Associated

ไม่ให้รู้ถึงบุคคลที่ไม่เกี่ยวข้องเข้าร่วมทดสอบ

เสี่ยง data รั่วไหล

Avoid bringing non-involved party to associate due to sensitive information disclosure led to break confidentiality of vulnerability within targeted system.

Illegal Software

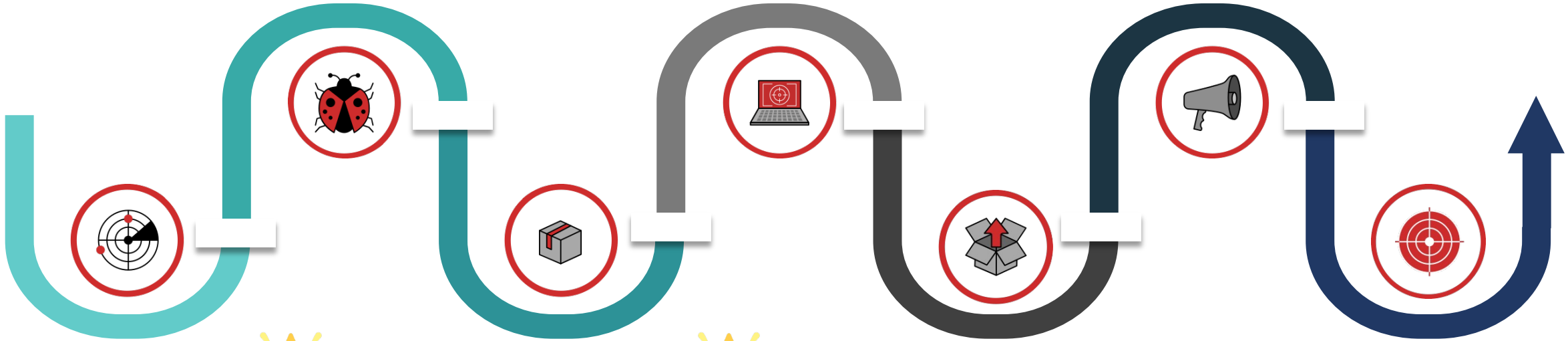
ไม่ควรใช้ software เถื่อนเสี่ยงมัลแวร์

Avoid using illegal software due to malware injection opportunity into targeted system.

05

Cyber Kill Chain

Cyber Kill Chain



1. Reconnaissance

Identify a target, explore vulnerabilities and weaknesses

2. Weaponization

Craft malware & exploit into deliverable payload to attack target system

3. Delivery

Deliver payload to launch attacks on target, may include user interaction

4. Exploitation

Malicious code within payload is executed within victim's target system

5. Installation

Malicious payload is installed into victim's target system to gain access

6. Command and Control

Use malware to remote control and lateral movement within system

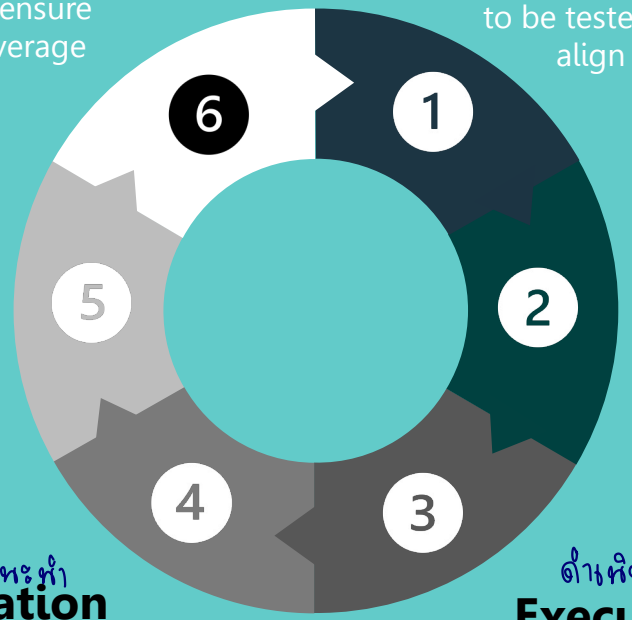
7. Actions on Objective

Carry out to achieve goal, i.e., exfiltration, modifying etc.

06

Security Testing Lifecycle

Security Testing Lifecycle (Recap)



วางแผน
Plan
Identify the asset(s) that need to be tested and time-period align with the plan

Scope
Determine testing depth method and define which assets to be in-scope testing

ดำเนินการทดสอบ
Execute & Test
Conduct execute test through manual by human and automated testing by tools

การแก้ไข/แนะนำ
Remediation
Co-operate with asset owners to resolve or mitigated identified vulnerabilities

ทดสอบซ้ำ
Retest
Tester check to ensure that asset owner has fixed or mitigated risk and vulnerabilities

กำหนดค.ถี่
Repeat
Define testing frequency to ensure continual coverage

Security Testing Phase 1 - Plan

① Stakeholder Management

keep contect กลุ่มคนที่เกี่ยวข้องให้ครบ

Establish all stakeholders that need to be involved for the project.

② Project Timeline

กำหนด ฉ กับผู้เกี่ยวข้องให้ครบ

Discuss project timeline to adjust the period to perform security testing.

#ไม่ทับtimelineคนอื่น กำหนดวันทดสอบให้ชัดเจน

③ Resource Planning

กำหนดว่าต้องใช้คนเท่าไหร่ เพื่อให้เพียงพอกับการทดสอบฯครบถ้วน

Plan available resources for assigning to be security tester of the project.

④ Schedule to execute

ระบุวันที่ทดสอบให้ชัดเจน กำหนดช่วงเวลา

Appoint with related parties on each security testing phase that aligned with project time plan.

⑤ Environment Preparation

เตรียมข้อมูลที่ใช้ในการทดสอบ

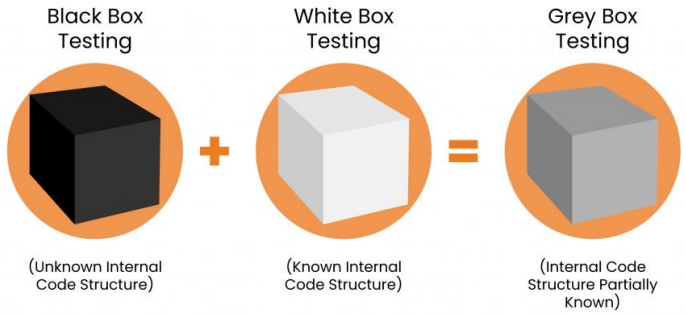
Discuss with asset owner what environment or assets to be used for testing that testers can design approaching method.

06-2

Security Testing Phase 2 - Scope

เลือก scope ให้เหมาะกับ ความละเอียด

Security Testing Methodology



เหมาะสุด
เสียเวลาคนโจมตีที่นานแต่ได้ข้อมูลอะไรเลย

Black-box Testing

หาช่องโหว่ = ไม่รู้อะไรเลย

- Tester accesses the system without knowing the target's internal structure
- Focus on input and output from behavior under testing, as user's perspective or real threat actor
- Based on "Dynamic Application Security Testing (DAST)"

-ไม่ลงลึกถึงรายละเอียดใน App นั้นๆ

Grey-box Testing

มีข้อมูลในส่วน Application

- Tester is partially known the target's internal structure
- Combines input from developers and can result more effective testing strategy
- Focus on the paths most likely to affect users or result in a defect
- Commonly used in penetration testing

ประเมินค.เสี่ยงธุรกิจ
ที่ละเอียดขึ้น

(เจ้าของระบบเล่าว่าระบบมีกระทำงานยังไง)

เห็นค.เสี่ยง function ขณะการทำงาน

White-box Testing

รู้ข้อมูลทั้งหมด (รู้ในสิ่นทั้งหมด)

- Tester is fully known the target's internal structure
- Allow tester to analyze the internal structure such as infrastructure and source code
- Based on "Static Application Security Testing (SAST)"
- Commonly used in Static Code Analysis or White-box Pentest

ลงลึกถึง source code

Security Testing Phase 2 - Scope

Defining Scope

Methodology

เลือกใช้แบบที่เหมาะสม

Define appropiate security testing methodology and approach (Black-box / White-box / Grey-box).

Environment & Data Preparation

เตรียม

Define testing environment and necessary test data for security testing e.g., IPs / URLs, Credentials, Application details, etc.

ให้เหมาะกับการทดสอบ (เตรียมข้อมูลให้ดีมากที่สุด)

Execution Period

กำหนดช่วงเวลาให้ชัดเจน

Define and discuss with stakeholders or related parties about time period for test execution, remediation, and retest period.

ควรกำหนดช่วงเวลา retest ด้วยเลย

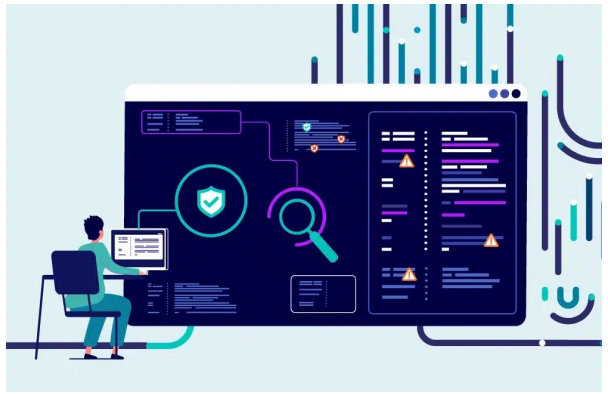
Security Testing Phase 3 – Execute & Test

KBTG KASIKORN BUSINESS-TECHNOLOGY GROUP

SAST vs DAST

เลือก metrology ในการทดสอบ

Static Application Security Testing (SAST)



* ตรวจสอบในระดับ source code
White-Box Security Testing

เพื่อให้เห็นค. เสี่ยงลึกๆ
Requires source code

พบได้จาก source code ที่รีวิวแล้ว
Vulnerability found in earlier development stage

ตรวจสอบได้เลย

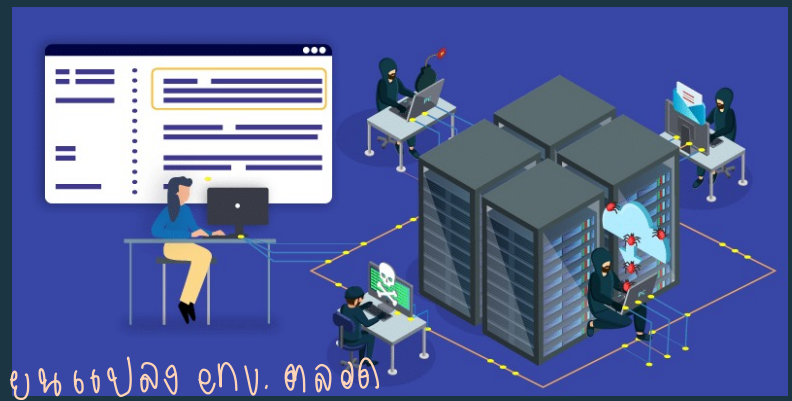
ผู้ตรวจสอบ ควรเชี่ยวชาญ ภาษาต่างๆ ที่ใช้งานในระบบนั้นๆ
Specific platform and language

อ่าน + วิเคราะห์ code

ไม่สามารถพบได้ พบได้ตอน runtime/env. ที่ทดสอบ นั่นเลย
Unable to identify runtime and environment issue

Test application at low level

Dynamic Application Security Testing (DAST)



ศักยภาพเปลี่ยนแปลง ตาม ค. ตลอดเวลา

Black-Box & Grey Box Security Testing

แค่เข้าใช้ระบบนั่นก็พอ
Requires running application

เจอได้ตอนทดสอบ/ใช้งานจริง
Vulnerability found in later development stage

พบได้ตอน runtime เลย
Able to identify runtime and environment issue

ไม่จำเป็นต้องเชี่ยวชาญในภาษานั้นๆ
Support various platforms and languages

แค่ user ที่เข้ามาทดสอบ (ไม่ลงลึก)
Test application at high level

Security Testing Phase 3 – Execute & Test

Static Application Security Testing (SAST)

Since the SAST is focused testing on application source code, we mostly used SAST on scope as white-box testing, DevSecOps, and perform static analysis on Mobile Application Package.

SonarQube

Scans source code for 15 languages for Bugs, Vulnerabilities, and Code Smells.

MobSF

An automated, all-in-one mobile application pen-testing, malware analysis and security assessment framework

Fortify

Pinpoint the root cause of vulnerabilities in source code, prioritizes issues, and provides detailed guidance on how to fix it.

CheckMarx

The enterprise software that scans, detects, & prioritizes vulnerabilities to reduce risk across software components.

Security Testing Phase 3 – Execute & Test

Dynamic Application Security Testing (DAST)

In VA and Penetration Testing use DAST method to find application vulnerabilities in various type such as Web Application, Mobile Application, API, and Network because we don't assessment only source code, but focus whole application.

Nessus

Platform that scans for security vulnerabilities in devices, applications, OS, and other network resources

Burp Suite

Integrate platform and graphical tool for performing security testing of web application

Nuclei

An automated scanner that offers scanning for variety protocol and be used to model all kinds of security check

Kali Linux

Open-source distribution that providing over 600 tools for penetration testing and security auditing

VA & Pentest
Use Cases in Security Testing Assessment

Security Testing Phase 3 – Execute & Test

หาช่องโหว่

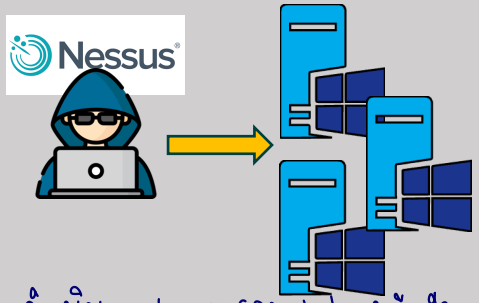
Vulnerability Assessment Use Case Example : Remote Desktop Protocol Vulnerability (BlueKeep)

In this VA use case, we will sample how we use a tool like Nessus to scan the network and discover the severe vulnerability in the target device.

① 

ควรอัพเดต plug-ins สม่ำเสมอ

Tester perform update plug-ins / KBs for VA scanner tool then set task for VA scanning.



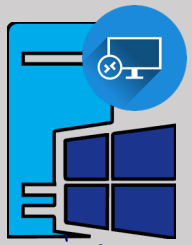
ตั้งค่าการ target asset ต่างๆ (อาจเป็น group)

Tester set scope for group assets, then perform VA scanning toward the targeted server.

เพื่อทำ VA

พบ service port ต่างๆ

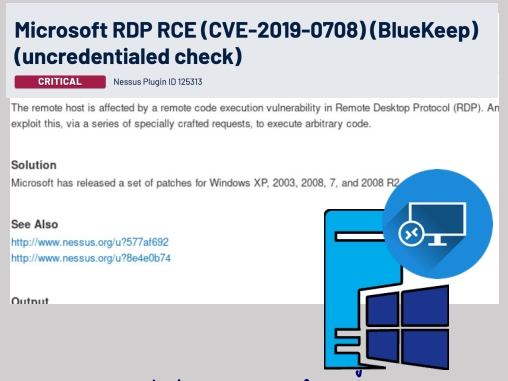
Found many service ports are opened.



นำข้อมูลที่ได้จากการ scan มา review

Tester review the detailed scan result, found some the targeted server has RDP port 3389 is the one of opened service ports. And reveal the critical vulnerability result.

ช่องโหว่ร้ายแรง



validate ว่าช่องโหว่ถูกต้องมั้ย

Tester found the RDP protocol is vulnerable to "BlueKeep" that is critical vulnerability.

ร้ายแรง

false protective => สิ่งที่พบจาก tool พบว่ามีปัญหา แต่จริงแล้วแก้ไขความเข้าใจผิดจาก tool ของ

Security Testing Phase 3 – Execute & Test

chatgpt again

แยกคำสั่งอันตราย เพื่อไปบ่งเกี่ยวกับ data base

Penetration Test Use Case Example : Web Application Pentesting - SQL Injection Vulnerability

In this pentest use case, we will sample how we identify web application vulnerabilities, such as SQL injection, using manual and automated tools.

Tester enumerates the target webapp to find vulnerability foothold.

Found that URL has parameter like: 'http://test.com/book.php?id=1'

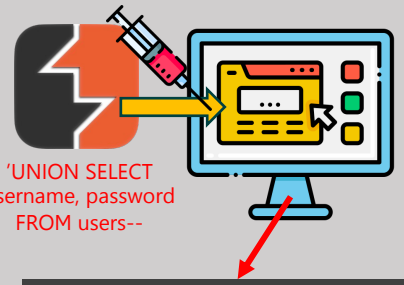
This URL probably use 'id' parameter to get book data from database.

Tester test this URL to find if there is SQL injection vulnerability by adding the payload like:

'http://test.com/book.php?id=1' OR '1 -- -'

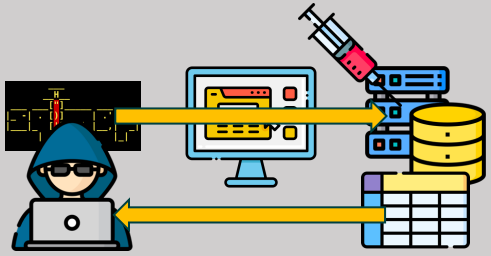
Logic เพื่อดู error

Found error message from SQL database, we can assume that this webapp is probably vulnerable to SQL injection.



SELECT book_id, name FROM book WHERE book_id = '$id' UNION SELECT username, password FROM users--

Tester will use more advance SQL injection payload to see the response from application, such as retrieve all data from database or can access data from other tables.

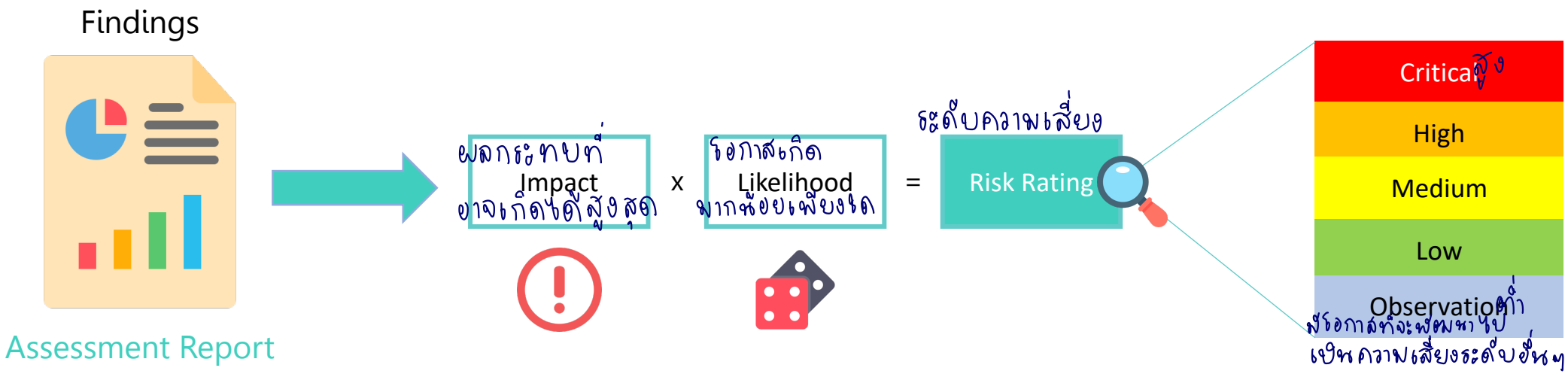


Tester may use automated tools like "sqlmap" to exploit the SQL injection vulnerability to proof more impact such as dumping all data from database or launching operating system shell prompt from execute command via SQL query etc.

Security Testing Phase 3 – Execute & Test

Assessment Report

- Every vulnerabilities issues from assessment will be called as “findings”.
  - Each finding item has different risk severity rating, based on the impact and likelihood
- After finishing assessment, the tester must write the assessment finding report to be used as artifact of evidence and remediation guidance for asset owners to fix the vulnerability.

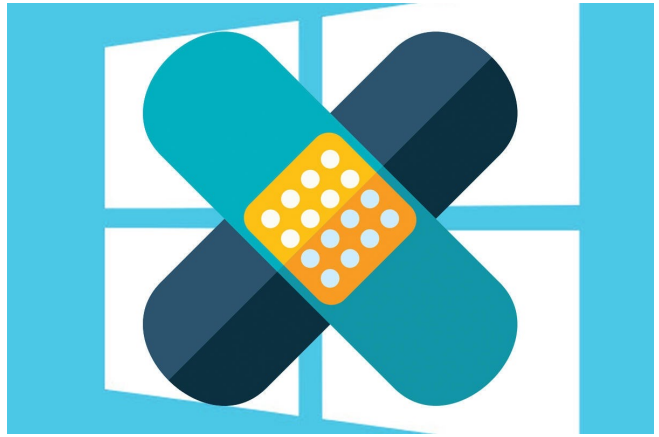


Security Testing Phase 4 – Remediation

Security finding remediation is meant to the methodology or action to remediate / eliminate the risks, caused by existing weakness or vulnerability on target assets.

Vulnerability Assessment (VA)

- Usually co-operate with asset owners to apply patches, update software version, or upgrade operating system to fix the vulnerabilities findings.
- Apply configuration to remediate the fixed issues.



Penetration Testing

- Usually co-operate with asset owners / developers to fix the issue by secure coding and hardening configuration to prevent the vulnerabilities from application bug and infrastructure flaws.
- Sometimes some issue findings cannot be eliminated due to business requirements or software package limitation, we will recommend to "mitigate" the issues to reduce the risk.

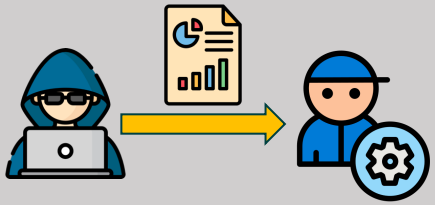
```
public PreparedStatement journalEntrySearch(
  Connection con,
  int personId,
  String wildcard) {

    String sql = "SELECT CreatedTimestamp, Body FROM journal_entries " +
      "WHERE PersonId = ? AND Body LIKE ?"
    PreparedStatement search = con.PrepareStatement(sql);
    search.setInt(1, personId);
    search.setString(2, "%" + wildcard + "%");

    return search;
}
```

Security Testing Phase 4 – Remediation

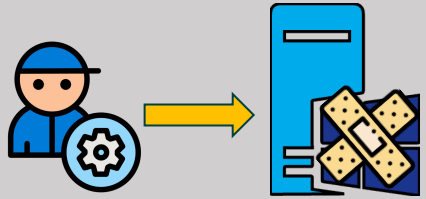
Use Case Example : Apply security patches (VA) & Fix vulnerable code (Pentest)

In this example, we will show how to co-operate with asset owners to fix the findings from assessment both VA and pentest.

VA: BlueKeep (RDP Vulnerability)

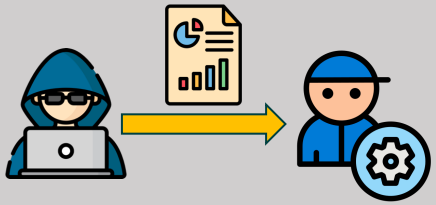


Tester reports assessment result to asset owner along with providing remediation guideline to fix the issue finding(s).

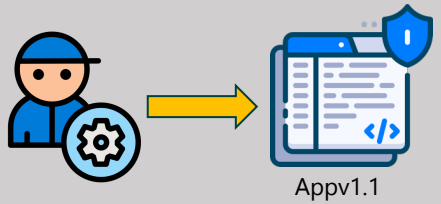


Asset owner install hotfixes / apply security patches to fix the critical vulnerability.

Pentest: SQL Injection



Tester reports assessment result to asset owner / developer along with providing remediation guideline to fix the issue finding(s).



Asset owner / developer fix vulnerability code to secure code to fix the vulnerability.

For example, the best practice to fix SQL injection is using “Prepare Statement” in input code.

Security Testing Phase 5 – Retest

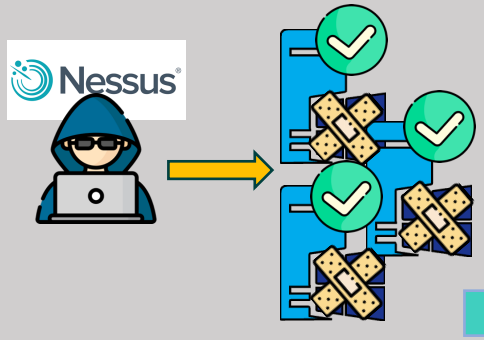
Why is re-test / re-visit validation important?

ต้องการยืนยันแน่ใจว่าช่องโหว่ที่พบนั้นถูกแก้ไข / risk ลดลง

- To ensure risk status of finding items. Is it fixed? Has the risk been reduced or not?
- To track and follow-up for finding status until it will be remediated.

จนกว่าจะมีการแก้ไข

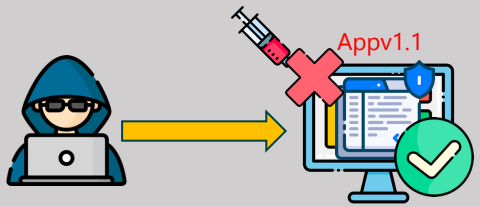
VA: BlueKeep (RDP Vulnerability)



Tester performs re-scan after informing by asset owner to verify that the vulnerability is fixed.

If the vulnerability has been fixed, the tester will confirm, and report result back to owner with re-assessment report.

Pentest: SQL Injection



Tester performs re-test after informing by developer to verify that the vulnerability is fixed and deployed on new app version.

If the vulnerability has been fixed, the tester will confirm, and report result back to developer with re-assessment report.

Security Testing Phase 5 – Retest

Defining Follow-up Cycle

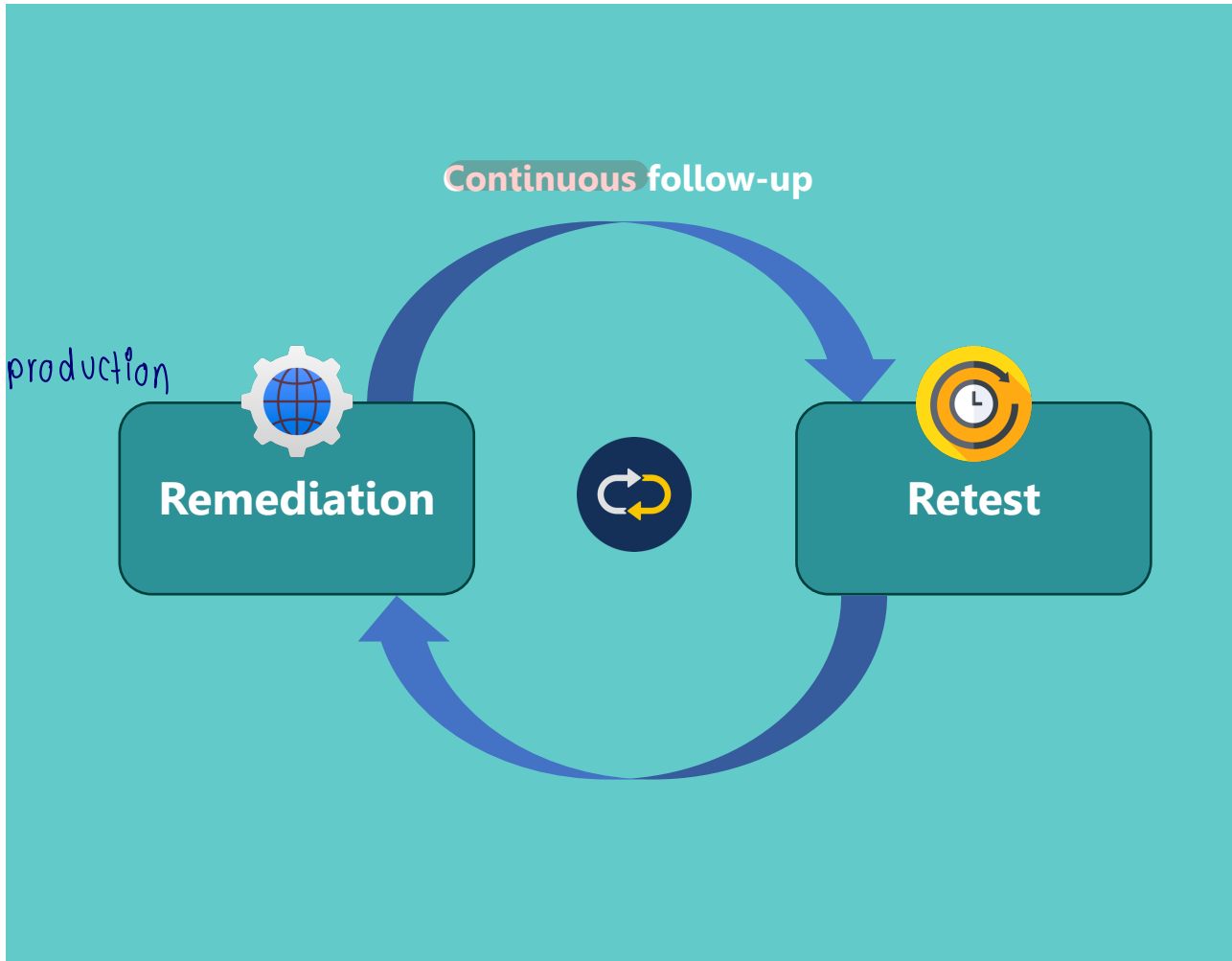
To define the frequency of findings follow-up cycle until they were remediated.

Project-Based

Define follow-up cycle frequency that focus only significant risk level of findings which have to be fixed before go-live.

SLA Based

Define follow-up cycle frequency by SLA (Service Level Agreement) for each risk level of each findings.



Security Testing Phase 6 – Repeat

How important of Repeat test ?

To establish the appropriate frequency of perform security testing cycle, and ensure target is audited continuously

Continuous

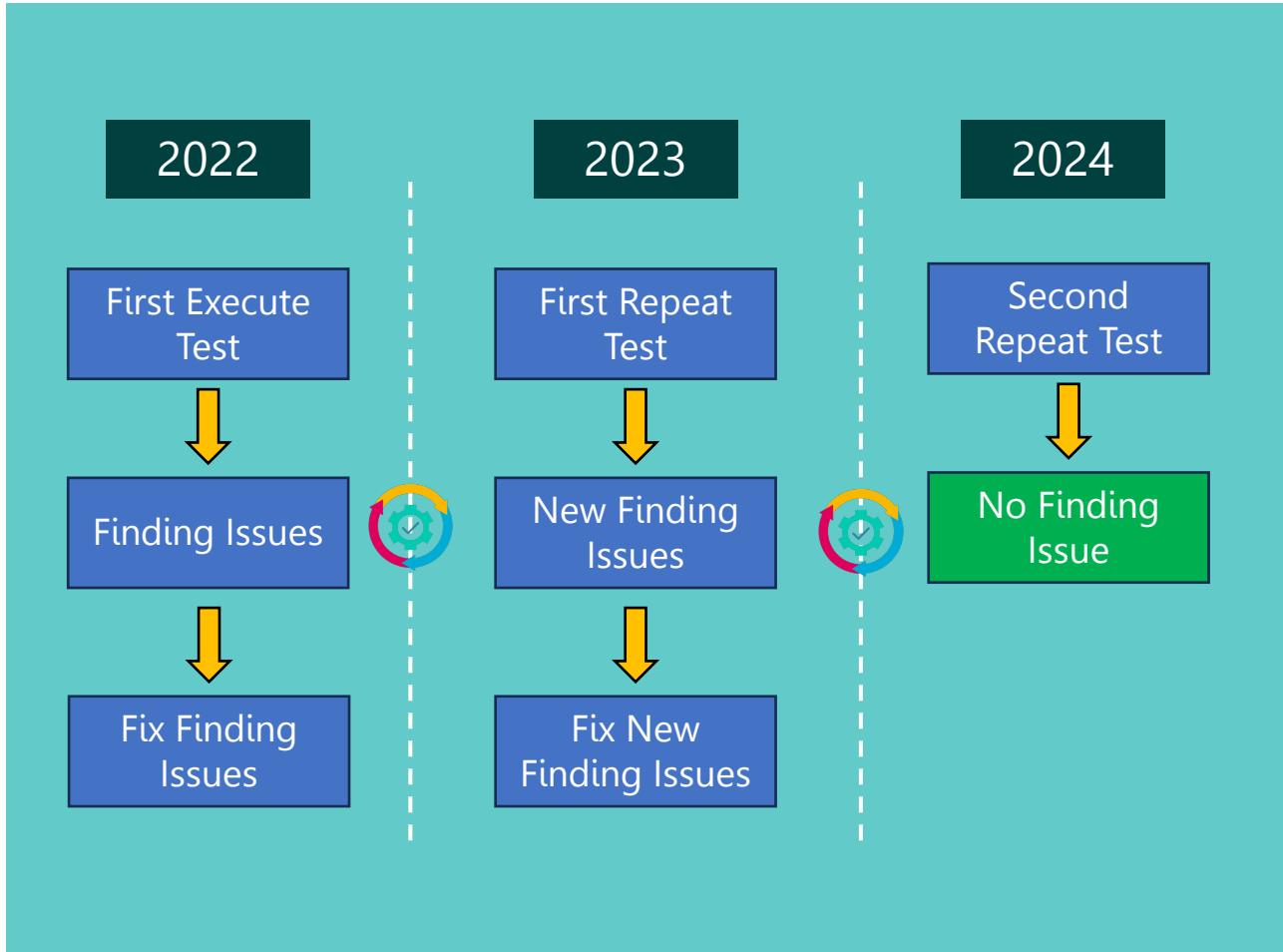
To perform continuous vulnerability validation on target assets or application

Frequency

To define frequency to perform retesting, based on agree with stake-holder (quarterly, annually, etc.)

Target Scope

To ensure no remaining scope that had not been audited on target assets or applications



07

Mitigation & Recommendation

Mitigation & Recommendation

What is risk mitigation and recommendation?

To discuss remediation and recommend solution, to reduce current risk in case of some finding issues cannot be fixed permanently within defined period (Both project-based and SLA-based).

Mitigation: Common practice

Apply security control: Define security control on assets to reduce attack surface from attacker

- **Implement Access Control List**: Ex. Restrict the assets to use only internal network or trusted network and ensure it is unable to access from public network.

Apply process: Define additional and change process of work to review, monitor, and make it more secure.

- Defense in depth
- Least privilege / Privilege Account
- Role-based Access Control

Mitigation & Recommendation

Recommendation: Best Practice

Secure Coding & Validation

Write code to develop software to secure against vulnerabilities attack.

- Prepare Statement
- Input Validation ทุกจุดที่เป็น ฟอร์ม
- Code sanitization
- Authentication & Authorization validation
  ถ้าสาย grey box จะเจอ Issue เยอะ

Hardening & Up-to-date Software

update ของล่าสุด

Not only the code, but the infrastructure must be secure too.

- Applying hardening configuration
- Using string cipher suite
- Do not use outdated and vulnerable software or product

08

Hands-on Training Recommend

Hands-on Training Recommend (Learning)

TryHackMe

Online, cloud-based, cybersecurity training platform used by individuals and academics alike.

HackTheBox Academy

Learning platform as real-world environment practice for every skill levels.

Portswigger Academy

Online training center for web application security from Burp Suite Creator.

Pentester Academy

Subscribe to learn hands-on and technically challenging courses in cybersecurity industries.

TCM Academy

Online training platform from The Cyber Mentor which provide various hacking topics to learn.

Sec Playground

Fulfill your cybersecurity journey from learning and practicing with intensive courses.

Hands-on Training Recommend (CTF-based)

VULNHUB
VULNERABLE BY DESIGN

VulnHub

Provide materials allowing anyone to gain practical hands-on experience with cybersecurity.

HACKTHEBOX

HackTheBox Lab

Gamified cybersecurity upskilling and talent assessment platform for hackers and organizations.

picoCTF

picoCTF

Gamified cybersecurity education program based on Capture-The-Flag (CTF) framework.

ATTACK DEFENSE

Attack-Defense

1800+ Labs from Pentester Academy. Covering various topic to practice hands-on.

OWASP Juice Shop

Open-source project by OWASP to used for web application security training

Root Me

Root-me

A platform for everyone to test and improve knowledge in computer security and hacking.

Key Takeaway

- The benefit of **“security testing”** is
  - To eliminate risk from vulnerabilities
  - To prevent damage from attackers
  - To protect sensitive data, privacy, reputation

- Security testing is categorized as 3 types:
  - Vulnerability Assessment
  - Penetration Test
  - Red Teaming

Key Takeaway

- Phase of security testing life cycle:
  - 1 - Plan
  - 2 - Scope
  - 3 - Execute and Test
  - 4 - Remediate
  - 5 - Retest / Revisit
  - 6 - Repeat

- There are many tools and techniques can be used in security testing based on agreement target. Be sure to cover it all.

- Remediation is the master key to eliminate and mitigate the risks from security testing.

Key Takeaway

- Being ethical hacker, remember to have strong **“ethics”** before doing any security testing.

of knowlegge + technical skill

- Ethical hacker prevents sensitive data and reputations from threat, while threat actor aims to defame and don’t care any cost they’ve committed.

Key Takeaway

- Offensive Security path requires fundamental knowledge and strong technical skills. Be sure to learn hands-on and practice various scenarios to improve knowledge.

- Stuck while practicing?? Don’t worry. Just **“TRY HARDER”.**

Thank You for Watching

Hope you enjoy offensive security path :)