

Università degli Studi di Salerno



Dipartimento di Informatica

# Penetration Testing & Ethical Hacking

## Tipi e Metodologie di Testing

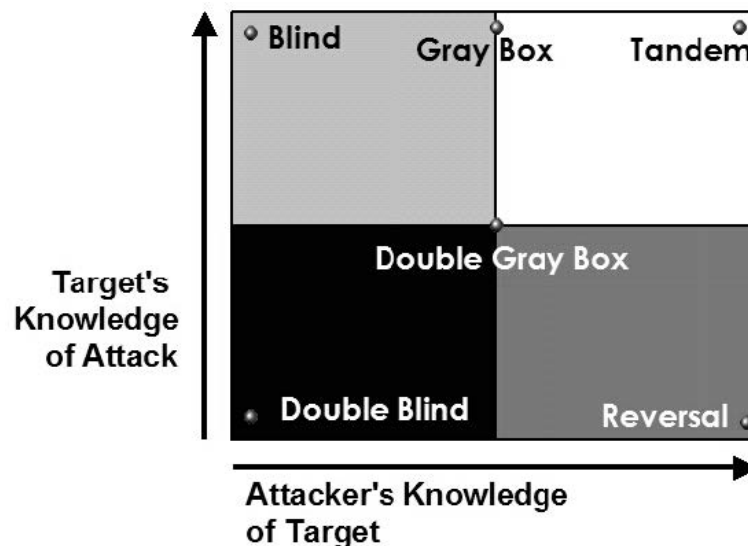
### Parte 3

Arcangelo Castiglione  
arcastiglione@unisa.it

# Metodologie di Testing

## OSSTMM – Tipi di Test

- I tipi di test si differenziano in base alla quantità di informazioni che
  - Il **pentester possiede sull'obiettivo** (*asset*) da valutare (**Asse X**)
  - L'**asset possiede sul pentester** (**Asse Y**)



Target = *asset*  
Attacker = *pentester*

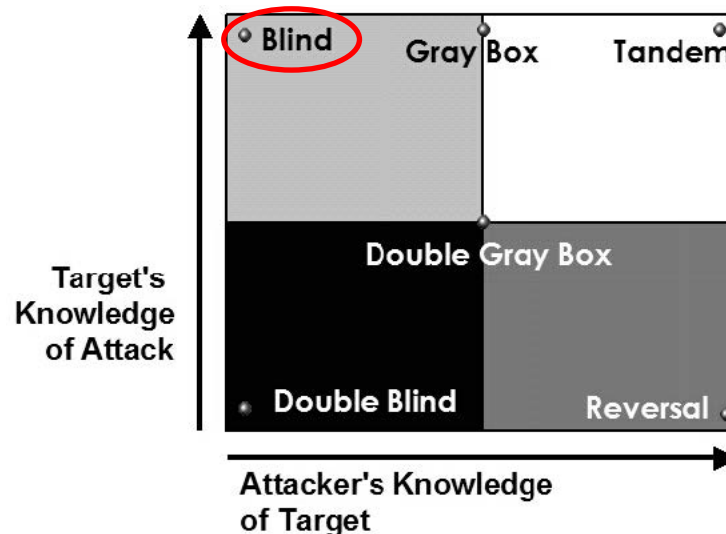


# Metodologie di Testing

## OSSTMM – Tipi di Test

### ➤ Blind

- Non richiede al pentester alcuna conoscenza preliminare sull'asset da valutare
- L'asset viene informato prima dell'esecuzione del test
- Ciò rende questo tipo di test ampiamente accettato

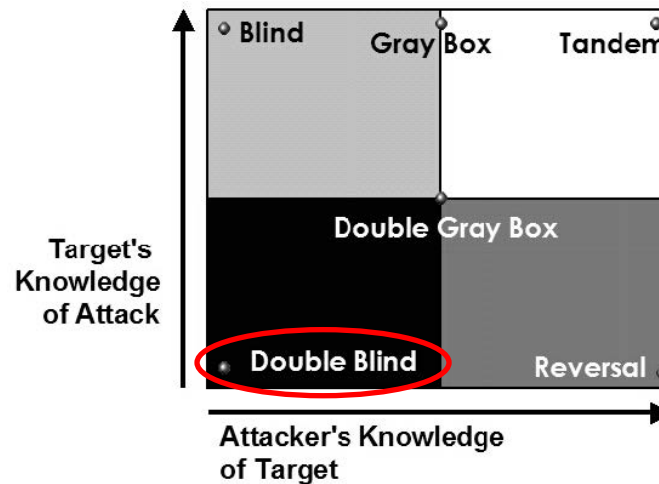


# Metodologie di Testing

## OSSTMM – Tipi di Test

### ➤ Double Blind

- Né il pentester ha alcuna conoscenza dell'asset né l'asset viene informato prima dell'esecuzione del test
- **N.B.** La maggior parte delle valutazioni di sicurezza oggi viene eseguita utilizzando questa strategia

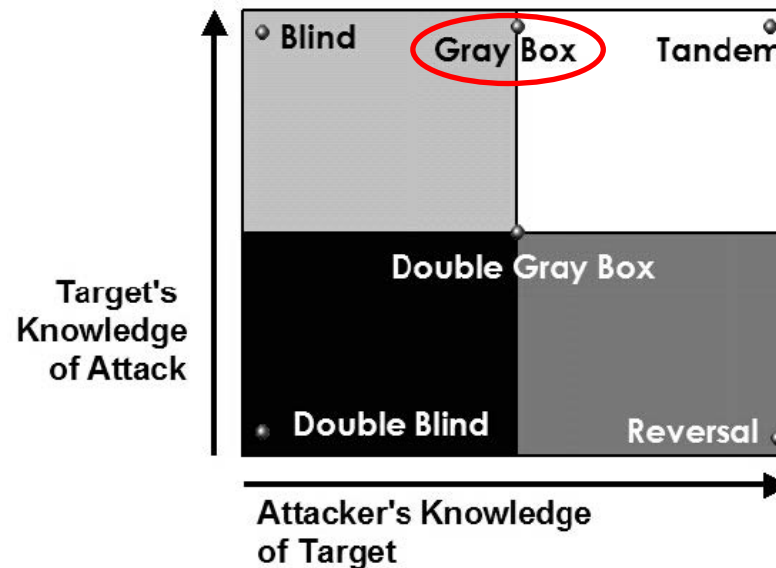


# Metodologie di Testing

## OSSTMM – Tipi di Test

### ➤ Gray Box

- Il pentester ha conoscenza limitata sull'asset
- L'asset viene informato prima dell'esecuzione del testing

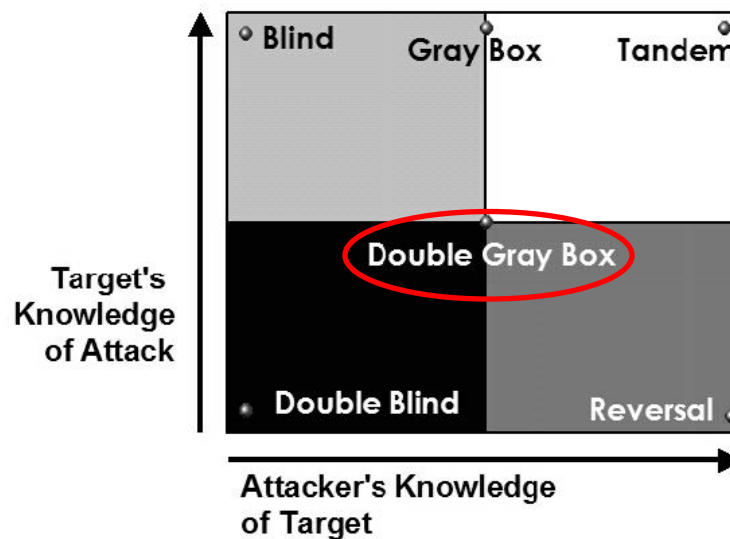


# Metodologie di Testing

## OSSTMM – Tipi di Test

### ➤ Double Gray Box

- Opera in modo analogo al Gray Box testing
- Ma pone specifici vincoli sulla durata del testing

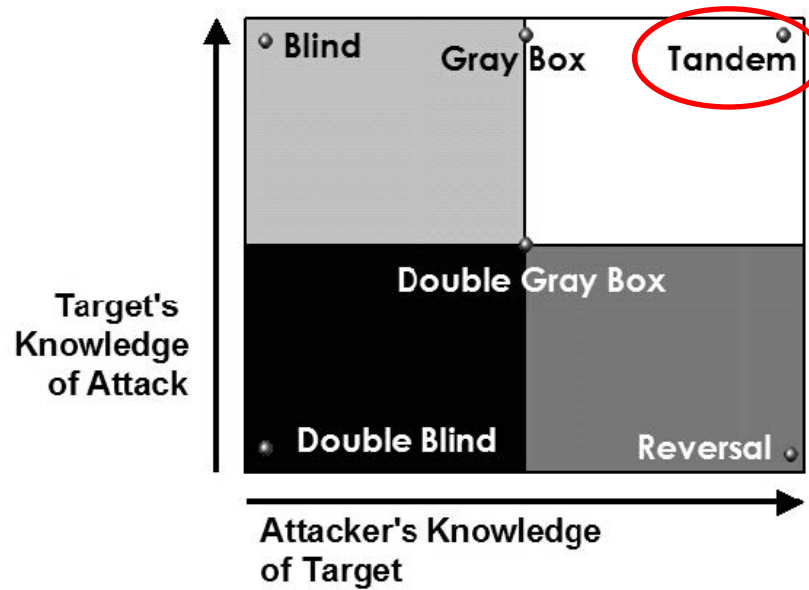


# Metodologie di Testing

## OSSTMM – Tipi di Test

### ➤ Tandem

- Il pentester ha piena conoscenza dell'asset
- L'asset è informato su come e quando verrà condotto il test

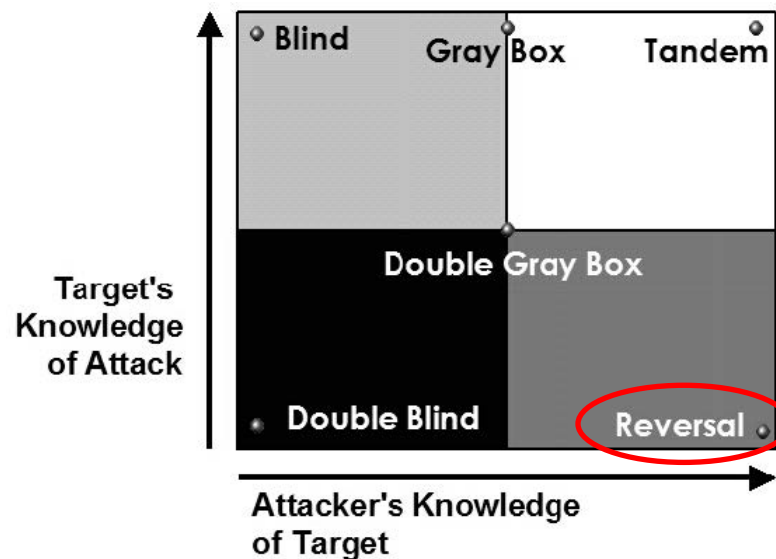


# Metodologie di Testing

## OSSTMM – Tipi di Test

### ➤ Reversal

- Il pentester ha piena conoscenza dell'asset
- L'asset non ha alcuna conoscenza del pentester





# Metodologie di Testing

## OSSTMM – Casi e Procedure di Test

---

- OSSTMM permette anche di definire **casi di test**, che generalmente valutano aspetti quali
  - Sicurezza del controllo accessi
  - Sicurezza dei processi
  - Controllo dei dati
  - Protezione perimetrale
  - Livello di consapevolezza della sicurezza da parte del personale
  - Etc



# Metodologie di Testing

OSSTMM – Casi e Procedure di Test

---

- Le **procedure di test** si concentrano su
  - Cosa deve essere valutato (asset)
  - Come deve avvenire la valutazione
  - Quali azioni devono essere messe in atto prima, durante e dopo la valutazione
  - Come devono essere interpretati e correlati i risultati ottenuti al termine della valutazione



# Metodologie di Testing

## OSSTMM – Risk Assessment Value (RAV) Score

---

- Al termine del processo di valutazione viene calcolato un valore (*metrica*) di sicurezza
  - **RAV (Risk Assessment Value) Score**
- **RAV Score**
  - Rappresenta lo stato dell'asset in termini di sicurezza
  - Può essere usato
    - Dal pentester per fornire un'idea precisa sulla sicurezza di un asset
    - Da un'organizzazione per ottimizzare la quantità di investimenti richiesti per la messa in sicurezza del proprio asset



# Metodologie di Testing

## OSSTMM – Risk Assessment Value (RAV) Score

---

- Mostra quanto un asset sia sicuro rispetto alle minacce
- È un valore quantitativo, di tipo numerico
  - Un **RAV Score** pari a **100** denota una «sicurezza perfetta»
    - Equilibrio «ottimale» tra **Vettori** e **Controlli**
  - Un **RAV Score inferiore a 100** evidenzia quali **controlli** sono **insufficienti o assenti**
  - Quando il **RAV Score** è 100 e vengono aggiunti ulteriori controlli esso **supera 100**
    - Ciò denota che si stanno «**sprecando**» **risorse**: «inutile» investire risorse per migliorare qualcosa che è già «perfettamente sicuro»




# Metodologie di Testing

## OSSTMM – Risk Assessment Value (RAV) Score

### ➤ RAV Calculator

- Un foglio di calcolo per semplificare la creazione di **RAV Score**
  - Metriche standard per misurare la **Superficie di Attacco** di un asset
- Necessario per completare il **Security Test Audit Report (STAR)**

Attack Surface Security Metrics					
OSSTMM version 3.0					
Fill in the white number fields for OPSEC, Controls, and Limitations with the results of the security test. Refer to OSSTMM 3 (www.osstmm.org) for more information.					
OPSEC					
Visibility	0				
Access	0				
Trust	0				
Total (Porosity)	0				
					
				OPSEC 0.000000	



[https://www.isecom.org/rav\\_calc\\_OSSTMM3.xls](https://www.isecom.org/rav_calc_OSSTMM3.xls)

# Metodologie di Testing

## OSSTMM – Security Test Audit Report (STAR) Sheet

---

### ➤ **Security Test Audit Report (STAR) Sheet**

- Riepilogo, in formato standard, dei risultati prodotti da un vulnerability assessment o da un penetration testing OSSTMM
- Fornisce in maniera strutturata
  - Indicazioni precise sulla **Superficie di Attacco**
  - Dettagli su cosa è stato testato e come
- Il documento STAR è necessario quando la sicurezza di un'organizzazione è certificata secondo la ISECOM OSSTMM



# Metodologie di Testing

## OSSTMM – Security Test Audit Report (STAR) Sheet – Esempio



### Security Test Audit Report

OSSTMM 3.0 Security Verification Certification

OSSTMM.ORG - ISECOM.ORG

### STAR Sheet

Report ID

Date

Lead Auditor

Test Date Duration

Scope and Index

Vectors

Channels

Test Type

I am responsible for the information within this report and have personally verified that all information herein is factual and true.

**SIGNATURE**

**COMPANY STAMP/SEAL**

OPST Certification #

OPSA Certification #



# Metodologie di Testing

## OSSTMM – Principali Vantaggi

---

- Si adatta a molti tipi di test di sicurezza
  - Penetration Testing, Vulnerability Assessment, Security Audit
- Riduce il verificarsi di falsi positivi e falsi negativi
- Fornisce metriche di sicurezza riproducibili
- Garantisce che
  - La valutazione di sicurezza sia condotta in maniera accurata
  - I risultati siano raccolti in modo coerente, quantificabile ed affidabile





# Metodologie di Testing

## OSSTMM – Principali Vantaggi

---

- «Aggiornata» in base alle nuove tendenze dei test di sicurezza, alle regolamentazioni ed alle questioni etiche
- Si adatta facilmente alle *best practice* del settore, alle politiche aziendali ed alle norme
- Una verifica di sicurezza certificata in base alla metodologia OSSTMM può essere accreditata direttamente dall'*ISECOM (Institute for Security and Open Methodologies)*



# Metodologie di Testing

## Open Source Security Testing Methodology Manual (OSSTMM)

- Fornisce varie tipologie di certificazione
- <https://www.isecom.org/certification.html>



### **OSSTMM Professional Security Analyst**

The OPSA is a technical, skills-based certification designed to accredit professional security analysts.



### **OSSTMM Professional Security Tester**

The OPST is a technical, skills-based certification designed to accredit professional penetration testers.



### **OSSTMM Professional Security Expert**

The OPSE is an introductory, knowledge-based certification designed to accredit security professionals working with the OSSTMM.



# Metodologie di Testing

## Open Source Security Testing Methodology Manual (OSSTMM)

- Fornisce varie tipologie di certificazione
- <https://www.isecom.org/certification.html>



### OSSTMM Wireless Security Expert

The OWSE is a technical, knowledge-based certification designed to accredit professional penetration testers.



### OSSTMM Certified Trust Analyst

The CTA is a knowledge-based certification designed to accredit professionals measuring trust or making trust-based decisions either in a business or security capacity.



### Certified Security Awareness Instructor

The SAI is a knowledge-based certification designed to accredit professionals teaching cybersecurity awareness.

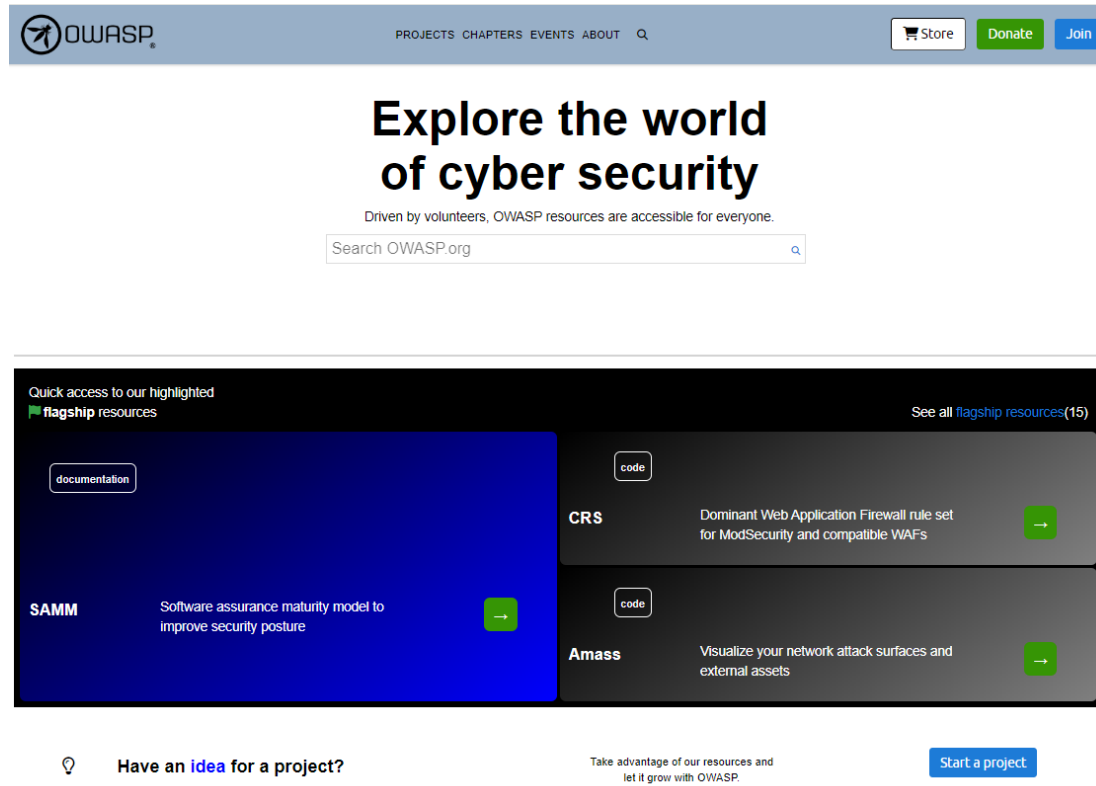


# Metodologie di Testing

## Open Web Application Security Project (OWASP)

➤ **Open Web Application Security Project (OWASP)**

➤ <https://owasp.org/>



# Metodologie di Testing

## OWASP – Caratteristiche

---

### ➤ Fornisce

- Linee guida a sviluppatori e pentester per gestire la sicurezza delle applicazioni Web e mobile, oltre che delle relative API, mediante
  - **OWASP Web Security Testing Guide (WSTG)**
  - **OWASP Mobile Application Security (MAS)**
  - **OWASP Software Assurance Maturity Model (SAMM)**
  - **OWASP Top 10 Project**
- Numerosi strumenti (tipicamente Open Source) per valutare la sicurezza delle Web e Mobile Application
- [https://owasp.org/www-community/Free\\_for\\_Open\\_Source\\_Application\\_Security\\_Tools](https://owasp.org/www-community/Free_for_Open_Source_Application_Security_Tools)

# Metodologie di Testing

OWASP – Web Security Testing Guide (WSTG)

---



OWASP Web Security Testing Guide 4.2



# Metodologie di Testing

## OWASP – Web Security Testing Guide (WSTG)

---

- Fornisce linee guida per
  - Integrare la sicurezza nelle Web Application e nei Web Service attraverso principi e pratiche di programmazione sicura
  - Effettuare il penetration testing di Web Application e Web Service
  
- È costituita da due sezioni principali
  - **Web Security Testing Framework**
  - **Web Application Security Testing**

# Metodologie di Testing

## OWASP – WSTG – Web Security Testing Framework

---

- Definisce generiche tecniche ed attività per il controllo della sicurezza nelle varie fasi del ciclo di vita dello sviluppo del software
  - Può essere utilizzato per sviluppare testing framework ad hoc
- Utilizzato per valutare la sicurezza di un software durante le sue fasi di analisi dei requisiti, progettazione, sviluppo, distribuzione, configurazione e manutenzione
  - Evitando così di attendere fino al completamento della creazione del software
- Non definisce una particolare metodologia di sviluppo e non fornisce indicazioni specifiche appartenenti ad una determinata metodologia
  - Modello di sviluppo generico che può essere seguito e adattato in base alle proprie esigenze



# Metodologie di Testing

## OWASP – WSTG – Web Security Testing Framework

---

- Definisce una serie di attività che dovrebbero aver luogo
  - Prima che inizi lo sviluppo del software
  - In fase di definizione e progettazione del software
  - Durante lo sviluppo del software
  - Durante la distribuzione del software
  - Durante la configurazione, il funzionamento e la manutenzione del software

# Metodologie di Testing

## OWASP – WSTG – Web Application Security Testing

---

- Si concentra sulla valutazione di sicurezza di una Web application
- Consente di effettuare *analisi* di sicurezza *passive o attive* dell'applicazione per rilevare eventuali punti deboli, difetti tecnici o vulnerabilità
- Eventuali problemi di sicurezza riscontrati verranno presentati al committente, insieme a
  - Una valutazione dell'impatto
  - Una proposta di mitigazione o una soluzione tecnica

# Metodologie di Testing

## OWASP – WSTG – Web Application Security Testing

---

- Raccoglie e descrive tutte le possibili tecniche di analisi della sicurezza per le applicazioni Web, mantenendosi costantemente aggiornato
- Si basa su un approccio «black box»
  - Il pentester non sa nulla (o ha pochissime informazioni) sull'applicazione da testare
- Tale framework è costituito da tre elementi principali
  - *Tester*: chi esegue le attività di testing
  - *Strumenti e Metodologie*: la parte più importante del Web Application Security Testing, che stabilisce in che modo deve essere condotta l'analisi
  - *Applicazione*: la «black box» da valutare

# Metodologie di Testing

## OWASP – WSTG – Web Application Security Testing

---

➤ L'attività di **testing** può essere di tipo **attivo** o **passivo**

➤ **Testing Passivo**

- Il pentester cerca di comprendere la logica dell'applicazione, esplorandola così come farebbe un normale utente
- Possono essere utilizzati strumenti per la raccolta di informazioni

➤ **Testing Attivo**

- Il pentester effettua un insieme di test, raggruppati in 12 categorie

- |   |                                   |
|---|-----------------------------------|
| 1. <i>Information Gathering</i>                           | 9. <i>Cryptography</i>            |
| 2. <i>Configuration and Deployment Management Testing</i> | 10. <i>Business Logic Testing</i> |
| 3. <i>Identity Management Testing</i>                     | 11. <i>Client-side Testing</i>    |
| 4. <i>Authentication Testing</i>                          | 12. <i>API Testing</i>            |
| 5. <i>Authorization Testing</i>                           |                                   |
| 6. <i>Session Management Testing</i>                      |                                   |
| 7. <i>Input Validation Testing</i>                        |                                   |
| 8. <i>Error Handling</i>                                  |                                   |

# Metodologie di Testing

## OWASP – Mobile Application Security (MAS)

---

- Fornisce uno standard di sicurezza per le App mobile (**MASVS**) ed una guida completa su come valutarle rispetto a tale standard (**MASTG**)
  - **OWASP Mobile Application Security Verification Standard (MASVS)**
  - **OWASP Mobile Application Security Testing Guide (MASTG)**
  - **OWASP Mobile Application Security Checklist (MAS Checklist)**
- **OWASP MASVS e OWASP MASTG** definiscono
  - I processi, le tecniche e gli strumenti da utilizzare durante la valutazione di sicurezza di un'App mobile
  - Una serie di casi di test che consentono ai pentester di fornire risultati coerenti e completi da una valutazione di sicurezza

➤ <https://mas.owasp.org/>

# Metodologie di Testing

## OWASP – MAS – Verification Standard

---

- **OWASP Mobile Application Security Verification Standard (MASVS)** è lo standard di settore per la sicurezza delle App mobile
- Può essere utilizzato da
  - Progettisti e sviluppatori di software mobile per sviluppare applicazioni sicure
  - Pentester per garantire la completezza e la coerenza dei risultati dei test di sicurezza

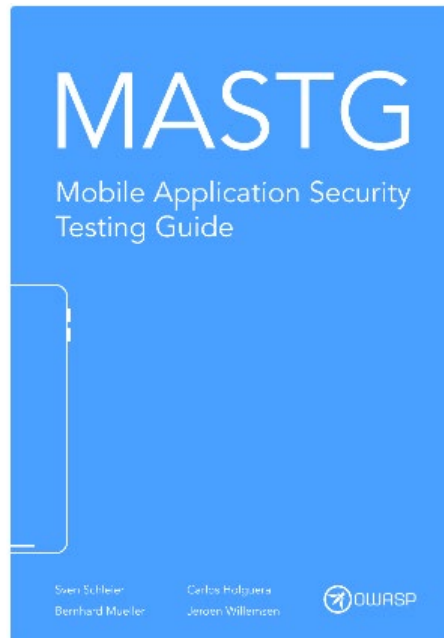


# Metodologie di Testing

## OWASP – MAS – Testing Guide

---

- La **OWASP Mobile Application Security Testing Guide (MASTG)** è un manuale completo per i test di sicurezza delle App mobile ed il reverse engineering
- Descrive i processi tecnici (**casi di test**) per la verifica dei controlli elencati nell'OWASP MASVS



# Metodologie di Testing

## OWASP – MAS – Checklist

- La **OWASP Mobile Application Security Checklist** permette di verificare i casi di test definiti nella **OWASP MASTG** per ciascuno dei controlli richiesti dal **OWASP MASVS**

Mobile Application Security Checklist  
MASVS-STORAGE: Storage  
OWASP MASTG x.x.x (commit: xxxxxx) OWASP MASVS y.y.y (commit: yyyyyy)

MASVS-ID	Platform	Description	L1	L2	R	Status
<a href="#">MASVS-STORAGE-1</a>		The app securely stores sensitive data.				
	android	<a href="#">Testing the Device-Access-Security Policy</a>				Fail
	android	<a href="#">Testing Local Storage for Sensitive Data</a>				Pass
	ios	<a href="#">Testing Local Data Storage</a>				N/A
<a href="#">MASVS-STORAGE-2</a>		The app prevents leakage of sensitive data.				
	android	<a href="#">Testing Logs for Sensitive Data</a>				Fail
	android	<a href="#">Determining Whether the Keyboard Cache Is Disabled for Text Input Fields</a>				
	android	<a href="#">Testing Backups for Sensitive Data</a>				

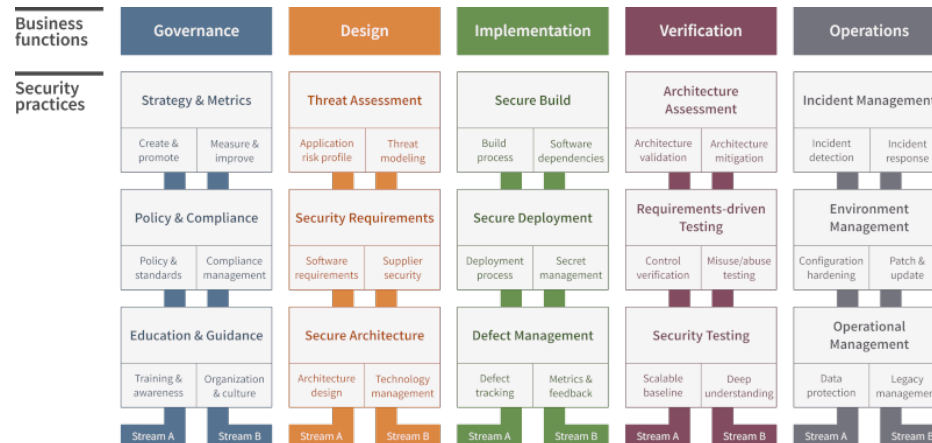
+ ≡ MASVS-STORAGE MASVS-CRYPTO MASVS-AUTH MASVS-NETWORK MASVS-PLATFORM MASVS-CODE MASVS-RESILIENCE



# Metodologie di Testing

## OWASP – Software Assurance Maturity Model (SAMM)

- Fornisce un metodo oggettivo e misurabile per analizzare e migliorare il ciclo di vita dello sviluppo sicuro del software
- Supporta l'intero ciclo di vita del software ed è indipendente dalla tecnologia e dai processi
- Si evolve in base alle diverse esigenze e segue un approccio basato sul rischio



➤ <https://owasp.org/www-project-samm/>

# Metodologie di Testing

## OWASP – Top 10 Project

---

- Mostra i 10 principali rischi per la sicurezza delle Web App
  
- Per ciascun rischio mostra
  - I principali scenari di attacco
  - Generici metodi di attacco, indipendenti dalla tecnologia utilizzata
  - Il suo impatto tecnico ed aziendale
  - Come tale rischio potrebbe essere prevenuto
  - Riferimenti a fonti esterne utili per meglio comprendere il rischio
  
- Si concentra sulle macro-aree dei problemi di sicurezza delle Web App
  - Piuttosto che affrontarne i dettagli

<https://owasp.org/www-project-top-ten/>



# Metodologie di Testing

OWASP – Top 10 Project

---

**2021**

**A01:2021-Broken Access Control**

**A02:2021-Cryptographic Failures**

**A03:2021-Injection**

**A04:2021-Insecure Design**

**A05:2021-Security Misconfiguration**

**A06:2021-Vulnerable and Outdated Components**

**A07:2021-Identification and Authentication Failures**

**A08:2021-Software and Data Integrity Failures**

**A09:2021-Security Logging and Monitoring Failures\***

**A10:2021-Server-Side Request Forgery (SSRF)\***

\* From the Survey

# Metodologie di Testing

OWASP – Top 10 Project

---

2021

- **A01:2021-Broken Access Control**
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures\*
- A10:2021-Server-Side Request Forgery (SSRF)\*

\* From the Survey

# Metodologie di Testing

## OWASP – Top 10 Project

A01:2021 – Broken Access Control



### Factors

CWEs Mapped	Max Incidence Rate	Avg Incidence Rate	Avg Weighted Exploit	Avg Weighted Impact	Max Coverage	Avg Coverage
34	55.97%	3.81%	6.92	5.93	94.55%	47.72%



### Overview

Moving up from the fifth position, 94% of applications were tested for some form of broken access control with the average incidence rate of 3.81%, and has the most occurrences in the contributed dataset with over 318k. Notable Common Weakness Enumerations (CWEs) included are *CWE-200: Exposure of Sensitive Information to an Unauthorized Actor*, *CWE-201: Exposure of Sensitive Information Through Sent Data*, and *CWE-352: Cross-Site Request Forgery*.

# Metodologie di Testing

## OWASP – Top 10 Project

---

### Description

Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification, or destruction of all data or performing a business function outside the user's limits. Common access control vulnerabilities include:

- Violation of the principle of least privilege or deny by default, where access should only be granted for particular capabilities, roles, or users, but is available to anyone.
- Bypassing access control checks by modifying the URL (parameter tampering or force browsing), internal application state, or the HTML page, or by using an attack tool modifying API requests.
- Permitting viewing or editing someone else's account, by providing its unique identifier (insecure direct object references)
- Accessing API with missing access controls for POST, PUT and DELETE.
- Elevation of privilege. Acting as a user without being logged in or acting as an admin when logged in as a user.
- Metadata manipulation, such as replaying or tampering with a JSON Web Token (JWT) access control token, or a cookie or hidden field manipulated to elevate privileges or abusing JWT invalidation.
- CORS misconfiguration allows API access from unauthorized/untrusted origins.
- Force browsing to authenticated pages as an unauthenticated user or to privileged pages as a standard user.

# Metodologie di Testing

## OWASP – Top 10 Project

---

### How to Prevent

Access control is only effective in trusted server-side code or server-less API, where the attacker cannot modify the access control check or metadata.

- Except for public resources, deny by default.
- Implement access control mechanisms once and re-use them throughout the application, including minimizing Cross-Origin Resource Sharing (CORS) usage.
- Model access controls should enforce record ownership rather than accepting that the user can create, read, update, or delete any record.
- Unique application business limit requirements should be enforced by domain models.
- Disable web server directory listing and ensure file metadata (e.g., .git) and backup files are not present within web roots.
- Log access control failures, alert admins when appropriate (e.g., repeated failures).
- Rate limit API and controller access to minimize the harm from automated attack tooling.
- Stateful session identifiers should be invalidated on the server after logout. Stateless JWT tokens should rather be short-lived so that the window of opportunity for an attacker is minimized. For longer lived JWTs it's highly recommended to follow the OAuth standards to revoke access.

Developers and QA staff should include functional access control unit and integration tests.

# Metodologie di Testing

## OWASP – Top 10 Project

---

### Example Attack Scenarios

**Scenario #1:** The application uses unverified data in a SQL call that is accessing account information:

```
pstmt.setString(1, request.getParameter("acct"));  
ResultSet results = pstmt.executeQuery( );
```

An attacker simply modifies the browser's 'acct' parameter to send whatever account number they want. If not correctly verified, the attacker can access any user's account.

```
https://example.com/app/accountInfo?acct=notmyacct
```

**Scenario #2:** An attacker simply forces browses to target URLs. Admin rights are required for access to the admin page.

```
https://example.com/app/getappInfo  
https://example.com/app/admin_getappInfo
```

If an unauthenticated user can access either page, it's a flaw. If a non-admin can access the admin page, this is a flaw.



# Metodologie di Testing

## OWASP – Top 10 Project

---

### References

- [OWASP Proactive Controls: Enforce Access Controls](#)
- [OWASP Application Security Verification Standard: V4 Access Control](#)
- [OWASP Testing Guide: Authorization Testing](#)
- [OWASP Cheat Sheet: Access Control](#)
- [OWASP Cheat Sheet: Authorization](#)
- [PortSwigger: Exploiting CORS misconfiguration](#)
- [OAuth: Revoking Access](#)

### List of Mapped CWEs

[CWE-22 Improper Limitation of a Pathname to a Restricted Directory \('Path Traversal'\)](#)

[CWE-23 Relative Path Traversal](#)

[CWE-35 Path Traversal: '..'/'.../'](#)

# Metodologie di Testing

## OWASP – Principali Vantaggi

---

- Valutare le Web App rispetto ai 10 principali rischi di sicurezza garantisce che vengano
  - Evitati o mitigati gli attacchi derivanti dalle vulnerabilità più comuni
  - Manteneute la confidenzialità, l'integrità e la disponibilità (*triade CIA*) della Web App
  - Maggiori dettagli in seguito...



# Metodologie di Testing

## OWASP – Principali Vantaggi

---

- Incoraggia pratiche di programmazione sicura, integrando la valutazione della sicurezza in ogni fase dello sviluppo di una Web App
  - Garantisce che l'applicazione messa in produzione sia (presumibilmente) robusta, priva di errori e sicura
- È ampiamente accettato a livello globale
  - I primi 10 rischi sono di solito allineati con altri standard di valutazione della sicurezza delle Web App
  - Permette di ottenere contemporaneamente la conformità rispetto a più di uno standard

# Metodologie di Testing

## NIST Special Publication (SP) 800-115

---

- Fornisce linee guida di sicurezza ed è rivolto ad organizzazioni di tutte le dimensioni e settori
  - Ogni organizzazione può utilizzare volontariamente le linee guida definite dal NIST SP 800-115 per migliorare la sicurezza del proprio asset
  - È obbligatorio che tutte le agenzie federali rispettino le linee guida del NIST
  - Anche gli appaltatori (e subappaltatori) che lavorano per le agenzie federali devono rispettare tali linee guida, altrimenti rischiano di perdere il contratto
  
- La NIST Special Publication (SP) 800-115 fornisce in particolare
  - Linee guida tecniche per condurre *penetration testing* e *vulnerability assessment*
  - Supporto nella pianificazione e nell'esecuzione dei test di sicurezza
  
- <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf>

# Metodologie di Testing

## NIST Special Publication (SP) 800-115

---

- Alcuni punti salienti del NIST SP 800-115 includono
  - **Pianificazione:** Fornisce indicazioni sulle attività di pianificazione, come la definizione degli obiettivi, l'individuazione delle regole di ingaggio, l'identificazione dei ruoli e delle responsabilità del team di pentesting e lo sviluppo di piani di testing
  - **Scoperta:** Definisce tecniche per la raccolta di informazioni (*information gathering*), identificazione di porte/servizi, rilevamento di vulnerabilità, sniffing di rete, sfruttamento delle credenziali predefinite, etc
  - **Attacco:** Definisce metodi per ottenere l'accesso, aumentare i privilegi, sfruttare le vulnerabilità, condurre attacchi *DoS* e spostarsi attraverso la rete (*pivoting*)
  - **Reporting:** Delinea gli elementi chiave che dovrebbero essere inclusi in un *penetration testing report*, come vulnerabilità, impatto ed azioni correttive
  - **Valutazione delle Competenze:** Fornisce elementi per valutare le capacità del team di pentesting in aree quali reti, sistemi operativi, Web app, tecnologie wireless, etc
  - **Considerazioni Legali:** Fornisce elementi di discussione su questioni legali e restrizioni che potrebbero applicarsi agli incarichi di penetration testing

# Metodologie di Testing

## NIST Special Publication (SP) 800-115

---

- NIST SP 800-115 è suddiviso in varie sezioni che coprono diversi aspetti dei test di sicurezza
  - *Security Testing and Examination Overview*
  - *Review Techniques*
  - *Target Identification and Analysis Techniques*
  - *Target Vulnerability Validation Techniques*
  - *Security Assessment Planning*
  - *Security Assessment Execution*
  - *Post-Testing Activities*

# Metodologie di Testing

## SP 800-115 – Security Testing and Examination Overview

---

- Una valutazione di sicurezza dovrebbe comprendere almeno le seguenti fasi
  - *Planning*
  - *Execution*
  - *Post-Execution*
  
- Vengono definite 3 tipologie di valutazione per un asset
  - *Testing*: Confrontare il comportamento reale con il comportamento atteso
  - *Examination*: Controllare, ispezionare, revisionare, osservare, studiare o analizzare un oggetto (*asset*) per migliorarne la comprensione
  - *Interviewing*: Discutere con il personale dell'organizzazione (in gruppi o individualmente) per ottenere chiarimenti

# Metodologie di Testing

## SP 800-115 – Review Techniques

---

- Vengono affrontati diversi aspetti e tecniche di revisione
  - Revisione della documentazione
  - Revisione dei log
  - Revisione delle regole
  - Revisione delle configurazioni
  
- Vengono anche fornite indicazioni su come effettuare
  - *Sniffing della rete* per identificare ed analizzare gli asset
  - *Controlli di integrità* per verificare se eventuali file di sistema o comunque file «critici» siano stati compromessi



# Metodologie di Testing

## SP 800-115 – Target Identification and Analysis Techniques

---

- Vengono fornite indicazioni su come identificare porte, servizi e sistemi nella rete
  - Il passo successivo consiste nell'identificare eventuali loro vulnerabilità
  
- Le tecniche trattate in questa sezione sono
  - *Network Discovery*
  - *Network Port e Service Identification*
  - *Vulnerability Scanning*
  - *Wireless Scanning (passive ed active scanning, wireless device location tracking, bluetooth scanning)*

# Metodologie di Testing

## SP 800-115 – Target Vulnerability Validation Techniques

---

- Vengono fornite indicazioni su come
  - Confermare l'esistenza di una vulnerabilità
  - Comprenderne l'impatto (rischio) se la vulnerabilità viene sfruttata
  
- Tale sezione copre sia debolezze tecniche che quelle dovute alla mancanza di consapevolezza e formazione (i.e., «*vulnerabilità umane*»)
  - *Password Cracking*
  - *Target Exploitation*
  - *Social Engineering*

# Metodologie di Testing

## SP 800-115 – Security Assessment Planning

---

- Definisce come pianificare il processo di valutazione della sicurezza
  
- Fornendo indicazioni su come
  - Sviluppare una politica di valutazione della sicurezza
  - Dare priorità e pianificare le valutazioni
  - Gestire lo sviluppo del piano di valutazione
  - Selezionare e personalizzare le tecniche di valutazione della sicurezza
  - Gestire la logistica della valutazione (selezione dei valutatori e delle loro competenze, dell'ubicazione, degli strumenti e delle risorse)
  - Affrontare le considerazioni legali

# Metodologie di Testing

## SP 800-115 – Security Assessment Execution

---

- L'esecuzione è ciò che segue la pianificazione
  
- È importante che i valutatori si attengano al piano di valutazione
  - Se è necessario «deviare» da tale piano, la situazione dovrebbe essere riesaminata per prendere una decisione
  
- Questa sezione copre aspetti quali
  - Coordinazione delle risorse e delle attività coinvolte nel processo di testing
  - Valutazione ed analisi dei risultati ottenuti dal processo di testing
  - Trattamento dei dati relativi a tale processo (raccolta, archiviazione, trasmissione e distruzione)

# Metodologie di Testing

## SP 800-115 – Post-Testing Activities

---

- Riguarda ciò che accade dopo il processo di valutazione della sicurezza
- I dati raccolti vengono «convertiti» in azioni da intraprendere
- Le attività di post-testing mirano a raccogliere i risultati della sezione precedente ed a creare un piano per mitigare le vulnerabilità rilevate
- Il NIST fornisce linee guida per le seguenti attività di post-testing
  - *Recommendation/Remediation* su come risolvere e/o mitigare le problematiche di sicurezza rilevate
  - *Reporting*