root@hacklab:~/parte-02/lezione-17

ETHICAL HACKING POTENZIATO DALL'IA // PARTE II

LEZIONE 17

Mobile Application Security - Android & iOS

Sicurezza delle applicazioni mobili, dall'analisi statica all'instrumentation dinamica.

docente **Vincenzo Calabro** fase Specializzazione: Mobile Application Security

root@hacklab:~/lezione-17/agenda $

LEZIONE 17 / AGENDA

In questa lezione

// PERCORSO

> Modelli Android/iOS
> OWASP MASVS/MASTG
> Analisi statica di APK/IPA
> Instrumentation dinamica (Frida)
> Storage, comunicazioni e pinning

metodo obiettivi -> teoria -> strumenti -> laboratorio -> caso di studio -> etica -> verifica

LEZIONE 17 / OBIETTIVI

Obiettivi di apprendimento

// AL TERMINE SAPRAI (TASSONOMIA DI BLOOM)

> Comprendere — l'architettura di sicurezza Android/iOS
> Applicare — analisi statica e dinamica di app
> Analizzare — storage, comunicazioni e superfici IPC
> Valutare — la conformita secondo OWASP MASVS
> Creare — un report di mobile penetration test

LEZIONE 17 / INQUADRAMENTO

Inquadramento

// DI COSA PARLIAMO

Le app mobili custodiscono dati sensibili su piattaforme con modelli di sicurezza propri. Vediamo analisi statica e dinamica, storage, comunicazioni e il metodo OWASP MASVS.

// PERCHE CONTA

Sicurezza delle applicazioni mobili, dall'analisi statica all'instrumentation dinamica.

LEZIONE 17 / MAPPA // INSIEME

Mappa d'insieme del modulo

Attacco/tecnica · Difesa · Norme & metodo · IA

FONDAMENTI

- 01 Modello di sicurezza mobile
- 02 OWASP MASVS e MASTG
- 03 Analisi statica di APK e IPA
- 04 Analisi dinamica e instrumentation
- 05 Storage, deep link, WebView e pinning

APPROFONDIMENTI

- 06 Isolamento e custodia dei segreti
- 07 Segreti e componenti esposti
- 08 Deep link, IPC e WebView
- 09 Certificate pinning e bypass
- 10 Reverse engineering e offuscamento

LEZIONE 17 / MAPPA // FONDAMENTI

Mappa dei concetti — Fondamenti

Attacco/tecnica · Difesa · Norme & metodo · IA

LEZIONE 17 / CONCETTO 1/10 [Difesa]

Modello di sicurezza mobile

// SPIEGAZIONE

Android e iOS isolano le app in sandbox, regolano l'accesso con permessi, garantiscono l'integrita con la firma e custodiscono i segreti in keystore e keychain. Chi non padroneggia questo modello non ne sa valutare le debolezze.

// ESEMPIO

> Un'app che salva un token fuori dal keystore lo espone ad altre app o al backup.

LEZIONE 17 / CONCETTO 2/10 [Difesa]

OWASP MASVS e MASTG

// SPIEGAZIONE

MASVS definisce i requisiti di sicurezza di un'app mobile per livelli; MASTG e la guida di test corrispondente. Insieme danno un metodo ripetibile per verificare storage, comunicazioni, crittografia e resistenza al reverse engineering.

// ESEMPIO

```
> Si verifica ogni requisito MASVS con i test del MASTG, documentando l'esito.
```

LEZIONE 17 / CONCETTO 3/10

[Attacco/tecnica]

Analisi statica di APK e IPA

// SPIEGAZIONE

Decompilare un pacchetto APK o IPA rivela manifest, permessi richiesti, endpoint e spesso segreti hardcoded come chiavi API. L'analisi statica trova questi errori senza nemmeno eseguire l'app.

// ESEMPIO

> Una chiave API in chiaro nel codice decompilato dell'APK.

LEZIONE 17 / CONCETTO 4/10

[Attacco/tecnica]

Analisi dinamica e instrumentation

// SPIEGAZIONE

A runtime si osserva e si modifica il comportamento dell'app: Frida inietta script per intercettare funzioni, Objection automatizza compiti comuni come il bypass di controlli. Si vede cosa l'app fa davvero, non solo cosa dichiara.

// ESEMPIO

```
$ frida -U -n app -l hook.js (intercetta la funzione di verifica a runtime)
```

LEZIONE 17 / CONCETTO 5/10

[Difesa]

Storage, deep link, WebView e pinning

// SPIEGAZIONE

Dati sensibili salvati in chiaro, deep link non validati, WebView che eseguono contenuti insicuri e certificate pinning aggirabile sono difetti ricorrenti. Il pinning protegge le comunicazioni, ma su device controllato puo essere bypassato in test.

// ESEMPIO

> Un deep link non validato apre una schermata interna con parametri controllati dall'attaccante.

LEZIONE 17 / MAPPA // APPROFONDIMENTI

Mappa dei concetti — Approfondimenti

Attacco/tecnica · Difesa · Norme & metodo · IA

Isolamento e custodia dei segreti

// SPIEGAZIONE

La sandbox impedisce a un'app di leggere i dati di un'altra; permessi e firma governano capacita e integrita; keystore e keychain proteggono le chiavi con supporto hardware. Aggirare o usare male questi meccanismi apre le porte.

// ESEMPIO

> Una chiave in keystore con protezione hardware non e estraibile neppure con root.

LEZIONE 17 / CONCETTO 7/10

[Attacco/tecnica]

Segreti e componenti esposti

// SPIEGAZIONE

Chiavi, URL e credenziali incorporate nell'app sono recuperabili da chiunque la decompili. Allo stesso modo, componenti Android esportati senza controlli permettono ad altre app di invocarli: due errori molto comuni.

// ESEMPIO

> Un'Activity esportata senza permessi viene lanciata da un'app malevola.

LEZIONE 17 / CONCETTO 8/10

[Attacco/tecnica]

Deep link, IPC e WebView

// SPIEGAZIONE

Deep link e meccanismi di comunicazione tra componenti sono porte d'ingresso se non validano l'input; le WebView che abilitano JavaScript o bridge nativi possono esporre funzioni sensibili a contenuti web non fidati.

// ESEMPIO

> Una WebView con bridge JavaScript espone metodi nativi a una pagina controllata.

LEZIONE 17 / CONCETTO 9/10

[Attacco/tecnica]

Certificate pinning e bypass

// SPIEGAZIONE

Il pinning lega l'app a un certificato atteso, ostacolando l'intercettazione del traffico. In test, su device controllato, si bypassa con Frida o Objection per ispezionare le API: tecnica legittima solo in ambiente autorizzato.

// ESEMPIO

> Uno script Frida disabilita il pinning per permettere l'analisi del traffico TLS.

LEZIONE 17 / CONCETTO 10/10

[Attacco/tecnica]

Reverse engineering e offuscamento

// SPIEGAZIONE

Decompilare e comprendere la logica di un'app serve a trovare debolezze; l'offuscamento rende il codice meno leggibile ma non e una misura di sicurezza in se: ritarda l'analista, non impedisce l'analisi determinata.

// ESEMPIO

> Codice offuscato rallenta il reverse, ma la logica di autenticazione resta ricostruibile.

LEZIONE 17 / STRUMENTI

Strumenti della lezione

// TOOLKIT

MobSF | Frida | Objection | jadx | apktool | Burp Suite

// NOTA OPERATIVA

> Ogni strumento si usa in ambiente virtualizzato e isolato, solo su bersagli di proprieta o autorizzati. Output e artefatti vanno protetti e, a fine ingaggio, gestiti o distrutti.

root@hacklab:~/lezione-17/lab-1 $

LEZIONE 17 / LABORATORIO 1

Laboratorio 1

$ Analisi statica di un APK (MobSF + jadx)

// PROCEDURA

[1] Analizza un APK con MobSF e jadx: segreti

[2] permessi

[3] componenti esposti

[!] Verifica: Quali informazioni espone il manifest?

root@hacklab:~/lezione-17/lab-2 $

LEZIONE 17 / LABORATORIO 2

Laboratorio 2

$ Instrumentation con Frida (bypass pinning/root)

// PROCEDURA

[1] Usa Frida per aggirare pinning e root detection e intercettare il traffico

[!] Verifica: Perche il pinning non basta da solo?

LEZIONE 17 / CASO DI STUDIO

Caso di studio

// SCENARIO

> Vulnerabilita mobile reale (insecure storage o pinning assente) e impatto sulla riservatezza.

// ANALISI RICHIESTA

> Ricostruisci la dinamica dei fatti con metodo e tracciabilita

> Individua errori, cause radice e punti di decisione critici

> Proponi contromisure, controlli e lezioni apprese

root@hacklab:~/lezione-17/etica $

LEZIONE 17 / ETICA & DIRITTO

Nota etico-legale

// PRINCIPI INDEROGABILI

> autorizzazione scritta scope definito ambiente isolato dati protetti responsible disclosure

// SPECIFICO DELLA LEZIONE

[!] Solo app di proprieta o autorizzate; attenzione a dati personali e proprieta intellettuale.

LEZIONE 17 / RICAPITOLAZIONE

Ricapitolazione

// PUNTI CHIAVE

> L'architettura di sicurezza Android/iOS
> Analisi statica e dinamica di app
> Storage, comunicazioni e superfici IPC
> La conformita secondo OWASP MASVS
> Un report di mobile penetration test

LEZIONE 17 / AUTOVALUTAZIONE

Autovalutazione

// DOMANDE DI VERIFICA

? Quali informazioni espone il manifest di un'app?

? Perche il certificate pinning non basta da solo?

? Dove un'app conserva in modo insicuro i dati?

// VALUTAZIONE

Report di mobile pentest mappato sui requisiti MASVS.

// LETTURE

OWASP MASTG/MASVS; doc. Frida.

lezione 17 completata

$ next

Mobile Application Security - Android & iOS

> **[!]** Le competenze di questa lezione si impiegano esclusivamente in contesti legali, autorizzati e deontologicamente corretti, entro lo scope e con autorizzazione scritta.

root@hacklab:~# _

root@hacklab:~/contatti $

CONTATTI / RISORSE

Restiamo in contatto

Vincenzo Calabro

cyber security / digital forensics /
docenza e divulgazione

// SCANSIONA O VISITA

> vincenzocalabro.it

Inquadra il QR con la fotocamera per aprire il sito.