

```
root@hacklab:~/parte-02/lezione-18
```

ETHICAL HACKING POTENZIATO DALL'IA // PARTE II

LEZIONE 18

Crittografia applicata e attacchi crittografici

Uso corretto della crittografia e sfruttamento delle implementazioni deboli.

docente **Vincenzo Calabro** fase Specializzazione: crittografia applicata

```
root@hacklab:~/lezione-18/agenda $
```

LEZIONE 18 / AGENDA

In questa lezione

// PERCORSO

- > Primitivi e garanzie
- > Modalità operative ed errori
- > Padding oracle e RSA
- > PKI e TLS
- > Gestione delle chiavi

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

```
root@hacklab:~/lezione-18/obiettivi $
```

LEZIONE 18 / OBIETTIVI

Obiettivi di apprendimento

// AL TERMINE SAPRAI (TASSONOMIA DI BLOOM)

- > Comprendere – i primitivi crittografici e le loro garanzie
- > Applicare – attacchi a implementazioni deboli
- > Analizzare – errori di modalita, IV e chiavi
- > Valutare – la robustezza di TLS/PKI
- > Creare – raccomandazioni di hardening crittografico

```
root@hacklab:~/lezione-18/inquadramento $
```

LEZIONE 18 / INQUADRAMENTO

Inquadramento

// DI COSA PARLIAMO

La crittografia protegge solo se usata correttamente: gli errori d'implementazione sono il vero punto debole. La lezione collega primitive, protocolli e attacchi pratici al cattivo uso.

// PERCHE CONTA

Uso corretto della crittografia e sfruttamento delle implementazioni deboli.

```
root@hacklab:~/lezione-18/mappa-insieme $
```

LEZIONE 18 / MAPPA // INSIEME

Mappa d'insieme del modulo

■ Attacco/tecnica ■ Difesa ■ Norme & metodo ■ IA

FONDAMENTI

- 01 Primitive crittografiche
- 02 Modalità operative ed errori
- 03 Padding oracle e RSA fragile
- 04 PKI e TLS nel mondo reale
- 05 Gestione chiavi e post-quantum

APPROFONDIMENTI

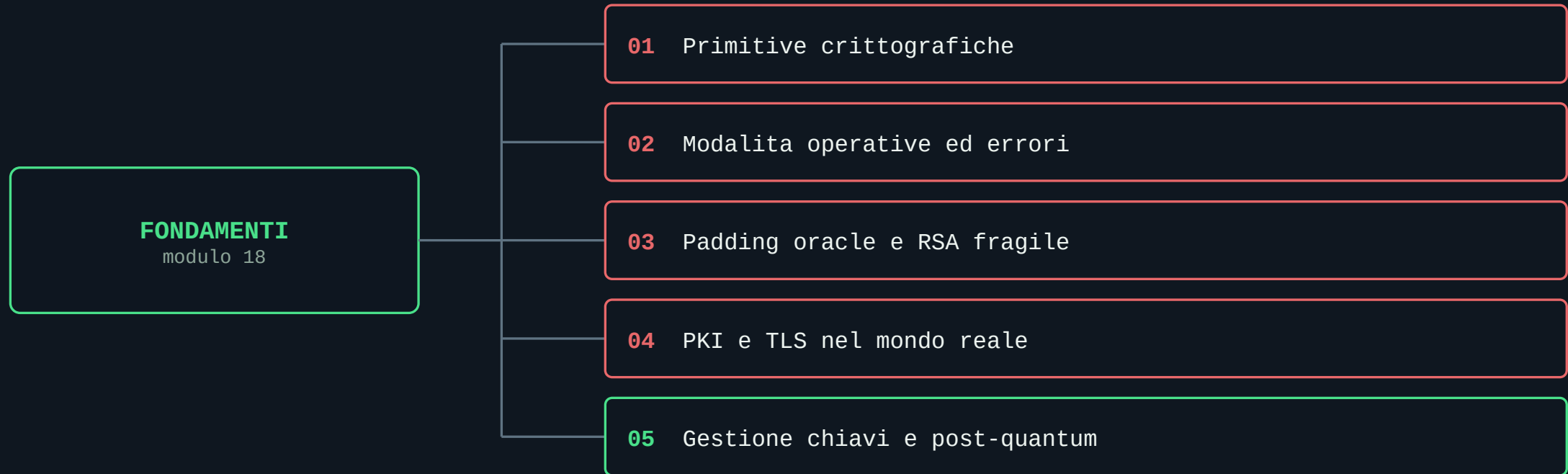
- 06 Integrità e autenticità
- 07 Errori di modalità
- 08 RSA mal implementato
- 09 Downgrade e validazione TLS
- 10 Ciclo di vita delle chiavi

```
root@hacklab:~/lezione-18/mappa-1 $
```

LEZIONE 18 / MAPPA // FONDAMENTI

Mappa dei concetti – Fondamenti

■ Attacco/tecnica ■ Difesa ■ Norme & metodo ■ IA



```
root@hacklab:~/lezione-18/concetto-1 $
```

LEZIONE 18 / CONCETTO 1/10

[Attacco/tecnica]

Primitive crittografiche

// SPIEGAZIONE

La crittografia simmetrica con una chiave condivisa e veloce; l'asimmetrica con coppia di chiavi risolve scambio e firme; gli hash garantiscono integrità, HMAC l'autenticità con chiave, le firme digitali autenticità e non ripudio. Conoscerne il ruolo evita usi sbagliati.

// ESEMPIO

> Si firma con la chiave privata e si verifica con la pubblica: autenticità e non ripudio.

```
root@hacklab:~/lezione-18/concetto-2 $
```

LEZIONE 18 / CONCETTO 2/10

[Attacco/tecnica]

Modalita operative ed errori

// SPIEGAZIONE

La modalita conta quanto l'algoritmo: ECB rivela pattern, un IV o nonce riutilizzato rompe la riservatezza. Le modalita AEAD come AES-GCM uniscono cifratura e autenticazione, evitando molti errori classici.

// ESEMPIO

> ECB su un'immagine lascia intravedere la figura originale nei blocchi ripetuti.

```
root@hacklab:~/lezione-18/concetto-3 $
```

LEZIONE 18 / CONCETTO 3/10

[Attacco/tecnica]

Padding oracle e RSA fragile

// SPIEGAZIONE

Un padding oracle sfrutta i messaggi d'errore sul padding per decifrare senza la chiave; RSA mal implementato, senza padding robusto o con parametri deboli, espone a recupero del testo o della chiave. Spesso il problema è l'implementazione, non l'algoritmo.

// ESEMPIO

> Risposte diverse su padding valido o no permettono di decifrare byte per byte.

```
root@hacklab:~/lezione-18/concetto-4 $
```

LEZIONE 18 / CONCETTO 4/10

[Attacco/tecnica]

PKI e TLS nel mondo reale

// SPIEGAZIONE

TLS è sicuro se configurato bene: attacchi di downgrade forzano protocolli deboli, la cattiva validazione dei certificati apre al MITM, i cipher obsoleti indeboliscono il canale. La PKI regge solo se la catena di fiducia è verificata.

// ESEMPIO

> Un client che non valida il certificato accetta un MITM con certificato falso.

```
root@hacklab:~/lezione-18/concetto-5 $
```

LEZIONE 18 / CONCETTO 5/10

[Difesa]

Gestione chiavi e post-quantum

// SPIEGAZIONE

La sicurezza dipende da come si custodiscono le chiavi: HSM e KMS le proteggono e ne gestiscono rotazione e accesso. All'orizzonte i computer quantistici minacciano gli algoritmi attuali, da cui l'interesse per la crittografia post-quantum.

// ESEMPIO

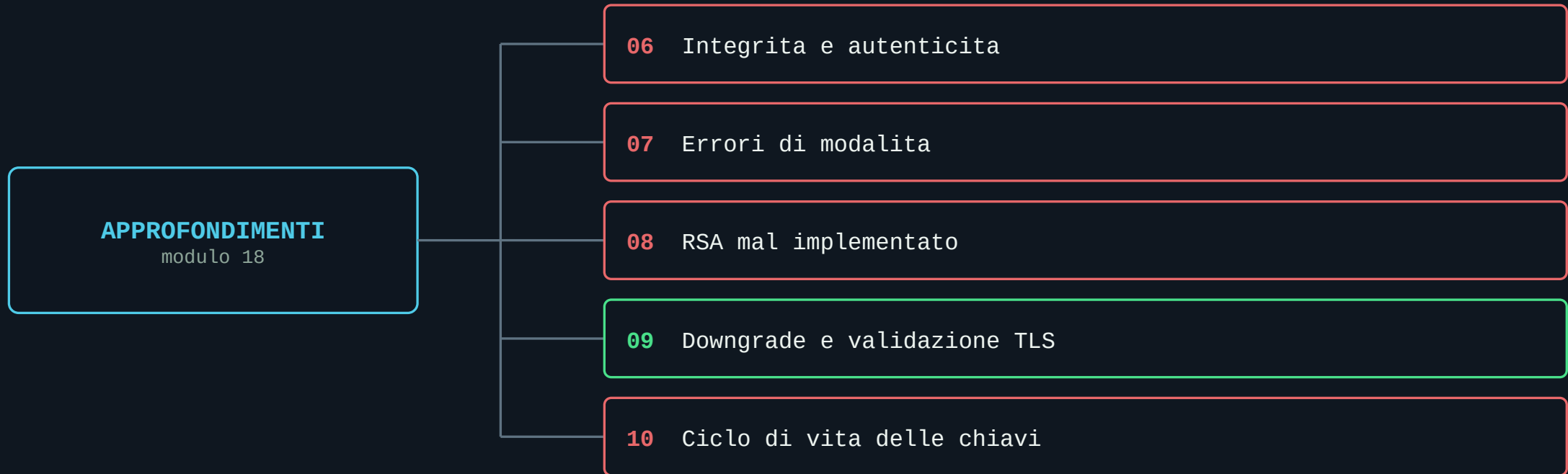
> Le chiavi generate e custodite in HSM non lasciano mai il dispositivo in chiaro.

```
root@hacklab:~/lezione-18/mappa-2 $
```

LEZIONE 18 / MAPPA // APPROFONDIMENTI

Mappa dei concetti – Approfondimenti

■ Attacco/tecnica ■ Difesa ■ Norme & metodo ■ IA



```
root@hacklab:~/lezione-18/concetto-6 $
```

LEZIONE 18 / CONCETTO 6/10

[Attacco/tecnica]

Integrità e autenticità

// SPIEGAZIONE

Un hash prova integrità ma non autenticità; HMAC aggiunge una chiave segreta per garantire chi ha prodotto il messaggio; le firme digitali danno anche non ripudio. AEAD unisce riservatezza e autenticità in un'unica operazione.

// ESEMPIO

> Usare HMAC invece del solo hash impedisce a un attaccante di ricalcolare l'impronta.

```
root@hacklab:~/lezione-18/concetto-7 $
```

LEZIONE 18 / CONCETTO 7/10

[Attacco/tecnica]

Errori di modalita

// SPIEGAZIONE

Riutilizzare un IV o un nonce, o usare ECB, fa trapelare informazioni sui dati cifrati: pattern, ripetizioni, talvolta il testo. Sono errori di implementazione frequenti che annullano la robustezza dell'algoritmo sottostante.

// ESEMPIO

> Due messaggi cifrati con lo stesso nonce in CTR permettono di ricavarne la differenza.

```
root@hacklab:~/lezione-18/concetto-8 $
```

LEZIONE 18 / CONCETTO 8/10

[Attacco/tecnica]

RSA mal implementato

// SPIEGAZIONE

RSA senza padding adeguato, con esponenti o moduli deboli o con chiavi condivise per errore, è vulnerabile a recupero del messaggio o fattorizzazione. L'algoritmo è solido; quasi sempre il difetto è nell'uso che se ne fa.

// ESEMPIO

> Moduli RSA che condividono un fattore primo si fattorizzano con il massimo comun divisore.

```
root@hacklab:~/lezione-18/concetto-9 $
```

LEZIONE 18 / CONCETTO 9/10

[Difesa]

Downgrade e validazione TLS

// SPIEGAZIONE

Attacchi di downgrade negoziano versioni e cipher deboli; una validazione lasca dei certificati annulla la protezione. Disabilitare i protocolli obsoleti e validare rigorosamente la catena sono difese essenziali e spesso trascurate.

// ESEMPIO

> Disabilitare TLS 1.0/1.1 e i cipher export blocca molti attacchi di downgrade.

```
root@hacklab:~/lezione-18/concetto-10 $
```

LEZIONE 18 / CONCETTO 10/10

[Attacco/tecnica]

Ciclo di vita delle chiavi

// SPIEGAZIONE

Le chiavi vanno generate, custodite, ruotate e revoked in modo sicuro: HSM e KMS automatizzano gran parte del ciclo. Pianificare la migrazione verso algoritmi post-quantum e già oggi una scelta prudente per i dati a lunga vita.

// ESEMPIO

> Rotazione periodica delle chiavi per limitare l'impatto di un'eventuale compromissione.

```
root@hacklab:~/lezione-18/strumenti $
```

LEZIONE 18 / STRUMENTI

Strumenti della lezione

// TOOLKIT

OpenSSL

cryptography

pycryptodome

CyberChef

testssl.sh

Hashcat

// NOTA OPERATIVA

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

```
root@hacklab:~/lezione-18/lab-1 $
```

LEZIONE 18 / LABORATORIO 1

Laboratorio 1

\$ Padding oracle (decifratura in lab)

// PROCEDURA

[1] Sfrutta un padding oracle in laboratorio per decifrare un testo cifrato

[!] Verifica: Perché l'oracle rivela il plaintext?

```
root@hacklab:~/lezione-18/lab-2 $
```

LEZIONE 18 / LABORATORIO 2

Laboratorio 2

\$ Analisi e hardening TLS (testssl.sh)

// PROCEDURA

[1] Valuta una configurazione TLS con testssl.sh e correggi le debolezze

[!] Verifica: Quali cipher suite vanno disabilitate e perche?

```
root@hacklab:~/lezione-18/caso-di-studio $
```

LEZIONE 18 / CASO DI STUDIO

Caso di studio

// SCENARIO

> Vulnerabilità crittografica storica (padding oracle o downgrade TLS) e lezioni di implementazione.

// ANALISI RICHIESTA

- > Ricostruisci la dinamica dei fatti con metodo e tracciabilità
- > Individua errori, cause radice e punti di decisione critici
- > Proponi contromisure, controlli e lezioni apprese

```
root@hacklab:~/lezione-18/etica $
```

LEZIONE 18 / ETICA & DIRITTO

Nota etico-legale

// PRINCIPI INDEROGABILI

> autorizzazione scritta scope definito ambiente isolato dati protetti responsible disclosure

// SPECIFICO DELLA LEZIONE

[!] Gli attacchi crittografici solo su sistemi/dati di laboratorio o autorizzati.

```
root@hacklab:~/lezione-18/ricapitolazione $
```

LEZIONE 18 / RICAPITOLAZIONE

Ricapitolazione

// PUNTI CHIAVE

- > I primitivi crittografici e le loro garanzie
- > Attacchi a implementazioni deboli
- > Errori di modalita, IV e chiavi
- > La robustezza di TLS/PKI
- > Raccomandazioni di hardening crittografico

```
root@hacklab:~/lezione-18/verifica $
```

LEZIONE 18 / AUTOVALUTAZIONE

Autovalutazione

// DOMANDE DI VERIFICA

- ? Perché un padding oracle rivela il plaintext?
- ? Quali cipher suite vanno disabilitate e perché?
- ? Perché riutilizzare un IV/nonce è pericoloso?

// VALUTAZIONE

Esercizio di attacco documentato + nota difensiva.

// LETTURE

Aumasson, Serious Cryptography; Cryptopals.

lezione 18 completata

\$ next

Crittografia applicata e attacchi crittografici

[!] 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

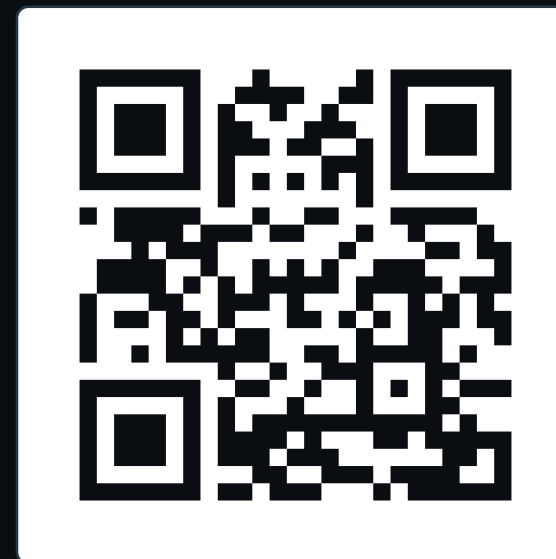


Vincenzo Calabro

cyber security / digital forensics /
docenza e divulgazione

// SCANSIONA O VISITA

> vincenzocalabro.it



Inquadra il QR con la fotocamera
per aprire il sito.