

INTERNET SECURITY

TUTORATO

DAVIDE CARNEMOLLA

DIPARTIMENTO DI MATEMATICA E INFORMATICA
UNIVERSITÀ DEGLI STUDI DI CATANIA

A.A. 2021/2022



PROPRIETÀ DI SICUREZZA



Confidenzialità
(Segretezza)



Integrità



Autenticazione

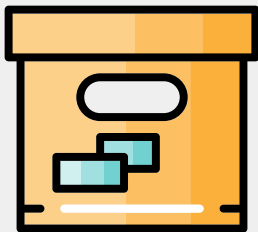


Definizione

L'informazione non sia rilasciata ad entità non autorizzate a conoscerla

Garantita da

- **Crittografia**
- **Steganografia**



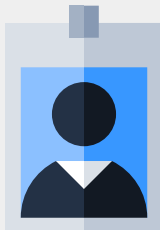
Definizione

L'informazione non sia modificata da entità non autorizzate

Garantita da

- **Firma elettronica**

PROPRIETÀ DI 1° LIVELLO: AUTENTICAZIONE



Definizione

Le entità siano esattamente chi dichiarano di essere

Garantità da

- **Conoscenza** (password, PIN)
- **Possesso** (Smart card, Smart Token)
- **Biometria** (impronte, iride)

AUTENTICAZIONE BASATA SU CONOSCENZA



Come funziona

La conoscenza di un segreto (password, PIN) comprova l'identità dell'utente

Problemi

Sensibile ad attacchi di **guessing, snooping, spoofing, sniffing**

COME VIENE SALVATA UNA PASSWORD?

- Memorizzate **in chiaro** su un file di sistema (CTS, 1960)
- Uso di funzioni hash crittografiche
- GNU/Linux

PASSWORD: GNU/LINUX

```
cat /etc/shadow | grep root
```

```
root:$1$Etg2ExUZ$F9NTP7omafhKIlqaBMqng1:15651:0:99999:7:::
```

Analisi dell'output

- **\$1\$** indica l'utilizzo di una funzione hash MD5
- **Etg2ExUZ** è il sale
- **F9NTP7omafhKIlqaBMqng1** è il segreto
- **15651** è la data in cui è stata impostata la password
- **0** sono i giorni che devono trascorrere prima di poter modificare la password
- **99999** sono i giorni dopo cui bisogna modificare la password
- **7** sono i giorni dopo l'utente viene disabilitato

AUTENTICAZIONE BASATA SU POSSESSO



Come funziona

Il possesso di un dispositivo fisico (Smart Card, Smart Token, YubiKey) comprova l'identità dell'utente

Caratteristiche

- Interamente leggibile nel caso di carte magnetiche
- Estrazione dei segreti gestita da un'interfaccia funzionale nel caso di carte elettroniche

AUTENTICAZIONE BASATA SU BIOMETRIA



Come funziona

Il possesso di caratteristiche biometriche (impronte digitali, impronta della retina, viso etc.) comprova l'identità

Caratteristiche

- Meno accurato ma più affidabile
- Utilizzato in combinazione con autenticazione basata su conoscenza



Non ripudio



Disponibilità



Definizione

L'entità non possa negare la propria partecipazione ad una transazione con uno specifico ruolo

Garantita da

- Protocolli di sicurezza appositi (ad esempio **PEC**)



Definizione

Il sistema sia operante e funzionante in ogni momento

Garantita da

- Autenticazione
- Accesso complicato impegnando il chiamante computazionalmente

CRITTOGRAFIA



Crittologia

Il termine **Crittologia** deriva dal greco *kryptòs* (nascosto) e *logos* (discorso) ed è la scienza che si occupa delle scritture nascoste. Comprende la **Crittografia** e la **Crittoanalisi**.



Crittografia

La **Crittografia** è la scienza che si occupa di costruire dei metodi per rendere un messaggio intelligibile.



Crittoanalisi

La **Crittoanalisi** è la scienza che si occupa di costruire metodi per “rompere” gli schemi crittografici

CRITTOGRAFIA SIMMETRICA



Bob

$$c = \mathcal{E}(m, k)$$



Alice



$$m = \mathcal{D}(c, k)$$



Definizione

Sia $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ uno schema di cifratura simmetrico. Diremo che \mathcal{SE} è *perfettamente sicuro* se

$$\forall M_1, M_2 \in \mathcal{M} \text{ e } \forall c \in \mathcal{C} \quad Pr[\mathcal{E}_{\mathcal{K}}(M_1) = c] = Pr[\mathcal{E}_{\mathcal{K}}(M_2) = c]$$

ONE-TIME PAD (OTP)

Specifiche

1. $\mathcal{M} = \{0,1\}^m, m \in \mathbb{N} : m > 0$
2. $\mathcal{K} \xleftarrow{\$} \{0,1\}^m$
3. $\mathcal{E}_{\mathcal{K}}(M) = \mathcal{K} \oplus M$
4. $\mathcal{D}_{\mathcal{K}}(C) = C \oplus \mathcal{K}$

Condizione per la perfetta sicurezza

OTP è perfettamente sicuro sotto l'ipotesi che la chiave venga utilizzata per cifrare un singolo messaggio.

ONE-TIME PAD: IMPLEMENTAZIONE (1)

```
import random
```

```
def generate_key(m):  
    return bytes(  
        random.randrange(0,256) for i in range(m)  
    )
```

```
def xor_bytes(key, message):  
    m = min(len(key), len(message))  
    return bytes(  
        [key[i] ^ message[i] for i in range(m)]  
    )
```

ONE-TIME PAD: IMPLEMENTAZIONE (2)

```
message = "OTP is perfect for a single message"
message = message.encode()
key = generate_key(len(message))
cipher = xor_bytes(key, message)
print(key)
print(cipher)
print(xor_bytes(key, cipher))
```

ONE-TIME PAD: ATTACCO

```
message2 = "I don't need another key"  
message2 = message2.encode()  
cipher2 = xor_bytes(key, message2)  
print(cipher2)
```

Attacco

- $c_1 \oplus c_2 = m_1 \oplus k \oplus m_2 \oplus k = m_1 \oplus m_2$
- Se m_1 è noto posso ottenere $m_2 = c_1 \oplus c_2 \oplus m_1$



Funzione Hash

Una *funzione hash* è una funzione matematica con le seguenti proprietà:

- prende in input una stringa (di bit) di qualsiasi dimensione
- restituisce in output una stringa (di bit) di dimensione fissata
- è efficiente dal punto di vista computazionale

Funzione Hash Crittografica

Una *funzione hash crittografica* è una funzione hash con le seguenti proprietà:

- **Collision Resistance**
- **Hiding**
- **Puzzle friendliness**

MESSAGE AUTHENTICATION CODE (MAC)



Bob

$m, T = \text{MAC}_k(m)$



Alice

$\text{VF}_k(m, T')$



<https://cryptohack.org>

COME POSSIAMO EFFETTUARE LO
SCAMBIO DI UNA **CHIAVE**?

CRITTOGRAFIA ASIMMETRICA



Bob

$$c = \mathcal{E}(m, pk)$$



pk



Alice

$$m = \mathcal{D}(c, sk)$$



sk

“La firma digitale è
l’inverso della cifratura”

Anonimo

“La firma digitale è
l’inverso della cifratura”

Anonimo





Bob



pk



Alice

$$m, \sigma = \text{Sign}(m, sk)$$



$$VF(pk, m, \sigma)$$



sk



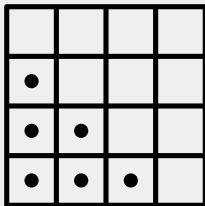
STEGANOGRAPHIA



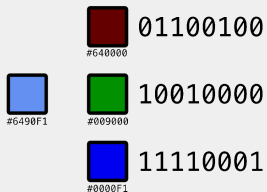
Steganografia

La steganografia è una tecnica che si prefigge di nascondere la comunicazione tra due interlocutori.

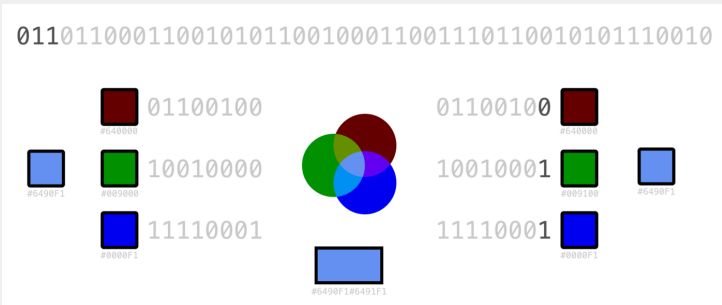
LEAST SIGNIFICANT BIT (LSB)



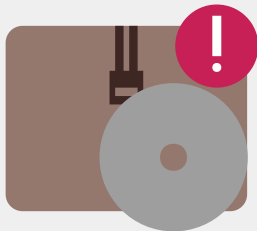
Imagine



LEAST SIGNIFICANT BIT (LSB)



CLASSIFICAZIONE SOFTWARE NOCIVI

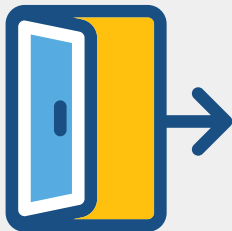


Definizione

Software scritto con l'esplicito scopo di violare alcune proprietà di sicurezza di un sistema.

Caratteristiche

- **Carico**
- **Propagazione**

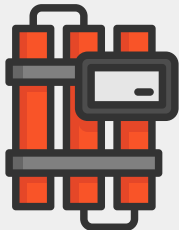


Definizione

Punto d'accesso segreto per bypassare l'autenticazione in un sistema.

Nota

Tipicamente le trapdoor vengono inserite dagli sviluppatori per testare il software.



Definizione

Porzione di codice di un software nocivo apparentemente innocua fino al verificarsi di particolari condizioni.



Definizione

Software utile che in fase di esecuzione compie violazioni di sicurezza.



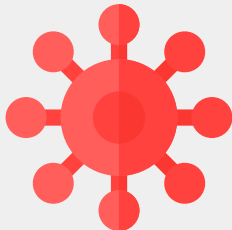
Definizione

Software che sfrutta una macchina remota già violata per lanciare nuovi attacchi.



Definizione

Software nocivo che infetta macchine remote, ciascuna delle quali a loro volta infetta altre macchine remote.



Definizione

Software nocivo che viola altri programmi non nocivi, sfruttandoli per propagarsi.



<https://app.hackthebox.com>

FIREWALL



Definizione

Un firewall è un componente software o hardware di difesa perimetrale di una rete.

Funzionalità

- Protegge le risorse interne
- Monitora il traffico
- Filtra i dati



GNU/Linux

iptables
shorewall
Firewalld
ufw



Windows

Windows Defender
GlassWire
Norton
Comodo



Mac Os

Apple Firewall
Total AV
Avira
Bitdefender

Informazioni

Iptables è un firewall per i sistemi GNU/Linux implementato a livello kernel (Netfilter).



Regole



Catene

Nota

iptables è stato sostituito da nftables in Debian a partire dalla versione 11 (Buster).

IPTABLES: CATENE DI DEFAULT



INPUT



OUTPUT



FORWARD



PREROUTING



POSTROUTING



filter

INPUT
OUTPUT
FORWARD



nat

OUTPUT
PREROUTING
POSTROUTING



mangle

Visualizzare le regole

- `iptables -t <table> -L`

Un po' di pulizia

- `iptables -F` # elimina tutte le regole
- `iptables -X` # elimina tutte le catene personalizzate
- `iptables -t nat -F` # elimina tutte le regole di nat

IPTABLES: STATI DELLE CONNESSIONI



NEW



ESTABLISHED



RELATED

Accettiamo le connessioni ESTABLISHED e RELATED

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j  
ACCEPT
```

Accettiamo le connessioni SSH

```
iptables -A INPUT -p tcp -dport 22 -m state --state NEW -j ACCEPT
```

Analisi

- **-A** aggiunge la regola in coda
- **INPUT** la catena a cui la regola fa riferimento
- **-p** indica il protocollo
- **-dport** indica la porta di destinazione
- **-m state --state NEW** indica di accettare le connessioni esterne in ingresso
- **-j** indica il target



Policy (default)

```
iptables -P INPUT DROP
```

Analisi

- **-P:** Policy
- **INPUT:** Chain
- **DROP:** Target



Una regola più precisa (address source)

```
iptables -A INPUT -p tcp -s 192.168.1.2 -dport 22 -j ACCEPT
```

Una regola più precisa (interface)

```
iptables -A INPUT -p tcp -i eth0 -dport 22 -j ACCEPT
```



GET YOUR HANDS DIRTY!

INTRUSION DETECTION SYSTEM



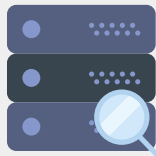
Definizione

Un Intrusion Detection System (IDS) è un dispositivo software o hardware per identificare accessi non autorizzati alla rete locale o alle macchine host.

IDS: CLASSIFICAZIONE



NIDS



HIDS



Hybrid IDS

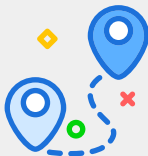
IDS: TECNICHE DI RILEVAMENTO



Signature based



Statistical anomaly-based

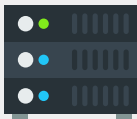


Stateful protocol analysis

IDS: DA COSA È COMPOSTO?



Sensors/Agents



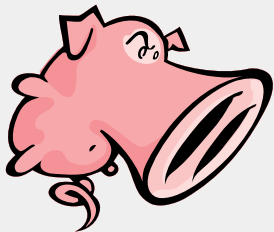
Management Server



Database Server



Console



Snort



Suricata

- Davide Carnemolla
- Herbrant (Telegram, Github, Discord, ...)
- **email:** herbrant@protonmail.org