CyberSec Talent Week Challenge

Cyber Crisis Response: Attacco Ransomware a un ospedale

bip.CyberSec

Introduzione

Un ospedale di medie dimensioni ha subito un attacco **ransomware** che ha cifrato cartelle cliniche elettroniche, sistemi di monitoraggio dei pazienti e infrastrutture di supporto critiche. Il **malware** sembra essersi diffuso rapidamente attraverso una combinazione di *exploit* di rete e *phishing* mirato al personale amministrativo. La direzione dell'ospedale ha incaricato il team **BIP** da te gestito di intervenire sull'incidente, ripristinare i dati e prevenire ulteriori danni.

Per completare la Challenge è necessario svolgere tutti i task riportati di seguito all'interno del **template standard**, rispondendo alle domande e alle richieste.

bip. CyberSec

01.

Task 1 | Analisi della diffusione del Malware

L'ospedale ha registrato un traffico di rete anomalo nelle ore precedenti l'attacco. Analizza i **log** ricevuti (allegato "**Allegato 1_Task 1_ log**") e **rispondi alle seguenti domande**:

- *Qual è stato il primo dispositivo infettato?*
- *Quale vulnerabilità è stata sfruttata per l'accesso iniziale?*
- *È possibile determinare l'IP dell'attaccante o il server C2?*
- *Se il ransomware ha cifrato file su una cartella condivisa, significa che tutti i PC della rete sono infetti? Ragiona sulle varie modalità di propagazione del ransomware e quindi sulle diverse eventualità.*

Usa il file di log per identificare i comportamenti sospetti e giustificare ogni risposta con riferimento alle evidenze trovate.

02.

Task 2 | Isolamento e contenimento

A partire dall'analisi effettuata nel **task 1**, il tuo team è ora incaricato di attuare misure rapide di contenimento. Utilizzando lo stesso file di log (allegato "**Allegato 1_Task 1_ log**") **scrivi uno script Python** per:

- identificare gli host che hanno effettuato connessioni al server di comando e che hanno attività sospette;
- isolare le macchine infette impedendo la comunicazione verso l'esterno e la propagazione interna, pur salvaguardando la **subnet** di fiducia.

Riporta sia lo **script** che l'**output** generato stampato a video.

In caso avessi difficoltà nella scrittura dello script, cerca informazioni sul web.

02.

Task 2 | Isolamento e contenimento

Input atteso dallo script riportato di seguito:

```
logfile: Allegato 1_Task 1_log.csv
trusted_subnet: 172.16.5.0/24
malicious_ip: 185.100.87.45
```

03.

Task 3 | Reverse Engineering del Ransomware

Il ransomware ha cifrato file critici utilizzando una combinazione di crittografia simmetrica e asimmetrica:

- **AES-256** per la cifratura dei dati,
- **RSA-4096** per la cifratura della chiave AES.

A causa di un errore nell'implementazione, è stato possibile sottrarre frammenti di chiavi rubate, che trovi nell'allegato "**Allegato 2_Task 3_keys_dump**". Il tuo compito è ricostruire le chiavi e simulare la decifratura.

Scrivi uno script Python per:

- **ricostruire** la chiave AES a partire dai frammenti riportati;
- **decifrare** un file critico con modalità CBC con IV 00000000000000000000000000000000.
  N.B. cerca di generare in automatico un contenuto cifrato a tua scelta all'interno dello script.

03.

Task 3 | Reverse Engineering del Ransomware

Riporta sia lo script che l'output generato.

In caso avessi difficoltà nella scrittura dello script, cerca informazioni sul web.

Output attesi

Per ricostruzione della chiave AES:

```
AES_KEY_RECOVERED: 5f2a1c3e9bf04d7a6b8e3c9d2e7a5f...
```

Per decifratura del file medico critico:

```
Decrypted_File: patient_data_recovered.txt
```

04.

Task 4 | Coordinamento e comunicazione

Immaginati ora di dover redigere, data la tua posizione da **team leader**, un report rivolto alla direzione dell’ospedale e anche alle autorità pubbliche.

Scrivi un rapporto testuale (max 20 righe) in cui:

- analizzi le perdite e le problematiche derivanti dal blocco del sistema;
- descrivi una strategia completa di ripristino.

05.

Task 5 | Implementazione sistema monitoraggio

NOTA: questo task è opzionale e dunque non obbligatorio ai fini del completamento della Challenge.

Dopo il contenimento dell'incidente, la direzione ospedaliera ti incarica di implementare un **sistema di monitoraggio** per rilevare in tempo reale eventuali **nuovi comportamenti sospetti** nella rete. Dato l'allegato "**Allegato 3_Task 5_network_logs**" contenente il log cronologico di successivi eventi di rete, **scrivi uno script Python** che generi **alert testuali** in presenza di comportamenti anomali come:

- **connessioni verso IP malevoli** già noti in precedenza;
- **query DNS** verso domini .**onion**;
- accessi ripetuti tra host interni tramite **protocollo SMB**.

05.

Task 5 | Implementazione sistema monitoraggio

Riporta sia lo script che l’output generato.

In caso avessi difficoltà nella scrittura dello script, cerca informazioni sul web.

Esempio generico di **output atteso**:

```
[ALERT] C2 Connection detected: 192.168.1.100 → 185.100.87.45
[ALERT] DNS Query to suspicious domain: 192.168.1.200 → darkserver.onion
[INFO] Possible lateral movement: 192.168.1.12 accessing multiple hosts via SMB
```

CyberSec Talent Week Challenge

Attività completata!

bip. CyberSec

CyberSec Talent Week Challenge

Cyber Crisis Response: Attacco Ransomware a un ospedale

Francesca Craievich

bip.CyberSec

Cyber Crisis Response Challenge

Questo documento è un template che devi utilizzare come **standard** per realizzare la **presentazione finale**, ovvero il documento conclusivo che racchiude tutti i **task svolti**.

Il template fornito ha puro scopo indicativo. È possibile cambiare liberamente ogni aspetto grafico, aggiungere altre slide ed elementi extra.

Una volta completata la presentazione finale dovrai caricarla in **formato PDF** nell'apposita **sezione di upload** della pagina della Talent Week.

bip.CyberSec

01.

Task 1

Il primo dispositivo infettato è: 192.168.1.12.

Per l’accesso iniziale è stato utilizzato il protocollo SMB.

L'IP del server C2 è 185.100.87.45.
L’IP dell’attaccante non è noto.

Anche se è stata cifrata una cartella condivisa, non necessariamente tutti i pc sono infetti.

02.

Task 2

Script Python:

```
import csv
import ipaddress
from io import StringIO

log_data = """(contenuto del log come stringa, vedi sopra)"""

trusted_subnet = ipaddress.ip_network("172.16.5.0/24")
malicious_ip = "185.100.87.45"

csv_reader = csv.DictReader(StringIO(log_data.strip()), delimiter=';')
suspicious_hosts = set()
infected_hosts = set()

for row in csv_reader:
    src = row["Source_IP"]
    dst = row["Dest_IP"]
    action = row["Action"]
    notes = row["Notes"]
```

02.

Task 2

```
 if malicious_ip in notes or "C2_Connection" in action or "encryption" in notes.lower() or "failed login" in notes.lower():
        suspicious_hosts.add(src)
        infected_hosts.add(src)
    if "lateral movement" in notes.lower() or "darkserver.onion" in notes.lower():
        suspicious_hosts.add(src)
        infected_hosts.add(src)
isolation_commands = []
for ip in infected_hosts:
    if not ipaddress.ip_address(ip) in trusted_subnet:
        isolation_commands.append(f"iptables -A OUTPUT -s {ip} -j DROP")
        isolation_commands.append(f"iptables -A FORWARD -s {ip} -j DROP")
        isolation_commands.append(f"iptables -A FORWARD -d {ip} -j DROP")

print("Host infetti:", infected_hosts)
print("\nComandi per l'isolamento:")
for cmd in isolation_commands:
    print(cmd)
```

02.

Task 2

Output generato:

Host infetti: {'10.0.0.5', '192.168.1.100', '192.168.1.12', '192.168.1.200'}

Comandi per l'isolamento:
iptables -A OUTPUT -s 192.168.1.12 -j DROP
iptables -A FORWARD -s 192.168.1.12 -j DROP
iptables -A FORWARD -d 192.168.1.12 -j DROP
iptables -A OUTPUT -s 192.168.1.100 -j DROP
iptables -A FORWARD -s 192.168.1.100 -j DROP
iptables -A FORWARD -d 192.168.1.100 -j DROP
iptables -A OUTPUT -s 10.0.0.5 -j DROP
iptables -A FORWARD -s 10.0.0.5 -j DROP
iptables -A FORWARD -d 10.0.0.5 -j DROP
iptables -A OUTPUT -s 192.168.1.200 -j DROP
iptables -A FORWARD -s 192.168.1.200 -j DROP
iptables -A FORWARD -d 192.168.1.200 -j DROP

03.

Task 3

Script Python:

```
from Crypto.Cipher import AES
from Crypto.Util.Padding import pad, unpad
import binascii

# Frammenti rubati (ricostruiti in base alla traccia)
AES_PART_1 = "5f2a1c3e9bf0"
AES_PART_2 = "fd7a6b8e"
AES_PART_3 = "3c9d2e7a5f"

# Ricostruzione della chiave AES (fittizia, per esempio)
aes_key_hex = AES_PART_1 + AES_PART_2 + AES_PART_3
# Padding per AES-256 (32 byte / 64 hex chars)
aes_key_hex = aes_key_hex.ljust(64, '0')
aes_key = bytes.fromhex(aes_key_hex)

print("AES_KEY_RECOVERED:", aes_key_hex)

# File critico simulato (contenuto inventato)
plaintext = b"Patient Name: Mario Rossi\nBlood Type: 0+\nAllergies: None"

# IV fisso come da traccia
iv = bytes.fromhex("00" * 16)
```

03.

Task 3

```
# Cifratura
cipher_enc = AES.new(aes_key, AES.MODE_CBC, iv)
ciphertext = cipher_enc.encrypt(pad(plaintext, AES.block_size))

# Scrivi il file cifrato (simulazione)
with open("patient_data_encrypted.bin", "wb") as f:
    f.write(ciphertext)

# Decifratura
cipher_dec = AES.new(aes_key, AES.MODE_CBC, iv)
decrypted = unpad(cipher_dec.decrypt(ciphertext), AES.block_size)

# Salva il file decifrato
with open("patient_data_recovered.txt", "wb") as f:
    f.write(decrypted)

print("Decrypted_File: patient_data_recovered.txt")
```

03.

Task 3

Output:

patient_data_recovered.txt

Patient Name: Mario Rossi
Blood Type: 0+
Allergies: None

patient_data_recovered.txt (in esadecimale)

f79f0551a5a7990de3f841dc77cc5356e1c306d4c51011b7b3c3ef24be8bfa3c383db2c6
dff26079f2aeaf8cf1723e94

04.

Task 4

Rapporto Tecnico – Coordinamento Incidente Ransomware

Il giorno 18/03/2025, alle ore 22:31, l’ospedale ha subito un attacco ransomware che ha compromesso la disponibilità delle cartelle cliniche elettroniche, i sistemi di monitoraggio pazienti e infrastrutture critiche. L’infezione si è propagata tramite exploit SMB e tecniche di phishing mirato. Sono stati identificati accessi remoti non autorizzati e comunicazioni verso un server di comando (IP: 185.100.87.45), con crittografia dei dati attraverso AES-256 e RSA-4096.

Impatto:
Inaccessibilità temporanea ai dati clinici e ai sistemi vitali.
Rischi gravi per la continuità assistenziale e la sicurezza dei pazienti.
Potenziale esposizione di dati sensibili e violazione GDPR.
Rallentamento delle attività ospedaliere e ritardi nelle cure.

Strategia di Ripristino:
1. Isolamento completo delle macchine infette e verifica dell’integrità della trusted subnet.
2. Ripristino dei dati da backup sicuri offline, previa scansione antimalware.
3. Reinstallazione dei sistemi compromessi con aggiornamento delle patch di sicurezza.
4. Audit completo dell’infrastruttura e rafforzamento delle policy di accesso.
5. Formazione urgente del personale su phishing e sicurezza informatica.
6. Collaborazione con autorità per la segnalazione, raccolta prove e investigazione.

05.

Task 5 - opzionale

Script Python:

```
import pandas as pd

# Caricamento del file CSV
df = pd.read_csv("Allegato_3_Task_5_network_logs.csv", sep='\t')

# IP malevoli noti
malicious_ips = ["185.100.87.45"]
# Verifica connessioni C2
for _, row in df.iterrows():
    if row["action"] == "C2_Connection" and row["dest_ip"] in malicious_ips:
        print(f"[ALERT] C2 Connection detected: {row['source_ip']} -> {row['dest_ip']}")

# Verifica DNS verso .onion
for _, row in df.iterrows():
    if row["protocol"] == "DNS" and ".onion" in str(row["bytes_transferred"]):
        print(f"[ALERT] DNS Query to suspicious domain: {row['source_ip']} -> {row['bytes_transferred']}")
```

05.

Task 5 - opzionale

Script Python:

```
# Verifica accessi SMB ripetuti da stesso host
smb_access = df[(df["protocol"] == "SMB") & (df["action"].str.contains("Access"))]
smb_counts = smb_access.groupby("source_ip")["dest_ip"].nunique()

for ip, count in smb_counts.items():
    if count > 1:
        print(f"[INFO] Possible lateral movement: {ip} accessing multiple hosts via SMB")
```

Output generato:

```
[ALERT] C2 Connection detected: 192.168.1.100 -> 185.100.87.45
[ALERT] DNS Query to suspicious domain: 192.168.1.200 -> darkserver.onion
[INFO] Possible lateral movement: 192.168.1.12 accessing multiple hosts via SMB
```

Challenge completata!

CyberSec Talent Week 2025

Durante questa challenge ho messo in pratica molte delle competenze che sto sviluppando nel mio percorso magistrale a Trieste. Ho imparato a decifrare file cifrati da un ransomware ricostruendo le chiavi AES da frammenti corrotti, e ho capito meglio come funzionano le tecniche di cifratura mista (simmetrica e asimmetrica). Ho scritto script Python per automatizzare sia la decifratura che l'analisi di log di rete, rilevando attività sospette come movimenti laterali e connessioni a domini .onion. Infine, ho redatto un report professionale per la direzione ospedaliera, rendendomi conto di quanto sia importante saper comunicare bene anche in contesti critici.

Francesca Craievich

bip. CyberSec