Svolgere i tre esercizi descritti di seguito, caricando i file richiesti via EOL esattamente coi nomi specificati.

È possibile ricaricare ogni file un numero illimitato di volte (ogni volta verrà proposto di sovrascrivere la versione precedente) entro il tempo limite assegnato.

Esercizio 1 - Buffer Overflow

Scaricare il file **bof.zip**

Unzippare il file con l'eseguibile compilato

$ unzip bof.zip

L'eseguibile **es** ha bisogno dei permessi di esecuzione.

$ cd bof

$ chmod +x ./es

Compito dello studente è sfruttare il buffer overflow e farsi stampare la flag.

La tipologia del buffer overflow è una di quelle viste a lezione.

Lo studente dovrà quindi consegnare 3 file

- Un file **bof.txt** contenente:
  - La flag
  - Il payload finale con il quale viene lanciato l'eseguibile
  - Una descrizione il più possibile dettagliata dei passaggi eseguiti per ricavare l'overflow che permette di sovrascrivere l'indirizzo di ritorno. Questa descrizione può includere tutti i tentativi che sono stati fatti oppure soltanto il meccanismo che è stato eseguito.
- Uno screenshot **overflow.png** che mostra il payload che esegue l'overflow che dimostra che siate in grado di controllare l'indirizzo di ritorno ad esempio: AAAA..AA+BBBB
- Uno screenshot **payload.png** che mostra il payload eseguito su **es** e relativa stampa del flag

Il livello di dettaglio e precisione della descrizione sarà utilizzato come valutazione della prova.

Esercizio 2 - Suricata

L'esercitazione consiste nel creare una serie di regole per l'IDS visto a lezione, Suricata. Le regole saranno testate per verificarne la correttezza.

I punti 1 e 2 richiedono di definire semplici regole di base, il punto 3 richiede di studiare un tracciato di traffico e di definire una regola per ricreare i log corrispondenti e da essi individuare la flag.

Lo studente dovrà consegnare un unico file suricata.txt con le soluzioni degli esercizi. Dettagli e descrizioni che motivano una determinata soluzione saranno valutati positivamente.

1) Scrivere una regola suricata in modalità alert per il traffico icmp SOLTANTO IN ENTRATA sulla rete 192.168.56.X

Hint.: Se pensate sia necessario cambiare la configurazione di suricata è bene specificarlo inserendo le righe da modificare.

2) Scrivere una regola suricata in modalità alert per qualsiasi richiesta a **evilcorp.com**. Nota bene NON è possibile utilizzare il protocollo http o la porta 80 per creare questa regola.

3) Ricavare da questo file di tracciato di traffico **nc_esame_25_giugno.pcapng** le seguenti informazioni

- Protocollo del traffico (hint: non è SSH)
- I due Indirizzi IP in gioco (hint: considerate che ovviamente il traffico è in due direzioni)
- La/Le porte del protocollo in gioco

Hint: Sulla VM avete un tool valido per recuperare queste informazioni

Successivamente lo studente deve scrivere una (o più di una se necessario) regola suricata in modalità alert per il traffico del protocollo in questione, specificandone gli IP trovati precedentemente.

Se creata correttamente la regola e configurato correttamente suricata (si ricordi l'esercitazione in classe su MQTT) nei log dovrebbe essere possibile leggere il contenuto del tracciato di traffico.

Parsare a piacimento il contenuto e ricomporre la flag.

Hint. La flag di solito è una frase di senso compiuto

Esercizio 3 - Web

Unzippate il file **web.zip**

$ unzip web.zip

Unzippate l'archivio dei pacchetti di installazione docker

$ unzip docker_packages.zip

Installateli con

$ sudo dpkg -i *.deb

Portatevi nella cartella con la challenge

$ cd web

Diventate root

$ sudo -s

A questo punto si lanci il container, se non sono stati fatti errori dovrebbe restituire un warning su python-xmlm ma tutto è avviato correttamente

 # docker build . -t seclab && docker run -p 8000:8000 seclab

Collegandovi col browser all'indirizzo

http://localhost:8000

dovreste avere la challenge visibile, con la relativa descrizione.

Consegnate il file report.txt con la flag e la descrizione del procedimento utilizzato per ricavarla, il file /etc/passwd (con nome passwd), e uno screenshot screenshot.png come descritto nella pagina web vulnerabile

Informazioni aggiuntive visibili una volta aperta la pagina web:

Web Challenge Esame 25 Giugno 2021:
La challenge consiste nello sfruttare una LFI (Local File Inclusion)

Attraverso una richiesta GET alla root / con il parametro path, e' possibile aprire un file locale. e.g. /?path=file_locale.txt

Scopo dell'esercitazione e' quello di sfruttare la LFI e recuperare/leggere il file /etc/passwd

Sono presenti alcuni filtri sui caratteri che e' possibile inviare come percorso; per ogni filtro "matched" verra' proposto un piccolo suggerimento.

Lo studente dovra' quindi consegnare 3 file

1) Il file report.txt dove dovra' spiegare in maniera comprensibile il concetto dietro alla vulnerabilita' di questa challenge. Lo studente dovra' inoltre elencare i vari passaggi e tentativi eseguiti per sfruttare la vulnerabilita', con i ragionamenti effettuati per "bypassare" i filtri, incluso ovviamente il payload finale

2) Lo screenshot dove e' possibile vedere chiaramente la chiamata all'applicazione con il payload e il risultato finale.

3) Il file etc/passwd

La qualita' del report incidera' sulla valutazione della parte pratica.