Gabriele Di giampietro **Data 2/12/2022**

Obbiettivo Web application hacking:

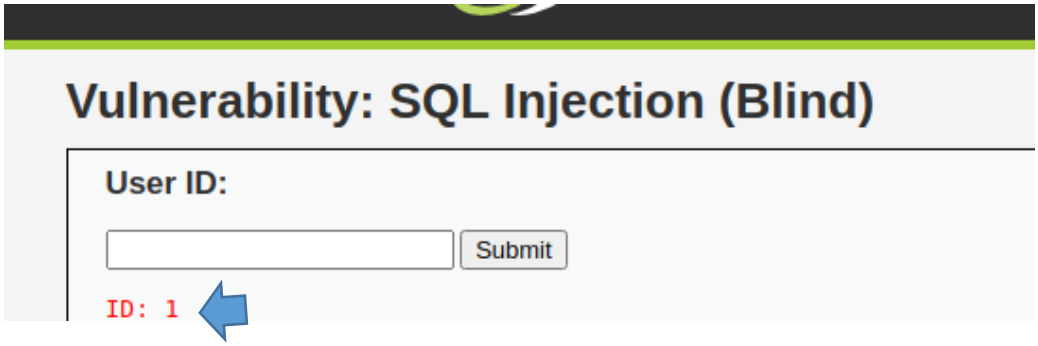
Exploitare le vulnerabilità presenti sulla DVWA della VM Metasploitables, preconfigurando il livello di sicurezza in Low, utilizzando SQL injection Blind e XSS Stored.

Recuperare le password degli utenti presenti nel Database con SQL e recuperare i cookie di sessione delle vittime di XSS ed inviarli verso un server sotto il controllo dell'attacante.

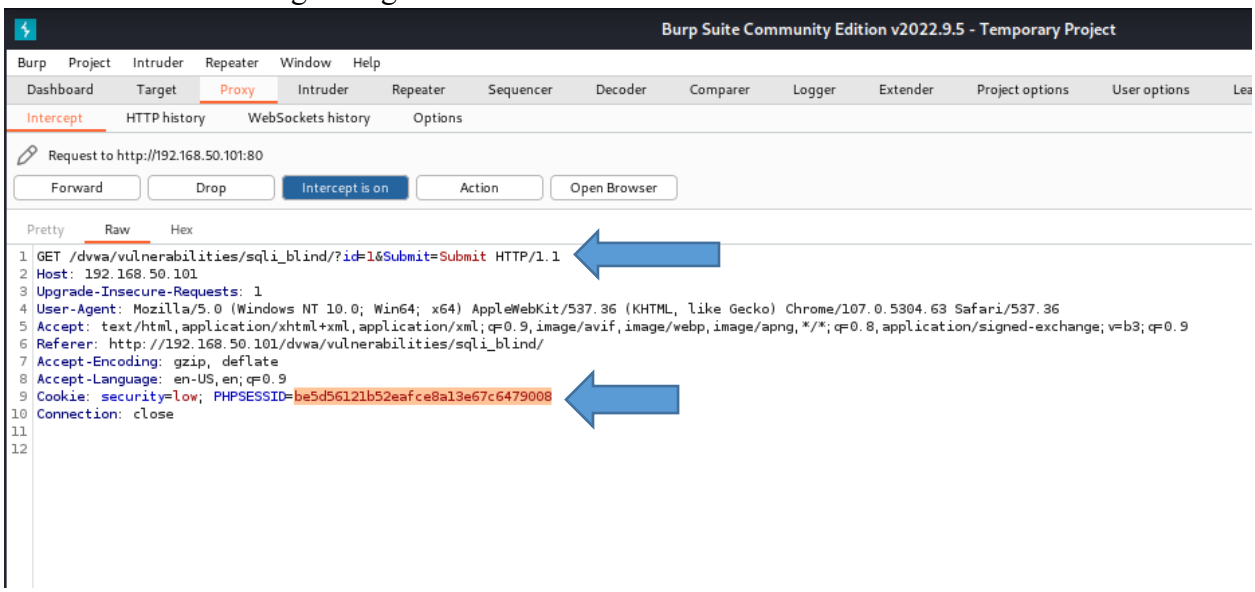
Prima Fase SQL injection Blind

Per prima cosa andiamo nella sezione DVWA security ed inseriamo il livello Low e ci collegheremo dalla Vm Kali, come da traccia.

Successivamente nella prima fase della nostra exploit andiamo a sfruttare le vulnerabilità del DB della VM Metasploitable attraverso la SQL injection per ricavare le password degli utenti al suo interno. Nel nostro caso però ci troviamo nella modalità Blind e quindi inserendo sulla stringa della DVWA una querry non avremo nessuna informazione sui dati contenuto all'interno, come possiamo vedere nell'immagine seguente.



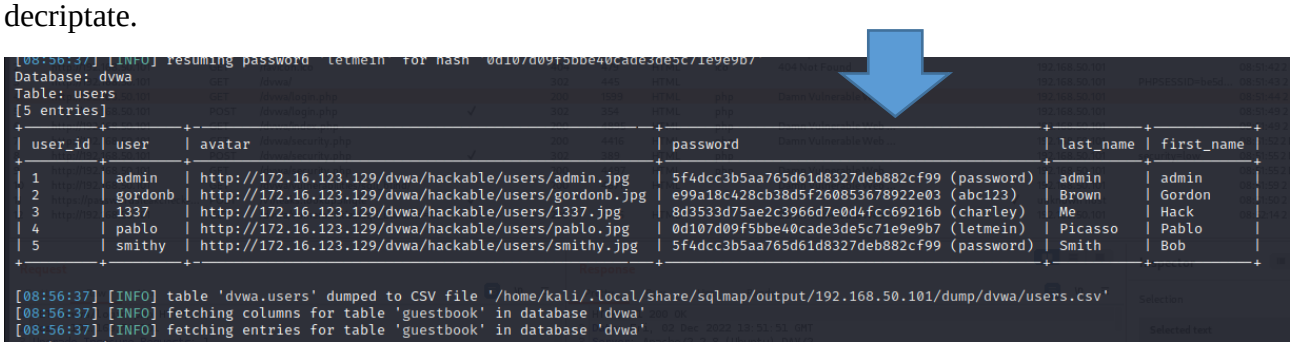
Andiamo allora a sfruttare il cookie di sessione per usarlo successivamente con il tool Sqlmap. Utilizzando il tool Burpsuite intercettiamo i dati di comunicazione sulla DVWA, che oltre a mostrarci la richiesta di Get nell'inserimento della querry, ci mostrerà anche come vediamo evidenziato nell'immagine seguente con le scritte rosse il nostro cookie.



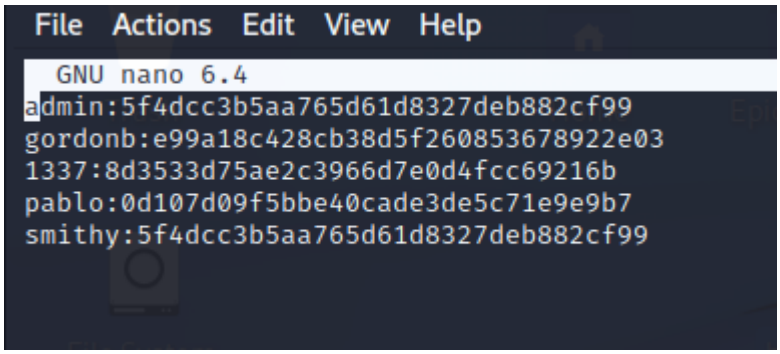
Dopo aver ottenuto il cookie di sessione, come abbiamo detto lo andremo ad utilizzare per il tool Sqlmap, il quale ci permette di vedere il contenuto delle tabelle e delle colonne che contengono i dati dei nostri Users sulla DVWA. Ci permetterà di scegliere anche un attacco dizionario e la decifrazione delle password che vengono mostrate criptate in md5.

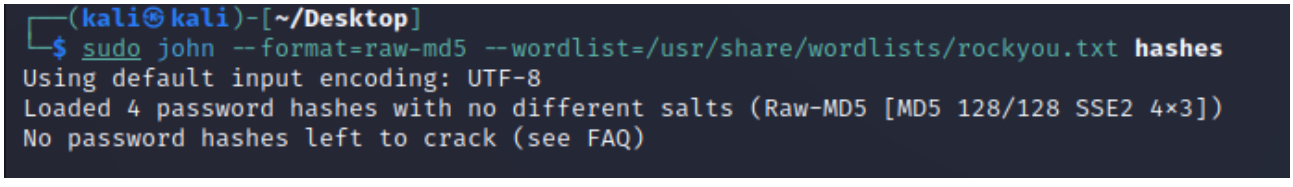
Possiamo vedere nell'immagine seguente che utilizziamo lo switch –u per indicare il nostro URL target e di seguito il cookie sottolineato in rosso recuperato con Burpsuite, con l'ulteriore comando dump per avere la stampa finale a schermo.

Come possiamo vedere abbiamo ottenuto i dati della tabella degli User della DVWA e delle password corrispondenti, possiamo notare la password mostrata si in codice md5 e di fianco decriptate.



Per avere un ulteriore conferma delle password decriptate, abbiamo utilizzato anche il tool John the ripper. Abbiamo prima creato un file con i codici in Hash ricavati e lo abbiamo inserito nella riga di comando per il tool jhon, avendo già decriptato queste password, per mostrarle a schermo utilizziamo lo switch --show. Possiamo vedere i passaggi nelle immagini seguenti:

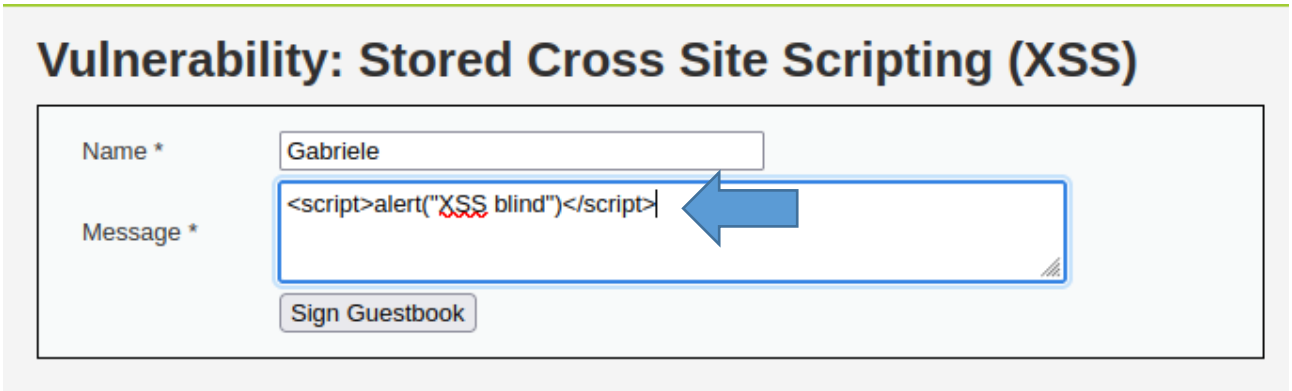


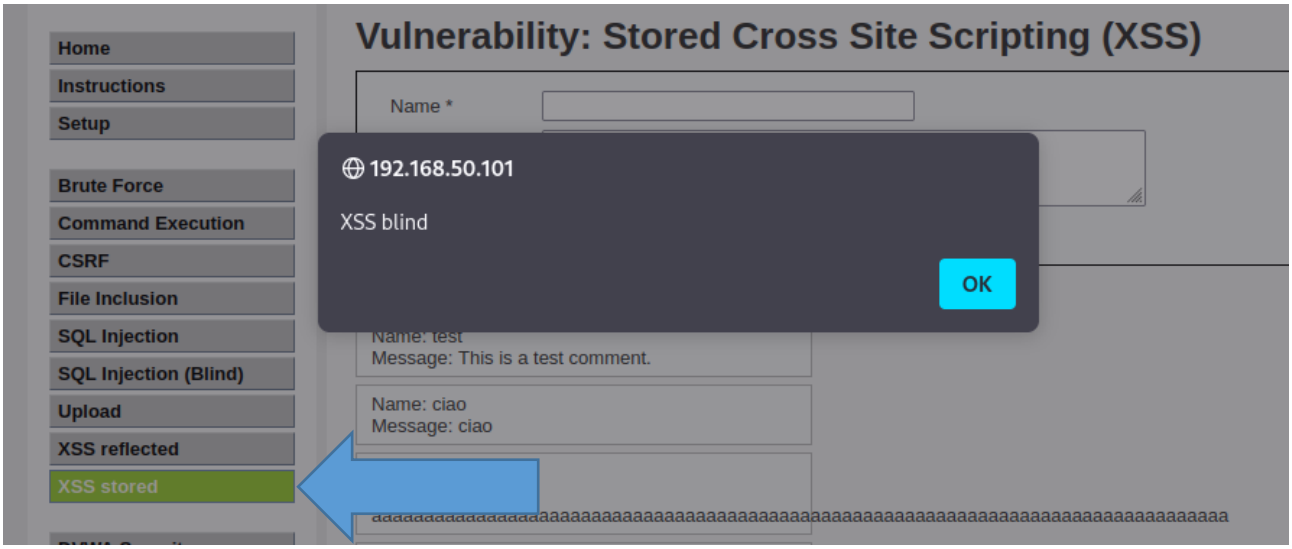


```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo john --show --format=raw-md5 /home/kali/Desktop/hashes
admin:password
gordonb:abc123
1337:charley
pablo:letmein
smithy:password
```

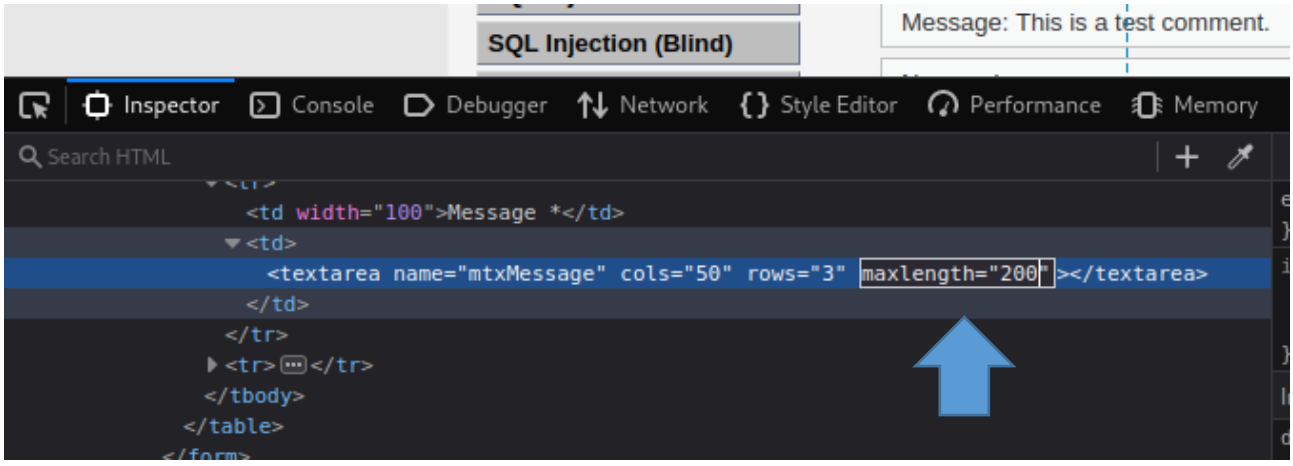
Seconda fase XSS Stored

Nella seconda fase siamo passati all'exploit di XSS Stored (persistente), in questo caso i payload che andremo a inserire verranno salvati permanentemente a differenza dell XSS riflesso, così ogni utente che entrerà nella Web application sarà la nostra vittima. Come prima cosa andiamo sempre ad impostare il livello di security della DVWA sul Low. Andiamo nella sezione della DVWA XSS Stored e andiamo ad inserire qualche stringa e qualche script come ad esempio l'alert e come possiamo vedere nelle immagini seguenti i payload vengono salvati ed ogni volta che riproviamo ad entrare nella sezione XSS Stored ci viene mostrato il nostro script alert.

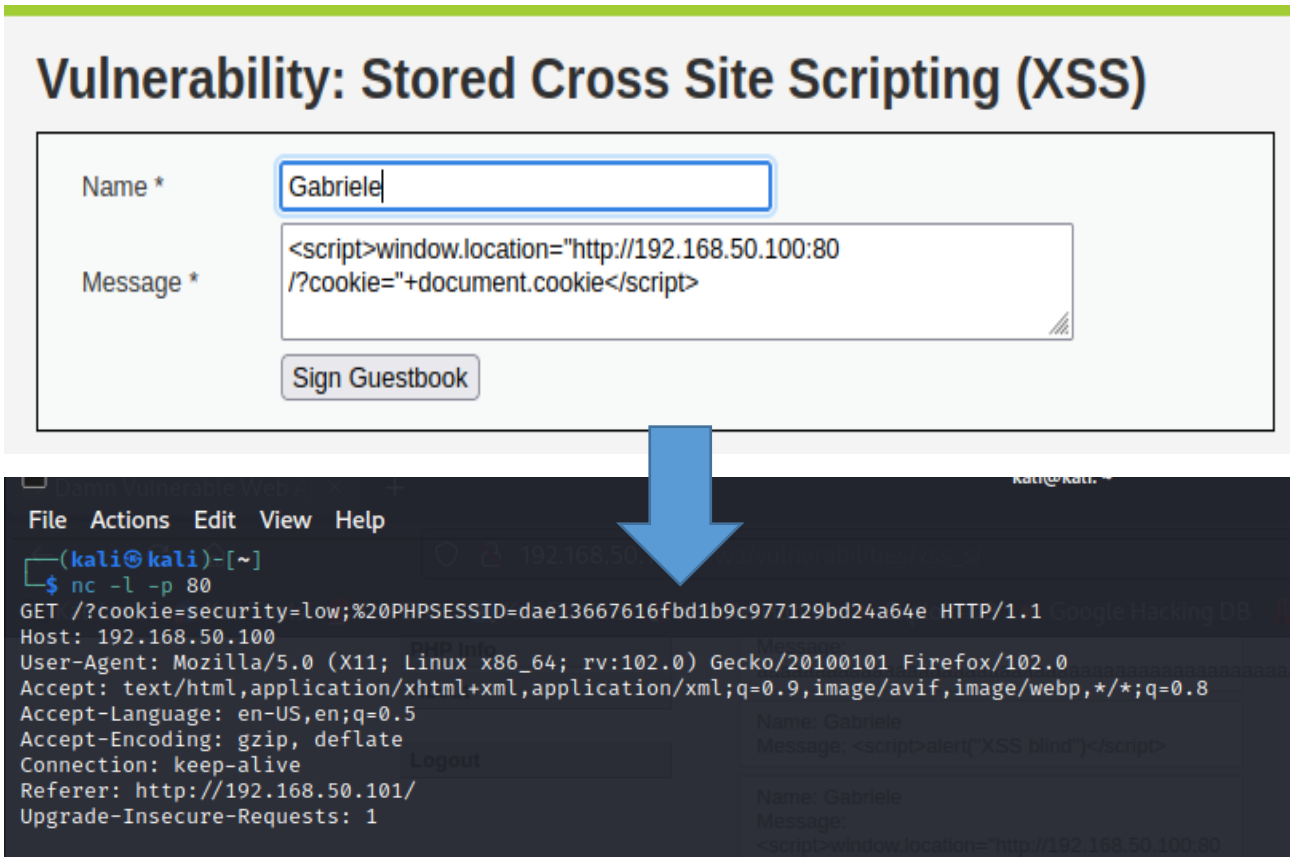




Ci siamo anche accorti che i caratteri nell'inserimento nel corpo del messaggio sono limitati, così ispezionando la pagina abbiamo modificato il "maxlenght", aggiungendo 200 e aumentando il limite di inserimento caratteri.



Aumentare il limite di caratteri ci ha permesso di inserire il nostro script per il recupero del cookie di sessione che andremo ad inviare sul nostro server in ascolto. Nel mio caso ho utilizzato Netcat in ascolto sulla Porta 80 e come possiamo vedere nelle immagini seguenti, ci viene mostrato il cookie sul terminale di Kali.



Ovviamente essendo un XSS Stored, questo dinamiche rimarranno salvate e ogni utente vittima che accederà alla web app inviare il suo cookie di sessione al nostro server in ascolto, oltre ad incontrare tutti gli altri payload inseriti come lo script alert.

Infatti come possiamo vedere nelle immagini che seguono, entrando anche con un altro utente (conosciamo la Password ottenute da Sqlmap), il suo cookie di sessione diverso sarà inviato al nostro Netcat in ascolto sulla Porta 80.