TECNICHE DI EXPLOIT CON #XSS SHELL PHP: #DVWA + BURPSUITE

TRACCIA:

1. RIPETERE L'ESERCIZIO DI IERI UTILIZZANDO QUESTA VOLTA AL POSTO DI UNA SHELL BASE UNA PIÙ SOFISTICATA E COMPLESSA

2. È POSSIBILE REPERIRE DELLE SHELL ANCHE ONLINE O EVENTUALMENTE DENTRO LA STESSA MACCHINA KALI

AL FINE DELL'ESERCIZIO È SUFFICIENTE CHE LE DUE MACCHINE (KALI E METASPLOITABLE) SIANO SULLA STESSA RETE, LASCIO QUINDI L'IP DI METASPLOITABLE SIA 192.168.50.101, MENTRE QUELLO DI KALI LINUX RESTERÀ 192.168.50.100. E FACCIO UN PING DI PROVA DA KALI.

PROSEGUENDO SULLE BASI DEL PRIMO ESERCIZIO CERCHERÒ UNA SHELL PIÙ STRUTTURATA CON CUI SPERIMENTARE. PARTO DI BASE DALL'ESEMPIO DELLE SLIDE CON UNA SHELL RIGUARDANTE I COOKIES. E PROSEGUO CON UN PAIO DI SHELL TROVARE SU INTERNET BASANDOMI SULLA VERSIONE DEL PROTOCOLLO PHP IMPLEMENTATO NELLA WEB APP DVWA.

#VERSIONE PHP:

PHP Version 5.2.4-2ubuntu5.10

System	Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Build Date	Jan 6 2010 21:50:12
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
additional .ini files parsed	/etc/php5/cgi/conf.d/gd.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/mysqli.ini, /etc/php5/cgi/conf.d/pdo.ini, /etc/php5/cgi/conf.d/pdo_mysql.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	zip, php, file, data, http, ftp, compress.bzip2, compress.zlib, https, ftps
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls
Registered Stream Filters	string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, convert.iconv.*, bzip2.*, zlib.*

This server is protected with the Suhosin Patch 0.9.6.2
Copyright (c) 2006 Hardened-PHP Project

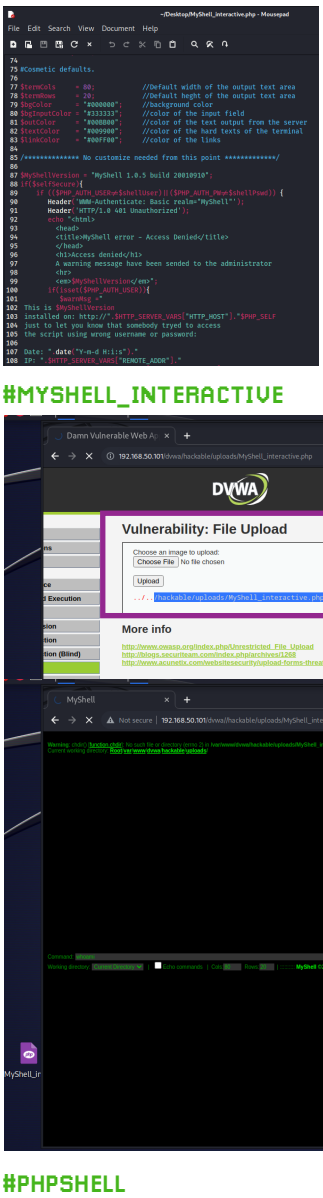
수호신

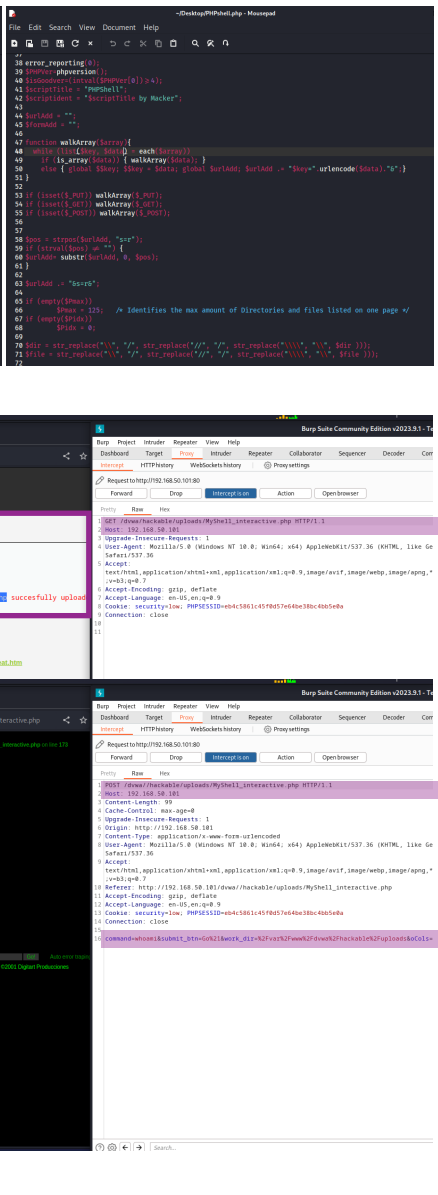
#SHELL PIÙ COMPLESSA:

QUESTE SONO UN PAIO DI SHELL IN PHP ON INTERFACCIA GRAFICA E CODICE +COMPLESSO CHE HO TROVATO IN RETE.
LA PRIMA SHELL CHIAMATA #MYSHELL_INTERACTIVE, CHE MI CONVINCE DI MENO, SI UTILIZZA PER INSERIRE COMANDI, MA A DIFFERENZA DELLA SHELL SEMPLICE POSSIEDE APPUNTO UN'INTERFACCIA GRAFICA WAREZ PARZIALMENTE PROGRAMMABILE SIA A LIVELLO ESTETICO CHE A LIVELLO DI ACCESSO (CON PASSWORD O SENZA AD ES.).
MENTRE LA SECONDA SHELL OLTRE AD UN'INTERFACCIA GRAFICA PER I COMANDI (CHE TROVO MOLTO PIÙ INTUITIVA ED EFFICIENTE) OFFRE ANCHE LA POSSIBILITÀ DI GESTIRE FILES E DIRECTORIES SU UN SERVER POCO SICURO...

#MYSHELL_INTERACTIVE

#PHPSHELL



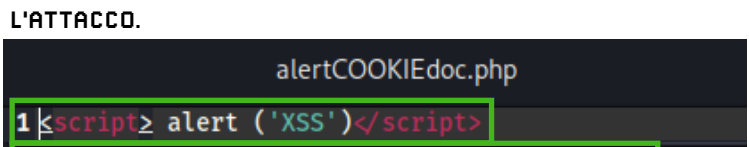


#MYSHELL_INTERACTIVE

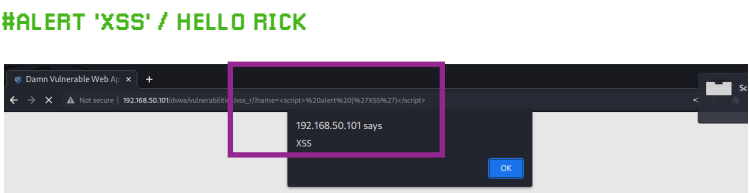
#PHPSHELL

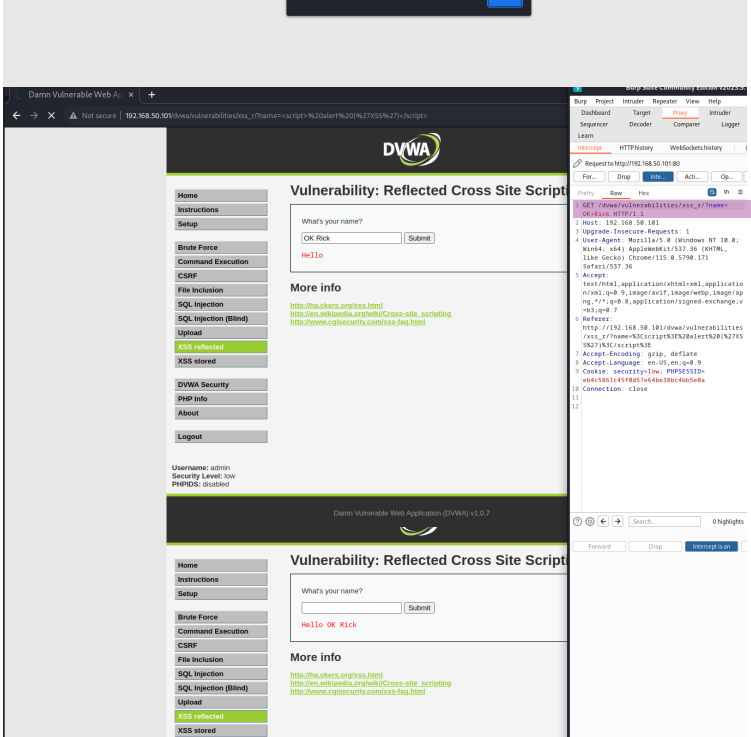
#XSS REFLECTED:

PROVIAMO ANCHE QUESTO TIPO DI ATTACCO, DOVE È SUFFICIENTE ANDARE AD INSERIRE LO SCRIPT NEL CAMPO DI INPUT DELLA PAGINA. NE HO SCELTI DUE (DI CUI UNO PRESENTATO NELLE SLIDE DURANTE IL CORSO) CHE PRESENTANO DUE TIPI DI ATTACCO E OBIETTIVI DIVERSI.
IL PRIMO VISUALIZZA NEL CAMPO IL TESTO DESIDERATO.
IL SECONOD RESTITUISCE L'ID DELLA SESSIONE UTENTE. IL LINK MALEVOLO VA INVIATO TRAMITE L'UTILIZZO DI TECNICHE SOCIAL ENGINEERING ALL'UTENTE BERSAGLIO PER COMPLETARE L'ATTACCO.



#ALERT 'XSS' / HELLO RICK





#DOCUMENT.COOKIE