Guida della vulnerabilità
Failure To Restrict URL Access

Introduzione

Questa è la guida della vulnerabilità di tipo **Failure To Restrict URL Access**, seguendo questa guida riuscirai a sfruttare la vulnerabilità all'interno di HackerLab.

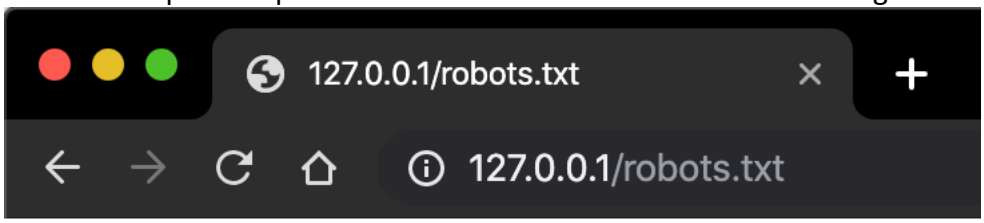
Questo tipo di vulnerabilità permette ad un malintenzionato di poter accedere a delle risorse non accessibili al pubblico di un applicativo sfruttando delle falle nei criteri di autenticazione o di controllo di accesso.

Requisiti

- Browser (Nella guida viene utilizzato Chrome)
  - https://support.google.com/chrome/answer/95346

Guida

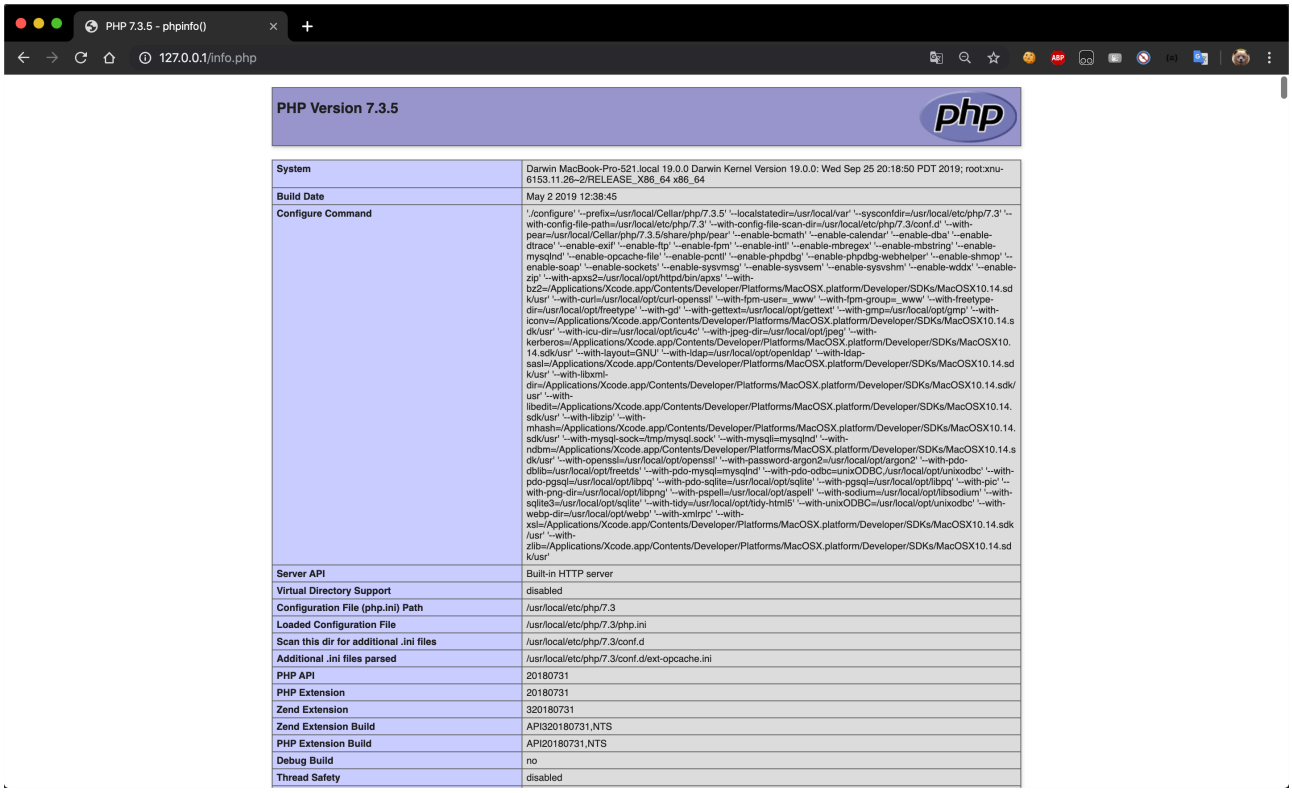
Questa vulnerabilità è molto semplice, per eseguirla al meglio è richiesta una conoscenza base di come sono strutturati i siti web. Comunemente all'interno dei siti web è presente un file chiamato robots.txt che imposta delle regole ai programmi utilizzati dai motori di ricerca per indicizzare i siti web. Queste regole definisco a questi bot se i percorsi inseriti devono essere indicizzati dal motore di ricerca oppure no (questo file ha comunque molte altre funzionalità). In questo caso all'interno di HackerLab è possibile accedere a questo file, che si trova nella cartella principale del sito web, quindi accessibile al percorso /robots.txt.
Accedendo quindi al percorso attraverso un browser si ottiene il seguente risultato:




```
# Regola da applicare a tutti i robot
User-agent: *
# Non fare accedere alle pagine di amministrazione
Disallow: info.php
Disallow: /admin/
```

Figura 1 Contenuto del file robots.txt

All'interno di questo file troviamo quindi le regole da applicare ai robot di indicizzazione. La prima riga del file robots.txt indica che le regole dovranno essere applicati a qualsiasi robot che visiterà il sito web. Le righe seguenti dicono a questi programmi di non accedere al file info.php e alla cartella /admin/. Per eseguire questa vulnerabilità proveremo ad accedere in modo diretto nei percorsi specificati all'interno del file robots.txt.
Quindi proseguiamo ad accedere al file info.php:



System	Darwin MacBook-Pro-521.local 19.0.0 Darwin Kernel Version 19.0.0: Wed Sep 25 20:18:50 PDT 2019; root:xnu-6153.11.26~2/RELEASE_X86_64 x86_64
Build Date	May 2 2019 12:38:45
Configure Command	'./configure' '--prefix=/usr/local/Cellar/php/7.3.5' '--localstatedir=/usr/local/var' '--sysconfdir=/usr/local/etc/php/7.3' '--with-config-file-path=/usr/local/etc/php/7.3' '--with-config-file-scan-dir=/usr/local/etc/php/7.3/conf.d' '--with-pear=/usr/local/Cellar/php/7.3.5/share/php/pear' '--enable-bcmath' '--enable-calendar' '--enable-dba' '--enable-dtrace' '--enable-exif' '--enable-ftp' '--enable-fpm' '--enable-intl' '--enable-mbregex' '--enable-mbstring' '--enable-mysqlnd' '--enable-opcache-file' '--enable-pcntl' '--enable-phpdbg' '--enable-phpdbg-webhelper' '--enable-shmop' '--enable-soap' '--enable-sockets' '--enable-sysvmsg' '--enable-sysvsem' '--enable-sysvshm' '--enable-wddx' '--enable-zip' '--with-apxs2=/usr/local/opt/httpd/bin/apxs' '--with-bz2=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-curl=/usr/local/opt/curl-openssl' '--with-fpm-user=_www' '--with-fpm-group=_www' '--with-freetype-dir=/usr/local/opt/freetype' '--with-gd' '--with-gettext=/usr/local/opt/gettext' '--with-gmp=/usr/local/opt/gmp' '--with-iconv=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-icu-dir=/usr/local/opt/icu4c' '--with-jpeg-dir=/usr/local/opt/jpeg' '--with-kerberos=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-layout=GNU' '--with-ldap=/usr/local/opt/openldap' '--with-ldap-sasl=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-libxml-dir=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-libedit=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-libzip' '--with-mhash=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-mysql-sock=/tmp/mysql.sock' '--with-mysqli=mysqlnd' '--with-ndbm=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-openssl=/usr/local/opt/openssl' '--with-password-argon2=/usr/local/opt/argon2' '--with-pdo-dblib=/usr/local/opt/freetds' '--with-pdo-mysql=mysqlnd' '--with-pdo-odbc=unixODBC,/usr/local/opt/unixodbc' '--with-pdo-pgsql=/usr/local/opt/libpq' '--with-pdo-sqlite=/usr/local/opt/sqlite' '--with-pgsql=/usr/local/opt/libpq' '--with-pic' '--with-png-dir=/usr/local/opt/libpng' '--with-pspell=/usr/local/opt/aspell' '--with-sodium=/usr/local/opt/libsodium' '--with-sqlite3=/usr/local/opt/sqlite' '--with-tidy=/usr/local/opt/tidy-html5' '--with-unixODBC=/usr/local/opt/unixodbc' '--with-webp-dir=/usr/local/opt/webp' '--with-xmlrpc' '--with-xsl=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr' '--with-zlib=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.14.sdk/usr'
Server API	Built-in HTTP server
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/local/etc/php/7.3
Loaded Configuration File	/usr/local/etc/php/7.3/php.ini
Scan this dir for additional .ini files	/usr/local/etc/php/7.3/conf.d
Additional .ini files parsed	/usr/local/etc/php/7.3/conf.d/ext-opcache.ini
PHP API	20180731
PHP Extension	20180731
Zend Extension	320180731
Zend Extension Build	API320180731,NTS
PHP Extension Build	API20180731,NTS
Debug Build	no
Thread Safety	disabled

Figura 2 Contenuto del file info.php

Come possiamo notare, siamo riusciti ad accedere ad un file di prova dedicato solamente allo sviluppo. Attraverso questo file possiamo ricavare moltissime informazioni riguardanti la macchina che ospita l'applicativo web, attraverso le quali possiamo ricercare vulnerabilità specifiche. Quindi possiamo considerare di aver già sfruttato la falla di tipo Failure To Restrict URL Access.
Proseguiamo dunque ad andare alla cartella /admin/:

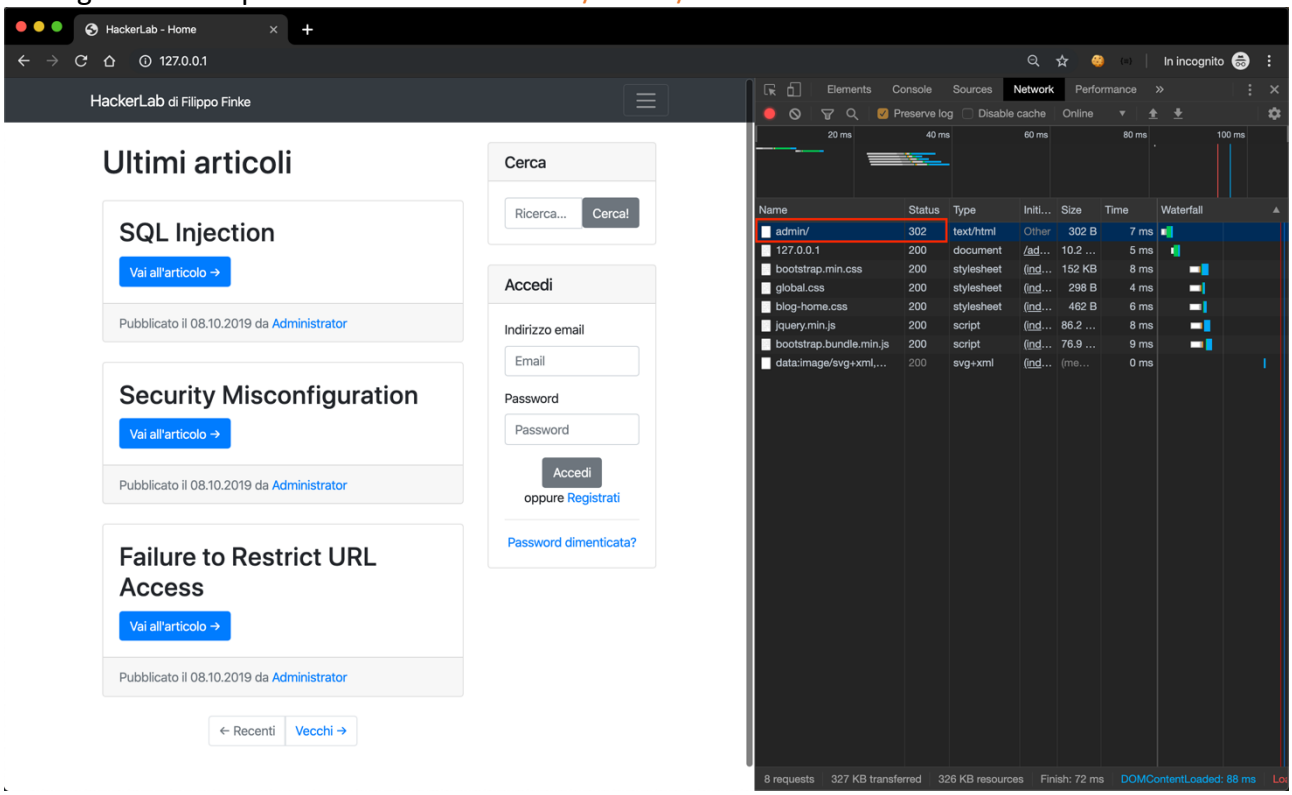


Figura 3 Accesso alla cartella admin

In questo caso possiamo notare che accedendo alla cartella admin si viene reindirizzati (Status 302 nella scheda delle richieste del browser) alla pagina principale del sito web, possiamo quindi dire che in questo caso i controlli di accesso siano stati eseguiti correttamente.