PENETRATION TEST REPORT

EESMSAFE.MO.COM

Penetration Tester – Neamul Kabir Emon
Phone Number – +8801741286840
Email – contact@neamulkabirllc.us

Document Control

Document Version	Owner & Role	Status & comments
V2.0	Adrian Owen	Penetration test report

Disclaimer

The content of this report is highly confidential and may include critical information on EESMSAFE.MO.COM systems, network, and applications. The report should be shared only with intended parties.

Although maximum effort has been applied to make this report accurate, Neamul Kabir Emon cannot be held responsible for inaccuracies or systems changes after the report has been issued since new vulnerabilities may be found once the tests are completed.

Moreover, Neamul Kabir Emon cannot be held responsible on how the report is implemented and changes made to EESMSAFE.MO.COM. Systems based on the recommendations of this report. Guidance should be taken from a network and security expert on how best to implement the recommendations.

All other information and the formats, methods, and reporting approaches is the intellectual property of Neamul Kabir Emon and is considered proprietary information and is provided in confidence to EESMSAFE.MO.COM for the purpose of internal use only.

Any copying, distribution, or use of any of the information set forth herein or in any attachments hereto form outside of EESMSAFE.MO.COM authorized representatives is strictly prohibited unless Example EESMSAFE.MO.COM obtains prior written consent of Neamul Kabir Emon.

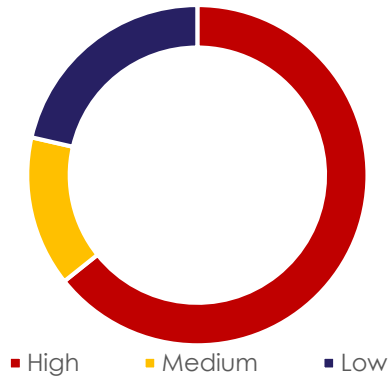
Table of Contents

Executive Summary

I was tasked with performing a black box penetration test towards https://92.3.102.27:4444 which revealed a need for Immediate Attention. The test was conducted on total 1 targets under an emergency 1 Days period.

Security tests were conducted from internet over the period from 19 May, 2022 to 20 May, 2022 with no prior knowledge of https://92.3.102.27:4444 state of security for the systems under tests. I have performed full penetration test but there is no weakness in taking complete access to the site.

The environment was found to some vulnerabilities, including one very serious security flaws such as *Microsoft IIS tilde directory enumeration* which makes them possible sensitive information disclosure. Putting the https://92.3.102.27:4444 at it is not possible to do much damage with this weakness



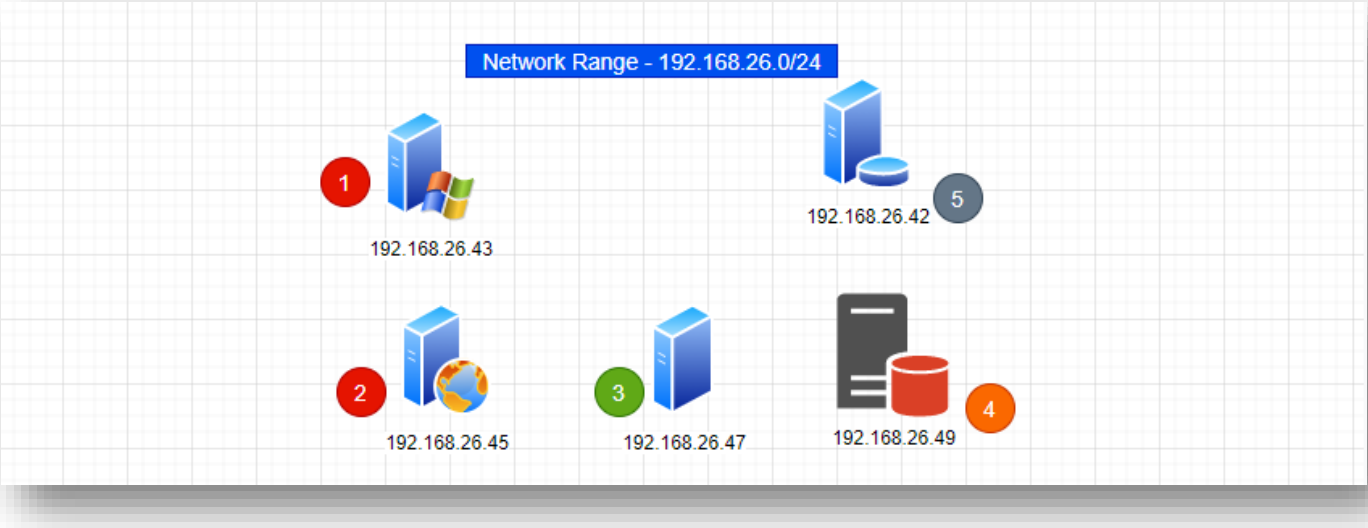
It appears that the overall security posture is extremely great, and no compliance to best practices. These servers and sites are secure don't have direct accessible vulnerability. https://92.3.102.27:4444.

In conclusion, based on the results of the tests, I believe that the https://92.3.102.27:4444 presents 1 High severity vulnerability & 2 medium vulnerability of performing complete penetration test .below the expected level of security, therefore the overall assessment was rated as **"GREAT".**

Security Posture

The scope was to exploit vulnerabilities on https://92.3.102.27:4444 servers and apps that may be exploited by malicious attack. The aim of the tests was to go as far as possible.

NOTE:- Dots Color Signify ➢ Red - High Risk Orange - Mid Risk Green - Low Risk Grey - Safe



By this map, it is extremely clear that the organizational security measures, policies, practices and procedures are aligned with the industry best practices. More than 10% of the tested infrastructure is in a critical state with High level of Risk. Total Site is a very secure new website with no such triticale weakness. Fix high and medium weaknesses first

TOTAL NUMBER OF VULNERABILITIES (including all 1 target machines)

Total Findings	High	Medium	Low
0	0	0	0

Overall Security Rating – Immediate Attention and Action Required

Methodology

I utilized a widely adopted approach to performing penetration testing during the tests to test how well the target environment is secured. Below, a breakdown of the applied methodology is provided.

- Information Gathering – Reconnaissance [Foot printing, Scanning and Enumeration]
- Vulnerability Analysis – Researching Potential Vulnerabilities and Analyzing them
- Exploitation – Using Exploits in order to validate the vulnerabilities of the target
- Post Exploitation – Everything that should be performed after successful exploitation
- House Cleaning – Ensuring that the Remnants of the Penetration Test are removed

Tools Utilized

Tools used by me were Industry Grade in a combination of Open Source and Commercial Licenses.

1. Nmap – Industry's Most Commonly used Open-Source Scanning Tool
2. Metasploit Framework – Industry Grade Most Popular Pen-Testing Framework Toolset
3. BurpSuite Professional – Best in Class Suite of Tools for Web Application Assessment
4. Nikto – Web Server Audit Tool
5. Dirbuster – Directory & Web Files Enumeration Tool
6. Wpscan – Most popular Word press Website Security scanning tool

Detailed Findings

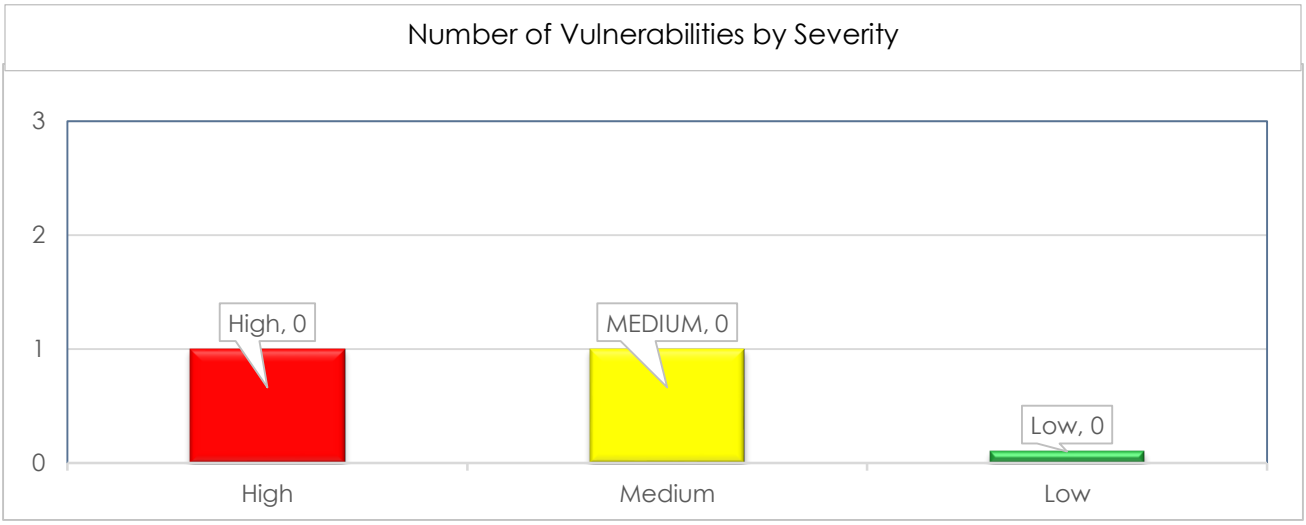
HOST – https://92.3.102.27

Name: https://92.3.102.27:4444
IP: 92.3.102.27
Type: **Windows** server

This host contains –

1. **Microsoft-IIS/10.0**
2. **Identified Technologies ASP.NET**

Operating System: **Windows**

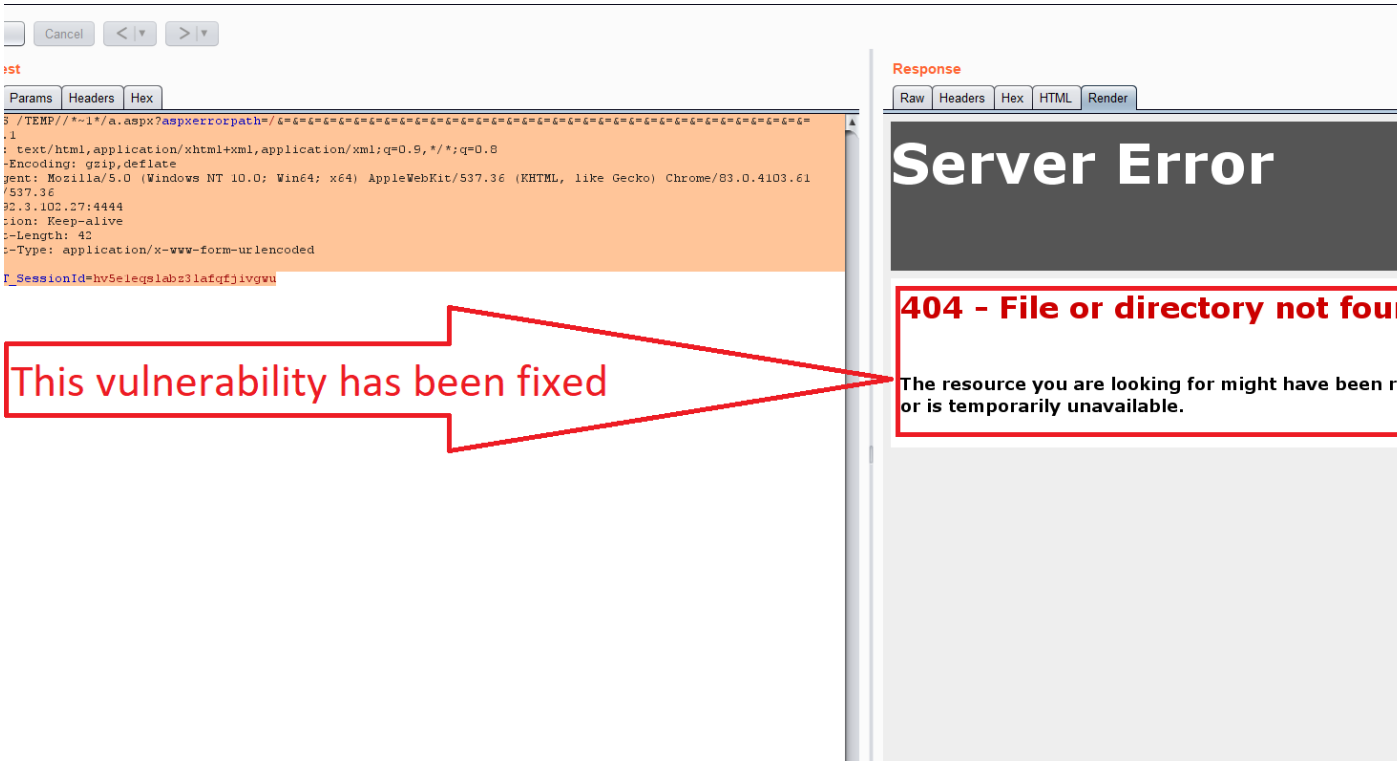


1. Microsoft IIS tilde directory enumeration – High [Resolved]

[Note: Your developer has solved this vulnerability – Renamed or removed this IIS Directory or file]

- ~~Vulnerability – Possible sensitive information disclosure.~~

  ~~URL: https://92.3.102.27:4444/~~



~~Description~~

~~It is possible to detect short names of files and directories which have an 8.3 file naming scheme equivalent in Windows by using some vectors in several versions of Microsoft IIS. For instance, it is possible to detect all short names of ".aspx" files as they have 4 letters in their extensions. This can be a major issue, especially for the .Net websites which are~~

vulnerable to direct URL access as an attacker can find important files and folders that are not normally visible.

Discovered by **Microsoft IIS tilde directory enumeration**

The impact of this vulnerability

This may cause the leakage of files containing sensitive information such as credentials, configuration files, maintenance scripts and other data.

Remediation

Discarding all web requests using the tilde character. Add a registry key named NtfsDisable8dot3NameCreation to HKLM\SYSTEM\CurrentControlSet\Control\FileSystem. Set the value of the key to 1 to mitigate all 8.3 name conventions on the server.

References

- Windows Short (8.3) Filenames - A Security Nightmare?
- Detectify KB: Microsoft IIS Tilde Vulnerability
- Microsoft IIS Short name Scanner PoC
- Microsoft IIS tilde character "~" Vulnerability/Feature – Short File/Folder Name Disclosure
- IIS Short File Name Disclosure is back! Is your server vulnerable?

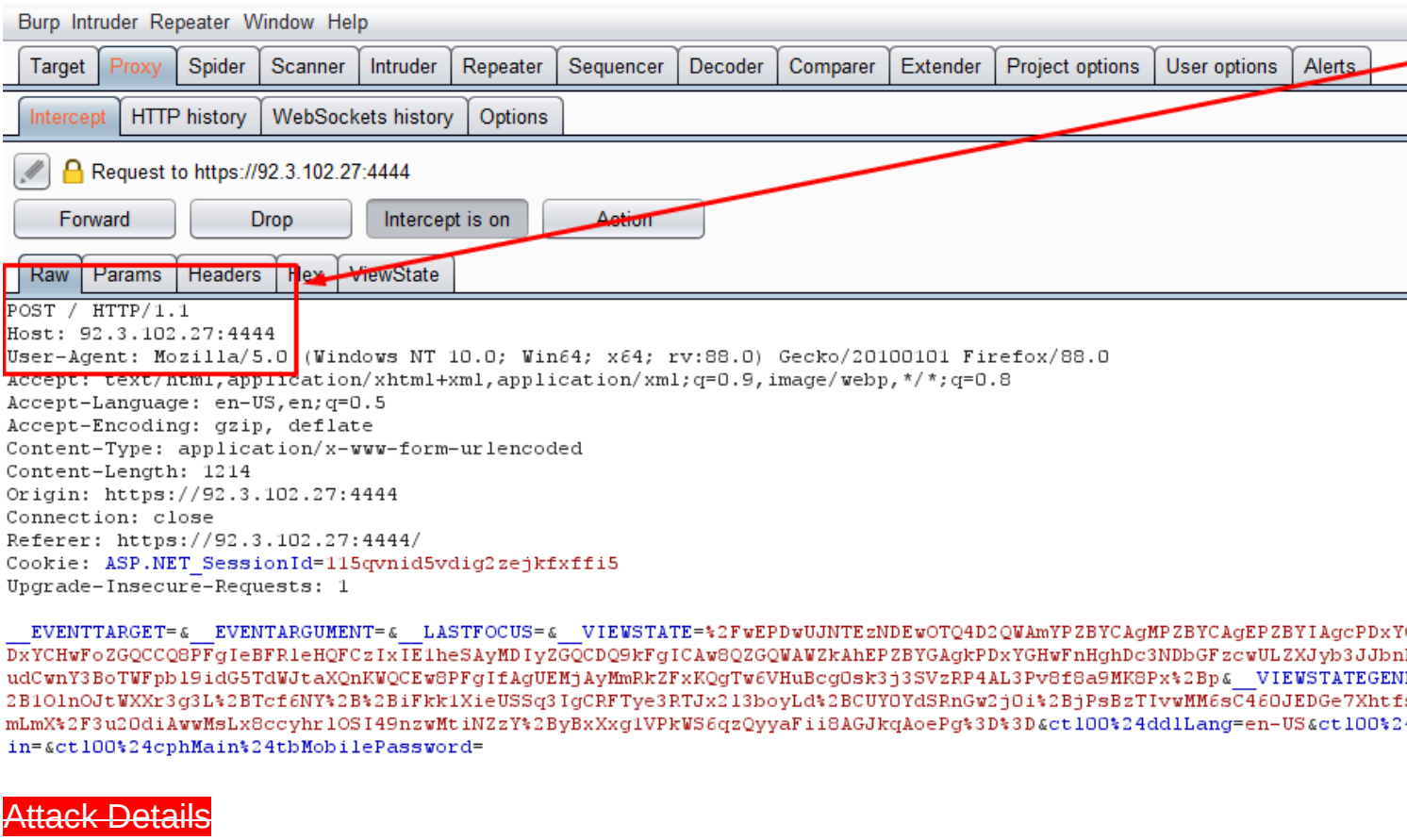
Classification

CWE CWE-20
CVSS Base Score: **7.5** - CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N
Attack Vector: Network
Attack Complexity: Low
Privileges Required: None
User Interaction: None
Scope: Unchanged
Confidentiality: High
Integrity: None
Availability: None

2. Password field submitted using GET method –Medium [Resolved]

[NOTE: Your developer has solved this vulnerability His Changed password submitted method GET to POST]

- ~~Vulnerability – Brute Force Attacks~~
- ~~URL:~~ ~~https://92.3.102.27:4444~~



~~Attack Details~~

~~Forms with password field submitted via GET:~~

- https://92.3.102.27:4444/

- Form name: <empty>
- Form action: <empty>
- Form method: GET

  Password input: ctl00$cphMain$tbMobilePassword
- https://92.3.102.27:4444/login.aspx

- Form name: <empty>
- Form action: <empty>
- Form method: GET

  Password input: ctl00$cphMain$tbMobilePassword

Description

These page(s) contain a form with a password field. The form's method attribute is either set to GET, or not defined at all, in which case it defaults to GET. This configuration may lead to user data being submitted using the GET method, causing the contents of the password field to appear in the URL. Sensitive information should not be passed via the URL. URLs could be logged or leaked via the Referer header.

Discovered by Password field submitted using GET method

The impact of this vulnerability

Possible sensitive information disclosure.

Remediation

The HTML form's method attribute should be defined and set POST rather than GET.

Classification

CWE CWE-200
CVSS Base Score: **5.3** - CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N
Attack Vector: Network
Attack Complexity: Low
Privileges Required: None
User Interaction: None

Scope: Unchanged
Confidentiality: Low
Integrity: None
Availability: None

References

Possible sensitive information disclosure.

3. TLS 1.0 enabled. Medium [Resolved] [NOTE : This vulnerability has also Resolved]Now this server don't using TLSv1.0.

[NOTE: Your developer has solved this vulnerability Disabled TLS 1.0]

- URL:https://92.3.102.27:4444/

Attack Details the SSL server (port: 4444) encrypts traffic using TLSv1.0.

Description

The web server supports encryption through TLS 1.0. TLS 1.0 is not considered to be "strong cryptography" as defined and required by the PCI Data Security Standard 3.2(.1) when used to protect sensitive information transferred to or from web sites. According to PCI, "30 June 2018 is the deadline for disabling SSL/early TLS and implementing a more secure encryption protocol – TLS 1.1 or higher (TLS v1.2 is strongly encouraged) in order to meet the PCI Data Security Standard (PCI DSS) for safeguarding payment data. Discovered by TLS 1.0 enabled

Remediation

It is recommended to disable TLS 1.0 and replace it with TLS 1.2 or higher.

Classification

CWE CWE-16

CVSS Base Score: 3.1 - CVSS:3.0/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N
- Attack Vector: Network
- Attack Complexity: High
- Privileges Required: None
- User Interaction: Required
- Scope: Unchanged
- Confidentiality: Low
- Integrity: None
- Availability: None

Web References

- Are You Ready for 30 June 2018? Saying Goodbye to SSL/early TLS
- PCI 3.1 and TLS 1.2 (Cloud flare Support)

4. Click jacking: X-Frame-Options header– LOW Resolved

URL: https://92.3.102.27:4444/

Attack Details

Paths without secure XFO header:
https://92.3.102.27:4444/
https://92.3.102.27:4444/ScriptResource.axd
https://92.3.102.27:4444/WebResource.axd
https://92.3.102.27:4444/login.aspx

Description

Click jacking (User Interface redress attack, UI redress attack, UI redressing) is a malicious technique of tricking a Web user into clicking on something different from what the user perceives they are clicking on, thus potentially revealing confidential information or taking control of their computer while clicking on seemingly innocuous web pages. The server did not return an X-Frame-Options header with the value DENY or SAMEORIGIN, which means that this website could be at risk of a click jacking attack. The X-Frame-Options HTTP response header can be used to indicate whether or not a browser should be allowed to render a page inside a frame or frame. Sites can use this to avoid click jacking attacks, by ensuring that their content is not embedded into untrusted sites. Discovered by Click jacking: X-Frame-Options header

The impact of this vulnerability

The impact depends on the affected web application.This Vulnerability is low severity and this website don't have any risk with this vulnerability.

How to fix this vulnerability

Configure your web server to include an X-Frame-Options header and a CSP header with frame-ancestors directive. Consult Web references for more information about the possible values for this header.

Classification

CWE CWE-693
CVSS Base Score: **5.8** - CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:L/A:N

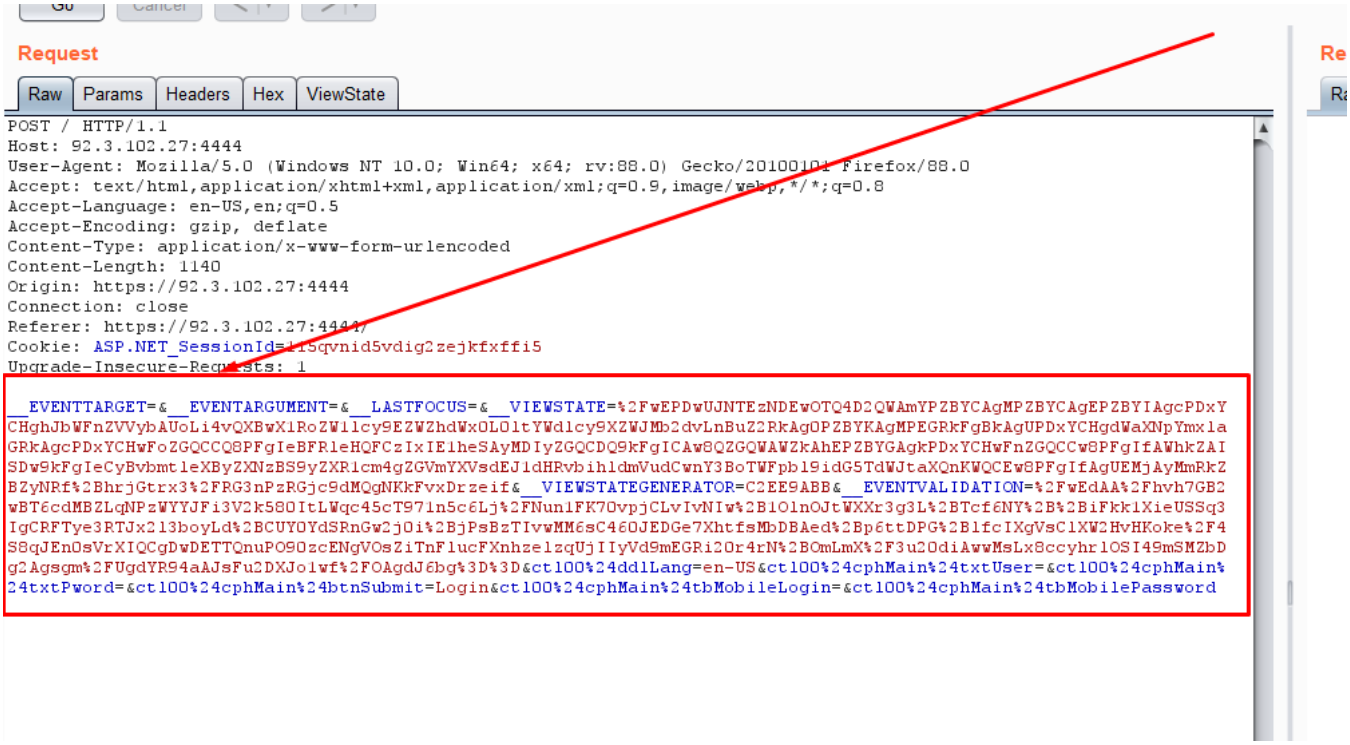
- Attack Vector: Network
- Attack Complexity: Low
- Privileges Required: None
- User Interaction: None
- Scope: Changed
- Confidentiality: None
- Integrity: Low
- Availability: None

Web References

- The X-Frame-Options response header
- Click jacking
- OWASP Clickjacking
- Frame Buster Buster

5. Cookies with missing, inconsistent or contradictory properties

LOW [Resolved]



URL: https://92.3.102.27:4444/

Attack Details

List of cookies with missing, inconsistent or contradictory properties:

- https://92.3.102.27:4444/

  Cookie was set with:

  Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:36:16 GMT; path=/

This cookie has the following issues:

- Cookie without SameSite attribute.
When cookies lack the SameSite attribute, Web browsers may apply different and sometimes unexpected defaults. It is therefore recommended to add a SameSite attribute with an appropriate value of either "Strict", "Lax", or "None".

- https://92.3.102.27:4444/

Cookie was set with:
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:37:05 GMT; path=/

This cookie has the following issues:

- Cookie without SameSite attribute.
When cookies lack the SameSite attribute, Web browsers may apply different and sometimes unexpected defaults. It is therefore recommended to add a SameSite attribute with an appropriate value of either "Strict", "Lax", or "None".

- https://92.3.102.27:4444/

Cookie was set with:
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:33:56 GMT; path=/

This cookie has the following issues:

- Cookie without SameSite attribute.
When cookies lack the SameSite attribute, Web browsers may apply different and sometimes unexpected defaults. It is therefore recommended to add a SameSite attribute with an appropriate value of either "Strict", "Lax", or "None".

- https://92.3.102.27:4444/

Cookie was set with:
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:37:06 GMT; path=/

This cookie has the following issues:

- Cookie without SameSite attribute.
When cookies lack the SameSite attribute, Web browsers may apply different and sometimes unexpected defaults. It is therefore recommended to add a SameSite attribute with an appropriate value of either "Strict", "Lax", or "None".

- https://92.3.102.27:4444/login.aspx

  Cookie was set with:

  ```
  Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:41:11 GMT; path=/
  ```

  This cookie has the following issues:

  ```
  - Cookie without SameSite attribute.
  When cookies lack the SameSite attribute, Web browsers may apply different and
  sometimes unexpected defaults. It is therefore recommended to add a SameSite
  attribute with an appropriate value of either "Strict", "Lax", or "None".
  ```

Vulnerability Description

At least one of the following cookies properties causes the cookie to be invalid or incompatible with either a different property of the same cookie, of with the environment the cookie is being used in. Although this is not a vulnerability in itself, it will likely lead to unexpected behavior by the application, which in turn may cause secondary security issues.

Discovered by Cookies with missing, inconsistent or contradictory properties

The impact of this vulnerability

Cookies will not be stored, or submitted, by web browsers.

How to fix this vulnerability

Ensure that the cookies configuration complies with the applicable standards.

Web References

- MDN | Set-Cookie
- Securing cookies with cookie prefixes
- Cookies: HTTP State Management Mechanism
- SameSite Updates - The Chromium Projects
- draft-west-first-party-cookies-07: Same-site Cookies

6. Cookies without Http only flag set– Low Resolved

URL: https://92.3.102.27:4444/

Attack Details

Cookies without HttpOnly flag set:

- https://92.3.102.27:4444/

```
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:36:16 GMT; path=/
```

- https://92.3.102.27:4444/

```
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:37:05 GMT; path=/
```

- https://92.3.102.27:4444/

```
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:33:56 GMT; path=/
```

- https://92.3.102.27:4444/

```
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:37:06 GMT; path=/
```

- https://92.3.102.27:4444/login.aspx

```
Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:41:11 GMT; path=/
```

Vulnerability Description

One or more cookies don't have the HttpOnly flag set. When a cookie is set with the HttpOnly flag, it instructs the browser that the cookie can only be accessed by the server and not by client-side scripts. This is an important security protection for session cookies.Discovered by Cookies without HttpOnly flag set

Impact: Cookies could be sent over unencrypted channels.

- https://www.latamairlines.com/ Verified Cookies without Secure flag set: https://www.latamairlines.com/ Set-Cookie:]

The impact of this vulnerability

Cookies can be accessed by client-side scripts.

The impact of this vulnerability

If possible, you should set the Http Only flag for these cookies.

Classification

CWECWE-16 CVSS
Base Score: **0** - CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:N/I:N/A:N
Attack Vector: Network
Attack Complexity: Low
Privileges Required: None
User Interaction: Required
Scope: Unchanged
Confidentiality: None
Integrity: None
Availability: None

7. Cookies without Secure flag set– Low Resolved

URL: https://92.3.102.27:4444/

Attack Details

Cookies without Secure flag set:
https://92.3.102.27:4444/

Cookies without Secure flag set:

- https://92.3.102.27:4444/

Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:36:16 GMT; path=/

- https://92.3.102.27:4444/

Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:37:05 GMT; path=/

- https://92.3.102.27:4444/

Set-Cookie: ASP.NET_SessionId=cxwkub2kxg41n3nuuwdrtudd; path=/; HttpOnly; SameSite=Lax

- https://92.3.102.27:4444/

Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:33:56 GMT; path=/

- https://92.3.102.27:4444/

Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:37:06 GMT; path=/

- https://92.3.102.27:4444/login.aspx

Set-Cookie: fsLang=en-US; expires=Wed, 19-May-2032 20:41:11 GMT; path=/

-

Vulnerability Description

One or more cookies does not have the Secure flag set. When a cookie is set with the Secure flag, it instructs the browser that the cookie can only be accessed over secure SSL/TLS channels. This is an important security protection for session cookies.

Discovered by **Cookies without Secure flag set**

The impact of this vulnerability

Cookies could be sent over unencrypted channels.

The impact of this vulnerability

If possible, you should set the secure flag for these cookies.

Classification

CWECWE-16CVSS

Base Score: **0** - CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:N/I:N/A:N

Attack Vector: Network

Attack Complexity: Low

Privileges Required: None

User Interaction: Required

Scope: Unchanged

Confidentiality: None

Integrity: None

Availability: None

8. HTTP Strict Transport Security (HSTS) not implemented– Low Resolved

URL: https://92.3.102.27:4444/

Attack Details

URLs where HSTS is not enabled:

- https://92.3.102.27:4444/

- https://92.3.102.27:4444/ScriptResource.axd

- https://92.3.102.27:4444/WebResource.axd

- https://92.3.102.27:4444/login.aspx

Vulnerability Description

HSTS can be used to prevent and/or mitigate some types of man-in-the-middle (MitM) attacks

Impact: Cookies could be sent over unencrypted channels.

- https://www.latamairlines.com/ Verified Cookies without Secure flag set: https://www.latamairlines.com/ Set-Cookie:]

The impact of this vulnerability

HSTS can be used to prevent and/or mitigate some types of man-in-the-middle (MitM) attacks

The impact of this vulnerability

It's recommended to implement HTTP Strict Transport Security (HSTS) into your web application. Consult web references for more information

Classification

arrow_drop_up
CWE CWE-16CVSS
Base Score: **0** - CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:N/I:N/A:N
Attack Vector: Network
Attack Complexity: Low
Privileges Required: None
User Interaction: Required
Scope: Changed
Confidentiality: None

Integrity: None
Availability: None

Web References

- hstspreload.org
- Strict-Transport-Security

9. Login page password-guessing attack– Low [Note: This web application using Strong/Encrypted Password I will run brute force – Password guessing attack using 1 lakh common password so this vulnerability not exploitable doesn't effect anymore]

URL: https://92.3.102.27:4444/

Vulnerability Description

A common threat web developer's face is a password-guessing attack known as a brute force attack. A brute-force attack is an attempt to discover a password by systematically trying every possible combination of letters, numbers, and symbols until you discover the one correct combination that works.

This login page doesn't have any protection against password-guessing attacks (brute force attacks). It's recommended to implement some type of account lockout after a defined number of incorrect password attempts. Consult Web references for more information about fixing this problem.

Discovered by Login page password-guessing attack

The impact of this vulnerability

An attacker may attempt to discover a weak password by systematically trying every possible combination of letters, numbers, and symbols until it discovers the one correct combination that works.

How to fix this vulnerability

It's recommended to implement some type of account lockout after a defined number of incorrect password attempts.

Classification

CWE CWE-307 CVSS

Base Score: **5.3** - CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L

Attack Vector: Network

Attack Complexity: Low

Privileges Required: None

User Interaction: None

Scope: Unchanged

Confidentiality: None

Integrity: None

Availability: Low

Web References

Blocking Brute Force Attacks

Conclusion

https://92.3.102.27:4444/ don't have a series of control failures, which led to a complete compromise of many in-scope machines. I would recommend you to fix these by looking at the weakness of High and Mediums every.

The overall risk identified to https://92.3.102.27:4444/ as a result of the penetration test is Medium. Attackers or hackers do not have the vulnerability to gain access from outside. I did not find any vulnerabilities in this system that would allow hackers to gain direct access to the server, it's a Medium risky situation.

The primary goal of this penetration test was stated as identifying if there is any weakness in https://92.3.102.27:4444/ .

These goals of the pen test were met and in-fact much more than this. 1 High Severity and 2 Medium severity vulnerabilities were found during the test that directly affect confidentiality, integrity and availability of the information and systems. Majority of the findings have occasional prevalence, easy exploitability, and divesting impact with simple prevention.

It was found that your security architecture has few patterns:

- Operating Systems are Outdated and Unpatched.
- Software's and Services are Outdated.
- Passwords are either defaults or very weak.
- Security Controls are either not defined or implemented in most cases.
- All the vulnerabilities found have easy mitigation

In conclusion, these vulnerabilities should not be there in the first place. https://92.3.102.27:4444/ Corporation needs to redefine their Information Security Management Program and rethink their processes.

Recommendations

Due to the impact to the overall organization as uncovered by this penetration test, appropriate actions should be taken to remediate and safeguard your IT infrastructure.

Though mitigation for specific vulnerabilities has already been given in this report, **additionally, we recommend the following:**

1. Establishment of Updates & Patch Management Program
2. Implementation of WAF and IPS
3. Source Code Review of Deployed Applications and Sanitization
4. Alignment of Security Policies with Industry's Best Practices
5. Use a Custom 404 (Not Found Error) Page
6. Social Engineering training for every employee
7. Vulnerability Scanning on at least monthly basis (Scan – Patch – Scan Again)
8. Install a HIPS and DLP to stop common attacking payloads like meter-preter

Additional Items

Appendix A - References:

There are some concepts and special tools I used, to which I have given the links below -

- **Kali Linux -** https://www.kali.org/downloads/
- **Vsftpd Exploit -** rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/
- **Rooting Guide -** blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/

Appendix B - Glossary:

There are some technical terms in the report which are important to be explained here -

- **Black Box Penetration Test -** In penetration testing, black-box testing refers to a method where an ethical hacker has no knowledge of the system being attacked. The goal of a black-box penetration test is to simulate an external hacking. It is the most unreliable form of penetration testing.

- **Social Engineering –** It is the art of using deception to con someone into providing information or access they would not normally have provided. It's the "human side" of breaking into a network and preys on the qualities of human nature, such as the desire to be helpful, the tendency to trust people and the fear of getting in trouble. According to recent statistics, 98% of all cyber-attacks rely on social engineering.

Thank you brother

Please let me know if you have any confusion.