

Lab12_19bce7450

VULNERABILITY REPORT

WEDNESDAY, JUNE 09, 2021



MODIFICATIONS HISTORY

Version	Date	Author	Description
1.0	06/09/2021	slvnyaswanth	Initial Version

TABLE OF CONTENTS

1.	General Information	4
1.1	Scope	4
1.2	Organisation	4
2.	Executive Summary	
		5
3.	Technical Details	6
3.1	title	11
4.	Vulnerabilities summary	6

GENERAL INFORMATION

SCOPE

VIT-AP has mandated us to perform security tests on the following scope:

- The following audit report comprises of the reported vulnerabilities in the following applications:
 - 1)StreamRipper32
 - 2)Frigate
-

ORGANISATION

The testing activities were performed between 05/03/2021 and 05/18/2021.

EXECUTIVE SUMMARY

VULNERABILITIES SUMMARY

Following vulnerabilities have been discovered:

Risk	ID	Vulnerability	Affected Scope
High	IDX-001	Shell Code Injection	Frigate "FIND COMPUTER" input field is vulnerable and on payload execution it triggers Calculator and Control Panel.
Medium	VULN-002	Buffer Overflow	Frigate and StreamRipper32 Both possess buffer overflow vulnerability where the application terminates due to overwriting of adjacent memory locations. Stream ripper 32 does not trigger the shell code while frigate does.

TECHNICAL DETAILS

SHELL CODE INJECTION

CVSS SEVERITY	High	CVSSv3 SCORE	8.0
CVSSv3 CRITERIAS	Attack Vector : Network Attack Complexity : High Required Privileges : Low User Interaction : Required	Scope : Changed Confidentiality : High Integrity : High Availability : High	
AFFECTED SCOPE	Frigate "FIND COMPUTER" input field is vulnerable and on payload execution it triggers Calculator and Control Panel.		

DESCRIPTION	<p>Command injection is an attack in which the goal is execution of arbitrary commands on the host operating system via a vulnerable application. Command injection attacks are possible when an application passes unsafe user supplied data (forms, cookies, HTTP headers etc.) to a system shell. In this attack, the attacker-supplied operating system commands are usually executed with the privileges of the vulnerable application. Command injection attacks are possible largely due to insufficient input validation.</p>
OBSERVATION	<p>When a payload generated by a malicious code has the trigger to any of the host operating system applications and utility, shell code injection attack exists. Attacker tries to gain access to a host OS utility and tries to run it by executing the payload which triggers the particular utility on execution.</p>
TEST DETAILS	

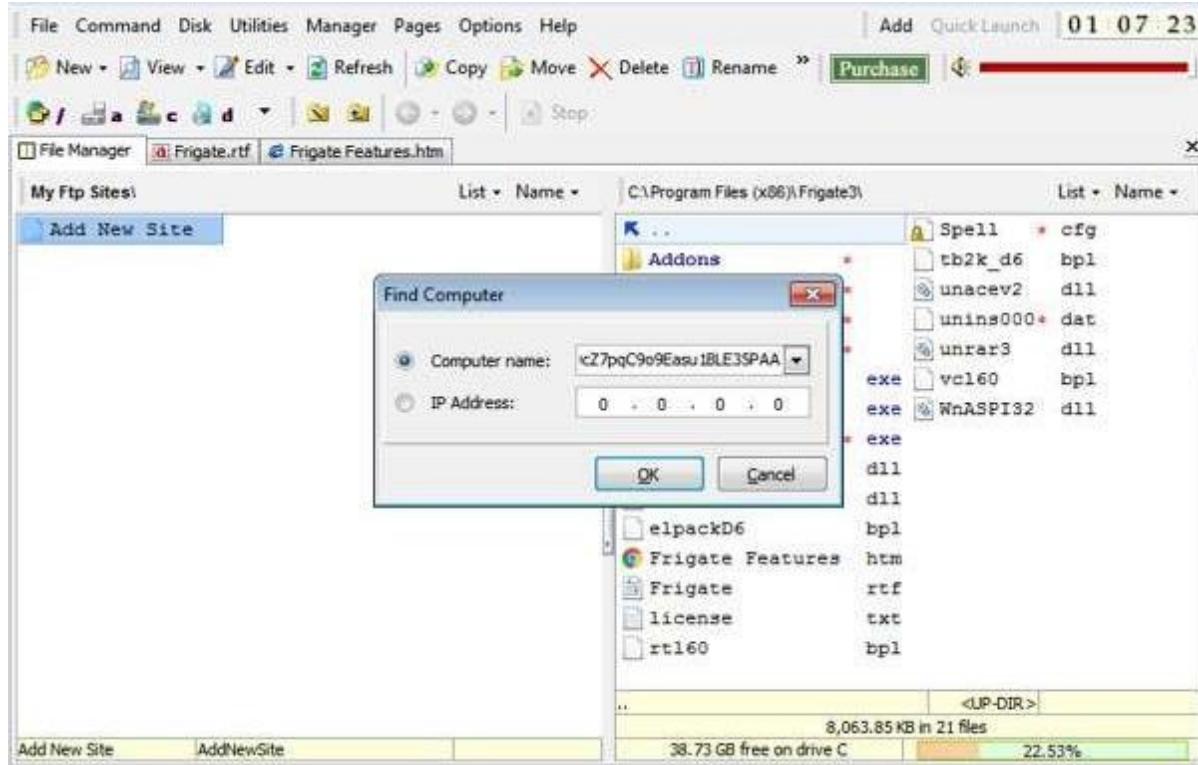


Image 1 – 1cnlab13.png



Image 2 – 2cnlab13.png



Image 3 – 3cnlab13.png

REMEDIATION	1) Installing a proper antivirus to monitor the application and the System. 2) Input validation and sanitation 3) Fixing application bugs by patching the application with the appropriate patch 4) Discard usage of third party applications
-------------	--

REFERENCES

BUFFER OVERFLOW

CVSS SEVERITY	Medium		CVSSv3 SCORE	5.4
CVSSv3 CRITERIAS	Attack Vector : Physical	Scope : Unchanged		
	Attack Complexity : Low	Confidentiality : Low		
	Required Privileges : None	Integrity : Low		
	User Interaction : Required	Availability : High		
AFFECTED SCOPE	Frigate and StreamRipper32 Both possess buffer overflow vulnerability where the application terminates due to overwriting of adjacent memory locations. Stream ripper 32 does not trigger the shell code while frigate does.			
DESCRIPTION	A buffer overflow condition exists when a program attempts to put more data in a buffer than it can hold or when a program attempts to put data in a memory area past a buffer. In this case, a buffer is a sequential section of memory allocated to contain anything from a character string to an array of integers. Writing outside the bounds of a block of allocated memory can corrupt data, crash the program, or cause the execution of malicious code.			
OBSERVATION	When attacker tries to exploit any of the vulnerable input fields in an application by injecting a payload, the allocated adjacent memory portions are overwritten causing it to grant access to the attacker. Attacker can have access control to the system if buffer overflow exists. It happens due to improper input validation and sanitization.			
TEST DETAILS				

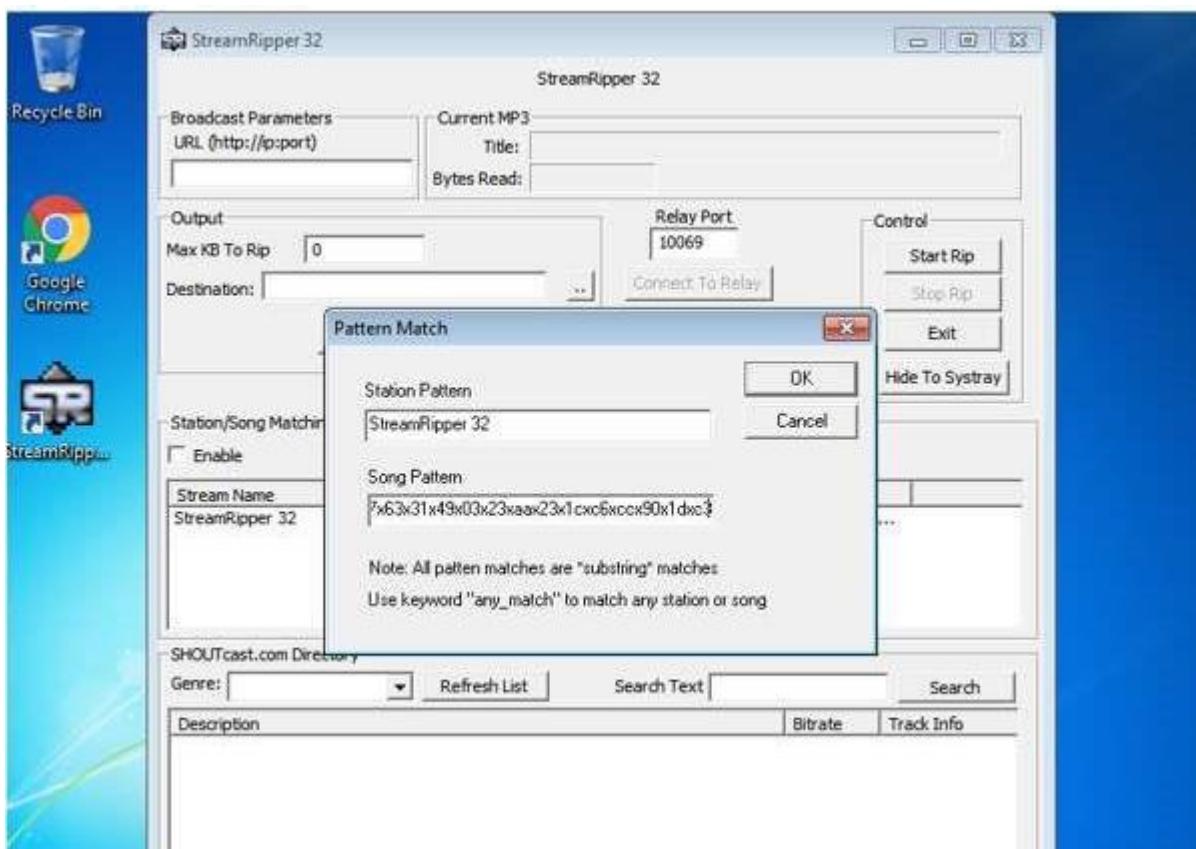


Image 4 – 4cnlab13.png

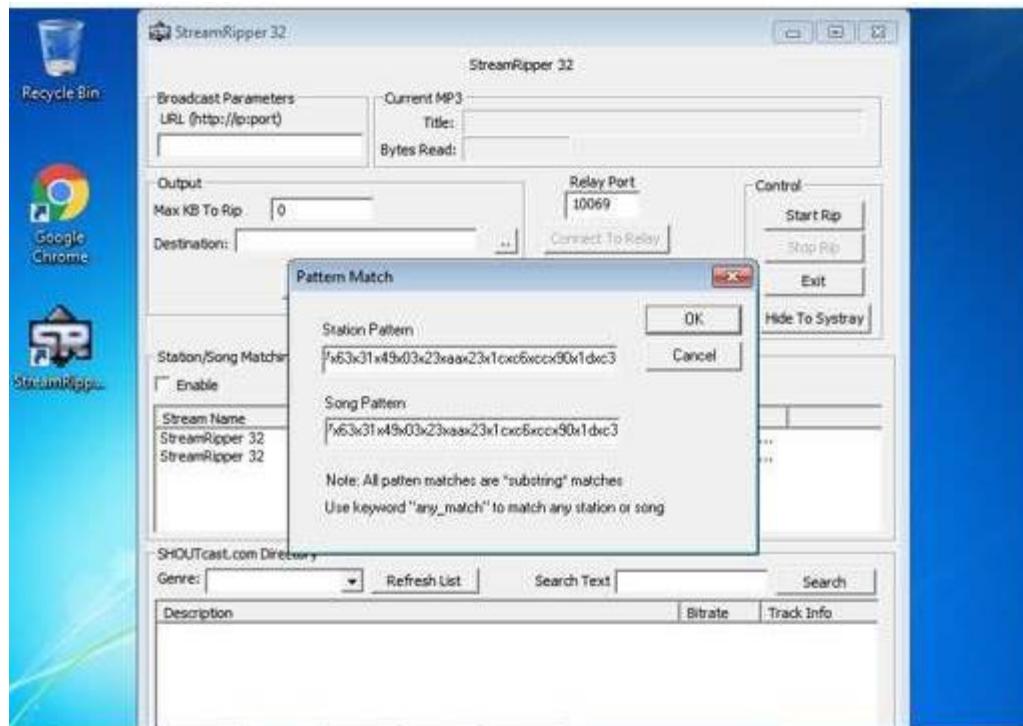


Image 5 – 5cnlab13.png

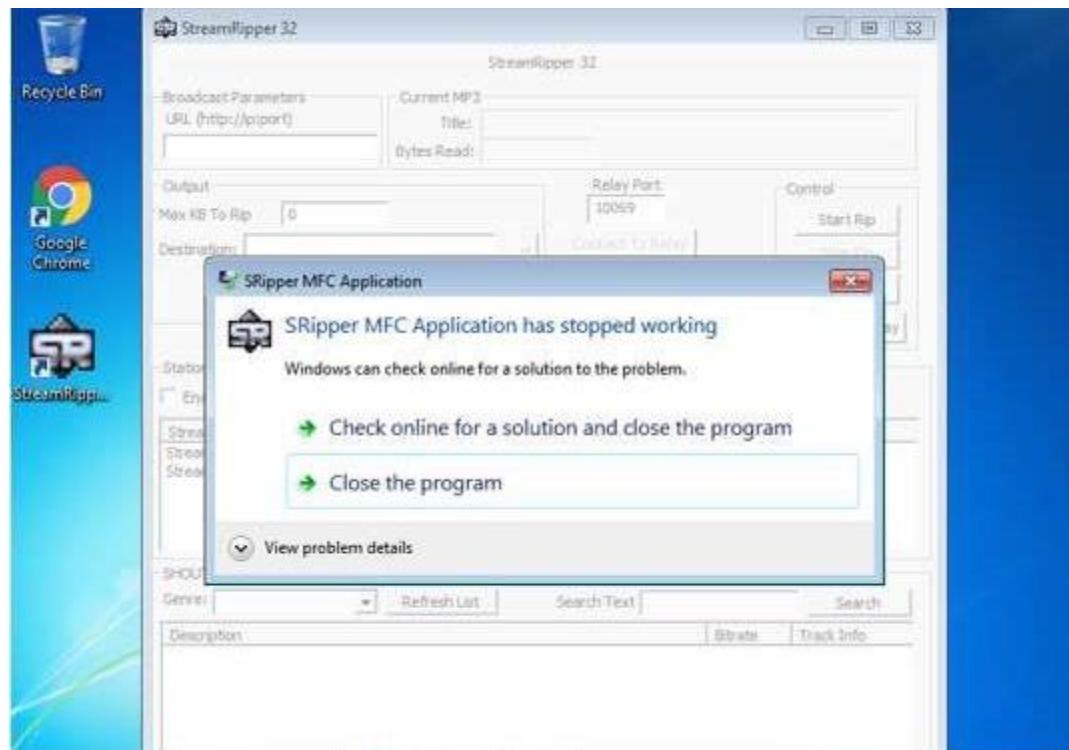


Image 6 – 6cnlab13.png

REMEDIATION	1) Enabling DEP, ASLR AND SEHOP 2) Patching up the applications to the latest versions 3) Input fields sanitation and validation
REFERENCES	