GoodSecurity Penetration Test Report

Sean.James@GoodSecurity.com

February 12, 2021

# High-Level Summary

GoodSecurity was tasked with performing an internal penetration test on GoodCorp’s CEO, Hans Gruber. An internal penetration test is a dedicated attack against internally connected systems. The focus of this test is to perform attacks, similar to those of a hacker and attempt to infiltrate Hans’ computer and determine if it is at risk. GoodSecurity’s overall objective was to exploit any vulnerable software and find the secret recipe file on Hans’ computer, while reporting the findings back to GoodCorp.

When performing the internal penetration test, there were several alarming vulnerabilities that were

identified on Hans’ desktop. When performing the attacks, GoodSecurity was able to gain access to his machine and find the secret recipe file by exploit two programs that had major vulnerabilities. The details of the attack can be found in the ‘Findings’ category.

All information within the summary is to assist GoldCorp on their quest to add layers of additional safety and security within their own environment. The author of this document made informed suggestions for improvements. The author used the phrase *~Not a Doctor~* throughout, suggesting that although the problems he saw were glaring in nature; he was by no certain means prescribing a specific path to resolve!

# Findings

Machine IP:

192.168.0.20

Hostname:

MSEGEWIN10

Vulnerability Exploited:

The Icecast exploit

Vulnerability Explanation:

There are a few vulnerabilities within this machine. The first being the ease at which someone seeking out an exploit can discover the use of Icecast as a method of communication on this machine. Icecast has known vulnerabilities that were easily discovered by using open-source applications. Once the exploit was run, the machine and its command line were accessible. This allowed access to files with sensitive information and PII.

Once inside the environment, I was able to traverse the directory and move around and escalate to different directories. Since Icecast is commonly used for media streaming, it is vulnerable to buffer overflow attacks. This may be allowing someone seeking an exploit to conduct a Man-in-the-Middle (MItM) attack.

This particular IP Address did not require signing in on remote SMB. Additionally, the terminal services does not use Network Level Authentication (NLA). This again is vulnerable to MItM attacks.

The SSL certificate for this cannot be trusted. In fact, it appears to an outsider that the certificate was signed as MSEDGEWIN10, which is the same as the host name. Are you signing your own certificates? We at GoodSecurity would recommend against that.

We at GoodSecurity also noted that the SSL supported medium-strength ciphers. This makes you vulnerable to SWEET32 attacks. This should be enhanced to a stronger encryption. Additionally, the remote service accepts connections that are encrypted using TLS 1.0. We would recommend this be upgraded to a newer, more recent version such as 1.3.

Severity:

The list above offers significant vulnerabilities that would allow for someone seeking out an exploit to gain access to sensitive information, PII, and set themselves up for a MItM attack.

Proof of Concept:

*-To get the IP address of the machine I want to run scans on, I went into the command prompt of that machine and ran:*

*<ipconfig> to get the IP of 192.168.0.20*

*A picture containing text

Description automatically generated*

*To make sure that one was up and reachable, I then went to my Kali machine and ran a ping command:*

*<ping -c 3 192.168.0.20>*

*Text

Description automatically generated*

*That gave me the results of all 3 packets received, with 0% loss.  Although this step is not necessary, I wanted to ensure I was working within the live network.*

*-To run a service scan, the first thing to do is go to nmap on the command line and type:  
<nmap -sV 192.168.0.20>*

*This allowed me to see the ports that are open and specifically, the port that would be associated with Icecast.  This allowed me to see that Icecast was a streaming media server running and open on Port 8000.*

*Text

Description automatically generated*

*From the previous step, we see that the Icecast service is running. I started by attacking that service. I searched for Icecast exploits and then ran the SearchSploit commands to show available Icecast exploits:*

*-To run the searchsploit command to see the Icecast exploits, I ran the following:*

*<searchsploit Icecast>*

*Text

Description automatically generated*

*I then ran the command that starts Metasploit:*

*<msfconsole>*

*I searched for the Icecast module and load it for use.  I ran the command to search for the Icecast module:*

*<search icecast>*

*Text

Description automatically generated*

*Then the command to use the Icecast module:*

*<use 0>*

*I Set the RHOST to the target machine. I ran the command that sets the RHOST:*

*Since we know the host IP, as well as the target port, I used the following:  
  
<set RHOST 192.168.0.20>   <set RPORT 8000>*

*Text

Description automatically generated*

*I ran the Icecast exploit:*

*<exploit>*

*This allowed me to have a Meterpreter session open. I ran the command that performs a search for a file containing the string secretfile on the target, and command to performs a search for a file containing the string recipe on the target:*

*Once gaining access to the machine via meterpreter, I first ran a <?> to see what I had access to.  Then I ran <shell> to gain control of the command line.  Once there, I was able to run some basic commands such as:*

*<pwd>*

*<ipconfig>*

*This allowed me to see where I was at when I entered the machine. Additionally, I was able to discover things like the MAC address which is part of layer 2 within the OSI model. Once I stretched out and took a look around, I realized that I was actually a few layers down within the machine, so I used the following command to have a mapping or route overview:*

*<cd ../../../../>*

*Since I was personally working on a Linux machine, and then remoted into a Windows environment on the host, I had to use appropriate commands to match. I mined down layer by layer, looking at different files with the limited access you granted me. I did not open files or attempt to access information that was forbidden to me from the contract stated. Mapping paths or vulnerabilities often means taking a larger view to understand the paths within data. Since every company is different, with different names and folders, this helps to get an idea of why the files I was asked to seek out or find are named as they are.*

*Once the exploit target was confirmed, I was able to view documents within:*

*<C:\Users\IEUsers\Documents>*

*Once there, I could see the three text files that I had access to. I opened them to view their contents to find:*

*Text

Description automatically generated*

*Text

Description automatically generated with medium confidence*

*The above screenshots are of the vulnerabilities discovered to include passwords and PII. There are a few different paths to get to this data. When first entering the virtual environment, a person could also enter the command:*

*<search -f \*secretfile\* -d /Users>*

*This revealed the information needed as well as the path to locate them as demonstrated below.*

*Text

Description automatically generated*

*I was able to find the logged in users by using the following command while in meterpreter:*

*<run post/windows/gather/enum\_logged\_on\_users>*

*Text

Description automatically generated*

*Additionally, once within the system, I entered into a PowerShell to use the windows command line easily. I was able to use the Windows command of:*

*<SYSTEMINFO>*

*This gave me critical information about the machine to include the host name, operating system and version, as well as the install date. This shows a period of almost (2) years since the system has been brought up to current. Although there may be patches in place, this is often a sought-after vulnerability for those who seek to exploit systems.*

*Text

Description automatically generated*

# Recommendations

Icecast is a vulnerable streaming media software that leaves your operating system vulnerable. We at GoodSecurity recommend that you either stop using such an application, or that you ensure the latest updates to your system and firewalls are in place.

Additionally, we at GoodSecurity would recommend that personal data and PII should be kept out of text files when possible. If there is a need to keep such data in a file, it would be best to password protect or encrypt that file.

We also recommend that you separate out specific servers for specific purposes. When entering an environment with bad intentions, it is much easier to create attacks with access to multiple servers that have open communication.

For additional reference, please see the attached Nessus Executive Summary that will reference the details of vulnerabilities above.

Graphical user interface, table

Description automatically generated

*Screenshot only to represent full report\*\*\**