linux kernel pwn系列(一)

原创

obfuscation123 于 2018-09-26 16:17:21 发布 2523 收藏 7

kernel pwn题目不准备以总结的方式来写，准备每一道题做一个专门的分析。网上有不少exp，准备在他人的exp的基础上进行复现(因为能力不够所以在比赛时候做不出来，只能通过事后复现)计划写
强网杯core 和 solid_core，两道改编了csaw 2010kernel pwn的题目以及ciscn babydriver。总共三道题，入门kernel pwn绰绰有余。

第一篇主要介绍linux kernel pwn前置的准备知识，主要包括使用qemu起系统，并附加远程gdb调试。

如何起系统？

ctf的kernel pwn中一般会给出qemu起系统的脚本 随便举一例

qemu-system-x86_64
-m 256M
-kernel ./bzImage
-initrd ./initrd.cpio
-append “root=/dev/ram rw console=ttyS0 oops=panic panic=1 kaslr”
-cpu qemu64,+smep,+smap
-netdev user,id=t0, -device e1000,netdev=t0,id=nic0
-s
-nographic -enable-kvm \

比较重要的是qemu-system-x86制定处理器体系，-m指定内存，
-s选项默认指定 开启更gdb远程调试端口 1234

比较常见的如何解包？

```
$ mkdir core
$ mv core.cpio ./core/core.cpio.gz
$ cd core
$ gunzip core.cpio.gz
$ cpio -idmv < core.cpio
```

这个时候，将exp放入系统的一个目录中

```
$ nano init
```

这条命令用于编辑init，一般用于删除定时关机

```
$find . | cpio -o -H newc | gzip > ../core.cpio
```

用于重新打包

也可以用这几条：

$ ./gen_cpio.sh core.cpio

$ mv core.cpio …/core.cpio

$ cd …

$ rm -rf core

gdb调试的时候遇到一些问题：

```
target remote:1234
```

这个时候如果返回一堆字符显示过长，

```
set architecture i386:x86-64:intel
```

使用这条命令设置架构

在运行时载入模块的符号表：

```
grep 0 /sys/module/your_module/sections/.text

add-symbol-file ./your_module.ko text
```

其中your_module是你要加载的模块，text为第一行命令的返回值。

如何得到kernel rop

如果有ELF形式的vmlinux映像可以直接用ROPgadgets，但更多时候我们只有bzImage，这个时候需要用extract-vmlinux进行提权，它在内核源码的scripts中，搜索自己linux系统的内核源码就能找到。

今天算是开坑了第一篇kernel pwn系列，下一篇准备介绍一个简单的实例：强网杯 core，极其类似于csaw2010的kernel pwn。