命令执行漏洞简析

Nullam eu tempor purus. Nunc a leo magna, sit amet consequat risus.

2018.8.26

个人简介

刘懿莹，2016级计算机科学与技术在读，东北大学信息安全实验室成员，在校期间曾获2018年美国大学生数学竞赛三等奖和2017年全国大学生英语竞赛等奖项，目前主要的研究方向是工控安全。

CONTENTS

漏洞简介

Nullam eu tempor purus. Nunc a leo magna, sit amet consequat risus.

命令执行的定义

当WEB应用需要调用一些外部程序去处理内容的情况下，会用到一些执行系统命令的函数。当用户可以控制命令执行函数中的参数时，可以注入恶意系统命令到正常的命令中，从而造成命令执行攻击。

在Windows中，"&&"的作用是将两条命令连接执行，在Linux系统下同样适用
在DVWA提供的命令执行漏洞测试模块下，输入"www.xxser.com&&Command"
系统将会执行Command

漏洞原理

用户通过浏览器提交执行命令，由于服务器端没有针对执行函数做过滤，导致在没有指定绝对路径的情况下就执行命令.

PHP可调用外部程序的常见函数：system,exec,shell_exec,passthru,popen,proc_popen

system() 输出并返回最后一行shell结果。

exec() 不输出结果，返回最后一行shell结果，所有结果可以保存到一个返回的数组里面。

passthru() 只调用命令，把命令的运行结果原样地直接输出到标准输出设备上。

popen()、proc_open() 不会直接返回执行结果，而是返回一个文件指针

利用条件

应用调用执行系统命令的函数

将用户输入作为系统命令的参数拼接到了命令行中

没有对用户输入进行过滤或过滤不严

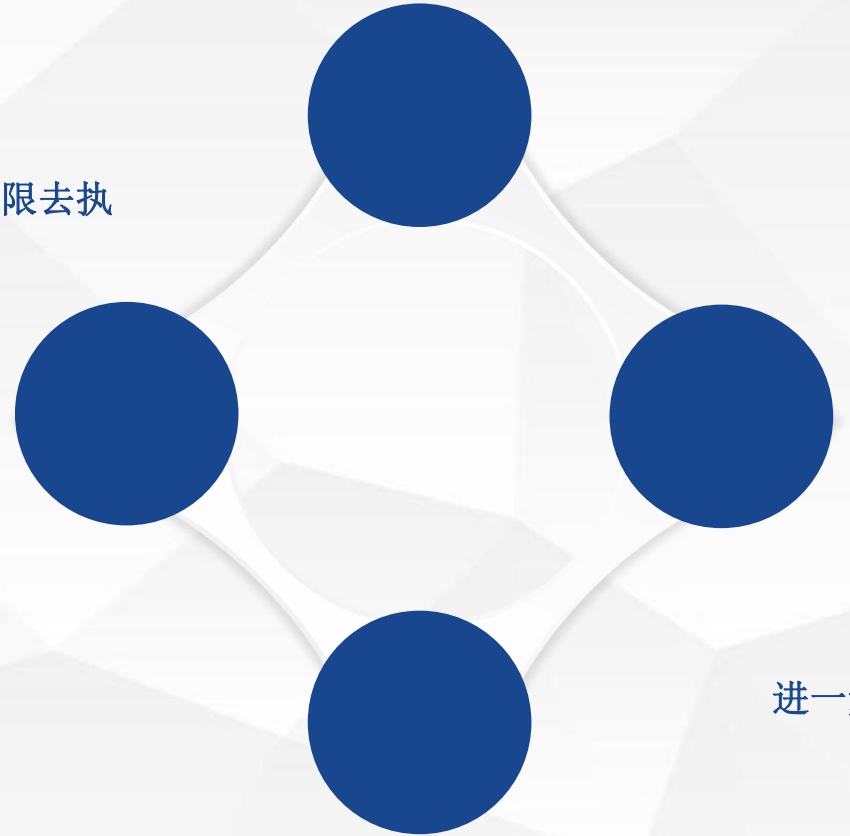
危害

继承Web服务程序的权限去执行系统命令

读取敏感信息

篡改数据

进一步内网渗透



02

分类

三个类别

代码层过滤不严

商业应用的一些核心代码封装在二进制文件中，在web应用中通过system函数来调用：system("/bin/program --arg $arg");

系统的漏洞造成命令注入

bash破壳漏洞(CVE-2014-6271)
http://www.freebuf.com/articles/system/50065.html

调用的第三方组件存在代码执行漏洞

如WordPress中用来处理图片的ImageMagick组件
JAVA中的命令执行漏洞
(struts2/ElasticsearchGroovy等)
https://www.anquanke.com/post/id/83872

03

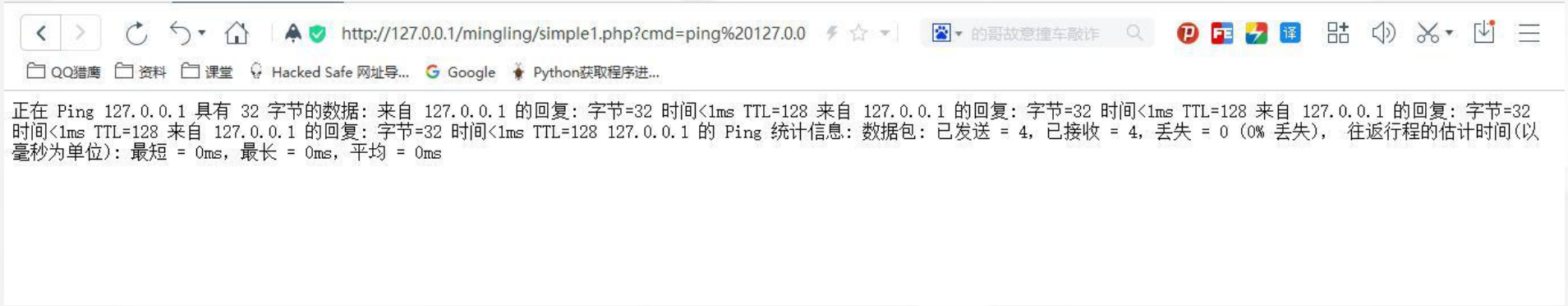
简单示例

示例一

php代码

```
<?php
   $arg = $_GET['cmd'];
   if ($arg) {
      system("$arg");
   }
?>
```

访问：
http://127.0.0.1/mingling/simple1.php?cmd=ping%20127.0.0.1
也就是执行了“ping 127.0.0.1”命令

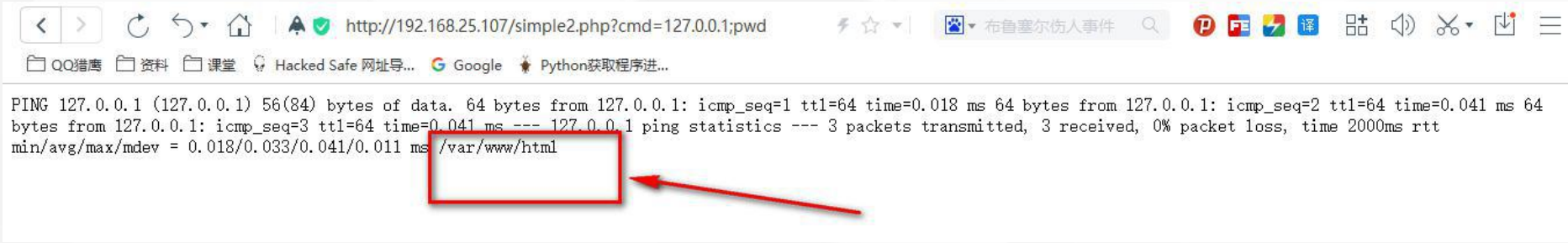


示例二

php代码

```
<?php
   $arg = $_GET['cmd'];
   if ($arg) {
      system("ping -c 3 $arg");
   }
?>
```

访问：
http://192.168.25.107/simple2.php?cmd=127.0.0.1;pwd
执行了pwd命令



示例三

php代码

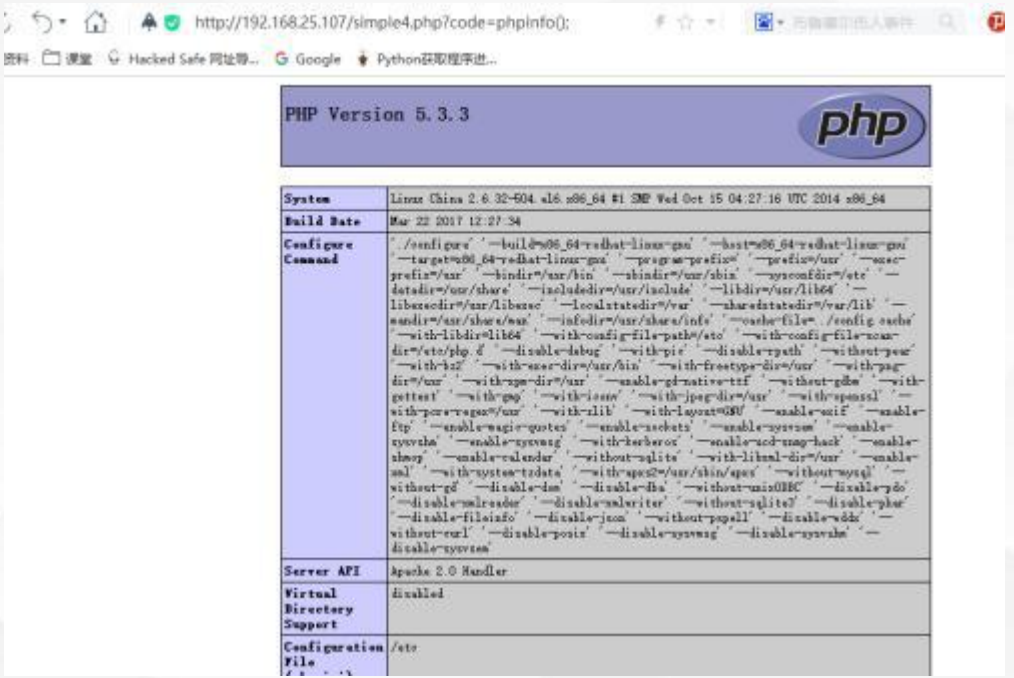
```
<?php
   $arg = $_GET['cmd'];

   if ($arg) {
      system("ls -al \"$arg\"");
   }

?>
```

访问：
http://192.168.25.107/simple3.php?cmd=/home%22;ifconfig%22
执行了ifconfig命令

示例四

php代码

```
<?php
eval($_REQUEST
['code']);
?>
```

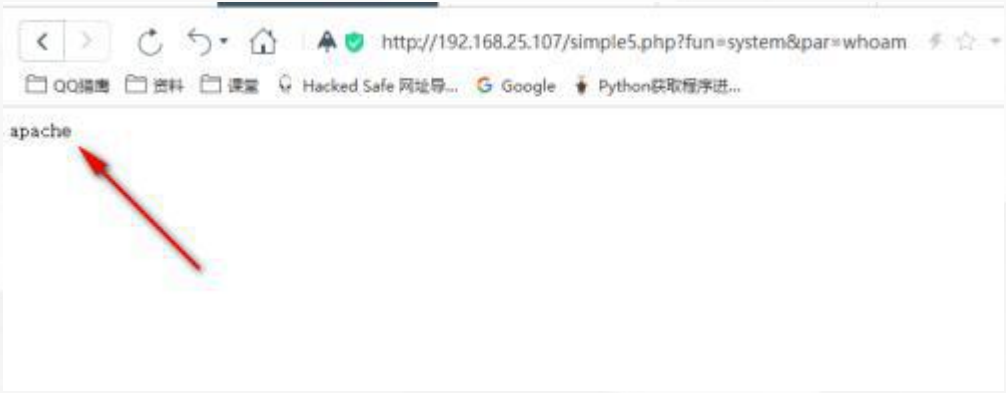


示例五 动态函数调用

php代码

```
<?php
   $fun = $_GET['fun'];
   $par = $_GET['par'];
   $fun($par);
?>
```

当访问：
http://192.168.25.107/simple5.php?fun=system&par=whoami
相当于执行了：
system（"whoami"）



04

防御方法

两个函数

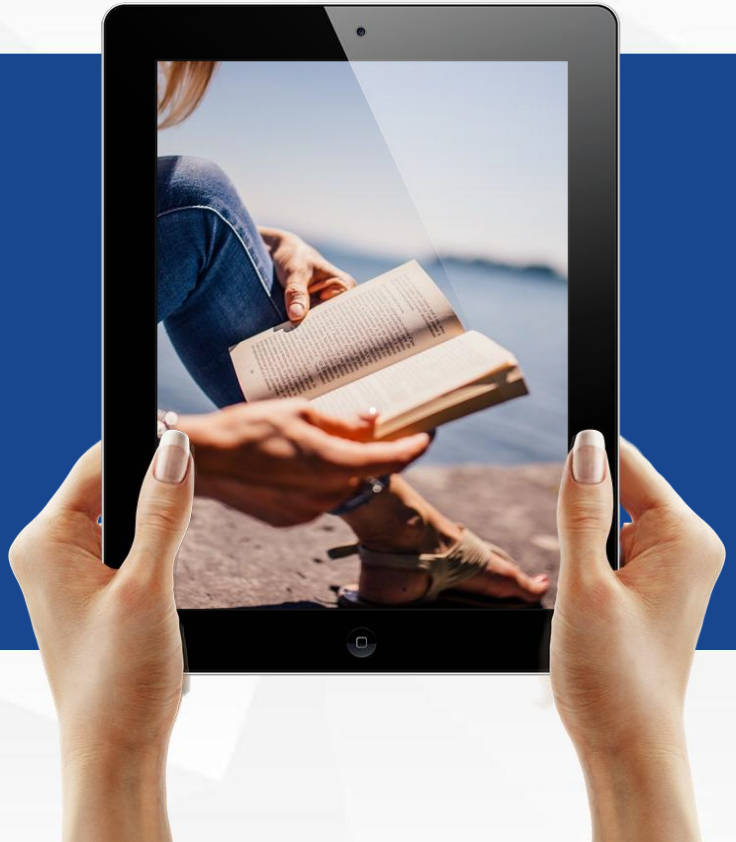
php的两个内置函数可以有效防止命令执行

escapeshellarg()

将给字符串增加一个单引号并且能引用或者转码任何已经存在的单引号，这样以确保能够直接将一个字符串传入 shell 函数，并且还是确保安全的。

escapeshellcmd()

对字符串中可能会欺骗 shell 命令执行任意命令的字符进行转义。 此函数保证用户输入的数据在传送到 exec() 或 system() 函数，或者 执行操作符 之前进行转义。



防御方法

02

尽量少用执行命令的函数或者直接禁用

04

在进入执行命令的函数/方法之前，对参数进行过滤，对敏感字符进行转义

01

参数值尽量使用引号包括

03

在使用动态函数之前，确保使用的函数是指定的函数之一

NeuqSec

演示完毕，谢谢观看

Nullam eu tempor purus. Nunc a leo magna, sit amet consequat risus.

by 刘懿莹