بسمه‌تعالی

شبکه پنهان: تشخیص شبکه‌های پنهان ایجادشده توسط دستگاه‌های USB

Hidden Network:

Detecting Hidden Networks created with USB Devices

خلاصه جامع

امروزه بسیاری از شرکت‌ها و سازمان‌های دولتی ارتباطات مجزایی دارند و با شبکه‌های مختلفی در ارتباط هستند. این شبکه‌های کامپیوتری برای موقعیت‌های خاصی ایجاد می‌شوند و قادرند ویژه باشند یا شامل اطلاعات مهمی مانند سیستم کنترل کارخانه، محیط امن جهت پردازش داده‌های خاص و یا شبکه‌هایی که با استاندارد ایمنی مطابقت دارند، باشند. طبق گذشته امنیت سایبری, مشاهده شده که یک نرم‌افزار مخرب مانند استاکس‌نت[1] به شبکه کامپیوتری یک نیروگاه اتمی نفوذ کرده است. براساس این واقعیت می‌توان مشاهده کرد که داشتن شبکه‌های کامپیوتری که از طریق کابل یا وای‌فای به اینترنت متصل نباشند، به تنهایی کافی نیست و هر اتصال خارجی دیگری نیز ممکن است برای کامپیوترها، تهدیدی ایجاد کند. این مقاله نشان دهنده امکانات ارائه شده از سوی به‌اصطلاح "شبکه پنهان"[2] می‌باشد و همچنین چگونگی شناسایی و محافظت از آن را در داخل یک شبکه سازمانی، نشان می‌دهد.

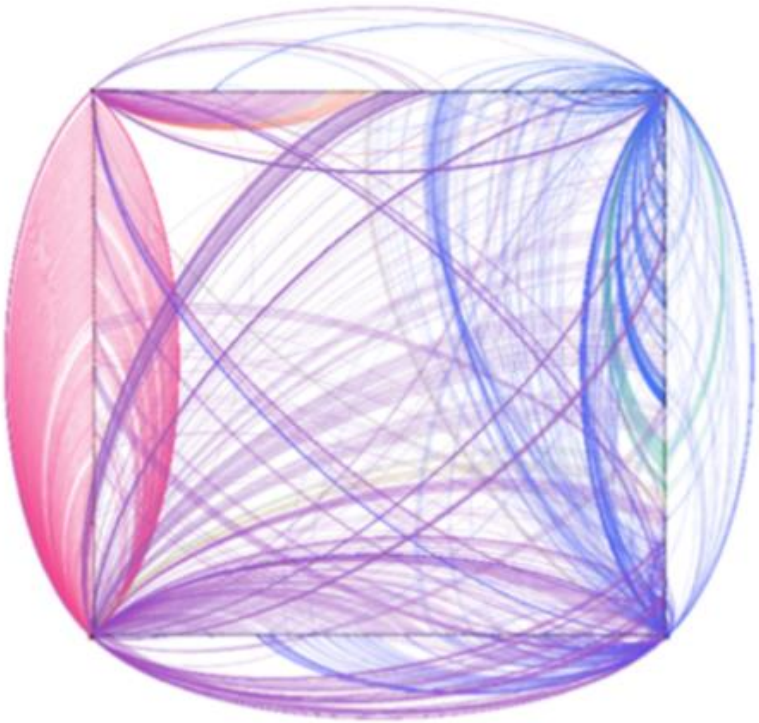
1. خطرهای ناشی از اتصالات

با توجه به امنیت شبکه‌های داده، روند طراحی شبکه‌ها و اتصالات آنها در سطح لینک، مانند اترنت، اتصالات WiFi و شبکه‌های دیگر، بسیار پیچیده هست و نیاز به تجزیه و تحلیل از جنبه‌های متفاوتی دارد.

تجزیه و تحلیل ترافیک در شبکه‌های سازمانی، ابزار اصلی موجود جهت درک آنچه که در شبکه در حال انجام است، می‌باشد. همچنین یک روش موثر جهت تعیین گره‌هایی از شبکه که بیشترین استفاده را از پروتکل‌ها می‌کنند، هست و این‌که کدام از آنها با بیشترین سرویس‌های ضروری سازگاری دارند یا کدام از آنها به عنوان تنگنا، عمل می‌کنند.

تهیه نقشه از شبکه بهترین روش جهت درک گره‌ها و تنظیمات شبکه است، بنابراین یک تجزیه و تحلیل خوب می‌تواند شبکه کامپیوتری را در برابر خطرات محافظت کند. با توجه به این موارد، می‌توان سطح بالاتری را برای درک شبکه و تهدیدات آن به دست آورد.

[1] Stuxnet
[2] Hidden Network



شکل۱: نقشه شبیه سازی شده شبکه به همراه گره ها و اتصالات مختلف بین آنها

ساختارهای ارتباطی گره‌ها در شناسایی مرزهای شبکه، به‌منظور کاهش نفوذ, شناسایی حملات و یا اقدامات امنیتی پیشگیرانه، نقش حیاتی دارد.

مشکل بیشتر این است که درک کنیم این یک شبکه است. در بسیاری موارد، یک شبکه به عنوان یک گروه از کامپیوترهای متصل به هم تعریف شده است که با فناوری‌ها و پروتکل‌های مختلفی با یکدیگر در ارتباط می‌باشند. در بیشتر موارد کاربران و ادمین‌های سیستم یا شبکه براین باور هستند که داشتن اتصال به شبکه از طریق کابل اترنت یا وای‌فای یک ویژگی خوب در وسایل دیجیتال می‌باشد. این مورد در اینجا مورد بحث نیست, همینطور که ممکن است یک سازمان اقدامات پیشگیرانه مربوط به استفاده از دستگاه‌های USB را اجرا نکند, ممکن است آنچه را که به عنوان "شبکه پنهان" شناخته می‌شود، اجرا کند. این شبکه‌ها از طریق استفاده از دستگاه‌های USB ایجادشده و اجازه برقراری ارتباط بین کامپیوترهای مختلف را می‌دهند.

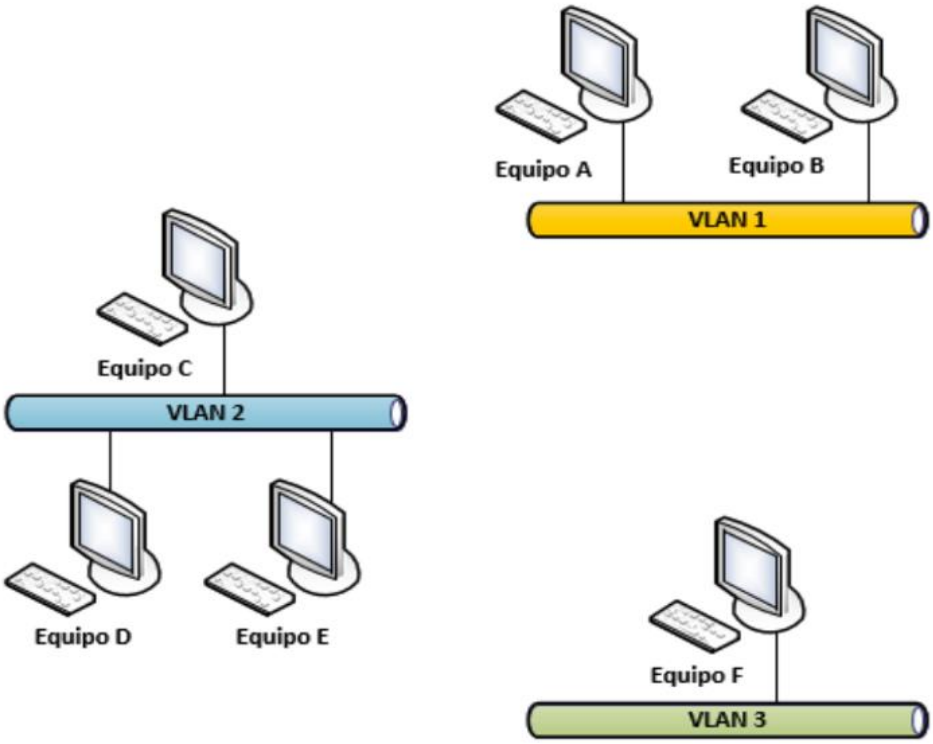
2. جداسازی شبکه و اتصال USB

برای درک خطرهای ناشی از شبکه‌های پنهان که توسط دستگاه‌های USB ایجادشده‌اند به یک مثال توجه کنید. فرض کنید در یک سازمانی سه نوع VLAN داریم:

در VLAN اول کامپیوترهای A, B قرار دارند که با هم در ارتباط می‌باشند.

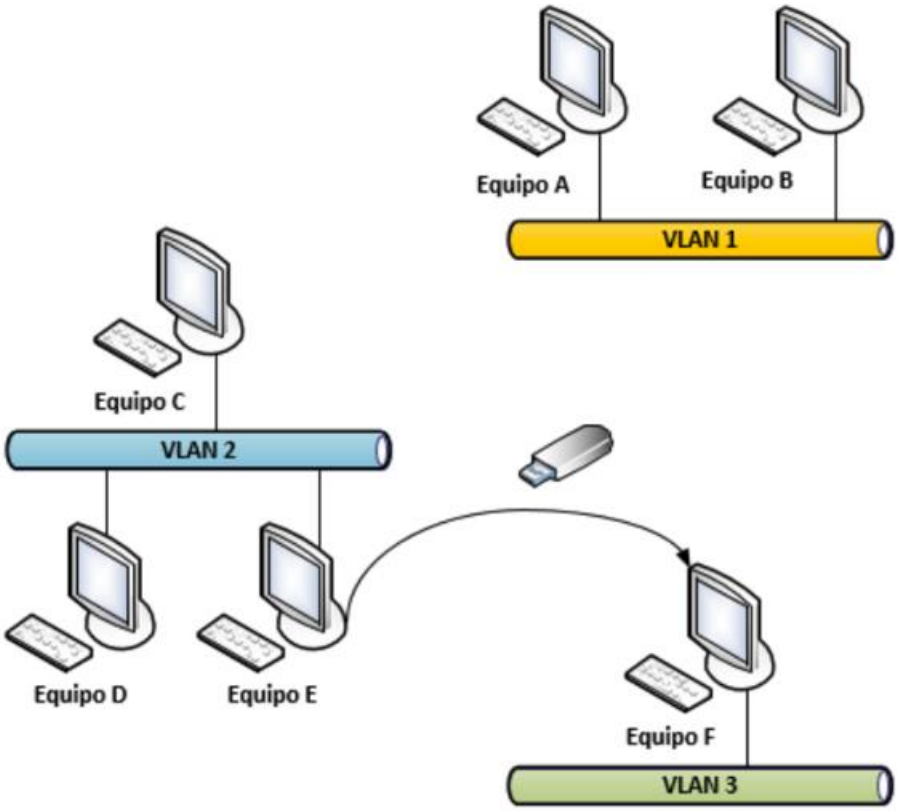
در VLAN دوم کامپیوترهای C, D, E قرار دارند که با هم در ارتباط می‌باشند.

در VLAN سوم کامپیوتر F قرار دارد.



شکل ۲: شبکه خارجی بوسیله کامپیوترهای متصل به هم در شبکه های مختلف

فرض‌کنید کاربران کامپیوترهای E, F درحال انتقال اطلاعات از طریق USB می‌باشند. یک شبکه مخفی بین این دو کامپیوتر ایجاد می‌شود. این اطلاعات را می‌توان بوسیله دو گره E, F نشان‌داد و یک قوس بین دو نقطه از کامپیوترهای E, F پیش‌بینی شده است.



شکل ۳: هم پوشانی شبکه پنهان در شبکه خارجی

3. اتصالات USB در سیستم‌های کامپیوتری

هنگامی که یک دستگاه USB از یک رایانه به رایانه دیگر متصل می‌شود, واژه گردافشانی پدیدار می‌شود. این مفهوم شبیه به آن است که در جاهای دیگر استفاده می‌شود و مربوط به خطرهای ناشی از یک دستگاه USB در میان کامپیوترهای مختلف، حتی زمانی که به شبکه‌های مختلف متصل هستند, می‌باشد.

هنگامی که کاربر یک دستگاه USB را متصل می‌کند، یک سری از ورودی‌ها در سیستم رجیستری ویندوز ایجاد می‌شود. این اطلاعات ارزشمند می‌باشد, برای مثال در یک تجزیه و تحلیل قانونی جهت اینکه بدانند این اطلاعات از کجا آمده است و یا تهدید واردشده در کجا موثر است, مورد استفاده می‌باشد.

کلید USBStor ایجادشده در رجیستری ویندوز اطلاعات مربوط به دستگاه‌های مختلف متصل‌شده به کامپیوتر را ذخیره می‌کند و شامل تمام اطلاعات مورد نیاز جهت شناسایی دستگاه‌های مربوطه می‌باشد.

شکل ٤ :بخش مربوط به دستگاهای متصل شده در رجیستری ویندوز

اطلاعات زیر از طریق دستگاه‌های USB متصل‌شده به یک کامپیوتر قابل شناسایی می‌باشد:

- نام دستگاه
- کلاس
- راهنمای کلاس[3]
- شناسه سخت‌افزار[۴]
- خدمات ارائه‌شده توسط دستگاه به عنوان مثال هارددیسک
- درایور

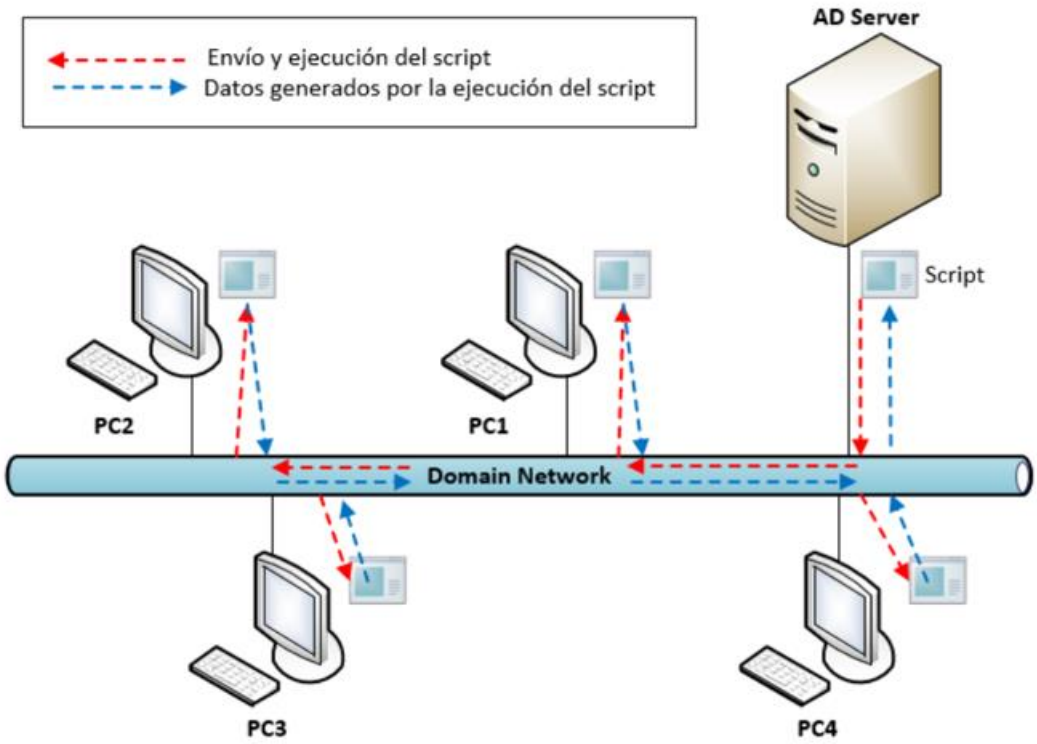
4. اتصالات پنهان: تشخیص این نوع از شبکه‌ها

با دانستن این‌که کجا و چگونه اطلاعات یک دستگاه USB در سیستم‌عامل مایکروسافت ذخیره شده است، می‌توان دانست که چه کسی دستگاه USB را با چه کسی به اشتراک می‌گذارد. به این ترتیب، ما می‌توانیم دو گره را که دو کامپیوتر تشکیل می‌دهند، تولید کنیم و قوس ارتباط بین آن دو کامپیوتر را نشان می‌دهد. یک شبکه پنهان به خاطر پیوند پنهان، تشخیص داده شده است. علاوه بر این به عنوان نتیجه‌ای از تعداد زیادی رویداد

[3] https://technet.microsoft.com/en-us/library/cc957340.aspx
[4] https://docs.microsoft.com/en-us/windows-hardware/drivers/install/hardware-ids

که از سیستم‌عامل بدست می‌آید, می‌توان تشخیص داد که کدام کامپیوتر از قبل به این شبکه متصل بوده است. این چگونگی هدایت قوس بین گره‌ها می‌باشد.

برای انجام خودکار تشخیص اتصالات پنهان, برنامه زیر مطرح شده است:



شکل ۵: نمودار انتشار کد در اکتیودایرکتری

در تصویر بالا می‌توان مشاهده کرد که چگونه برنامه در یک گره مرکزی اجرا می‌شود و چگونه چندین تکنولوژی مایکروسافت جهت اجرای دستورات در هر یک از کامپیوترهای موجود در دامنه، مورد استفاده قرار می‌گیرند. فناوری‌های مورد نظر که متناسب با این راه حل می‌باشد به شرح بعدی است:

- WinRM[5]

[5] https://msdn.microsoft.com/en-us/library/aa384426(v=vs.85).aspx

- SMB[6]
- WMI[7]

Powershell یک خط فرمان شی‌گرا از مایکروسافت است و یک برنامه ساده و قدرتمند می‌باشد که درون ساختار سیستم‌عامل مایکروسافت وجود دارد.

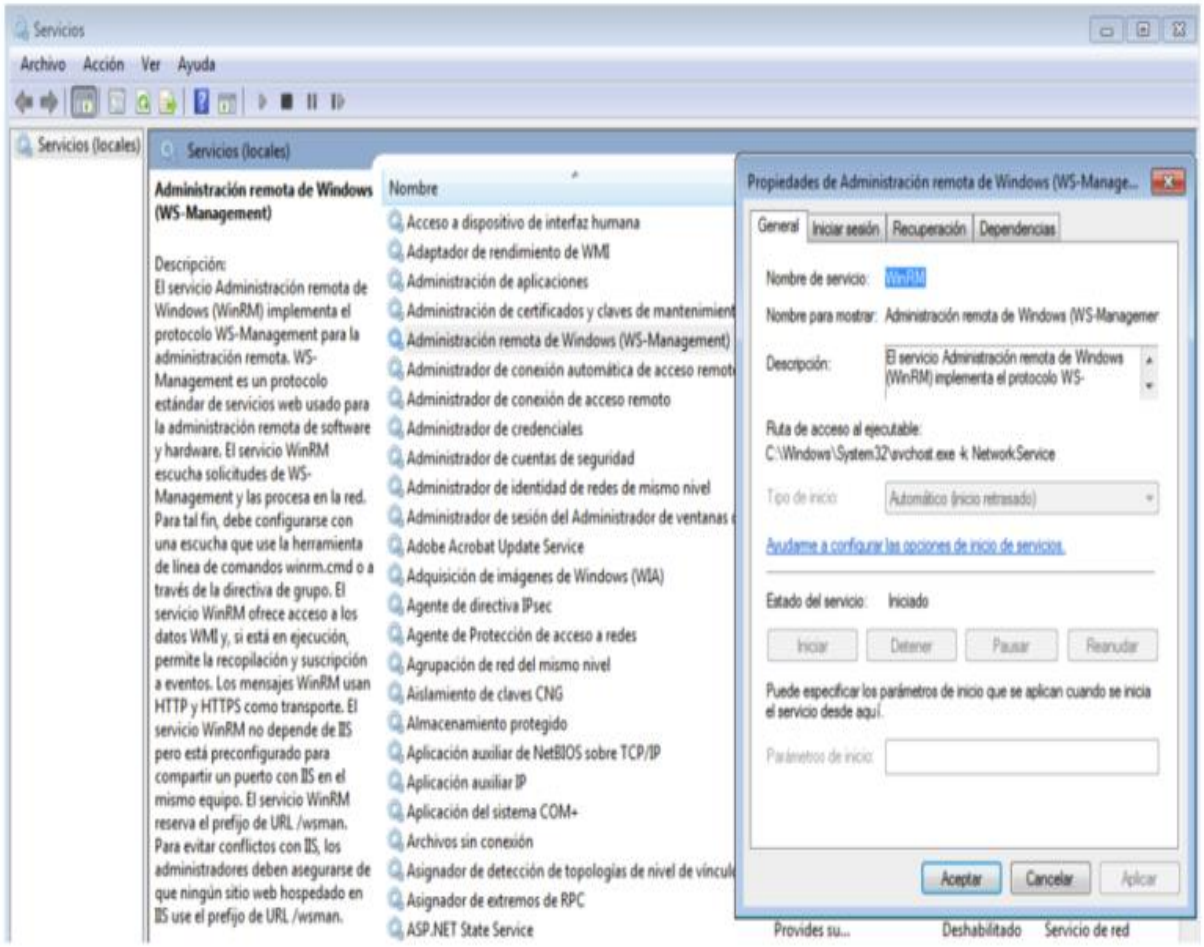
```
PS C:\> Get-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Enum\USBSTOR\*\*
        | Select FriendlyName
FriendlyName
------------
SanDisk U3 Cruzer Micro USB Device
WD Virtual CD 1110 USB Device
USB DISK 2.0 USB Device
USB DISK 2.0 USB Device
ADATA USB Flash Drive USB Device
Corsair Voyager USB Device
FLASH Drive AU_USB20 USB Device
hp USB Flash Drive USB Device
Kingston DT 101 G2 USB Device
Kingston DT 101 G2 USB Device
SanDisk U3 Cruzer Micro USB Device
USB Flash Disk USB Device
WD 3200BEV External USB Device
WD My Book 1110 USB Device
WDC WD25 00JB-00GVA0 USB Device
```

شکل ٦ :مجموعه ای از دستگاهای متصل شده در پاورشل

5. شبکه‌های مخفی USB در WinRM

کد مخصوص WinRM در PowerShell نیاز به فعال‌سازی سرویس Windows Remote Management (WinRM) در کامپیوترهای موجود در شبکه دارد:

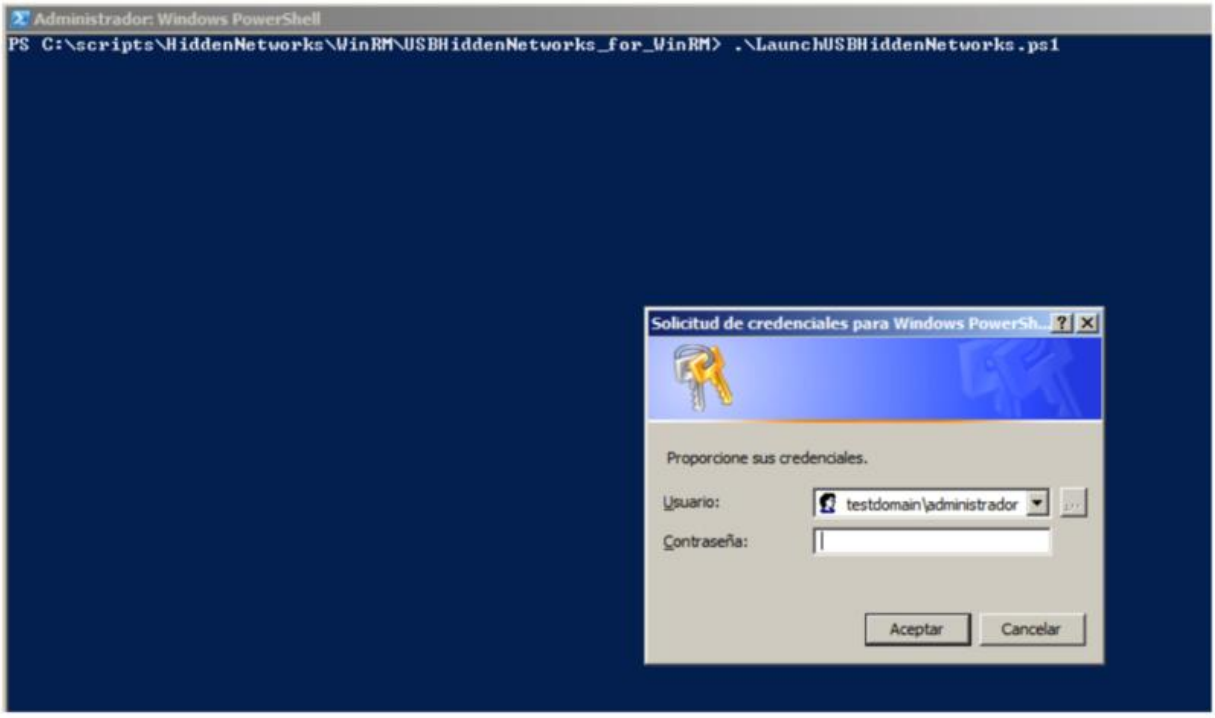
[6] https://msdn.microsoft.com/en-us/library/windows/desktop/aa365233(v=vs.85).aspx
[7] https://msdn.microsoft.com/en-us/library/aa394582(v=vs.85).aspx



شکل ۷: سرویس WinRM

علاوه بر این، کد در یک شبکه تک دامنه با یک مسیر فعال (AD) مورد آزمایش قرار گرفته است تا مجموعه اطلاعات را به صورت اتوماتیک تا آنجا که ممکن است، جمع‌آوری کند. دامنه اعتبارهای مدیریتی[8] برای تأیید اجرا در رایانه‌های راه‌دور در شبکه محلی مورد استفاده قرار می‌گیرند. هنگام اجرای کد، مجوز لازم است.

[8] Domain administrator credentials



شکل ۸ : مجوزهای مربوط به استفاده از کد

اجرای اولیه کد از طریق برنامه "LaunchUSBHiddenNetworks.ps1" انجام می‌شود, که کامپیوترهای راه‌دور را با استفاده از کد "RecollectUSB.ps1" به هم متصل می‌کند, که به عنوان پارامتر برای جمع‌آوری اطلاعات از دستگاه‌های USB منتقل می‌شود. بنابراین، اسکریپت باید به صورت جداگانه در هر کدام از رایانه‌های اختصاص‌داده‌شده، اجرا شود.

6. کد: LaunchUSBHiddenNetworks

اجرای این دستور براساس دستور PowerShell Invoke می‌باشد این فرمان اجازه می‌دهد تا به یک رایانه در شبکه، که آدرس IP یا نام کامپیوتر را به عنوان پارامتر عبور می‌دهد، متصل شوید و از سوی دیگر، اسکریپت PoweShell باید اجرا شود:

```
$salida=invoke-command -ComputerName (Get-Content servers.txt) -FilePath
'PathToScript\RecollectUSBData.ps1'-Credential testdomain\administrador
```

با پارامتر ComputerName، نام رایانه(ها) مورد بررسی قرارگرفته در داخل AD اختصاص داده می‌شود. امکان نام‌گذاری کامپیوترها به طور مستقیم بعد از کاما وجود دارد، اما در این مورد، یک فایل TXT (server.txt) با لیستی از رایانه‌ها مورد استفاده به عنوان پارامتر قرار گرفته‌اند.

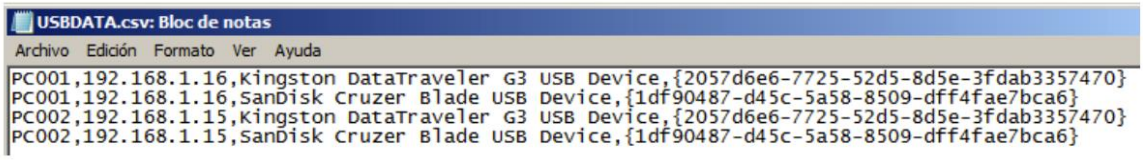
پارامتر FilePath مسیر اسکریپت PowerShell را برای عمل جمع‌آوری داده‌ها، تعیین می‌کند. در نهایت، پارامتر Credential اجازه استفاده از مجوز سرپرست دامنه برای تایید عمل کامپیوتر راه‌دور، دامنه testdomain و کاربر "administrator" می‌باشد.

نتیجه اجرا در شی $salida ذخیره می‌شود. اطلاعات بازیابی‌شده به همین ترتیب در یک فایل CSV به نام "USBDATA.csv" به صورت زیر ذخیره می‌شود:

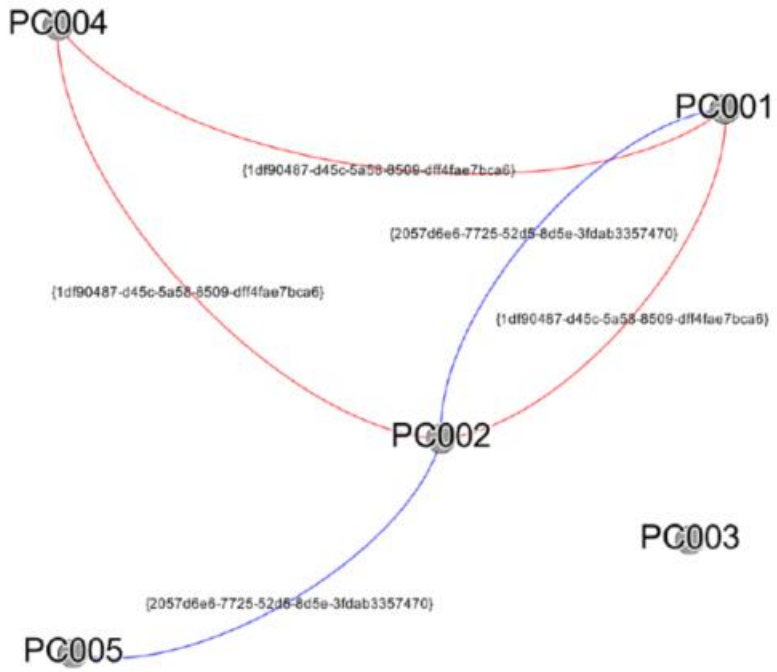
```
$salida | Out-File USBDATA.csv
```

قالب‌بندی فایل CSV پس از اجرای اسکریپت از ساختار زیر استفاده می‌کند:

شناسه(ID)/ نام USB/ آدرس IP/ نام کامپیوتر



با استفاده از این اطلاعات می‌توان گرافی از وضعیت شبکه به شکل بعدی رسم کرد:



شکل ۹ : گراف اتصالات مخفی بین دستگاهای متصل در شبکه را نشان میدهد

7. کد: RecollectUSBData

این اسکریپت مسئول جمع‌آوری تمامی اطلاعات مربوط به دستگاه‌های USB متصل به رایانه است و آن به طور محلی در رایانه‌های مورد استفاده برای حسابرسی اجرا می‌شود. داده از شاخه خاصی از رجیستری ویندوز، بازیابی می‌شود.

```
$USBDevices = @()
$USBContainerID = @()
$USBComputerName = @()
$USBComputerIP = @()
$SubKeys2        = @()
$USBSTORSubKeys1 = @()
```

جایی که اطلاعات مربوط به رایانه‌های مورد بررسی، ذخیره می‌شود و برای داده‌ای که اشاره به دستگاه‌های USB دارد در رجیستری ویندوز است.

```
$Hive    = "LocalMachine"
$Key     = "SYSTEM\CurrentControlSet\Enum\USBSTOR"
```

که $Hive و $Key مسیر کامل شاخه رجیستریی را ذخیره می‌کنند که در آن، جستجوی داده مربوط به دستگاه‌های USB در حال انجام است. متغیر $Hive با مقدار LocalMachine برابر با HKLM یا HKEY_LOCAL_MACHINE است.

```
$ComputerName = $Env:COMPUTERNAME
$ComputerIP = $localIpAddress=((ipconfig | findstr [0-9].\.)[0]).Split()[-1]
```

نام کامپیوتر محلی و همچنین آدرس IP آن در متغیرهای $ComputerName و $ComputerIP ذخیره شده است.

```
$Reg = [Microsoft.Win32.RegistryKey]::OpenRemoteBaseKey($Hive,$Computer)
$USBSTORKey = $Reg.OpenSubKey($Key)
$nop=$false
```

در مورد شی $Reg، Query رجیستریی که با استفاده از دستور OpenRemoteBaseKey اجرا می‌شود, با استفاده از متغیرهای $Hive و $Computer به عنوان پارامتر شاخه، ایجاد می‌کند. متغیر $nop بعدها برای کنترل جریان اجرا، استفاده می‌شود.

c

```
Try {
    $USBSTORSubKeys1  = $USBSTORKey.GetSubKeyNames()
}
Catch
{
    Write-Host "Computer: ",$ComputerName -foregroundcolor "white" -
backgroundcolor "red"
    Write-Host "No USB data found"
    $nop=$true
}
```

بلوک Try-Catch در صورتی که اطلاعاتی در مورد دستگاه‌های USB یافت نشود, مسئول مدیریت خطاها است. اگر هیچ اطلاعاتی یافت نشد، مقدار $true به منظور جلوگیری از اجرای کل فرآیند شناسایی و بازیابی اطلاعات دستگاه USB به متغیر $nop اختصاص داده می‌شود.

```
if(-Not $nop)
```

در صورت وجود هر ورودی مرتبط با اتصال دستگاه USB، مقدار متغیر $nop برابر با $true می‌باشد و بلاک‌های زیر اجرا خواهند شد:

بلاک 1 :

```
ForEach($SubKey1 in $USBSTORSubKeys1)
    {
        $Key2 = "SYSTEM\CurrentControlSet\Enum\USBSTOR\$SubKey1"
        $RegSubKey2 = $Reg.OpenSubKey($Key2)
        $SubkeyName2 = $RegSubKey2.GetSubKeyNames()
        $Subkeys2  += "$Key2\$SubKeyName2"
        $RegSubKey2.Close()

    }
```

هر یک از آیتم‌های موجود در بخش رجیستری که در آن جستجو انجام می‌شود یک دستگاه USB مختلف است. هر آیتم در matrix @ Subkeys2 ذخیره می‌شود.

بلاک 2 :

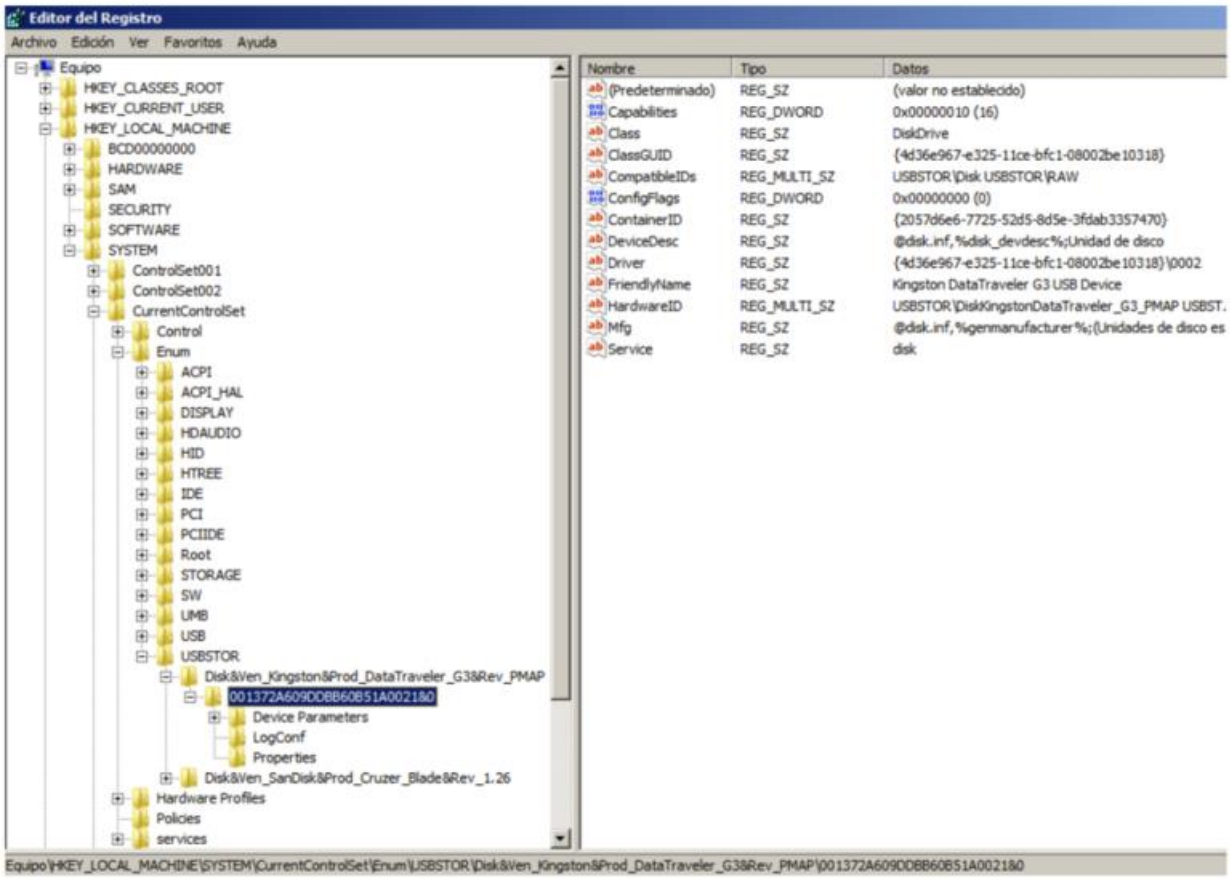
```
ForEach($Subkey2 in $Subkeys2)
    {
        $USBKey = $Reg.OpenSubKey($Subkey2)
        $USBDevice = $USBKey.GetValue('FriendlyName')

        $USBContainerID = $USBKey.GetValue('ContainerID')

        If($USBDevice)
        {
            $USBDevices += New-Object -TypeName PSObject -Property @{
            USBDevice = $USBDevice
            USBContainerID = $USBContainerID
            USBComputerName= $ComputerName
            ComputerIP = $ComputerIP
            }
        }

        $USBKey.Close()
    }
```

این بلوک هر دستگاه USB که قبلا در بلوک 1 شناسایی شده است را بررسی و در ماتریس Subkeys2 @ ذخیره می‌کند. برای هر آیتم که مقداری در $USBDevic وجود دارد، شناسه دستگاه بازیابی می‌شود: (USBContainerID). نام و آدرس IP کامپیوتر آن نیز به منظور اضافه‌کردن به فایل خروجی CSV اختصاص داده می‌شود.



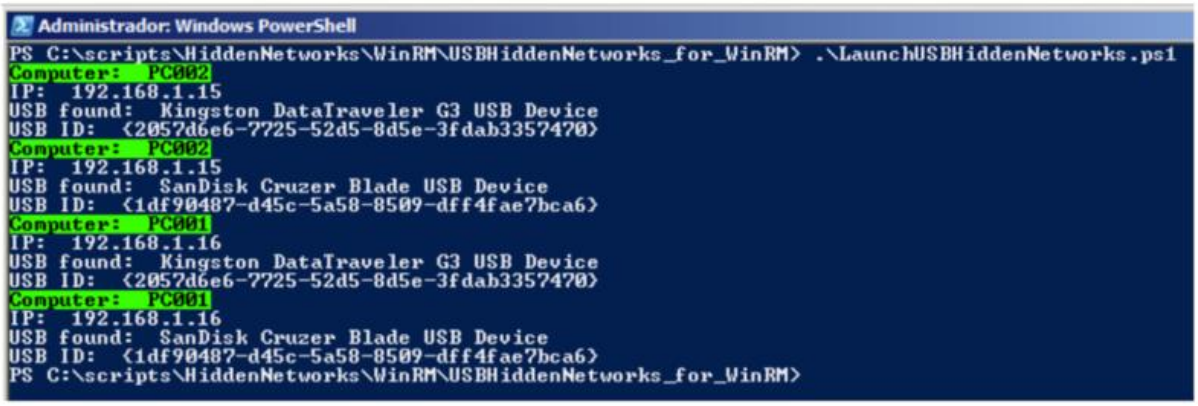
شکل ۱۰ : شاخه رجیستری مربوط به دستگاهای متصل USB

بلاک 3 :

```
for ($i=0; $i -lt $USBDevices.length; $i++) {
        $IDUnico=$USBDevices[$i] | Select -ExpandProperty "USBContainerID"
        $USBNombre=$USBDevices[$i] | Select -ExpandProperty "USBDevice"
        Write-Host "Computer: ",$ComputerName -foregroundcolor "black" -
backgroundcolor "green"
        Write-Host "IP: ",$ComputerIP
```

```
        Write-Host "USB found: ",$USBNombre
        Write-Host "USB ID: ",$IDUnico
        Echo "$ComputerName,$ComputerIP,$USBNombre,$IDUnico"
    }
```

در نهایت، این بلوک اطلاعات به دست آمده از کامپیوتر راه‌دور را نشان می‌دهد. دستور print-write-host بر روی سرور، جایی که اسکریپت اجرا می‌شود، استفاده می‌شود. دستور Echo به عنوان خروجی داده برای نوشتن اطلاعات در فایل CSV استفاده می‌شود.



شکل ۱۱ : خروجی بعد از اجرای کد

منابع:

1. https://blogs.technet.microsoft.com/heyscriptingguy/2012/05/18/use-powershell-to-find-the- history- of-usb-flash-drive-usage
2. http://www.elladodelmal.com/2017/06/brutal-kangaroo-y-la-infeccion-por-usb.html
3. https://github.com/ElevenPaths/USBHiddenNetworks