Puertos USB como Vector de Ataque con Digispark

Elías Orozco
@eliasorozco

Abstract— Al dejar una computadora desbloqueada mientras un usuario está distraído por unos segundos puede dar todo el tiempo necesario para obtener información sensible o persona de su computadora.

El objetivo de este documento es detallar la investigación y el desarrollo necesario para crear de un clon USB Rubber Ducky usando un Digispark (Attiny85), para obtener las credenciales de red Wifi de una máquina Windows, en cuestión de segundos.

I. Introducción

Casi todos los ordenadores, incluidos los de escritorio, portátiles, tabletas y teléfonos inteligentes reciben la información de los humanos a través de los teclados. Esto significa que cualquier dispositivo USB que afirme ser un teclado será automáticamente detectado y aceptado por los sistemas operativos más modernos, incluyendo Windows, Mac OS, Linux o Android.

Una herramienta de ataque de inyección de teclado (keystroke injection tool) es un dispositivo USB especialmente diseñado, a menudo disfrazado como una unidad de memoria USB, que ejecuta automáticamente código en cualquier ordenador host en el que esté conectado. Lo hace simulando ser un dispositivo de interfaz humana, HID por sus siglas en inglés “Human Interface Device” , que podría ser un teclado y/o ratón , y luego "escribiendo" los atajos de teclado y comandos a ser ejecutados.

Esto puede ser un vector de ataque para código malicioso, siendo potencialmente peligroso y destructivo. El objetivo de este documento es educar tanto al lector cómo a su "víctima" sobre los peligros de estos dispositivos. Considero que no hay lección de seguridad más memorable o impactante como conectar un dispositivo de aspecto inofensivo y ser recibido con una ráfaga de ventanas de terminal y texto apareciendo, y recibiendo un fondo de escritorio desplegado automáticamente que dice "You just got PWND, be more careful next time".

En este documento presento los detalles de la implementación de un clon del dispositivo USB rubber ducky usando un DigiSpark (Attiny 85), cuyo objetivo es vulnerar una máquina Windows a través del ya mencionado USB rubber ducky y scripting. El mecanismo permite al atacante vulnerar una máquina desatendida y recuperar datos sensibles o ejecutar código malicioso.

II. Dispositivos USB como Vector de Ataque

En esta sección se repasan algunas investigaciones previas en el área de utilizar USB como vector de ataque.

En la convención Black Hat del 2015, Nohl y Lell presentaron varios escenarios de ataque usando USB [1]. Los autores demostraron que es posible usar un USB para redirigir las consultas DNS de un usuario comprometido al servidor DNS del atacante. En un trabajo similar Kamkar [2] demuestra con un diminuto controlador, configurado para instalar una puerta trasera en el sistema comprometido que es posible cambiar las configuraciones DNS de una máquina desbloqueada. Más recientemente, un método que utiliza BadUSB desarrollado por Nikhil Mittal llamado Kautilya [3]. La herramienta permite al atacante recopilar información y ejecutar scripts que conlleva a vulnerar la máquina atacada.

Las computadoras confían inherentemente en cualquier dispositivo que clame ser HID. Ya que es con estos dispositivos que los usuarios interactúan con las computadoras incluyendo computadoras de escritorios, laptops, tablets y smartphones. Es por esto que los puertos USB han sido usados como vector de ataque por profesionales, pen testers y hackers maliciosos y se ha convertido una de las plataformas de ataque por inyección de tecleo en el negocio.

De manera comercial existen dispositivos en el mercado para llevar a cabo este tipo de vulneraciones.

Pero una de las historias ya contadas de los entusiastas de seguridad o hackers es que cuando una herramienta a la sale a la venta y su precio es alto, los hackers construyen su propia versión a una fracción del costo.

Un ejemplo de esto es el conocido USB Rubber Ducky desarrollado por Hak5 [4]. El cual esta compuesto por un microcontrolador programable de 60MHZ y una ranura SD. Este se comporta como un teclado pero aparenta ser un pendrive USB. Este puede ejecutar hasta 1000 palabras por minuto.

Pero El USB Rubber Ducky no es una herramienta del todo económica, por lo que afortunadamente se puede crear un clon casi idéntico, más con algunas limitantes, por una fracción del precio utilizando una placa de desarrollo Digispark.

El Digispark es una placa de desarrollo basada en el microcontrolador Attiny85 [5], similar a un Arduino Uno, sólo que más barata, más pequeña y un poco menos poderoso.

Uno de los limitantes es que para poder utilizar un dispositivo USB como vector de ataque es necesario tener acceso físico a la computadora de la víctima y necesitamos escribir el malware que muchas veces debe ser escrito a la medida.

No es tan poco común que alguien deje su computadora desatendida, y tampoco es común la desconfianza hacia dispositivos USB. Y es que, abusando ese descuido o confianza, se pueden robar contraseñas, o ejecutar código malicioso.

Para los propósitos de esta investigación se explotan vulnerabilidades en windows utilizando el dispositivo digispark (Attiny85). La máquina a vulnerar estará corriendo windows 7 con windows defender como su antivirus.

III. Herramientas

Para este proyecto se utilizan varias herramientas tanto de software como hardware. En esta sección se revisan las herramientas y tecnologías utilizadas.

A. Maquina a vulnerar

La máquina que será objetivo del ataque estará corriendo windows 7, 64 bits con todos los parches disponibles aplicados y usando Windows defender como protección de antivirus.

B. Dispositivo USB Digispark (Attiny85)

Se utiliza un dispositivo Digispark (Attiny85) como dispositivo de ataque, el cual puede ser conectado a la máquina de la víctima. El dispositivo cuenta solamente con 6K de memoria utilizables lo cual se debe tener en cuenta ya que no hay espacio en memoria para scripts muy grandes ni para guardar información en el mismo.

C. Arduino IDE

Antes de empezar a escribir el script que será ejecutado en la máquina de la víctima se necesita instalar el IDE que se encarga de compilar e instalar el script en el dispositivo Digispark.

Luego de instalar el Arduino IDE es necesario descargar el paquete de compatibilidad para la placa Digispark. Además es necesario instalar el Digispark Bootloader Driver. El driver es solamente necesario, para programarlo con arduino. Una vez que la programación esté lista este, funcionará como un USB Rubber Ducky en cualquier dispositivo en el que lo conecte sin ningún controlador adicional.

El proceso de configuración será abordado en la siguiente sección.

D. Powershell

PowerShell es una interfaz de consola (CLI) para Windows con posibilidad de escritura y unión de comandos por medio de instrucciones o scripts. Powershell automatiza las tareas del sistema, como el procesamiento por lotes, y crear herramientas de gestión de sistemas para procesos implementados.

IV. Configuración del Entorno de Desarrollo para Digispark

En esta sección se explican los pasos[6] a llevar a cabo para la correcta preparación del Arduino IDE para la programación del digispark.

A. Instalación del paquete de compatibilidad

Después de la instalación, abra la aplicación IDE de Arduino, navegue a Archivo -> Preferencias. En el campo de entrada "Gestor de URLs Adicionales de Tarjetas" introduzca la siguiente URL.

http://digistump.com/package_digistump_index.json

Seguidamente, navegue hasta Herramientas->Placa->Gestor de Placas. Seleccione la opción de “Contribución” del menú de Tipo, seleccione el paquete Digistump AVR Boards y proceda con la instalación..

B. Instalación del Digispark Bootloader Driver

Utilice el siguiente link para efectuar la instalación:

https://github.com/digistump/DigistumpArduino/releases/download/1.6.7/Digistump.Drivers.zip

Seguidamente ejecute el archivo “Install Drivers.exe” y proceda con la instalación.

Finalmente dentro del Arduino IDE navegue hasta herramientas->Placa y seleccione Digispark (Default -- 16.6mhz) cómo placa predeterminada.

V. Programación y despliegue del Script

En esta sección se procederá a convertir el dispositivo Digispark en un clon del USB Rubber Ducky, mediante programación.

Para el ejemplo a utilizar se creó un script cuyo objetivo será robar las credenciales del Wifi de la computadora y enviarlas por correo electrónico.

El código utilizado puede ser encontrado en el siguiente repositorio de github:

https://github.com/ExplosiveGalloPinto/PwnedCrDigisparkExample

El ejemplo utiliza la biblioteca DigiKeyboard.h la cual se encarga de que el DigiSpark actúe como un teclado y ejecute una variedad de acciones.

A. Ejecución del código

En el ejemplo a analizar, el script inicia su ejecución abriendo una ventana de simbolo de sistema (CMD), seguidamente usando comandos de powershell crea un archivo con extensión .csv de nombre temp.csv el cual será enviado posteriormente a una dirección de correo que se especifica en el código. Al finalizar el programa borra el archivo temp.csv y cierra las ventanas del símbolo del sistema.

B. Comandos de DigiKeyboard.h

Algunos comandos a analizar:

1) DigiKeyboard.sendKeyStroke()

Este comando simula la presión de una o varias teclas por parte de un usuario.

2) DigiKeyboard.delay()

Se encarga de pausar la ejecución del script por una cantidad de tiempo.

3) DigiKeyboard.print()

Encargado de la simulación del tecleo de una palabra por parte del usuario.

C. Comandos de PowerShell

En esta sección me refiero a líneas específicas en el archivo StealWifiScript.ino por motivos de formato y facilidad de lectura.

1) Linea 22

Esta línea simula en ingreso de un comando powershell por parte del usuario el cual crea un archivo .csv y lo llena con las credenciales de la víctima.

2) Linea 36

Es la encargada de enviar el archivo que se creó en la línea 22 a una dirección de correo especificada.

C. Despliegue

Una vez completado el script se procede a subirlo al Digispark, primero se debe hacer click en el botón de “Subir” una vez se compile el script se conecta el dispositivo Digispark y se espera a que se suba.

El despliegue en la computadora de la víctima debe realizarse aprovechando un momento de descuido, en el que se conecte el dispositivo a uno de los puertos USB de la máquina.

VI. Conclusiónes

En este proyecto se demostró cómo utilizando el dispositivo Digispark y herramientas de programación es posible vulnerar cualquier computadora que acepte un dispositivo USB.

Además de que construir un dispositivo de inyección de teclado esta ala alcance de cualquier entusiasta, pentester o incluso hacker.

También se demostró cómo usando un dispositivo USB un atacante necesita de solamente unos segundos para insertar el dispositivo, revelar y robar información sensible del usuario.

Además se encontraron algunas limitaciones o desventajas sobre usar un dispositivo como lo es el Digispark para crear un Rubber Ducky, algunas de ellas son:

- El poco tamaño de memoria para almacenar payloads
- El dispositivo está limitado nada más por lo que se pueda lograr mediante el uso de un teclado
- A la hora de crear el script este debe ser hecho a la medida tanto del sistema operativo como al de la máquina

VIII. Referencias

[1]BlackHat USA 2014, Karsten Nohl and Jakob Lell, BadUSB - On Accessories that Turn Evil, https://srlabs.de/badusb/, accesado el 13 Nov 2018

[2]S. Kamkar, USBDriveBy, http://samy.pl/usbdriveby/, accesado el 13 de Nov 2018

[3]Nikhil "SamratAshok" Mittal, Kautilya, https://github.com/samratashok/Kautilya, accesado el 13 Nov 2018

[4] USB Rubber Ducky. (2018). Hak5. https://shop.hak5.org/products/usb-rubber-ducky-deluxe, accesado el 13 Nov 2018

[5] Digispark USB Dev Board (Attiny85). CRCibernética. https://www.crcibernetica.com/digispark-usb-dev-board-attiny85/, accesado el 13 Nov 2018

[6] "Digispark:Tutorials:Connecting [Digistump Wiki]". 2018. Digistump.Com. https://digistump.com/wiki/digispark/tutorials/connecting, accesado el 13 Nov 2018

[7] CedArctic/DigiSpark-Scripts. (2018). GitHub. Retrieved 14 November 2018, from https://github.com/CedArctic/DigiSpark-Scripts