Configuring and running Radare2 on Android mobile phones

author: Alexandre Borges
revision: 1

Few days ago, I received a question about the possibility in running Radare2 on mobile phones. Yes, it is possible and handy. This is a procedure known by many reverse engineers, but sometimes could be hard to put all steps together.

Therefore, for running **Radare2** on **the Android mobiles**, perform the following steps:

1. Install the**Termux** from Google Play.

2. Run the **Termux**.

3. Install the following useful packages:

    a. **pkg install libllvm**
    b. **pkg install openssl**
    c. **pkg install openssh**
    d. **pkg install util-linux**
    e. **pkg install binutils**
    f. **pkg install libgcc**
    g. **pkg install readline**
    h. **pkg install dos2unix**
    i. **pkg install radare2**

4. Check the installed packages by running the following command:

    $ **pkg list-installed**

5. It is recommended to enable extra-keys view (**Volume Up+Q**) for an easier access to ESC/CTRL/ALT/TAB keys.

6. Remember that:

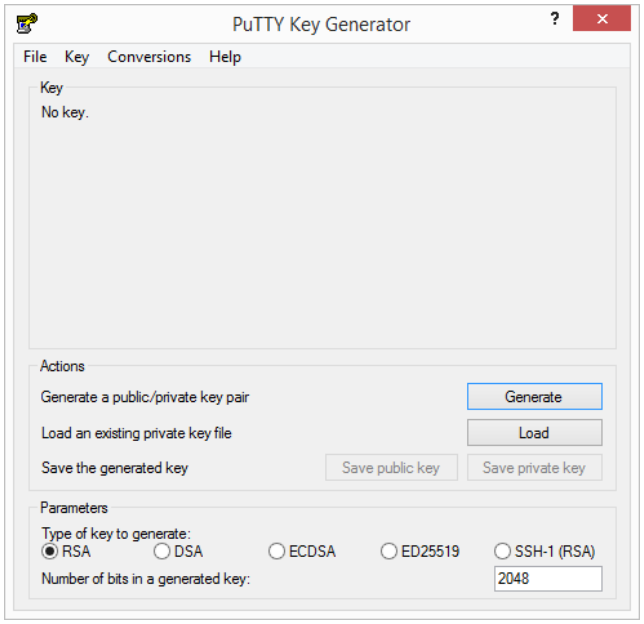
    a. **Volume Up+W** → Up arrow key
    b. **Volume Up+A** → Left arrow key
    c. **Volume Up+S** → Down arrow key
    d. **Volume Up+D** → Right arrow key
    e. More information on: https://termux.com/touch-keyboard.html

7. Generate SSH keys by running the **ssh-keygen** command.

8. As typing command on mobile is not easy, so it is better to connect to mobile through the wireless network by using **ssh** command. Thus, at this point, we have the possibility of connecting either from Windows (most people use **Putty.exe**) or from Linux (Kali Linux, in my case).

9. **(WINDOWS)** On Windows systems, to configure the **Putty.exe** (it can be downloaded from https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html), execute the following steps:

    a. Run the **Puttygen.exe** for generating the key pairs (**RSA – 2048 bits**, at least):



    b. Save both **public and private keys** into **C:\Program Files\PuTTY** directory. In my case, I used **pub_putty** and **priv_putty** as file names, respectively.

    c. Unfortunately, the public key file (**pub_putty**) generated by **Putty.exe** has an inappropriate format:

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20171129"
AAAAB3NzaC1yc2EAAAABJQAAAQEAure7/7iq9l7uuzmZaT1U8h+2Zi2YHAXxZylV
aNN5bPwqKgB4iXbeW+U2DIs+pI0sGZHZeIUbSSjWlAzUD6XNjYjHwE98tYo3QlGt
/CgPgRjjTTT0Vq0OARpPJHFTjGFM+ORP7zpe9ARwc/3kOPrxvFvoWQ7OMNwzBp0f
c/OKrO+O72BZEX7yqELB/45BdqQvJz6twlIOQ10umIn4tos5K4XZxCKdav2whIpX
PvHfQoVl5nnYn3wXBa+cLPrljKsmhDjQVDX+QeiEQyG119pqGMFLgtitCChLUBbp
BzsAmD2LOB2apHqfs2g/zOo9wUR+bLILGDmYKaZRqml1wyLbdw==
---- END SSH2 PUBLIC KEY ----
```

Copy this file to another place (for example, **your Desktop**) and edit it as shown below (put everything at the same line):

ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAure7/7iq9l7uuzmZaT1U8h+2Zi2YHAXxZylVaNN5bPwq
KgB4iXbeW+U2DIs+pI0sGZHZeIUbSSjWlAzUD6XNjYjHwE98tYo3QlGt/CgPgRjjTTT0Vq0OARp
PJHFTjGFM+ORP7zpe9ARwc/3kOPrxvFvoWQ7OMNwzBp0fc/OKrO+O72BZEX7yqELB/45Bd
qQvJz6twlIOQ10umIn4tos5K4XZxCKdav2whIpXPvHfQoVl5nnYn3wXBa+cLPrljKsmhDjQVDX+
QeiEQyG119pqGMFLgtitCChLUBbpBzsAmD2LOB2apHqfs2g/zOo9wUR+bLILGDmYKaZRqml
1wyLbdw== **administrator@hphacker**

It is interesting to realize that:

- The following lines were removed:

  ---- BEGIN SSH2 PUBLIC KEY ----
  Comment: "rsa-key-20171129"
  ---- END SSH2 PUBLIC KEY ----

- The string **ssh-rsa** was added at beginning of the line.
- The string **administrator@hphacker** was appended at the end of the line. In this case, **administrator** means the username and **hphacker** is the machine name.
- Again, everything is at **only ONE line**.

d. Upload this modified **pub_putty** file (from your Desktop) to an online repository (in my case, I used the website of my company, www.blackstormsecurity.com).

e. On the mobile, download the **pub_putty** file (containing the public key) by using similar commands (remember, in your case the URL is another one):

$ **cd**
$ **cd .ssh**
$ **wget www.blackstormsecurity.com/pub_putty**

f. Convert the downloaded file (**pub_putty)** to Unix format by running the following command:

$ **dos2unix pub_putty**

g. Add the Windows public key into the **authorized_keys** file by running the following command:

$ **cat pub_putty >> authorized_keys**

h. Check the **authorized_keys** file contents as shown below:

$ **cat authorized_keys**

```
bash-4.4$ more authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDgjPTtyGdkQPx3qGEgDj3YFLcd3ZMn2PIfqk0jhNQc
nGtKREOVMGXvHX+Huv5LvmMLLVBX+ytq/mb6yxnpasl6wFgaGwBvCkzLzEJ+S+MaMwbM3GbTMDRkYtbe
WGsTMjcvPV9JmRuer0ch10xnw4DO8Hi13CMc+V1mhLRfuE9HOMHdVuFU3DjLUaYLgAGEp1mjhjXLQs4I
XJcbnBaGAp+75CVZLsoqVjonZV9WsEcopPVzxsnxsBkAk/Ho2WLe/y0RKnr9DD3MdeIEZz516HtvqMF4
RU6QSOa/p96Q1Be+7We2EqALbvRfhdI4aeTDhcjzPP0XyCB4jl8gqOawcHof root@kali
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAure7/7iq9l7uuzmZaT1U8h+2Zi2YHAXxZylVaNN5bPwq
KgB4iXbeW+U2DIs+pI0sGZHZeIUbSSjWlAzUD6XNjYjHwE98tYo3QlGt/CgPgRjjTTT0Vq0OARpPJHFT
jGFM+ORP7zpe9ARwc/3kOPrxvFvoWQ7OMNwzBp0fc/OKrO+O72BZEX7yqELB/45BdqQvJz6twlIOQ10u
mIn4tos5K4XZxCKdav2whIpXPvHfQoVl5nnYn3wXBa+cLPrljKsmhDjQVDX+QeiEQyG119pqGMFLgtit
CChLUBbpBzsAmD2LOB2apHqfs2g/zOo9wUR+bLILGDmYKaZRqml1wyLbdw== administrator@hphac
ker
```

Take care: there can not be any **^M** at end of the line!

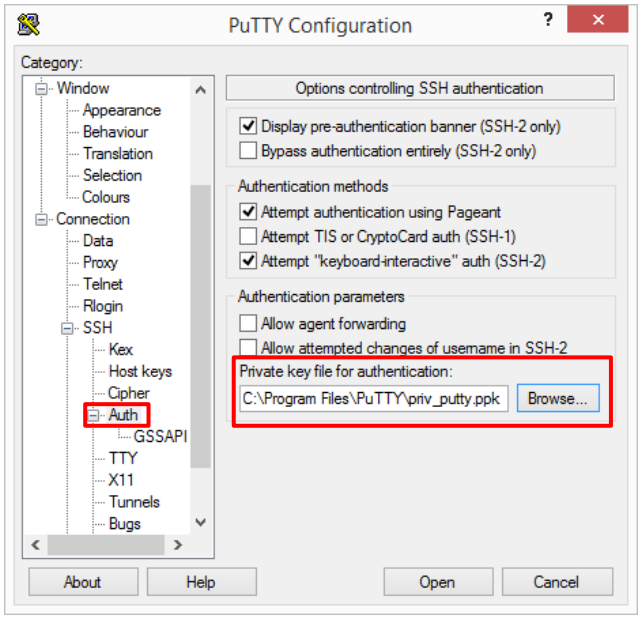
i. Check the IP address of the mobile phone by running the following command:

$ **ifconfig wlan0 | grep inet**

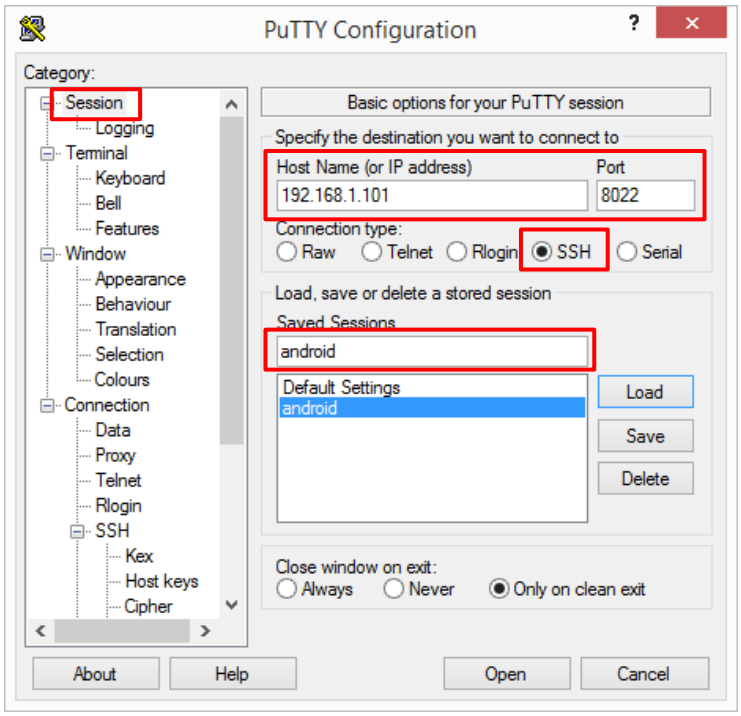
j. Run the **sshd daemon** (the **default port** is **8022** because the Termux is not running as root) by executing the following commands:

$ **sshd**

k. Configure the **Putty connection**. First**, browse the private key** as shown below:



Now, at the **Session** configuration, fill the text boxes as shown below:



Clicking on Open button, we have the following output from the Putty screen:

login as: administrator
Authenticating with public key "rsa-key-20171129"
Passphrase for key "rsa-key-20171129":
Welcome to Termux!

Wiki: https://wiki.termux.com
Community forum: https://termux.com/community
IRC channel: #termux on freenode
Gitter chat: https://gitter.im/termux/termux
Mailing list: termux+subscribe@groups.io
Search packages: pkg search <query>
Install a package: pkg install <package>
Upgrade packages: pkg upgrade
Learn more: pkg help

bash-4.4$ **id**
uid=10013(u0_a13) gid=10013(u0_a13)
groups=3003(inet),9997(everybody),50013(all_a13)
bash-4.4$ **uname -a**
Linux localhost 3.4.0-8347901 #1 SMP PREEMPT Fri Jun 10 04:50:23 KST 2016 armv7l
Android

10. (LINUX) On the Kali Linux, connecting to the Android mobile is infinitely easier than Windows, so perform the following steps:

    a. Generate the **SSH public** and **private key** pair as shown below:

```
root@kali:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:RpMa9khbvXcS49CqDGXCMS1irtdm7y+lZxtyT97XFuE root@kali
The key's randomart image is:
+---[RSA 2048]----+
|      o.         |
|    o..o.o .     |
|   o .*.B o +    |
|    .o % . = o . |
|   . .= S o + o .|
|  . . ++ ... o E |
|   . o .+oo .   o|
|        +oo= .  +|
|       ..=o.o .o |
+----[SHA256]-----+
root@kali:~#
```

    b. Check the generated keys as shown below:

```
root@kali:~# cd
root@kali:~# cd .ssh
root@kali:~/.ssh# ls
id_rsa  id_rsa.pub  known_hosts
root@kali:~/.ssh# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC862RqP1NbBOgpc0xFDa8RZv2BWss9
v6ojBzQGsAaF2i7bYcmGAe5GuiP1BRQsuCd759jQfRSJL3jr2tNaYwqXdZghy3lzccH5
310qThryMlJfSekZj5DNzri+bpLxIpT9U/GcZLaQ7mmIEHM0hsn0FiUZdcnl7sQ8Hyn1
3hGOVJQv9Km6BR5zMOsnFfVShpWEmoFLR1vKLo+CMjRZU/Q8Sbtt8IAn+gzqNlIcdB7Y
zNcDprs2OJ97gRQ/HhEKVSyX8i4B8/WsHuScFD/Uw6EsZFOzD1ZBouyZIO0D/ShvWBcS
eeLTwaxvcqLzG1nAbwOf8rHNrsZT5ebwfcAYJAe7 root@kali
root@kali:~/.ssh#
```

    c. As we have done for Windows, upload the public key file (**id_rsa.pub**) to an online repository and, on the Android mobile phone, download it by executing the following commands:

    $ **cd**
    $ **cd .ssh**
    $ **wget www.blackstormsecurity.com/id_rsa.pub**

    d. Append this new public key into the **authorized_keys** file by running the following command:

$ **cat pub_putty >> authorized_keys**

e. Check the **authorized_keys** file content by executing the following command:

$ **cat authorized_keys**

```
bash-4.4$ more authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDgjPTtyGdkQPx3qGEgDj3YFLcd3ZMn2PIfqk0jhNQc
nGtKREOVMGXvHX+Huv5LvmMLLVBX+ytq/mb6yxnpasl6wFgaGwBvCkzLzEJ+S+MaMwbM3GbTMDRkYtbe
WGsTMjcvPV9JmRuer0ch10xnw4DO8Hi13CMc+V1mhLRfuE9HOMHdVuFU3DjLUaYLgAGEp1mjhjXLQs4I
XJcbnBaGAp+75CVZLsoqVjonZV9WsEcopPVzxsnxsBkAk/Ho2WLe/y0RKnr9DD3MdeIEZz516HtvqMF4
RU6QSOa/p96Q1Be+7We2EqALbvRfhdI4aeTDhcjzPP0XyCB4jl8gqOawcHof root@kali
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAure7/7iq9l7uuzmZaT1U8h+2Zi2YHAXxZylVaNN5bPwq
KgB4iXbeW+U2DIs+pI0sGZHZeIUbSSjWlAzUD6XNjYjHwE98tYo3QlGt/CgPgRjjTTT0Vq0OARpPJHFT
jGFM+ORP7zpe9ARwc/3kOPrxvFvoWQ7OMNwzBp0fc/OKrO+O72BZEX7yqELB/45BdqQvJz6twlIOQ10u
mIn4tos5K4XZxCKdav2whIpXPvHfQoVl5nnYn3wXBa+cLPrljKsmhDjQVDX+QeiEQyG119pqGMFLgtit
CChLUBbpBzsAmD2LOB2apHqfs2g/zOo9wUR+bLILGDmYKaZRqml1wyLbdw== administrator@hphac
ker
```

f. Connect to the Android mobile phone as shown below:

```
root@kali:~# ssh 192.168.1.101 -p 8022
Welcome to Termux!

Wiki:            https://wiki.termux.com
Community forum: https://termux.com/community
IRC channel:     #termux on freenode
Gitter chat:     https://gitter.im/termux/termux
Mailing list:    termux+subscribe@groups.io

Search packages:   pkg search <query>
Install a package: pkg install <package>
Upgrade packages:  pkg upgrade
Learn more:        pkg help
bash-4.4$
bash-4.4$ id
uid=10013(u0_a13) gid=10013(u0_a13) groups=3003(inet),9997(everybody
),50013(all_a13)
bash-4.4$
bash-4.4$ uname -a
Linux localhost 3.4.0-8347901 #1 SMP PREEMPT Fri Jun 10 04:50:23 KST
 2016 armv7l Android
```

11. We have finished our the network steps for making the connection to the Android mobile phone simple and quick.

    Obviously, we want to show the **radare2** running, so the screen below proves our objective:

```
bash-4.4$ whereis ls
ls: /data/data/com.termux/files/usr/bin/applets/ls
bash-4.4$
bash-4.4$ r2 /data/data/com.termux/files/usr/bin/applets/ls
[0x00004914]> ie

[Entrypoints]
vaddr=0x00004914 paddr=0x00004914 baddr=0x00000000 laddr=0x00000000
haddr=0x00000018 type=program

1 entrypoints

[0x00004914]> pd 5

            ;-- entry0:
            ;-- pc:
            ;-- r15:
            0x00004914      5cc09fe5        ldr ip, [0x00004978]
   ; [0x4978:4]=0x52278 "lor" ; "x\"\x05"
            0x00004918      5c209fe5        ldr r2, [0x0000497c]
   ; [0x497c:4]=0xffffff88
            0x0000491c      00482de9        push {fp, lr}
            0x00004920      0cc08fe0        add ip, pc, ip
            0x00004924      04b08de2        add fp, sp, 4
[0x00004914]>
```

It has worked! ☺ If you have curiosity, read my previous article: **Overview about a typical bank-trojan --** http://www.blackstormsecurity.com/docs/FOAATTB.pdf

Radare your mind every day.

- **Malware and Security Researcher.**
- **Consultant, Instructor and Speaker on Malware Analysis, Memory Analysis, Digital Forensics, Rootkits and Software Exploitation.**
- **Instructor at Oracle, (ISC)2 and EC-Council.**
- **Ex-instructor at Symantec.**
- **Member of Digital Law and Compliance Committee (CDDC / SP)**
- **Member of the CHFI Advisory Board in EC-Council.**
- **Reviewer member of the The Journal of Digital Forensics, Security and Law**
- **Referee on Digital Investigation: The International Journal of Digital Forensics & Incident Response**
- **Author of "Oracle Solaris Advanced Administration book"**

Alexandre Borges