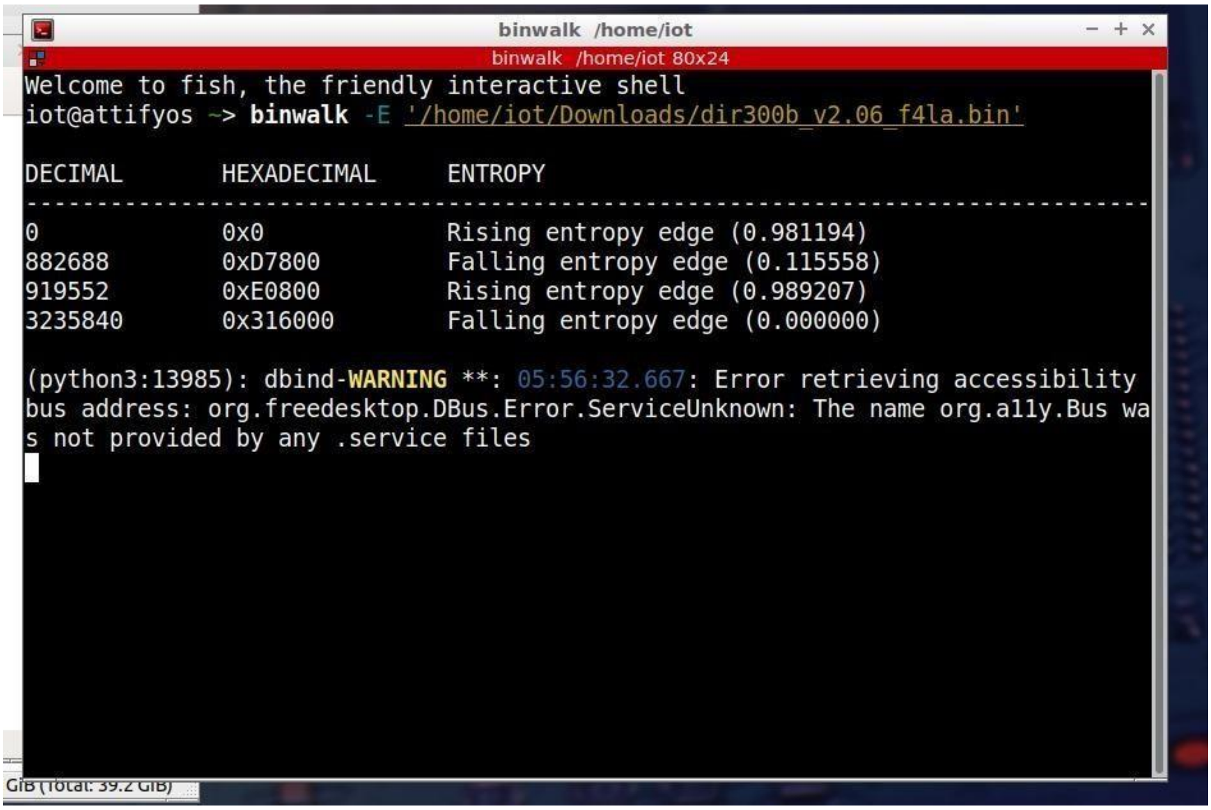
Experiment-4

AIM: Finding Vulnerabilities in IoT system.

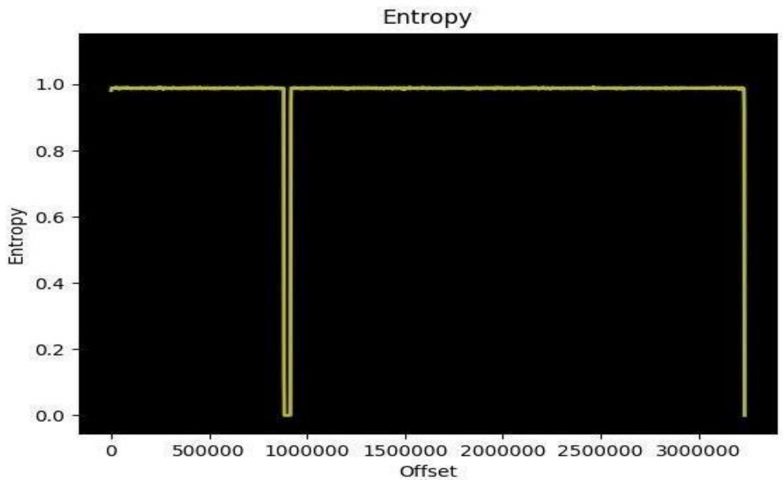
Step 1: First we have download the firmware name **DIR300**

Step 2: then we will check this firmware is encrypted or not . for this we will use

· Binwalk -E ‘file path’
· After check we will know thatthis firmware is not encrypted .

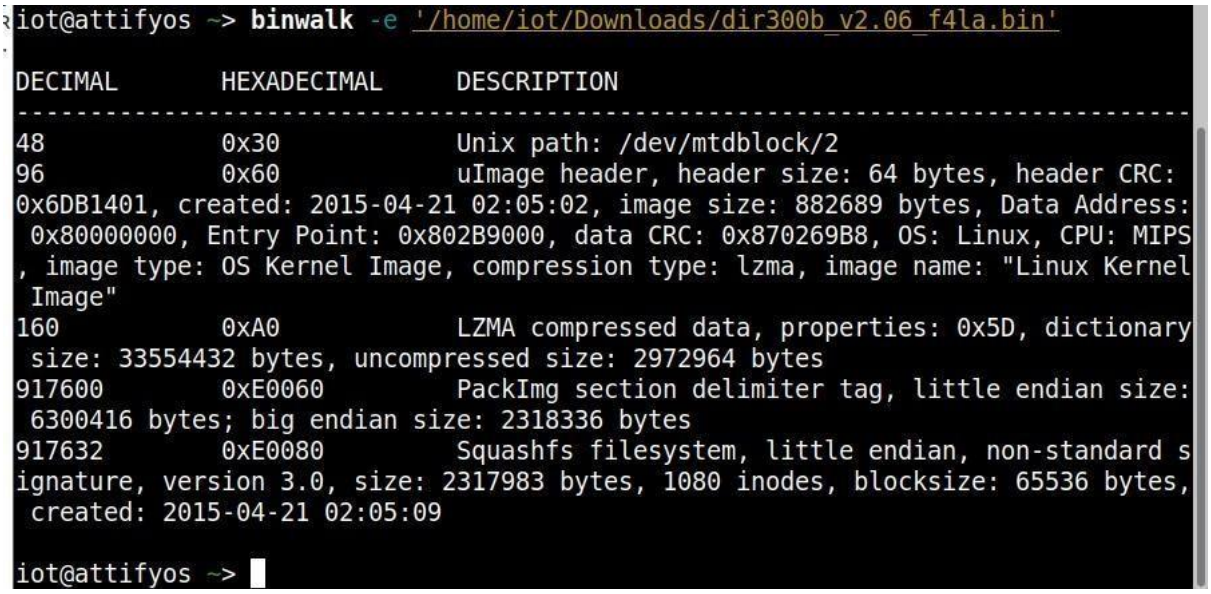


```
Welcome to fish, the friendly interactive shell
iot@attifyos ~> binwalk -E '/home/iot/Downloads/dir300b_v2.06_f4la.bin'

DECIMAL       HEXADECIMAL     ENTROPY
--------------------------------------------------------------------------------
0             0x0             Rising entropy edge (0.981194)
882688        0xD7800         Falling entropy edge (0.115558)
919552        0xE0800         Rising entropy edge (0.989207)
3235840       0x316000        Falling entropy edge (0.000000)

(python3:13985): dbind-WARNING **: 05:56:32.667: Error retrieving accessibility
bus address: org.freedesktop.DBus.Error.ServiceUnknown: The name org.a11y.Bus wa
s not provided by any .service files
```



Step 3: after checking that firmware is not encrypted . we can extract the firmware so that we can use

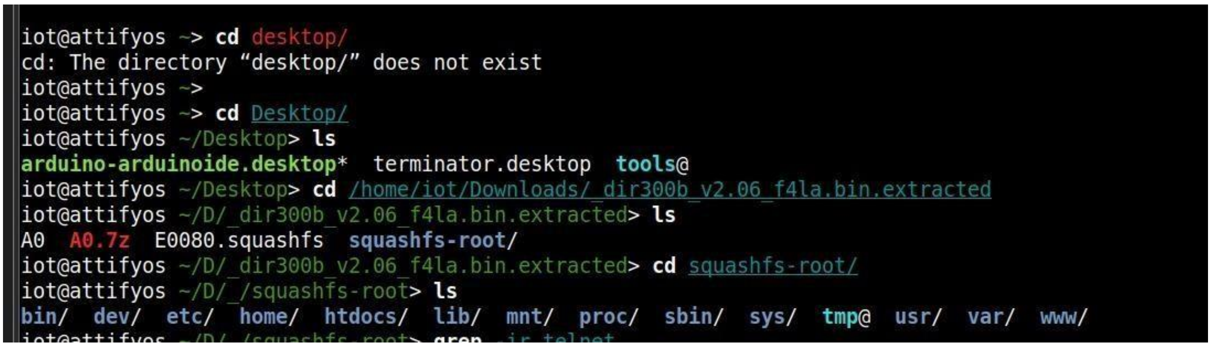
· **‘binwalk -e filepath’**



```
iot@attifyos ~> binwalk -e '/home/iot/Downloads/dir300b_v2.06_f4la.bin'

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
48            0x30            Unix path: /dev/mtdblock/2
96            0x60            uImage header, header size: 64 bytes, header CRC:
0x6DB1401, created: 2015-04-21 02:05:02, image size: 882689 bytes, Data Address:
 0x80000000, Entry Point: 0x802B9000, data CRC: 0x870269B8, OS: Linux, CPU: MIPS
, image type: OS Kernel Image, compression type: lzma, image name: "Linux Kernel
 Image"
160           0xA0            LZMA compressed data, properties: 0x5D, dictionary
 size: 33554432 bytes, uncompressed size: 2972964 bytes
917600        0xE0060         PackImg section delimiter tag, little endian size:
 6300416 bytes; big endian size: 2318336 bytes
917632        0xE0080         Squashfs filesystem, little endian, non-standard s
ignature, version 3.0, size: 2317983 bytes, 1080 inodes, blocksize: 65536 bytes,
 created: 2015-04-21 02:05:09

iot@attifyos ~>
```

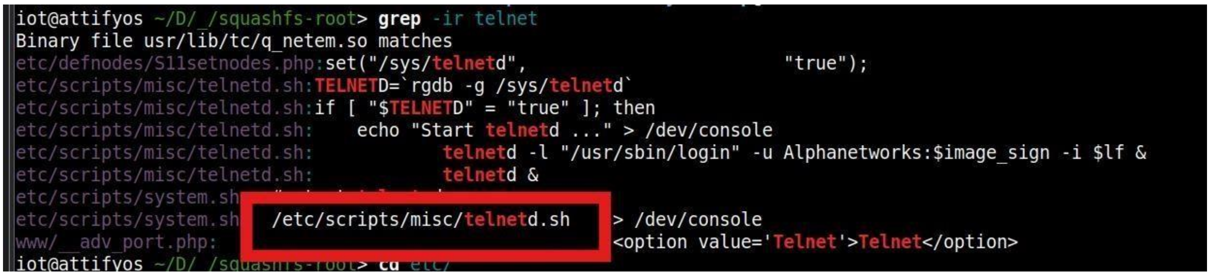
Step 4: after extracting the firmware use

· **cd desktop/ cd file path/ cd squashfs-root**



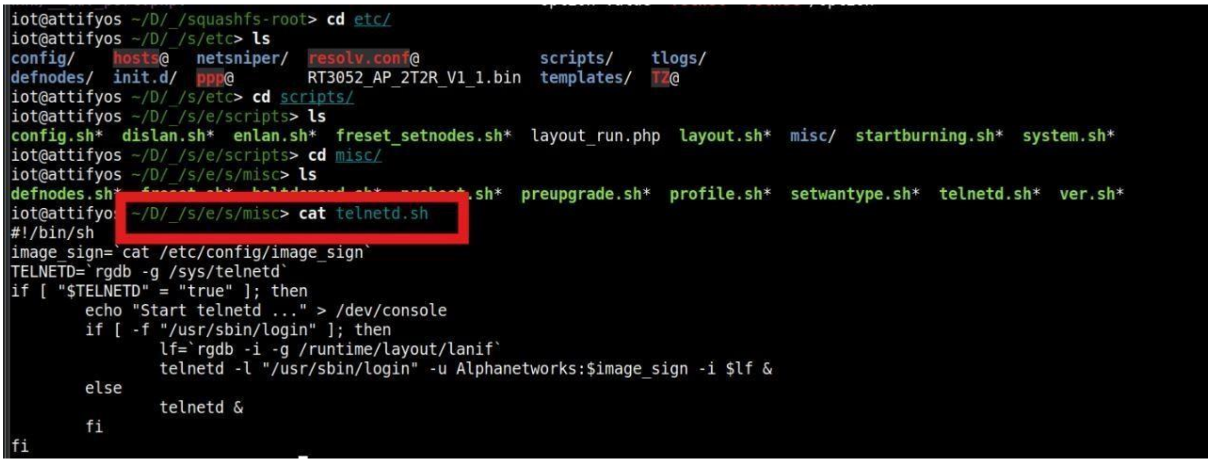
```
iot@attifyos ~> cd desktop/
cd: The directory “desktop/” does not exist
iot@attifyos ~>
iot@attifyos ~> cd Desktop/
iot@attifyos ~/Desktop> ls
arduino-arduinoide.desktop*  terminator.desktop  tools@
iot@attifyos ~/Desktop> cd /home/iot/Downloads/_dir300b_v2.06_f4la.bin.extracted
iot@attifyos ~/D/_dir300b_v2.06_f4la.bin.extracted> ls
A0  A0.7z  E0080.squashfs  squashfs-root/
iot@attifyos ~/D/_dir300b_v2.06_f4la.bin.extracted> cd squashfs-root/
iot@attifyos ~/D/_/squashfs-root> ls
bin/  dev/  etc/  home/  htdocs/  lib/  mnt/  proc/  sbin/  sys/  tmp@  usr/  var/  www/
iot@attifyos ~/D/_/squashfs-root> grep -ir telnet
```

Step 5: after entering in to squashfs -root folder . we can use grep -ir telnet to know location of password . **grep -ir telnet** location : **/etc/scripts /misc/telnetd.sh**



```
iot@attifyos ~/D/_/squashfs-root> grep -ir telnet
Binary file usr/lib/tc/q_netem.so matches
etc/defnodes/S11setnodes.php:set("/sys/telnetd",                    "true");
etc/scripts/misc/telnetd.sh:TELNETD=`rgdb -g /sys/telnetd`
etc/scripts/misc/telnetd.sh:if [ "$TELNETD" = "true" ]; then
etc/scripts/misc/telnetd.sh:    echo "Start telnetd ..." > /dev/console
etc/scripts/misc/telnetd.sh:            telnetd -l "/usr/sbin/login" -u Alphanetworks:$image_sign -i $lf &
etc/scripts/misc/telnetd.sh:            telnetd &
etc/scripts/system.sh
etc/scripts/system.sh   /etc/scripts/misc/telnetd.sh   > /dev/console
www/__adv_port.php:                                    <option value='Telnet'>Telnet</option>
iot@attifyos ~/D/_/squashfs-root> cd etc/
```

Step 6: after the getting the path of password. We can follow this path for find a password. **Path : cd etc/ls/cd scripts/ls/cd misc/ls/cat telnetd.sh**

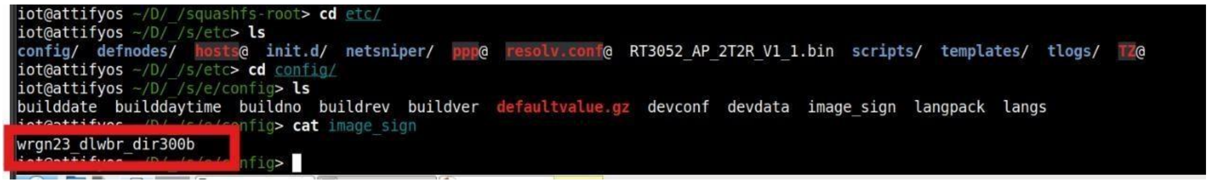


```
iot@attifyos ~/D/_/squashfs-root> cd etc/
iot@attifyos ~/D/_/s/etc> ls
config/    hosts@   netsniper/  resolv.conf@            scripts/    tlogs/
defnodes/  init.d/  ppp@        RT3052_AP_2T2R_V1_1.bin  templates/  TZ@
iot@attifyos ~/D/_/s/etc> cd scripts/
iot@attifyos ~/D/_/s/e/scripts> ls
config.sh*  dislan.sh*  enlan.sh*  freset_setnodes.sh*  layout_run.php  layout.sh*  misc/  startburning.sh*  system.sh*
iot@attifyos ~/D/_/s/e/scripts> cd misc/
iot@attifyos ~/D/_/s/e/s/misc> ls
defnodes.sh*  freset.sh*  haltdemand.sh*  preboot.sh*  preupgrade.sh*  profile.sh*  setwantype.sh*  telnetd.sh*  ver.sh*
iot@attifyos ~/D/_/s/e/s/misc> cat telnetd.sh
#!/bin/sh
image_sign=`cat /etc/config/image_sign`
TELNETD=`rgdb -g /sys/telnetd`
if [ "$TELNETD" = "true" ]; then
        echo "Start telnetd ..." > /dev/console
        if [ -f "/usr/sbin/login" ]; then
                lf=`rgdb -i -g /runtime/layout/lanif`
                telnetd -l "/usr/sbin/login" -u Alphanetworks:$image_sign -i $lf &
        else
                telnetd &
        fi
fi
```

Step 7: after follow path we can get image_ sign password file path.

· **Path : cd etc/ls/cd config/ls/cat image_sign** .

· after follow cd etc/ls/cd config/ls/cat image_sign this path we get the password



```
iot@attifyos ~/D/_/squashfs-root> cd etc/
iot@attifyos ~/D/_/s/etc> ls
config/  defnodes/  hosts@  init.d/  netsniper/  ppp@  resolv.conf@  RT3052_AP_2T2R_V1_1.bin  scripts/  templates/  tlogs/  TZ@
iot@attifyos ~/D/_/s/etc> cd config/
iot@attifyos ~/D/_/s/e/config> ls
builddate  builddaytime  buildno  buildrev  buildver  defaultvalue.gz  devconf  devdata  image_sign  langpack  langs
iot@attifyos ~/D/_/s/e/config> cat image_sign
wrgn23_dlwbr_dir300b
iot@attifyos ~/D/_/s/e/config>
```

Conclusion : The main disadvantage of this firmware is not encrypted. We can get the password and explore any file of this device.