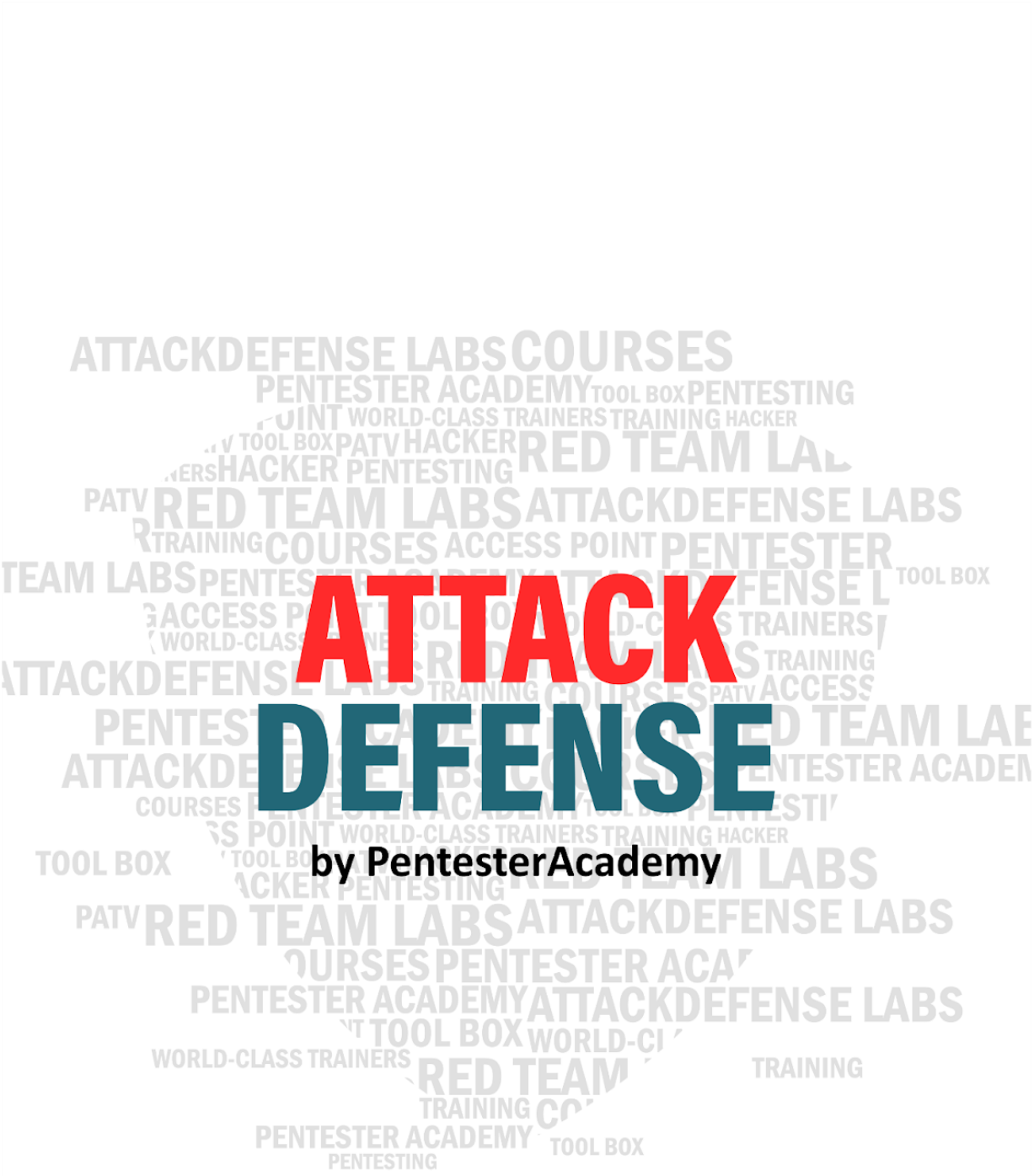
ATTACK
DEFENSE
by PentesterAcademy

Name	TFTP Boot using U-boot
URL	https://www.attackdefense.com/challengedetails?cid=1232
Type	IoT : Bootloader

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic.

Objective: Boot the machine using the provided U-boot and remote TFTP server.

Step 1: U-Boot binary is provided in /root directory.

Command: ls -l

```
root@attackdefense:~# ls -l
total 3340
-rwxr-xr-x 1 root root 3417040 Sep 17 09:49 u-boot
root@attackdefense:~#
```

Step 2: In this challenge, the kernel, DTB and rootfs files are on TFTP server. To know the IP address of TFTP server, first check the IP address of local machine.

Command: ip addr

```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
3300: eth0@if3301: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:03 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.3/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
3303: eth1@if3304: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:6e:72:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.110.114.2/24 brd 192.110.114.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

The local machine is on IP 192.110.114.2. So, according to the challenge description, the TFTP server should be on 192.110.114.3

Step 3: Start the emulation.

Command: qemu-system-arm -M vexpress-a9 -m 512M -kernel u-boot -nographic

Here,

-M vexpress-a9: Virtual machine selection
For more on Vexpress: https://crux-arm.nu/SupportedDevices/Vexpress
-m 512M : Memory to be allocated to virtual device
-kernel u-boot: Use U-boot to boot the machine
-nographic : To invoke qemu from CLI

```
root@attackdefense:~# qemu-system-arm -M vexpress-a9 -m 512M -kernel u-boot -nographic
pulseaudio: pa_context_connect() failed
pulseaudio: Reason: Connection refused
pulseaudio: Failed to initialize PA contextaudio: Could not init `pa' audio driver
ALSA lib confmisc.c:767:(parse_card) cannot find card '0'
ALSA lib conf.c:4528:(_snd_config_evaluate) function snd_func_card_driver returned error:
ALSA lib confmisc.c:392:(snd_func_concat) error evaluating strings
```

The emulation will start, u-boot will try to locate the kernel and fail. This will drop the user into u-boot console.

```
TFTP error: trying to overwrite reserved memory...
smc911x: MAC 52:54:00:12:34:56
Wrong Image Format for bootm command
ERROR: can't get kernel image!
=>
```

NOTE: When we let U-boot fail on its own, it takes IP from DHCP automatically. If the autoboot is interrupted, the user might have to assign it address and in some cases the setup might not work.

Step 4: Check the board information get the memory information and address range.

Command: bdinfo

```
=> bdinfo
arch_number = 0x000008e0
boot_params = 0x60002000
DRAM bank   = 0x00000000
-> start    = 0x60000000
-> size     = 0x20000000
DRAM bank   = 0x00000001
-> start    = 0x80000000
-> size     = 0x00000004
eth0name    = smc911x-0
ethaddr     = 52:54:00:12:34:56
current eth = smc911x-0
ip_addr     = <NULL>
baudrate    = 38400 bps
TLB addr    = 0x7fff0000
relocaddr   = 0x7ff8b000
reloc off   = 0x1f78b000
irq_sp      = 0x7fe8aee0
sp start    = 0x7fe8aed0
=>
```

Step 5: Define the IP address of remote TFTP server.

Command: setenv serverip 192.110.114.3

```
=>
=> setenv serverip 192.110.114.3
=>
```

Step 6: Fetch kernel, Device Tree Blob file and root filesystem from TFTP server and store these on valid memory addresses. Also, make sure that these don't overwrite each other.

Fetching Kernel

Command: tftp 0x61000000 zImage

```
=> tftp 0x61000000 zImage
smc911x: MAC 52:54:00:12:34:56
smc911x: detected LAN9118 controller
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
Using smc911x-0 device
TFTP from server 192.110.114.3; our IP address is 10.0.2.15; sending through gateway 10.0.2.2
Filename 'zImage'.
Load address: 0x61000000
Loading: #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         ##################################
         1.1 MiB/s
done
Bytes transferred = 4181384 (3fcd88 hex)
smc911x: MAC 52:54:00:12:34:56
```

Fetching DTB

Command: tftp 0x61a00000 vexpress-v2p-ca9.dtb

```
=> tftp 0x61a00000 vexpress-v2p-ca9.dtb
smc911x: MAC 52:54:00:12:34:56
smc911x: detected LAN9118 controller
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
Using smc911x-0 device
TFTP from server 192.110.114.3; our IP address is 10.0.2.15;
Filename 'vexpress-v2p-ca9.dtb'.
Load address: 0x61a00000
Loading: ###
         27.3 KiB/s
done
Bytes transferred = 14430 (385e hex)
smc911x: MAC 52:54:00:12:34:56
```

Fetching RootFS

Command: tftp 0x62000000 rootfs.cpio.uboot

```
=> tftp 0x62000000 rootfs.cpio.uboot
smc911x: MAC 52:54:00:12:34:56
smc911x: detected LAN9118 controller
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
Using smc911x-0 device
TFTP from server 192.110.114.3; our IP address is 10.0.2.15; sending through
Filename 'rootfs.cpio.uboot'.
Load address: 0x62000000
Loading: ##################################################################
         ##################################################################
         ##################################################################
         ##################################################################
         ##########
         1.2 MiB/s
done
Bytes transferred = 1379392 (150c40 hex)
smc911x: MAC 52:54:00:12:34:56
```

Step 7: Set kernel arguments

Command: setenv bootargs 'console=ttyAMA0 console=tty0 root=/dev/ram rw'

```
=> setenv bootargs 'console=ttyAMA0 console=tty0 root=/dev/ram rw'
=>
```

Here,

- console=ttyAMA0: Redirect first serial port (on ARM architecture) to current session
- console=tty0: Redirect Qemu virtual serial port to current session
- root=/dev/ram: RAM device location
- rw : Mounting disk image in read/write mode

Step 8: Finally, boot the machine by providing bootz command with locations of Kernel, RootFS and DTB respectively..

Command: bootz 0x61000000 0x62000000 0x61a00000

```
=> bootz 0x61000000 0x62000000 0x61a00000
Kernel image @ 0x61000000 [ 0x000000 - 0x3fcd88 ]
## Loading init Ramdisk from Legacy Image at 62000000 ...
   Image Name:
   Image Type:   ARM Linux RAMDisk Image (uncompressed)
   Data Size:    1379328 Bytes = 1.3 MiB
   Load Address: 00000000
   Entry Point:  00000000
   Verifying Checksum ... OK
## Flattened Device Tree blob at 61a00000
   Booting using the fdt blob at 0x61a00000
   Loading Ramdisk to 7fd39000, end 7fe89c00 ... OK
   Loading Device Tree to 7fd32000, end 7fd3885d ... OK

Starting kernel ...

Booting Linux on physical CPU 0x0
```

The machine will start and after going through boot sequence, eventually present console login to the user. The user has to use the following credentials:

Username: root
Password: <none>

```
Welcome to Buildroot
buildroot login: root
#
```

After logging into the machine, the user can run common Linux commands.

Command: ps

```
# ps
PID   USER     COMMAND
    1 root     init
    2 root     [kthreadd]
    3 root     [rcu_gp]
    4 root     [rcu_par_gp]
    5 root     [kworker/0:0-eve]
    6 root     [kworker/0:0H]
```

In this manner, one can boot a machine using U-boot and remote TFTP server.

References:

- U-boot source: https://www.denx.de/wiki/U-Boot/SourceCode