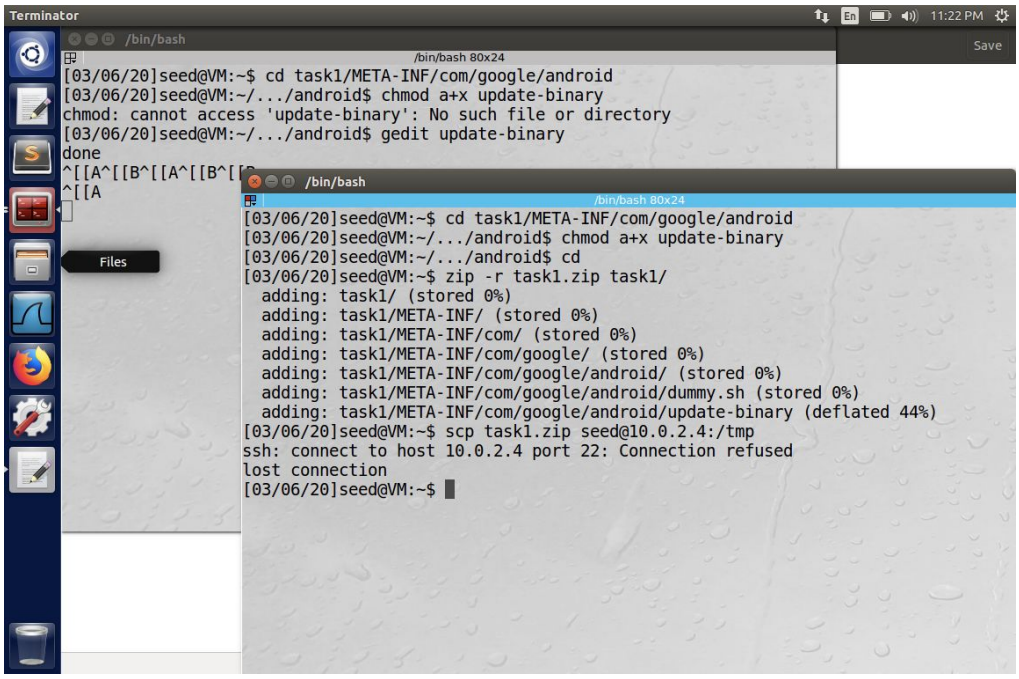
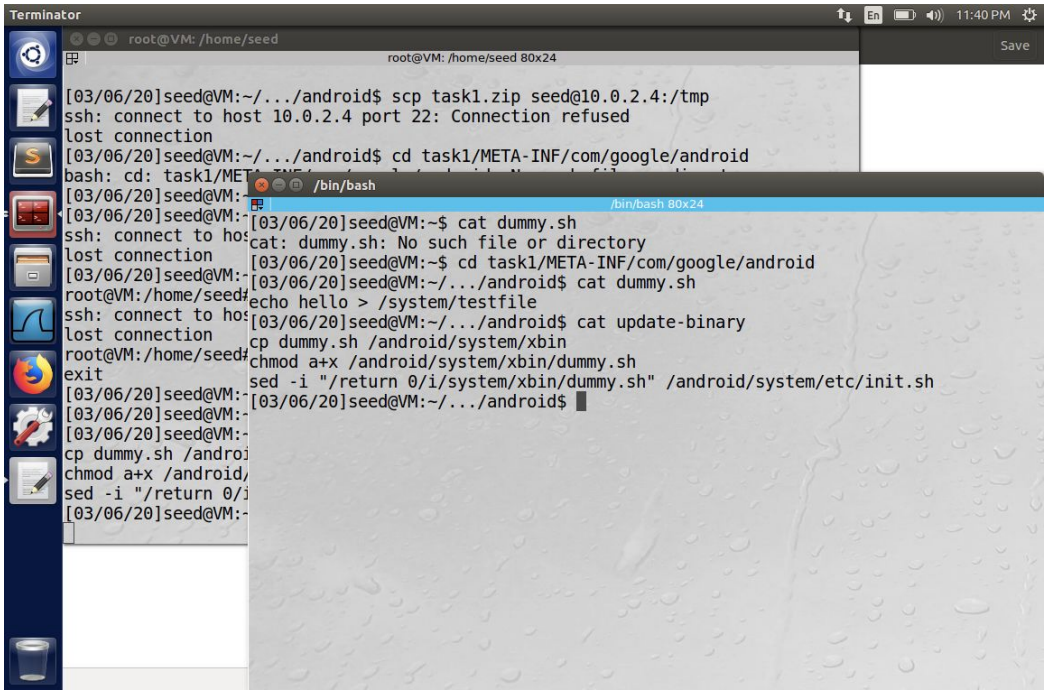
Lab 9
Karsen Diepholz
6 March 2020

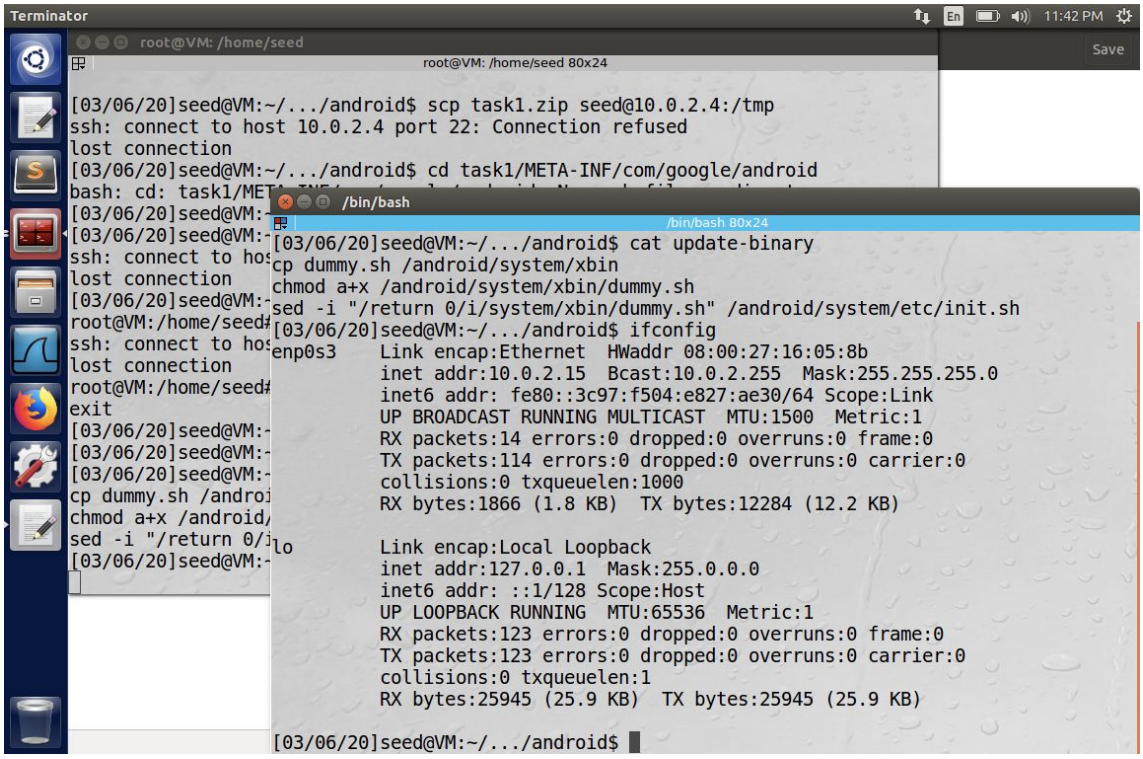
Task 1



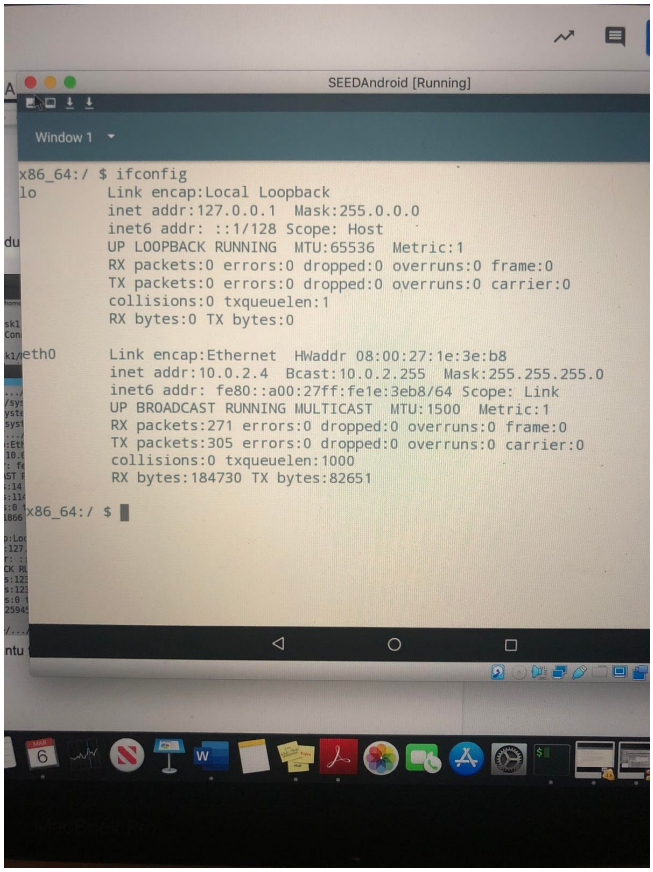
This shows me zipping the contents of task1 into a folder and sending them to the host (android) machine

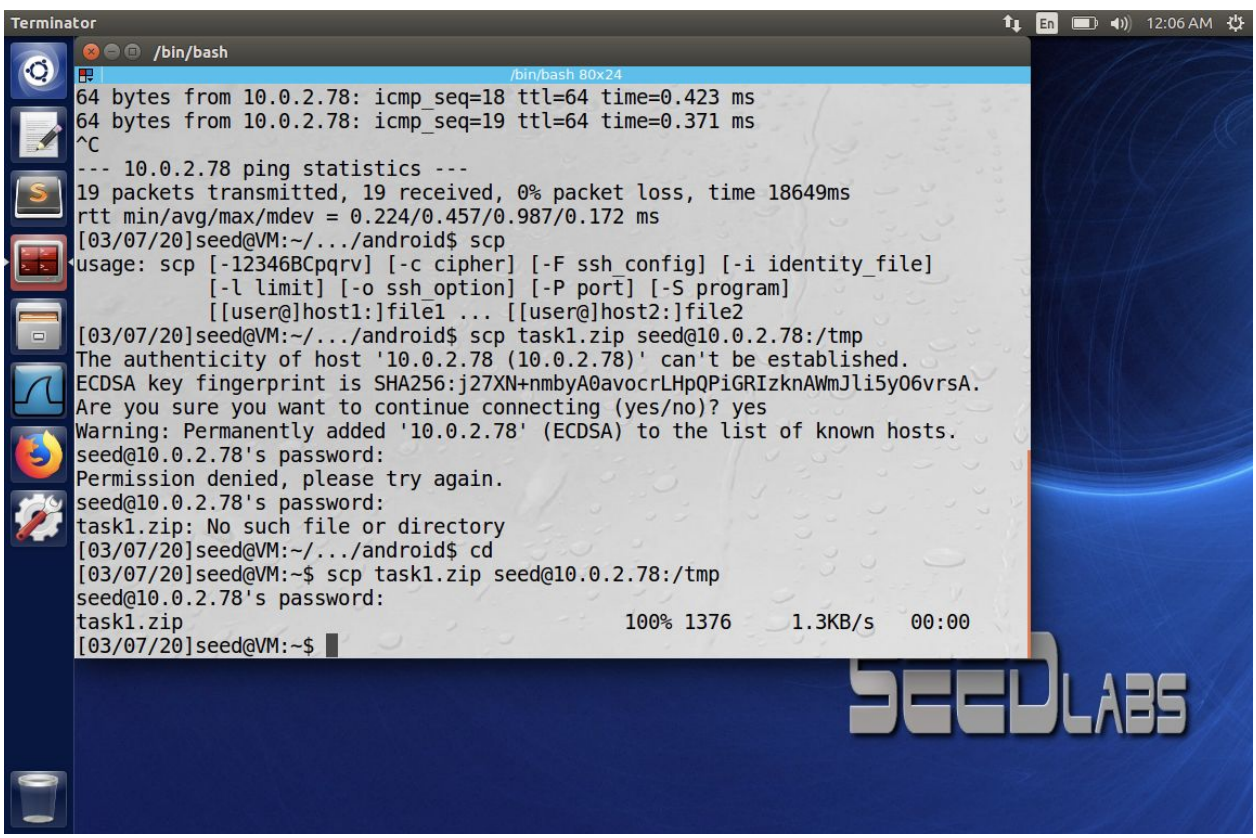


Using cat, we can see the contents of dummy.sh and update-binary in which i sent to the android VM

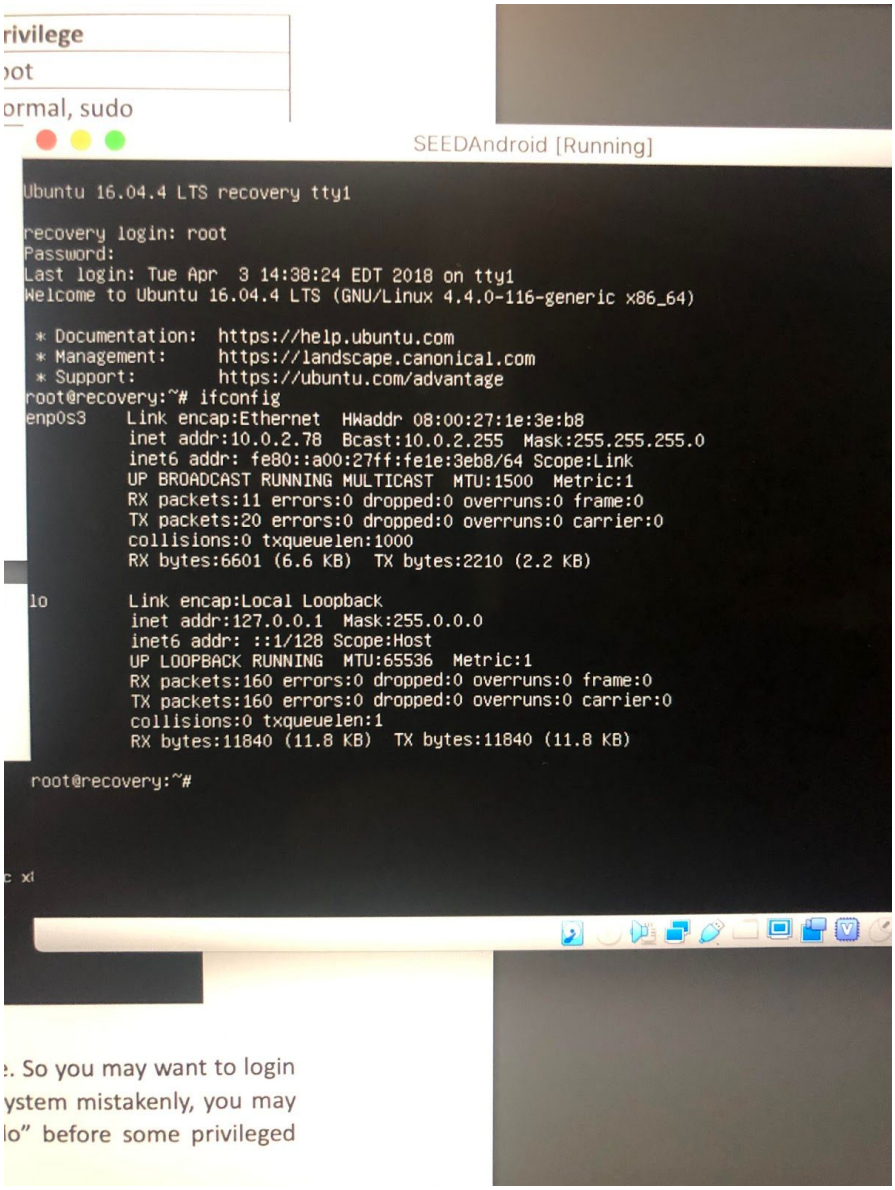


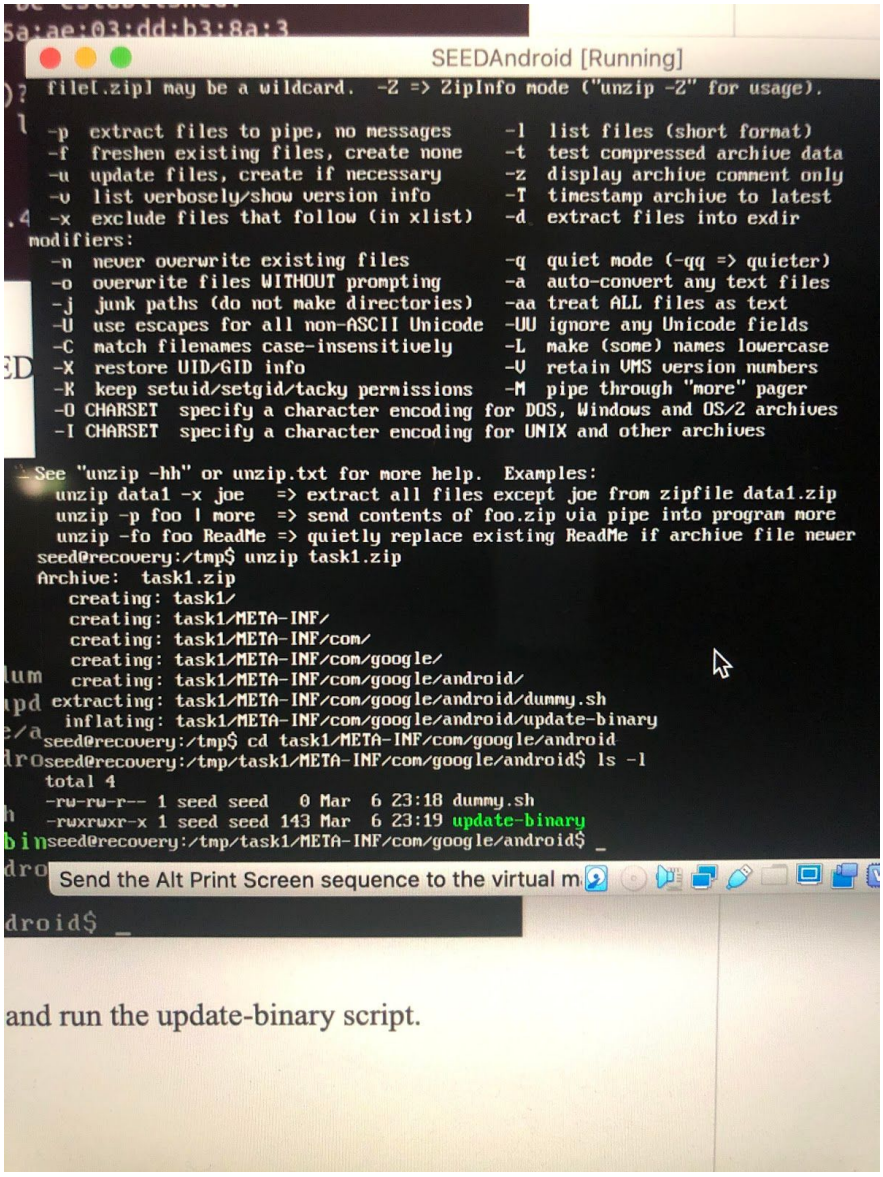
We are able to see the IP of our Ubuntu through this command


SEEDAndroid [Running]
Window 1
x86_64:/ $ ifconfig
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope: Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:0 TX bytes:0
eth0 Link encap:Ethernet HWaddr 08:00:27:1e:3e:b8
inet addr:10.0.2.4 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe1e:3eb8/64 Scope: Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:271 errors:0 dropped:0 overruns:0 frame:0
TX packets:305 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:184730 TX bytes:82651
x86_64:/ $


Terminator
12:06 AM
/bin/bash
/bin/bash 80x24
64 bytes from 10.0.2.78: icmp_seq=18 ttl=64 time=0.423 ms
64 bytes from 10.0.2.78: icmp_seq=19 ttl=64 time=0.371 ms
^C
--- 10.0.2.78 ping statistics ---
19 packets transmitted, 19 received, 0% packet loss, time 18649ms
rtt min/avg/max/mdev = 0.224/0.457/0.987/0.172 ms
[03/07/20]seed@VM:~/.../android$ scp
usage: scp [-12346BCpqrv] [-c cipher] [-F ssh_config] [-i identity_file]
[-l limit] [-o ssh_option] [-P port] [-S program]
[[user@]host1:]file1 ... [[user@]host2:]file2
[03/07/20]seed@VM:~/.../android$ scp task1.zip seed@10.0.2.78:/tmp
The authenticity of host '10.0.2.78 (10.0.2.78)' can't be established.
ECDSA key fingerprint is SHA256:j27XN+nmbyA0avocrLHpQPiGRIzknAWmJli5yO6vrsA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.78' (ECDSA) to the list of known hosts.
seed@10.0.2.78's password:
Permission denied, please try again.
seed@10.0.2.78's password:
task1.zip: No such file or directory
[03/07/20]seed@VM:~/.../android$ cd
[03/07/20]seed@VM:~$ scp task1.zip seed@10.0.2.78:/tmp
seed@10.0.2.78's password:
task1.zip 100% 1376 1.3KB/s 00:00
[03/07/20]seed@VM:~$
SEEDLABS

I successfully transferred the file using the android IP: 10.0.2.78





Successful unzip of task1 into the android VM

Task 2

Showing my processes...

Terminator 12:34 AM

Open Save

app_process.c | compile.sh | Application.mk | Android.mk

```
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := app_process
LOCAL_SRC_FILES := app_process.c
include $(BUILD_EXECUTABLE)
```

/bin/bash

/bin/bash 80x24

```
[03/07/20]seed@VM:~/.../android$ cat app_process.c
# include <stdio.h>
# include <stdlib.h>
# include <unistd.h>

extern char ** environ;

int main(int argc, char ** argv) {
 // Write the dummy file
 FILE * f = fopen("/system/dummy2", "w");
 if (f == NULL) {
    printf("Permission Denied.\n");
    exit(EXIT_FAILURE);
 }
 fclose(f);

 // Launch the original binary
 char * cmd = "/system/bin/app_process_original";
 execve(cmd, argv, environ);

 // execve () returns only if it fails
 return EXIT_FAILURE;
}
[03/07/20]seed@VM:~/.../android$
```

Terminator 12:35 AM

Open Save

app_process.c | compile.sh | Application.mk | Android.mk

```
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := app_process
LOCAL_SRC_FILES := app_process.c
include $(BUILD_EXECUTABLE)
```

/bin/bash

/bin/bash 80x24

```
 fclose(f);

 // Launch the original binary
 char * cmd = "/system/bin/app_process_original";
 execve(cmd, argv, environ);

 // execve () returns only if it fails
 return EXIT_FAILURE;
}
[03/07/20]seed@VM:~/.../android$ cat compile.sh
export NDK_PROJECT_PATH=.
ndk-build NDK_application_MK=./Application.mk
[03/07/20]seed@VM:~/.../android$ cat Application.mk
APP_ABI :=x86
APP_PLATFORM := android-21
APP_STL := stlport_static
APP_BUILD_SCRIPT := Android.mk
[03/07/20]seed@VM:~/.../android$ cat Android.mk
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := app_process
LOCAL_SRC_FILES := app_process.c
include $(BUILD_EXECUTABLE)
[03/07/20]seed@VM:~/.../android$
```

```
Terminator
Open  Save
app_process.c | compile.sh | Application.mk | Android.mk
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := app_process
LOCAL_SRC_FILES := app_process.c
include $(BUILD_EXECUTABLE)
```

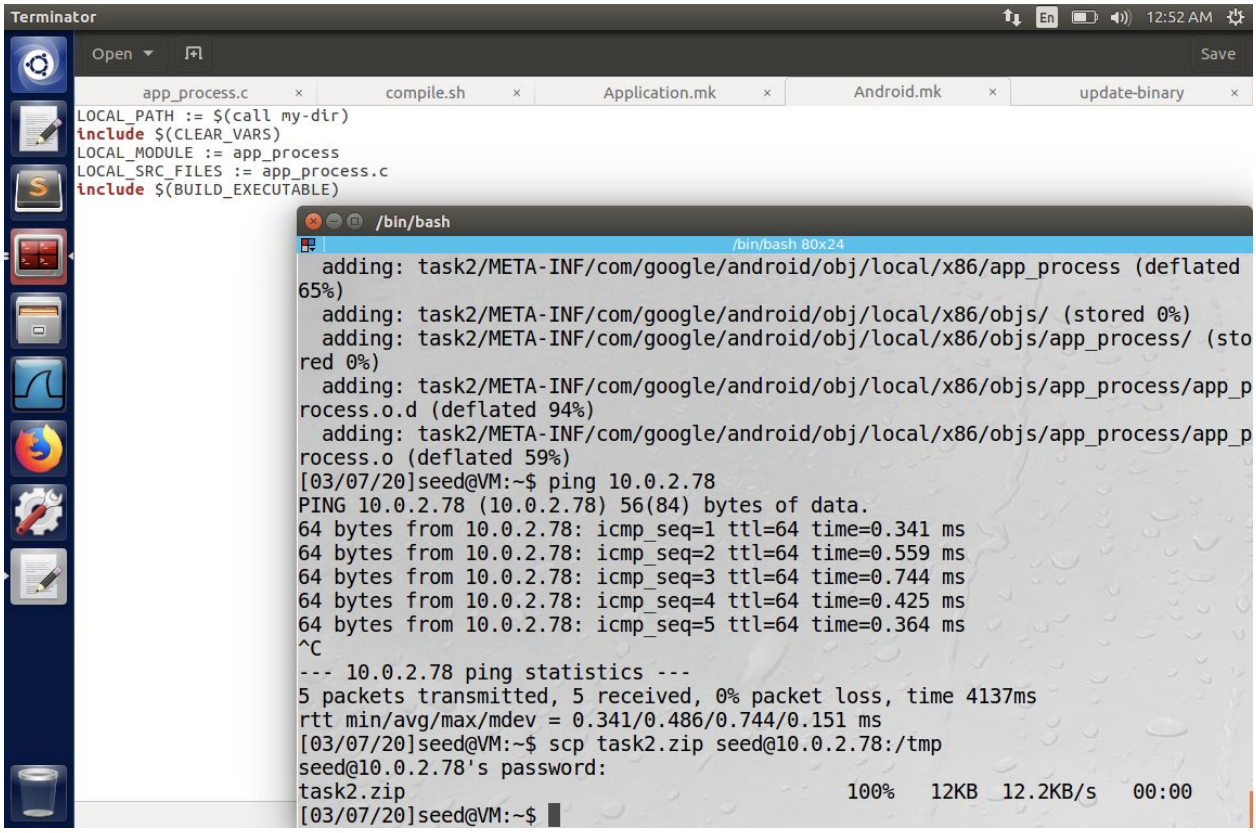
```
/bin/bash
/bin/bash 80x24
 return EXIT_FAILURE;
}
[03/07/20]seed@VM:~/.../android$ cat compile.sh
export NDK_PROJECT_PATH=.
ndk-build NDK_application_MK=./Application.mk
[03/07/20]seed@VM:~/.../android$ cat Application.mk
APP_ABI :=x86
APP_PLATFORM := android-21
APP_STL := stlport_static
APP_BUILD_SCRIPT := Android.mk
[03/07/20]seed@VM:~/.../android$ cat Android.mk
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := app_process
LOCAL_SRC_FILES := app_process.c
include $(BUILD_EXECUTABLE)
[03/07/20]seed@VM:~/.../android$ chmod a+x compile.sh
[03/07/20]seed@VM:~/.../android$ ls -l
total 16
-rw-rw-r-- 1 seed seed 140 Mar  7 00:34 Android.mk
-rw-rw-r-- 1 seed seed  98 Mar  7 00:33 Application.mk
-rw-rw-r-- 1 seed seed 461 Mar  7 00:32 app_process.c
-rwxrwxr-x 1 seed seed  72 Mar  7 00:33 compile.sh
[03/07/20]seed@VM:~/.../android$
```

Transferring task2.zip to android VM

```
Terminator
Open  Save
app_process.c | compile.sh | Application.mk | Android.mk | update-binary
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := app_process
LOCAL_SRC_FILES := app_process.c
include $(BUILD_EXECUTABLE)
```

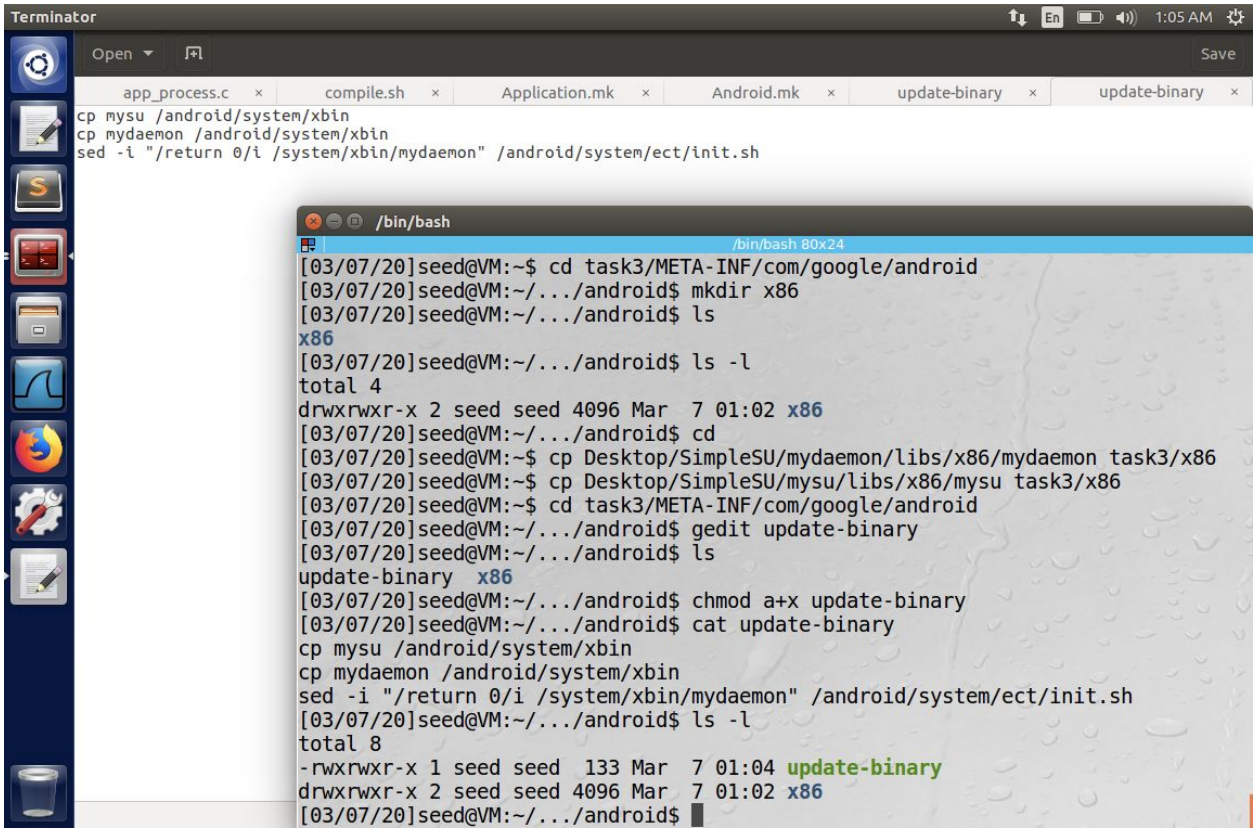
```
/bin/bash
/bin/bash 80x24
declare -x XMODIFIERS="@im=ibus"
[03/07/20]seed@VM:~/.../android$ export NDK_PROJECT_PATH=.
[03/07/20]seed@VM:~/.../android$ ndk-build NDK_APPLICATION_MK=./Application.mk
Compile x86    : app_process <= app_process.c
Executable     : app_process
Install        : app_process => libs/x86/app_process
[03/07/20]seed@VM:~/.../android$ ls -l
total 28
-rwxrwxr-x 1 seed seed  140 Mar  7 00:48 Android.mk
-rwxrwxr-x 1 seed seed   98 Mar  7 00:47 Application.mk
-rw-rw-r-- 1 seed seed  461 Mar  7 00:32 app_process.c
-rwxrwxr-x 1 seed seed   72 Mar  7 00:33 compile.sh
drwxrwxr-x 3 seed seed 4096 Mar  7 00:49 libs
drwxrwxr-x 3 seed seed 4096 Mar  7 00:49 obj
-rw-rw-r-- 1 seed seed  175 Mar  7 00:44 update-binary
[03/07/20]seed@VM:~/.../android$ ls
Android.mk  Application.mk  app_process.c  compile.sh  libs  obj  update-binary
[03/07/20]seed@VM:~/.../android$ cd libs
[03/07/20]seed@VM:~/.../libs$ ls
x86
[03/07/20]seed@VM:~/.../libs$ cd x86
[03/07/20]seed@VM:~/.../x86$ ls
app_process
[03/07/20]seed@VM:~/.../x86$
```

With this, we have given permission to execute compile.sh
Running compile.sh gives us app_process as well as the obj and libs file, which we will now zip and move into the android VM

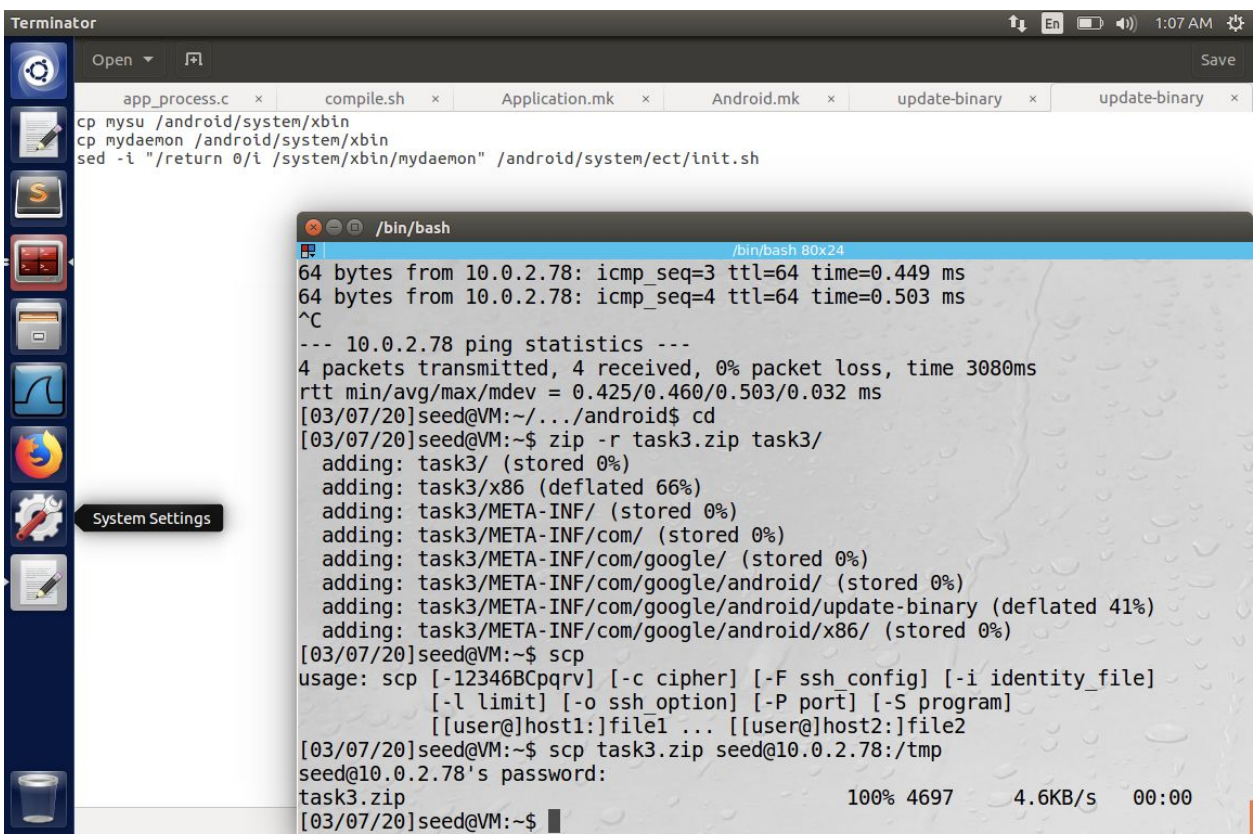


Task 2 has successfully been transferred to the android VM

Following running bash compile_all from the SudoSU.zip file provided, we create the following update-binary script and zip it to the task3 file



We send it to the Android VM…



Following sending task3.zip to the android VM, we use the recovery OS in order to launch the code we have sent and get root access #who am i

Sorry if the screenshots of android VM are low, it would not let me screenshot via my computer…