

Maszyna została skonfigurowana zgodnie z zaleceniami dla obu zadań.

Zadanie 1 Na maszynie Virbian2 strumień danych występuje w postaci niezaszyfrowanej jako pakiety przesyłane z ::1 port 22 do ::1 port 34796 poprzez IPv6.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	:::1	:::1	TCP	94	45066 -> 7777 [PSH, ACK] Seq=1 Ack=1 Win=512 Len=6 TSval=239
2	0.000154931	192.168.1.2	192.168.1.1	SSH	112	Client: Encrypted packet (len=44)
3	0.001308413	192.168.1.1	192.168.1.2	SSH	112	Server: Encrypted packet (len=44)
4	0.001422037	192.168.1.2	192.168.1.1	TCP	68	34796 -> 22 [ACK] Seq=45 Ack=45 Win=501 Len=0 TSval=19521937
5	0.001726695	:::1	:::1	TCP	94	7777 -> 45066 [PSH, ACK] Seq=1 Ack=7 Win=512 Len=6 TSval=239
6	0.001753115	:::1	:::1	TCP	88	45066 -> 7777 [ACK] Seq=7 Ack=7 Win=512 Len=0 TSval=239149831
7	5.129166698	PcsCompu_77:1d:cb		ARP	44	Who has 192.168.1.1? Tell: 192.168.1.2
8	5.129783924	PcsCompu_e6:4a:13		ARP	62	192.168.1.1 is at 08:09:27:e6:4a:13
9	5.132245929	PcsCompu_e6:4a:13		ARP	62	Who has 192.168.1.2? Tell: 192.168.1.1
10	5.132761815	PcsCompu_77:1d:cb		ARP	44	192.168.1.2 is at 08:09:27:77:1d:cb

Rysunek 1: Strumień danych w V2 w postaci niezaszyfrowanej

Pomiędzy maszyną Virbian2 a maszyną Virbian1 strumień danych występuje w postaci zaszyfrowanej jako pakiety przesyłane z 192.168.1.2 port 34796 do 192.168.1.1 port 22 poprzez IPv4.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	:::1	:::1	TCP	94	45066 -> 7777 [PSH, ACK] Seq=1 Ack=1 Win=512 Len=6 TSval=239
2	0.000154931	192.168.1.2	192.168.1.1	SSH	112	Client: Encrypted packet (len=44)
3	0.001308413	192.168.1.1	192.168.1.2	SSH	112	Server: Encrypted packet (len=44)
4	0.001422037	192.168.1.2	192.168.1.1	TCP	68	34796 -> 22 [ACK] Seq=45 Ack=45 Win=501 Len=0 TSval=19521937
5	0.001726695	:::1	:::1	TCP	94	7777 -> 45066 [PSH, ACK] Seq=1 Ack=7 Win=512 Len=6 TSval=239
6	0.001753115	:::1	:::1	TCP	88	45066 -> 7777 [ACK] Seq=7 Ack=7 Win=512 Len=0 TSval=239149831
7	5.129166698	PcsCompu_77:1d:cb		ARP	44	Who has 192.168.1.1? Tell: 192.168.1.2
8	5.129783924	PcsCompu_e6:4a:13		ARP	62	192.168.1.1 is at 08:09:27:e6:4a:13
9	5.132245929	PcsCompu_e6:4a:13		ARP	62	Who has 192.168.1.2? Tell: 192.168.1.1
10	5.132761815	PcsCompu_77:1d:cb		ARP	44	192.168.1.2 is at 08:09:27:77:1d:cb

Rysunek 2: Strumień danych V2 pomiędzy V1 w postaci zaszyfrowanej (trochę się ucieło)

Na maszynie Virbian1 strumień danych występuje w postaci niezaszyfrowanej jako pakiety przesyłane z 127.0.0.1 port 34796 do 127.0.0.1 port 7 poprzez IPv4.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	:::1	:::1	TCP	94	45066 -> 7777 [PSH, ACK] Seq=1 Ack=1 Win=512 Len=6 TSval=239
2	0.000154931	192.168.1.2	192.168.1.1	SSH	112	Client: Encrypted packet (len=44)
3	0.001308413	192.168.1.1	192.168.1.2	SSH	112	Server: Encrypted packet (len=44)
4	0.001422037	192.168.1.2	192.168.1.1	TCP	68	34796 -> 22 [ACK] Seq=45 Ack=45 Win=501 Len=0 TSval=19521937
5	0.001726695	:::1	:::1	TCP	94	7777 -> 45066 [PSH, ACK] Seq=1 Ack=7 Win=512 Len=6 TSval=239
6	0.001753115	:::1	:::1	TCP	88	45066 -> 7777 [ACK] Seq=7 Ack=7 Win=512 Len=0 TSval=239149831
7	5.129166698	PcsCompu_77:1d:cb		ARP	44	Who has 192.168.1.1? Tell: 192.168.1.2
8	5.129783924	PcsCompu_e6:4a:13		ARP	62	192.168.1.1 is at 08:09:27:e6:4a:13
9	5.132245929	PcsCompu_e6:4a:13		ARP	62	Who has 192.168.1.2? Tell: 192.168.1.1
10	5.132761815	PcsCompu_77:1d:cb		ARP	44	192.168.1.2 is at 08:09:27:77:1d:cb

Rysunek 3: Strumień danych w V1 w postaci niezaszyfrowanej

Zadanie 2 Opiszę tylko komendy, które nie zostały podane.

Za pomocą komendy „gpg –gen-key”generuje klucz prywatny dla user2 i powtarzam komendę eksportującą tylko dla user2.

Za pomocą komendy „scp 192.168.1.1:user1-pgp-key user1-pgp-key”kopiuje klucz user1 na maszynę V2.

Aby wejść w tryb edycji klucza i upewnić się, że jego funkcja skrótu i podpisanie w następujących krokach:

1. `gpg --edit-key user1@mail.example.com`
2. `fpr`
3. Funkcje skrótu sprawdzam po prostu skrót klucza na V1 i V2.
4. `sign` i `quit`

```

user@virbian: ~
gpg (GnuPG) 2.2.12; Copyright (C) 2018 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Secret key is available.

sec  rsa3072/434D536F6F9FCADF
     created: 2020-06-04  expires: 2022-06-04  usage: SC
     trust: ultimate    validity: ultimate
ssb  rsa3072/EFC8AF7C72F81B03
     created: 2020-06-04  expires: 2022-06-04  usage: E
[ultimate] (1). user1 <user1@mail.example.com>

gpg> fpr
pub  rsa3072/434D536F6F9FCADF 2020-06-04 user1 <user1@mail.example.com>
Primary key fingerprint: B915 5F6B C306 E98A AB98 0C03 434D 536F 6F9F CADF
gpg>
  
```

Rysunek 4: Klucz na maszynie V1

```

user@virbian: ~
gpg: key 434D536F6F9FCADF: public key "user1 <user1@mail.example.com>" imported
gpg: Total number processed: 1
gpg:      imported: 1
user@virbian: ~
gpg --edit-key user
user1@mail.example.com  user2@mail.example.com
gpg (GnuPG) 2.2.12; Copyright (C) 2018 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

pub  rsa3072/434D536F6F9FCADF
     created: 2020-06-04  expires: 2022-06-04  usage: SC
     trust: unknown     validity: unknown
sub  rsa3072/EFC8AF7C72F81B03
     created: 2020-06-04  expires: 2022-06-04  usage: E
[ unknown] (1). user1 <user1@mail.example.com>

gpg> fpr
pub  rsa3072/434D536F6F9FCADF 2020-06-04 user1 <user1@mail.example.com>
Primary key fingerprint: B915 5F6B C306 E98A AB98 0C03 434D 536F 6F9F CADF
gpg> B915 5F6B C306 E98A AB98 0C03 434D 536F 6F9F CADF
  
```

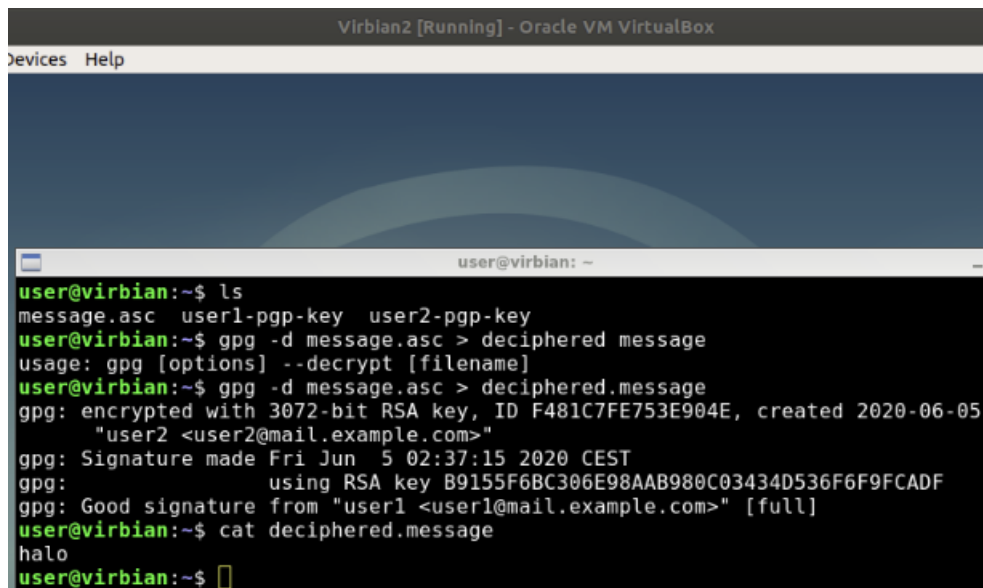
Rysunek 5: Klucz na maszynie V2

Następnie konfiguruje ssh na V2 i wykonuje podobne instrukcje, ale dla V1.

Tworzę plik message za pomocą:

1. `touch message`
2. `echo halo > message`

Następnie szyfruje, a następnie kopiuje na V2 za pomocą instrukcji `scp message.asc 192.168.1.2:message.asc`.



The screenshot shows a terminal window titled "Virbian2 [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```
user@virbian:~$ ls
message.asc user1-pgp-key user2-pgp-key
user@virbian:~$ gpg -d message.asc > deciphered message
usage: gpg [options] --decrypt [filename]
user@virbian:~$ gpg -d message.asc > deciphered.message
gpg: encrypted with 3072-bit RSA key, ID F481C7FE753E904E, created 2020-06-05
      "user2 <user2@mail.example.com>"
gpg: Signature made Fri Jun  5 02:37:15 2020 CEST
gpg:          using RSA key B9155F6BC306E98AAB980C03434D536F6F9FCADF
gpg: Good signature from "user1 <user1@mail.example.com>" [full]
user@virbian:~$ cat deciphered.message
halo
user@virbian:~$
```

Rysunek 6: Rozszyfrowana wiadomość