IPSec with RSA encription

Miguel Frade

Department of Informatics Engineering
Polytechnic Institute of Leiria

1. Introduction
   - Authentication modes on IPSec
   - Key sizes
2. RSA
   - Encription requirements
   - Encription configuration
3. Exercises
   - Configure R2
   - Another IPSec scenario for the same network
   - Configure 2 IPSec tunnels

Introduction

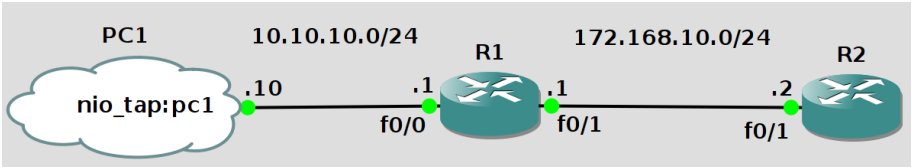
Authentication modes on IPSec

- `pre-share` – pre-shared key (PSK)
- `rsa-encr` – RSA encription
- `rsa-sig` – RSA signature

So far we've used `pre-share`
The goal for this class is to configure IPSec with `rsa-encr`

Introduction

Our goal is to configure this network



1. be able to ping from PC1 to R2
2. create an IPSec tunnel between R1 and R2 with **RSA encription** authentication

Introduction – Diffie-Hellman group

After authentication, DH is used to derivate a key between two IPSec peers

- DH Group 1 – 768-bits modulus
- DH Group 2 – 1024-bit modulus
- DH Group 5 – 1536-bit modulus
- DH Group 7 – 163-bit elliptical curve
- DH Group 14 – 2048-bit modulus
- DH Group 15 – 3072-bit modulus
- DH Group 16 – 4096-bit modulus
- DH Group 19 – 256-bit elliptical curve
- DH Group 20 – 384-bit elliptical curve
- DH Group 21 – 521-bit elliptical curve

Introduction – key robustness

Key size in bits with comparable cryptographic robustness:

Table: NIST recommended key sizes

Symmetric	RSA & DH	ECC
80	1 024	160
112	2 048	224
128	3 072	256
192	7 680	384
256	15 360	512

Symmetric keys should be “shared” with DH or ECC keys on the same row

Source: `http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57_part1_rev3_general.pdf`

RSA encription requirements

Routers must have a Full Qualified Domain Name (FQDN)

- setup domain `lab.org`

```
R1(config)$ ip domain-name lab.org
```

- do the same on router R2

And a RSA key pair

- generate a 1024 bits RSA key

```
R1(config)$ crypto key generate rsa general-keys modulus 1024
```

- do the same on router R2

RSA encription configuration

Share public keys between routers

- view R2 public key (needed to configure R1)

```
R2$ show crypto key mypubkey rsa
```

- result

```
% Key pair was generated at: 00:06:34 UTC Mar 1 2002
Key name: R2.lab.org
 Usage: General Purpose Key
 Key is not exportable.
 Key Data:

  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00BBE529      # this is the
  1EB6DA54 268420E5 42FAECC1 E0703D30 D600B96B 3BF16F87 DA8BEEEE 59405929      # public key
  DB919066 AC687C36 C8CB5EFA 1E982B41 256F5A82 C299A31B 7D7103D6 5AF0276A      # you must
  62C35076 90276B2C AD81F730 661B32B6 331F03A3 95FF8BAC 34C0C316 770895B7      # copy to
  7310726C 7726195D 81778D8A A71DAA90 1A199E2C 8C06A12A DB94F259 5F020301 0001 # router R1

% Key pair was generated at: 00:06:34 UTC Mar 1 2002
Key name: R2.lab.org.server
 Usage: Encryption Key
 Key is not exportable.
 Key Data:
  307C300D 06092A86 4886F70D 01010105 00036B00 30680261 00DBD2B1 2862BB25
  61134448 709EA9F2 589F4EE3 D151A971 7EAB4D6F 14E0A6EE C3E013DA 65D3009F
  2B6387D9 44F19338 D6923A00 C7D2B831 87B951C7 58FA8E62 56F86148 F8FF43C9
  B6E6A640 5F81FD70 EF0CE288 7560DA51 6207EA03 C0985D81 29020301 0001
```

RSA encription configuration

Setup R2 public key on R1

```
R1(config)$ crypto key pubkey-chain rsa
R1(config-pubkey-chain)$ addressed-key 172.168.10.2 encryption    # peer address (R2)
R1(config-pubkey-key)$ address 172.168.10.2                       # again peer address
R1(config-pubkey-key)$ key-string
Enter a public key as a hexidecimal number ....
# paste here the public key of router R2
```

```
R1(config-pubkey)$ 30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00BBE529
R1(config-pubkey)$ 1EB6DA54 268420E5 42FAECC1 E0703D30 D600B96B 3BF16F87 DA8BEEEE 59405929
R1(config-pubkey)$ DB919066 AC687C36 C8CB5EFA 1E982B41 256F5A82 C299A31B 7D7103D6 5AF0276A
R1(config-pubkey)$ 62C35076 90276B2C AD81F730 661B32B6 331F03A3 95FF8BAC 34C0C316 770895B7
R1(config-pubkey)$ 7310726C 7726195D 81778D8A A71DAA90 1A199E2C 8C06A12A DB94F259 5F020301 0001
```

```
R1(config-pubkey)$ quit                                           # exit config-pubkey mode
```

RSA encription configuration

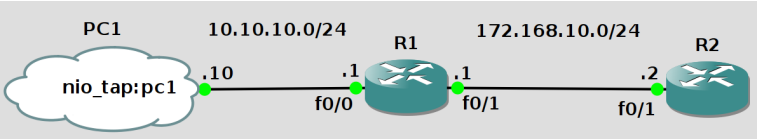
Rest of R1 the configuration

```
R1(config)$ crypto isakmp enable
R1(config)$ crypto isakmp policy 110
```

```
R1(config-isakmp)$ authentication rsa-encr    # rsa encription authentication
# Everything else is equal to pre-share mode
```

```
R1(config-isakmp)$ encryption des
R1(config-isakmp)$ group 5
R1(config-isakmp)$ hash sha
R1(config-isakmp)$ lifetime 86400
R1(config-isakmp)$ exit
R1(config)$ crypto isakmp key 0 cisco-ss address 172.168.10.2
R1(config)$ crypto ipsec transform-set TSET esp-des
R1(cfg-crypto-trans)$ mode tunnel
R1(cfg-crypto-trans)$ exit
R1(config)$ access-list 105 permit ip 10.10.10.0 0.0.0.255 host 172.168.10.2
R1(config)$ crypto map MYMAP 10 ipsec-isakmp
R1(config-crypto-map)$ match address 105
R1(config-crypto-map)$ set transform-set TSET
R1(config-crypto-map)$ set peer 172.168.10.2
R1(config-crypto-map)$ exit
R1(config)$ interface f0/1
R1(config-if)$ crypto map MYMAP
```

Exercise 1 – Configure R2



1. Configure R2, also with `rsa-encr` mode
2. Then test configuration
   - with `ping`
   - then do `show crypto ipsec sa`.
   - capture packets with Wireshark

Exercise 2 – For the same network scenario

Set up an IPSec tunnel between `R1` and `R2` to protect only the `HTTP` traffic originating from the network `10.10.10.0/24` with destination to `R2`.

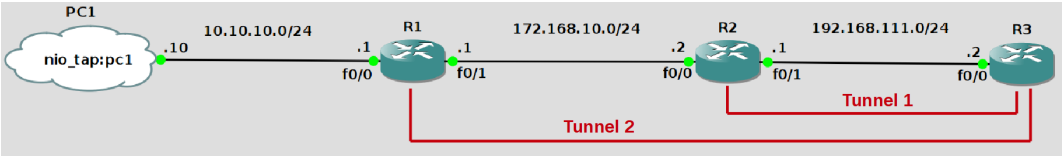
Data confidentiality of `HTTP` and `telnet` packets, must be ensured by `AES` and authentication through the `MD5` algorithm. The Security Associations lifetime must be `1 048 576` KB and the IKE tunnel should be configured with **rsa encription** and the key exchange must be done by the DH algorithm group `5`, with perfect forward secrecy. The integrity of ISAKMP communications should be guaranteed with the `SHA` and the confidentiality with `AES 256`.

For paramaters not specified above, use the defaults values.

Exercise 3 – Configure 2 IPSec tunnels

Setup 2 IPSec tunnels:

1. Tunnel 1: TSET = `ESP` with `AES` and `HMAC-SHA`, only for `ICMP` packets, with **rsa encription**
2. Tunnel 2: TSET = `AH` with `HMAC-MD5`, only for `HTTP` from `10.10.10.0/24` to host `192.168.111.2`, with the **pre shared key** `cisco-ss`



Test IPSec tunnels and verify the traffic with Wireshark