Asymmetric Cryptography: RSA

Juan Manuel Torres Palma

Universidad de Granada

January 4, 2016

Table of contents

Table of contents

Cryptography

Cryptography is a science that uses mathematics in a way that makes data impossible to read (Ciphertext) for those that are not in possesion of a key that allows them to read it.

- **Symmetric.** The same key is used to encrypt and decrypt.
- **Asymmetric.** Different keys are used to encrypt and decrypt.

Table of contents

Asymmetric Cryptography

- Bob: Uses Alice's **public key** to encrypt.
- Alice: Uses her **private key** to decrypt.
- Key is a number or set of numbers that applied to the message, makes it impossible to read.
- The bigger the key, the harder to break.

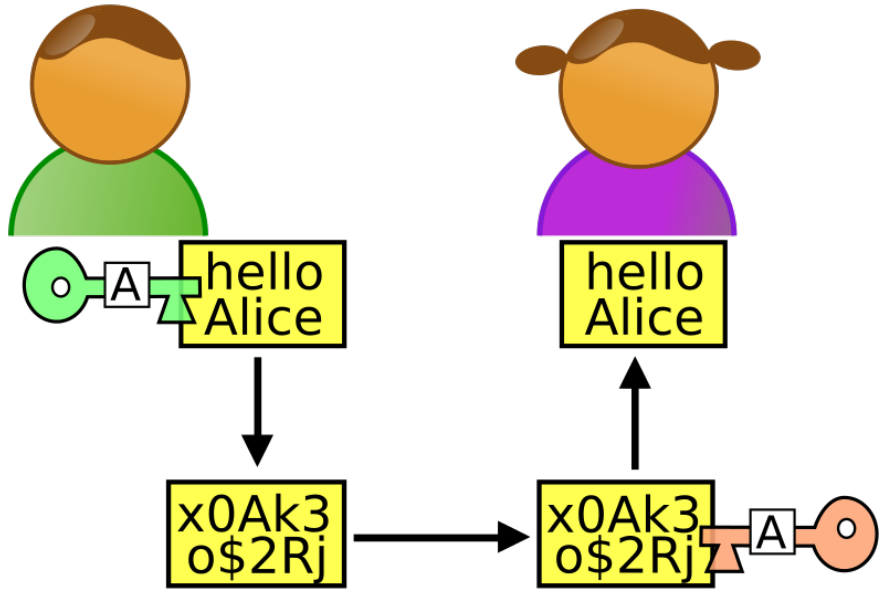

Table of contents

RSA-32[1]: Key generation

Key generation algorithm

$e \leftarrow 3$
repeat
    $p \leftarrow genprime()$
until $(p \ mod \ e) \neq 1$
repeat
    $q \leftarrow genprime()$
until $(q \ mod \ e) \neq 1$

$N \leftarrow p \times q$
$L \leftarrow (p-1)(q-1)$
$d \leftarrow modinv(e, \ L)$
return $(N, \ e, \ d)$

Generating keys

```
[jmtp@randy-betty src (master ✗)]$ ./gen-key
n = 3723700633, phi = 3723578548
Mod inverse: 2482385699
[jmtp@randy-betty src (master ✗)]$ cat key.pub
3723700633, 3
[jmtp@randy-betty src (master ✗)]$ cat key.pri
3723700633, 2482385699
```

[1] Used for simplicity, RSA-1024 recommended.

RSA-32: Encrypt

Encryption algorithm

$(n,\ e) \leftarrow readkey(pub)$
$c \leftarrow (m^e\ mod\ n)$
return c

Encypting a message

```
[jmtp@randy-betty src (master ✗)]$ echo -n Secret! > msg.txt
[jmtp@randy-betty src (master ✗)]$ ./encrypt -k key.pub msg.txt | tee crypt.txt
0008b98b000fb89d000ece3b00169b48000fb89d0017d14000008c61
[jmtp@randy-betty src (master ✗)]$ cat crypt.txt
0008b98b000fb89d000ece3b00169b48000fb89d0017d14000008c61
```

RSA-32: Decrypt

Decryption algorithm

$(n,\ d) \leftarrow readkey(pri)$
$m \leftarrow (c^d \ mod \ n)$
return m

Decrypting a message

```
[jmtp@randy-betty src (master ✗)]$ ./decrypt -k key.pri crypt.txt
Secret!
```

Table of contents

References

- **RSA algorithm theory explained:** Link.
- **SC-RSA implementation:** GitHub.