DE2: Homework 1

Lisa Halmschlager (1902224), Alexandra Lehner (1904041)

Nov 3, 2019

ASSIGNMENT:

Lets simulate a simplified message passing with https://www.ceu.edu Ceu.edu generates public and private keys and saves it to separate files. A visitor contacts ceu.edu, Ceu.edu sends over it's public key A visitor uses ceu.edu's public key to encrypt a message. CEU decrypts the message with its private key

Get in groups of two. One of the peers acts as ceu.edu, the other as visitor.

- - - CEU (Script 1) - - -

1. (At the webserver installation), ceu.edu generates a keypair. It saves it to two files in PEM format, private.pem and public.pem. Write an R script that does exactly this.
2. When a visitor contacts the CEU website, ceu.edu sends the public key to the visitor. Distribute the public PEM file among the team.

```r
library(PKI)

CEU_key <- PKI.genRSAkey(bits = 2048L) # generates RSA public/private key pair.

private.pem <- PKI.save.key(CEU_key, private=TRUE) # creates PEM representation of RSA key.
public.pem <- PKI.save.key(CEU_key, private=FALSE)

write(public.pem, file="ceu_RSA_pub") # Saves .pem public key to file
write(private.pem, file="ceu_RSA_prv") # Saves .pem public key to file
```

- - - VISITOR (Script 2) - - -

3. The visitor creates a message and encodes it with CEUs public key. Write an R script that does exactly this.
4. The visitor writes the encrypted message to a file and sends it to ceu.edu. Send the data back to your peer.

```r
library(PKI)

message = c("This is our first Data Engineering 2 assignment")

pub.pem.loaded <- scan("ceu_RSA_pub", what='list', sep='\n') # Load
pub.key.loaded <- PKI.load.key(pub.pem.loaded) # loads an RSA key in PKCS#1/8 PEM format.

encrypted.message <- PKI.encrypt(charToRaw(message), pub.key.loaded)

outf <- file("encodeddata.bin", "wb") # create wb (some kind of folder/file/Macbinary archive)
writeBin(encrypted.message, outf) # Write binary data to the outf connection
close(outf)
```

- - - CEU (Script 3) - - -

5. CEU (in a new R script) Reads back its private key from disk along with the encrypted message, decrypts the message and prints it to the screen. Write an R script that does exactly this.

```
prv.pem.loaded <- scan("ceu_RSA_prv", what='list', sep='\n') # Load private PEM
prv.key.loaded <- PKI.load.key(prv.pem.loaded) # loads an RSA key in PKCS#1/8 PEM format.

inf <- file("encodeddata.bin", "rb")
loaded.encrypted.message <- readBin(inf, what = "raw", n=1000) # read raw data
close(inf)

decrypted.message <- rawToChar(PKI.decrypt(loaded.encrypted.message, prv.key.loaded))
print(decrypted.message)
```

```
## [1] "This is our first Data Engineering 2 assignment"
```