Lab 1: Using Public Keys

Karlstad University

Mahdi Akil – mahdi.akil@kau.se
Samuel Wairimu – samuel.wairimu@kau.se

1 Introduction

Systems based on public-key cryptography use two different types of keys:

public key a key that can safely be shared with anyone

private key a key that should be kept secret

Data that has been encrypted with a public key can only be decrypted with the corresponding private key [4]. Similarly, for signing messages, a signature on a message can be generated by a private (signing) key and only be successfully verified using the corresponding public (verification) key.

In this lab you'll use public-key cryptography in two cryptosystems:

age or *Actually Good Encryption* is a simple and secure file encryption tool [1]

SSH or *Secure Shell* is a network protocol used to securely communicate with network services over un-trusted networks [5].

This lab should be done individually.

2 Part 1: Encrypted file

The goal of this part is to use *rage to encrypt a file* of your choosing (preferably a photo) and send an email containing the encrypted file to Mahdi and Samuel. You can find their email-addresses at the top of this lab.

To send the encrypted file:

1. Install *rage* by following the instructions from here. (Takes some time to finish)!
2. Create your own key-pair using the *rage-keygen* command.
3. Encrypt any file of your choosing using one public key from below:
   Mahdi's key: age195v6c8pkyt5erx5nl4malpuuec2gc8z3xewfkylvnan34g6q6u9sdzmckn
   Samuel's key: age1es0kgeef4gxutm0tu55a0lxp25gzflprr57va76wc83ez84d93asq35ug6
4. Send an email to both Mahdi and Samuel where only one of them should be able to decrypt the file.
   example: If you used Mahdi's public key to encrypt the file, your email should say: only Mahdi can decrypt the attached file.

3 Part 2: SSH Authentication

The goal of this part is to use SSH authentication to access a Linux server instead of your KauID. The Linux server you're going to use is hex.cse.kau.se. You can access it with `ssh kauid@hex.cse.kau.se`, where `kauid` is your KauID and by providing your password when prompted. On Linux and MacOS, `ssh` works from the command line. On Windows, PuTTY[1] is a popular client, or install the SSH client for PowerShell in Windows 10.

To change to SSH authentication:

1. On your client, generate a ssh key-pair using your client of choice. On Linux and MacOS, you have `ssh-keygen`. On Windows, you have `puttygen` as part of PuTTY. If possible, try to generate the key-pair using Ed25519, otherwise use RSA.
2. On the server, put your *public key* in the file "~/.ssh/authorized_keys". You likely have to create the file and its folder.
3. It should now be possible to `ssh kauid@hex.cse.kau.se` without entering your password.

[1] https://www.putty.org/

If you run into trouble, do your best to find a solution on your own. This will likely not be the last time you run into SSH-related issues. One good approach to debug is to run ssh with verbose output flags ("-vv").

4 Submission

Apart from completing the steps in the previous sections, we expect you to write a small report (at least 300 words). In this report you should:

- Provide evidence that you managed to *ssh* to *kauid@hex.cse.kau.se* and explain the steps that you took
- Read about [PGP] or *Pretty Good Privacy* is a widely known standard to sign, encrypt and decrypt documents, text, emails, etc [2, 3]. Explain how it works.
- What is the difference between *PGP, rage and SSH?* What protocol do you think is the hardest to use?

The report should be submitted as a PDF in Canvas by the end of the course. Late labs may not be corrected until the next examination period.

Acknowledgements

The following people have contributed to this lab specification: Mahdi Akil, Rasmus Dahlberg, Leonardo Martucci, Tobias Pulls, Artem Voronkov, and Samuel Wairimu.

References

[1] Ben Cox and Filippo Valsorda: age encryption tool. https://github.com/str4d/rage (2019), [Online; accessed 13-January-2022]

[2] Green, M.: Whatś the matter with pgp? https://blog.cryptographyengineering.com/2014/08/13/whats-matter-with-pgp/ (2014), [Online; accessed 13-January-2022]

[3] Wikipedia contributors: Pretty good privacy — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Pretty_Good_Privacy&oldid=935403723 (2020), [Online; accessed 13-January-2022]

[4] Wikipedia contributors: Public-key cryptography — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Public-key_cryptography&oldid=935659390 (2020), [Online; accessed 13-January-2022]

[5] Wikipedia contributors: Secure shell — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Secure_Shell&oldid=935567873 (2020), [Online; accessed 13-January-2022]