EITA25 - Lab 3

André Frisk

March 11, 2022

Preparatory Questions

1.
   - **Study the Gnu Privacy Handbook, available at http://www.gnupg.org/gph/en/manual.html. Do not go into any details, but make sure that you have a feeling for what it does and how to work with GPG. The handbook will be available at the lab computers**: GnuPG is a tool for secure communication. This chapter is a quick-start guide that covers the core functionality of GnuPG. This includes keypair creation, exchanging and verifying keys, encrypting and decrypting documents, and authenticating documents with digital signatures. It does not explain in detail the concepts behind public-key cryptography, encryption, and digital signatures.

     GnuPG uses public-key cryptography so that users may communicate securely. In a public-key system, each user has a pair of keys consisting of a private key and a public key. A user's private key is kept secret; it need never be revealed. The public key may be given to anyone with whom the user wants to communicate. GnuPG uses a somewhat more sophisticated scheme in which a user has a primary keypair and then zero or more additional subordinate keypairs. The primary and subordinate keypairs are bundled to facilitate key management and the bundle can often be considered simply as one keypair.

   - **Make sure you understand the concept of public key cryptography, symmetric cryptography, hash functions and signatures**: Public-key cryptography, or asymmetric cryptography, uses a public key to encrypt data and a private key to decrypt information.

     Symmetric encryption uses the same key to encrypt and decrypt data making it very easy to use.

     A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but output is always of fixed length. Computationally hash functions are much faster than a symmetric encryption.

     A digital signature is a mathematical scheme for verifying the authenticity of digital messages or documents. A valid digital signature, where the prerequisites are satisfied, gives a recipient very strong reason to believe that the message was created by a known sender (authenticity), and that the message was not altered in transit (integrity).

   - **In the handbook, look up the commands to: create keypairs; generate key revocation certificates; import, export, and get fingerprints of public keys; and encrypt, decrypt, sign, and verify files. You don't need to remember them, but you should know where to look for them**:

```
alice% gpg --gen-key
alice% gpg --output revoke.asc --gen-revoke mykey
alice% gpg --import blake.gpg,
alice% gpg --output alice.gpg --export alice@cyb.org
--fingerprint
alice% gpg --output doc.gpg --encrypt --recipient blake@cyb.org doc
blake% gpg --output doc --decrypt doc.gpg
alice% gpg --output doc.sig --sign doc
--verify ELLER blake% gpg --output doc --decrypt doc.sig
```

2.
   - **What is a key revocation certificate?** A key revocation certificate is a special, revoked copy of your public key. You can generate a key revocation certificate and store it for future use.
   - **Why is it a good idea to have a revocation certificate?** Key revocation certificates are useful if you've forgotten the passphrase to your private key and you need some way to "disable" or revoke that key.

3.
   - **What is the web of trust and how does it work? Note! This term is also used in some other contexts, ensure you read about the web of trust related to PGP/GnuPG!** In cryptography, a web of trust is a concept used in PGP, GnuPG, and other OpenPGP-compatible systems to establish the authenticity of the binding between a public key and its owner. In the web of trust model, responsibility for validating public keys is delegated to people you trust. For example, suppose Alice has signed Blake's key, and Blake has signed Chloe's key and Dharma's key. If Alice trusts Blake to properly validate keys that he signs, then Alice can infer that Chloe's and Dharma's keys are valid without having to personally check them. She simply uses her validated copy of Blake's public key to check that Blake's signatures on Chloe's and Dharma's are good. In general, assuming that Alice fully trusts everybody to properly validate keys they sign, then any key signed by a valid key is also considered valid. The root is Alice's key, which is axiomatically assumed to be valid.
   - **What are the different trust levels in the web of trust?**
     (a) **unknown**: Nothing is known about the owner's judgment in key signing. Keys on your public keyring that you do not own initially have this trust level.
     (b) **none**: The owner is known to improperly sign other keys.
     (c) **marginal**: The owner understands the implications of key signing and properly validates keys before signing them.
     (d) **full**: The owner has an excellent understanding of key signing, and his signature on a key would be as good as your own.
   - **Look up the commands used to set trust on keys in GnuPG**:

     ```
     alice% gpg --edit-key blake
     Command> trust
     ```

   - **What is the difference between trust and validity?** Trust in the key's owner and the key's validity are indicated to the right when the key is displayed. Example: q/f. Trust in the owner is displayed first and the key's validity is second. The four trust/validity levels are abbreviated: unknown (q), none (n), marginal (m), and full

(f). Validity is connected to trust, with the q/f example the persons key is fully valid since someone signed it. Validity means that how valid the key might be.

4. **Read about Nmap so that you feel comfortable working with it. A good online resource is http://nmap.org. Read the lab problems below, and figure out what commands you need to use during the lab.** Nmap is used to discover hosts and services on a computer network by sending packets and analyzing the responses (through port scanning). Can be used to detect network security in a computer. Typically it is used to see which TCP ports different computers answers to.

   ```
   Pingscan: maybe -sn (No port scan) (https://nmap.org/book/man-host-discovery.html)
   Hostscan (online): -sL (List scan)
   ```

5. **Read about firewall and iptables and different rule settings such as drop, accept and different options that can be applied to iptables rules and their usages.**: In computing, a firewall is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted network and an untrusted network, such as the Internet.

   iptables is a user-space utility program that allows a system administrator to configure the IP packet filter rules of the Linux kernel firewall, implemented as different Netfilter modules. The filters are organized in different tables, which contain chains of rules for how to treat network traffic packets.

   DROP will drop any packet from the incoming IP address which is specified. ACCEPT means that the default policy for that chain, if there are no matching rules, is to allow the traffic.

6. **Get acquainted with POP3, SMTP, telnet, FTP and SSH. You do not have to learn any details about the protocols, but you should know what they are and in which context they are used**:

   (a) **POP3** – In computing, the Post Office Protocol (POP) is an application-layer Internet standard protocol used by e-mail clients to retrieve e-mail from a mail server.[2] POP version 3 (POP3) is the version in common use. The Post Office Protocol provides access via an Internet Protocol (IP) network for a user client application to a mailbox (maildrop) maintained on a mail server.

   (b) **SMTP** – The Simple Mail Transfer Protocol (SMTP) is an internet standard communication protocol for electronic mail transmission. Mail servers and other message transfer agents use SMTP to send and receive mail messages.

   (c) **Telnet** – Telnet is an application protocol used on the Internet or local area network to provide a bidirectional interactive text-oriented communication facility using a virtual terminal connection. User data is interspersed in-band with Telnet control information in an 8-bit byte oriented data connection over the Transmission Control Protocol (TCP).

   (d) **FTP** – The File Transfer Protocol (FTP) is a standard communication protocol used for the transfer of computer files from a server to a client on a computer network. FTP is built on a client–server model architecture using separate control and data connections between the client and the server.

(e) **SSH** – The Secure Shell Protocol (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network. Its most notable applications are remote login and command-line execution.

7. **Get acquainted with Wireshark. Understand how to enter filter strings. Information can e.g., be found at http://wiki.wireshark.org.** To find a string within a packet, click on Edit > Find Packet. Under "Find By:" select "string" and enter your search string in the text entry box. You'll probably want to leave "Case sensitive" unchecked. Under "Search in", the default is "Packet list" but that will only find a string that appears in the Info column of the Packet List pane, which is the one-line-per-packet summary view. There is a lot more information in most packets than what appears in the packet list Info column, so try "Packet details" and "Packet bytes".

   Or klick on the "Apply a display filter".

Lab Questions

1. **Which public key algorithm do you use for signatures and encryption?** RSA
2. **What does the option armor do?** Causes output to be generated in an ASCII-armored format similar to uuencoded documents. It makes the key readable
3. **Create a key revocation certificate and store this in a secure place, i.e., only you should be able to read it**: Yes
4. **Why is the encrypted file much smaller than the original file? Is the text readable? (You can open file with vi or vim text editors. Which types of algorithms are used when encrypting the file? Hint: Read about Hybrid ciphers in the Gnu Privacy Handbook.)**: Encryption does the job by scrambling the letters to make it appear as if there are no patterns present. Now, consider this: Compression makes a file smaller by removing redundancy. Encryption erases redundancy by scrambling letters. After you encrypt a file, there's no redundancy for a compression algorithm to remove.

   No

   Uses both a symmetric cipher and a public-key cipher.
5. **Sign the file encrypt_me.txt. Then verify the signature using the option verify. What command did you use to create the signature?** gpg –output signed –sign encrypt_me.txt
6. **What different ways are there to create signatures in GPG? How do they differ?** –clearsign. The option –clearsign causes the document to be wrapped in an ASCII-armored signature but otherwise does not modify the document.

   –detached. A signed document has limited usefulness. Other users must recover the original document from the signed version, and even with clearsigned documents, the signed document must be edited to recover the original. Therefore, there is a third method for signing a document that creates a detached signature, which is a separate file.
7. **Change one letter in the signed document and then verify the signature again. Is the signature valid?** No, unknown compressed algorithm.
8. **Import the other group public key into your keyring and sign it using your private key. Note that signing a key is not the same thing as signing a file**: Yes

9. **Create a text file and insert a message inside it that you want to send to the other group. Encrypt the created file using gpg command but this time include two options in your command: -u "Sender username" and -r "Receiver user name". Replace the sender and receiver user names with your own name and the other group's name (the names you and the other group used when generating the keys)**: Yes

10. **Send the encrypted files to each other using email or scp command. Can you decrypt and read both the encrypted files you sent and received? Why/why not?** Yes we could. As we verified each other with the -u -r options, which makes it possible for Jonathan to open and read my file as I have gained him permission, this using his public key. Encrypt with their public key and they can use their private key to decrypt it. And visecersa for their situation.

11. **Consider the four cases in the table below (one for each row). For each row, set the trust in your friends according to the row, and fill in the corresponding validity of David's public key (note that you need to set the trust and check the validity of David's key with commands)**:

    (a) marginal

    (b) full

    (c) marginal

    (d) full

12. **Determine, using a ping scan, which hosts are online on the local network (local network to scan: 10.0.0.0/24)**: 9 hosts 10.0.0.9 (10.0.0.10) 10.0.0.30 10.0.0.62 10.0.0.67 10.0.0.72 10.0.0.76 10.0.0.105

13. **Do a port scan for the hosts that are online. (You can use the option -T4 if the port scan seems slow). What services are running on the different hosts? Give some examples from each host, you don't have to write down all output**: ssh, rpcbind, https-alt

14. **Now ask other group to ping your IP address. Are they able to ping your computer why or why not?** No, they are blocked in my iptable. DROPped the IPaddress

15. **Try to ping other group, are you able to ping them? why or why not?** No, as we drop the answer

16. **Does the new rule added in the beginning of the list or end of the list? what does -A option do? what if you use -I instead of -A? Would the order be different?** -A is added at the end of the list, -I is at the top of the list.

17. **Now you have both drop and accept rules for other group for outgoing traffic. Are you able to ping other group or it will be dropped? How about if the accept rule was before the drop rule in the list?** OUTPUT ACCEPT before INPUT DROP then it works, otherwise not. As DROP will drop the packets directly, highest priority in the list as iptables goes from up to down.

18. **Start Wireshark with by entering the command sudo wireshark, in the terminal, and start getting acquainted with the program. Visit a few web pages e.g., http://www.marsta.nu/ and analyze the traffic. Try a few different filters, e.g.**

http to only see HTTP traffic. Is http traffic secure? No, as HTTPS (Hypertext Transfer Protocol Secure) provide security, not HTTP.

19. **Log into the other group VM by using SSH (search the usage of SSH command online). What can you say about the traffic? Is it secure?** Yes, it uses TCP and SSHv2 protocols and the SSH packets are encrypted

20. **Visit the site https://google.com, which uses TLS, and look at the traffic in Wireshark. Look for the TLS Client Hello message. Can an eavesdropper know which site you are visiting? Can you think of a situation when this is undesirable?** Yes. You can see the IP.