Securing Data and Communication in OS Linux

- Basic terms
  - Untrusted network and security of communication
  - Symmetric cryptography
  - Asymmetric cryptography
  - Digital signature
  - Public Key Infrastructure (PKI)
- Gnu Pricavy Guard (GPG)
- Communication tunneling using SSH protocol
  - Local port forwarding
  - Remote port forwarding
  - Dynamic port forwarding

Untrusted Network

- Risk of containing untrusted nodes:
  - We do not have control over them,
  - They can sniff traffic, execute attacks, etc.
- Out of administrator’s reach.
- Typically behind a border router or firewall.
- Need for protection of transmitted data.
- Example: Internet, public WiFi network.

Security of Communication

- Encryption – hiding message contents from a potential attacker.
  - Symmetric encryption,
  - Asymmetric encryption.
- Authentication – verifying the identities of communication parties
  - Shared secret (password),
  - Asymmetric cryptography (key pair).
- Integrity – securing against unsolicited changes of message contents during transmission.

Symmetric Cryptography

- Using the same key for both encryption and decryption
- Communicating parties need to agree on the key securely (critical is the exchange of the key)
- The selected key must be hard to guess
- Strength of a symmetric key – 128-bit key means 2^128 possible keys

Disadvantages of Symmetric Cryptography

- Potential attacker can obtain the key during the key exchange between the communicating parties

- Difficult to scale
  - For n participants, n(n-1)/2 keys are needed to guarantee secure and private communication between each other.

Asymmetric Cryptography

- Using two types of keys (key pair)
  - one for encryption, one for decryption
- Public key
  - Message recipient provides it to other people
  - Sender uses it to encrypt the message
- Private key
  - Not known to the sender
  - Can be used to decrypt a message that was encrypted with a matching public key
  - Only the recipient knows it

Digital Signature

- Calculation of a unique value for a document
  - Many-to-one function = hash function
  - Used to verify authenticity of a document
- Result of hash function
- Hash function must meet two conditions:
  - Two documents must not have the same resulting hash
  - It must not be possible to obtain the original document from the resulting hash

Public Key Infrastructure (PKI)

- Set of rules, procedures, and technical and organisational measures related to:
  - creation,
  - management,
  - usage,
  - sharing,
  - storage,
  - revocation,
- of encryption keys and digital certificates.

Gnu Privacy Guard

- GnuPG is a tool that can be used to increase security of a system.
  - Command `gpg`
- In order to secure communication between the communicating parties, it is needed that both sides use it.
- GnuPG can be used to accomplish operations described by PKI.

GPG: Key Creation and Management

- options
  - `--gen-key`
    - Key creation, default settings
  - `--full-gen-key`
    - Possibility to specify key usage
    - Key size selection
  - `--edit-key`
    - CLI for key management
    - `help, fingerprint`
    - `disable, enable`
    - `passwd, addkey`

GPG: Key Exchange

- options
  - `--list-key`
    - list of public keys
  - `--export`
    - public key export
  - `--import`
    - public key import
  - `--armor`
    - export in ASCII armor format

GPG: Encrypting/Decrypting Files

- options
  - `--encrypt`
    - encryption using a key pair
    - asymmetric encryption
  - `--decrypt`
    - decryption of the file
  - `--symmetric`
    - symmetric encryption of the file

GPG: Signature Creation and Verification

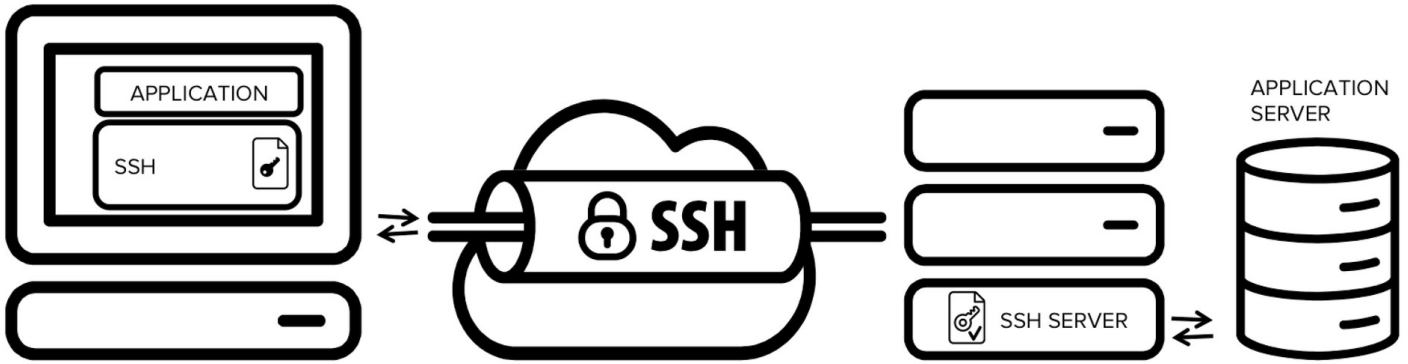
- options
  - `--sign`
    - signature of an encrypted file
  - `--clearsign`
    - signature without encryption
  - `--detach-sign`
    - separate the signature from the signed file during signing
  - `--verify`
    - signature verification

Traffic Tunneling

- Creation of a secure (encrypted) connection (“tunnel”) via an untrusted network.
- Possibility to secure any communication flowing through the “tunnel”.
  - Example: remote access, protocols that do not provide encryption, etc.
- SSH protocol supports “tunneling”.

Port Forwarding Using SSH

- The communication is forwarded via a secure SSH connection – i.e. “SSH tunnel”.
- Access to an available SSH server is needed.



https://www.ssh.com/ssh/tunneling/

SSH: Local Port Forwarding

```
ssh –L [laddr:]lport:host:port login@ssh.server
```

- Connections to the given local TCP port are forwarded to the given destination host and port via server “ssh.server”.
- Connection between local machine and server “ssh.server” is secured by protocol SSH.
- Privileged ports can be forwarded only by “root”.

SSH: Remote Port Forwarding

```
ssh –R [raddr:]rport:host:port login@ssh.server
```

- Connections to the given TCP port on the remote host (“ssh.server”) are forwarded to the given destination host and port on the local side.
- Created socket on the remote side typically listens on the local interface.
  - Possible to change, parameter `raddr`,
  - Setting parameter `GatewayPorts`.

SSH: Dynamic Port Forwarding

```
ssh –D [laddr:]lport login@ssh.server
```

- Specifies local dynamic forwarding on the application layer.
- Connections to the given local TCP ports are forwarded via server “ssh.server”.
- The destination of the forwarding is specified by the application protocol.
  - Supported protocols: SOCKS4, SOCKS5,
  - SSH functions as a SOCKS proxy.

References

- Manual pages
  - man gpg
  - man ssh
  - man sshd
  - man ssh_config
- GnuPG manual:
  - https://www.gnupg.org/gph/en/manual.html
- GnuPG Commands – Examples
  - http://www.spywarewarrior.com/uiuc/gpg/gpg-com-4.htm
- GPG Tutorial
  - https://futureboy.us/pgp.html
- SSH Tunnels
  - https://www.ssh.com/ssh/tunneling/