CS 35L

Software Construction Laboratory

Lecture 8.2

22nd May, 2019

Logistics

- Assignment 10 Signup Sheet
  - https://docs.google.com/spreadsheets/d/19bPoaFoi9rWZ-05hTJgUAqZPKWlAetRjFMniljwmZBs/edit?usp=sharing
- Assignment 8 Deadline
  - 29th May, 2019 – 11:55pm
- Hardware requirement for Week 8
  - Seeed Studio BeagleBone Green Wireless Development Board
  - Today's class

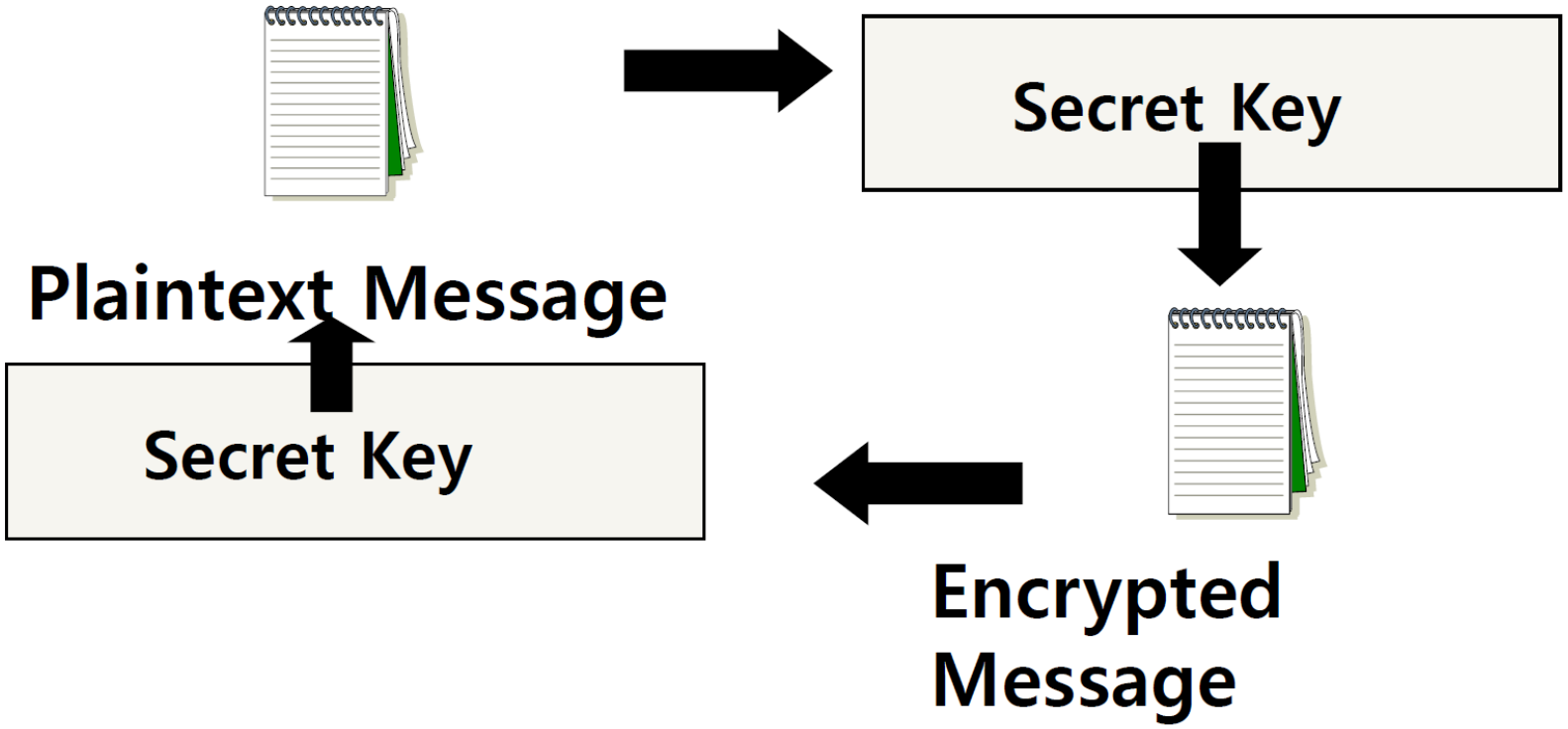
Review - Previous Lab

- SSH
  - Symmetric Key Encryption
  - Asymmetric Key Encryption
  - Server Validation
  - User Authentication

Digital Signature

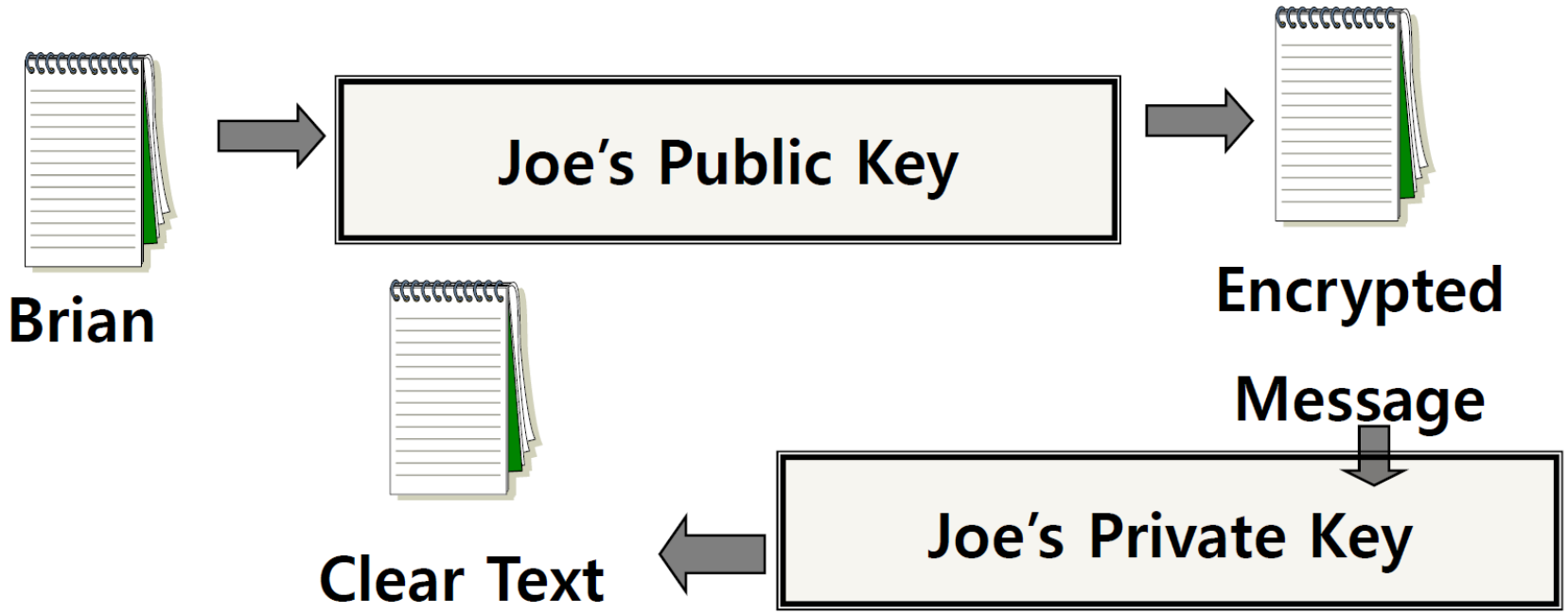
Secret Key (symmetric) Cryptography

- A single key is used to both encrypt and decrypt a message



Public Key (Asymmetric) Cryptography

- Two keys are used: a public and a private key. If a message is encrypted with one key, it has to be decrypted with the other.



Digital Signature

- An electronic stamp or seal
  - almost exactly like a written signature, except more guarantees!
- Is appended to a document
  - Or sent separately (detached signature)
- Ensures data integrity
  - document was not changed during transmission
  - intended to solve the problem of tampering and impersonation in digital communications.
- Based on Public Key Cryptography
- Reference

Steps for Generating a Digital Signature

SENDER:

- Generate a Message Digest
  - The message digest is generated using a set of hashing algorithms
  - Even the slightest change in the message produces a different digest
- Create a Digital Signature
  - The message digest is encrypted using the sender's private key. The resulting encrypted message digest is the digital signature
- Attach digital signature to message and send to receiver

Steps for Generating a Digital Signature

RECEIVER:

- Recover the Message Digest
  - Decrypt the digital signature using the sender's public key to obtain the message digest generated by the sender
- Generate the Message Digest
  - Use the same message digest algorithm used by the sender to generate a message digest of the received message
- Compare digests (the one sent by the sender as a digital signature, and the one generated by the receiver)
  - If they are not exactly the same => the message has been tampered with by a third party
  - We can be sure that the digital signature was sent by the sender (and not by a malicious user) because only the sender's public key can decrypt the digital signature and that public key is proven to be the sender's through the certificate.
  - If decrypting using the public key renders a faulty message digest, this means that either the message or the message digest are not exactly what the sender sent.

Digital Signature

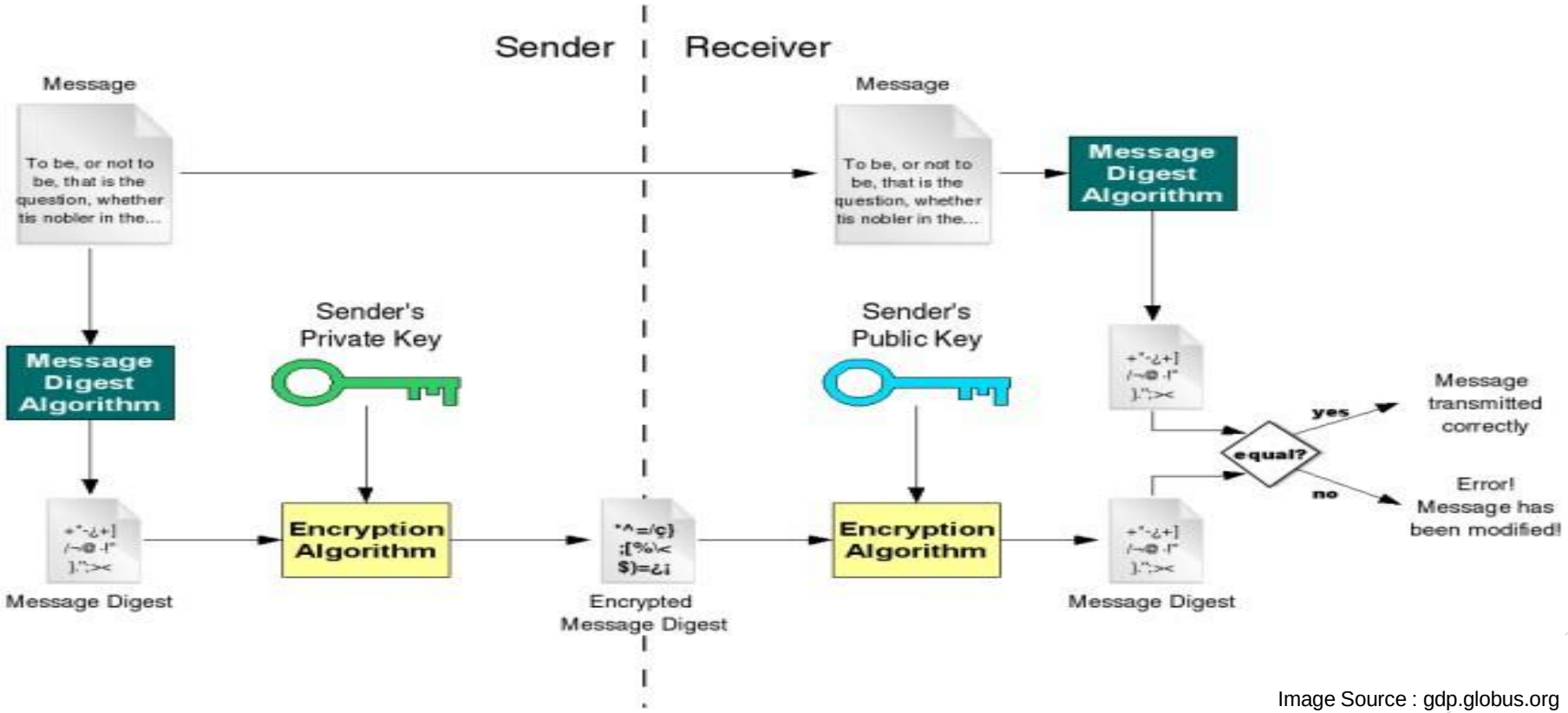


Image Source : gdp.globus.org

Detached Signature

- Digital signatures can either be attached to the message or detached
- A detached signature is stored and transmitted separately from the message it signs
- Commonly used to validate software distributed in compressed tar files
- You can't sign such a file internally without altering its contents, so the signature is created in a separate file

Assignment 8 - Homework

- Answer 2 questions in the file hw.txt
- A file eeprom that is a copy of the file /sys/bus/i2c/devices/0-0050/eeprom on your BeagleBone.
- https://www.gnupg.org/gph/en/manual.html
- Generate a key pair with the GNU Privacy Guard's commands (choose default options when prompted)
- Export public key, in ASCII format, into hw-pubkey.asc
- Use the private key you created to make a detached clear signature eeprom.sig for eeprom
- Use given commands to verify signature and file formatting
  - These can be found at the end of the assignment spec

Assignment 8 - Homework

- GNU Privacy Guard (GnuPG)
  - GnuPG allows you to encrypt and sign your data and communications
- It features a versatile key management system, along with access modules for all kinds of public key directories.
- GnuPG, also known as GPG, is a command line tool with features for easy integration with other applications.
- Reference: https://gnupg.org/gph/en/manual.html#INTRO

Assignment 8 - Homework

- GNU privacy guard (> gpg [option])
  - --gen key generating new keys
  - --armor ASCII format
  - --export exporting public key
  - --import import public key
  - --detach-sign creates a file with just the signature
  - --verify verify signature with a public key
  - --encrypt encrypt document
  - --decrypt decrypt document
  - --list-keys list all keys in the keyring
  - --send-keys register key with a public server/-keyserver option
  - --search-keys search for someone’s key

Presentations

- Today's Presentation:
  - Chester Hulse
  - Jackie Lam
- Next up:
  - Brian Phan
  - Henry Trinh

Questions?