Ecclesiastes (Eccl) 12:13

Now all has been heard;
here is the conclusion of the matter :

Have reverence for God, and obey his commands,
because this is all that man was created for.

Fear God and keep his commandments,
for this is the whole duty of man.

Cryptography

Cryptography

- Cryptography is the practice and study of Techniques for Secure Communication in the presence of Adversarial behavior
  - https://en.wikipedia.org/wiki/Cryptography
- Cryptography provides for Secure Communication in the presence of malicious third-parties known as Adversaries

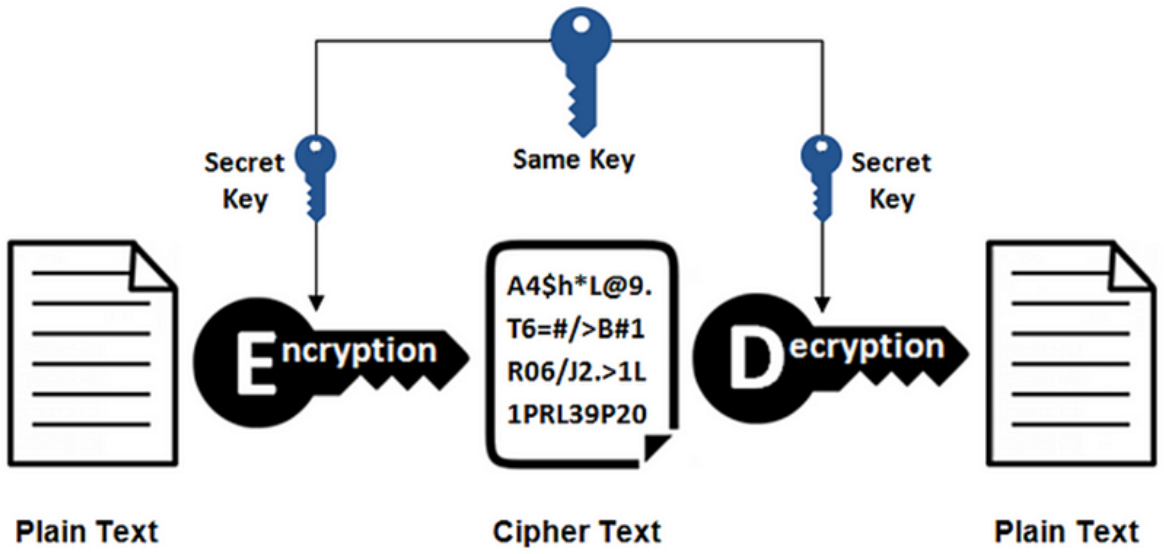
Types of Cryptography

- Cryptographic Algorithms (Encryption Algorithms)
  - Symmetric key Cryptography
  - Asymmetric (or Public) key Cryptography
    - RSA
- Hash functions
  - sha256

Symmetric Key Cryptography

- How to send Symmetric key ?
  - DES (Data Encryption Standard)
  - AES (Advanced Encryption Standard)
- Key distribution problem

- http://des.online-domain-tools.com/
- https://encode-decode.com/

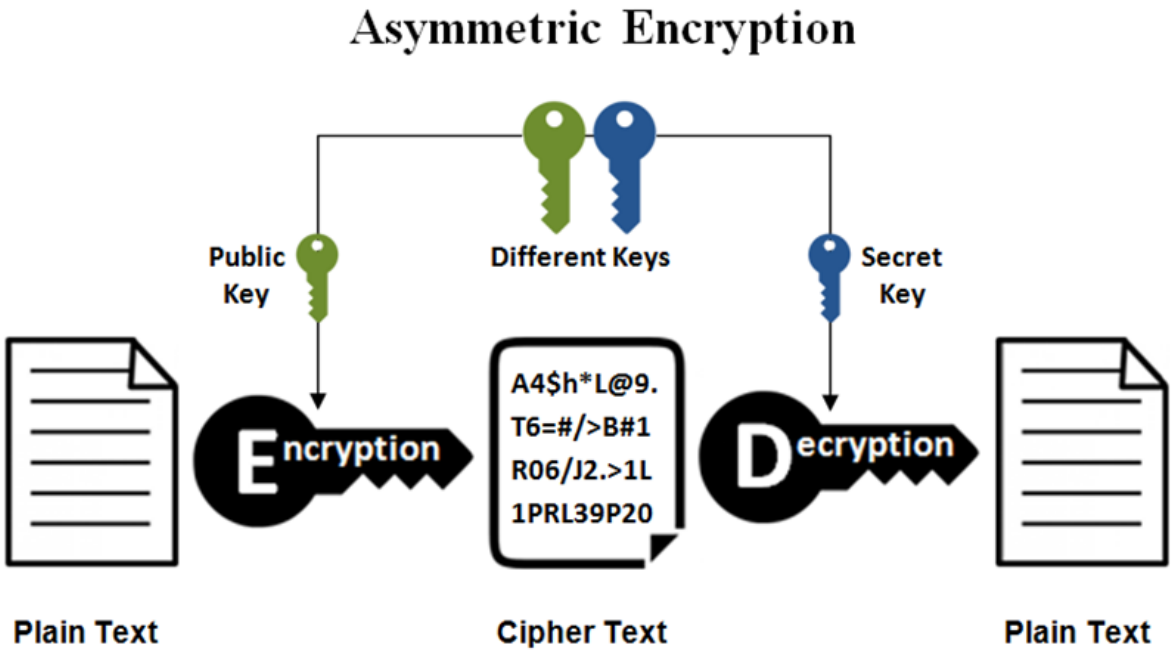


✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

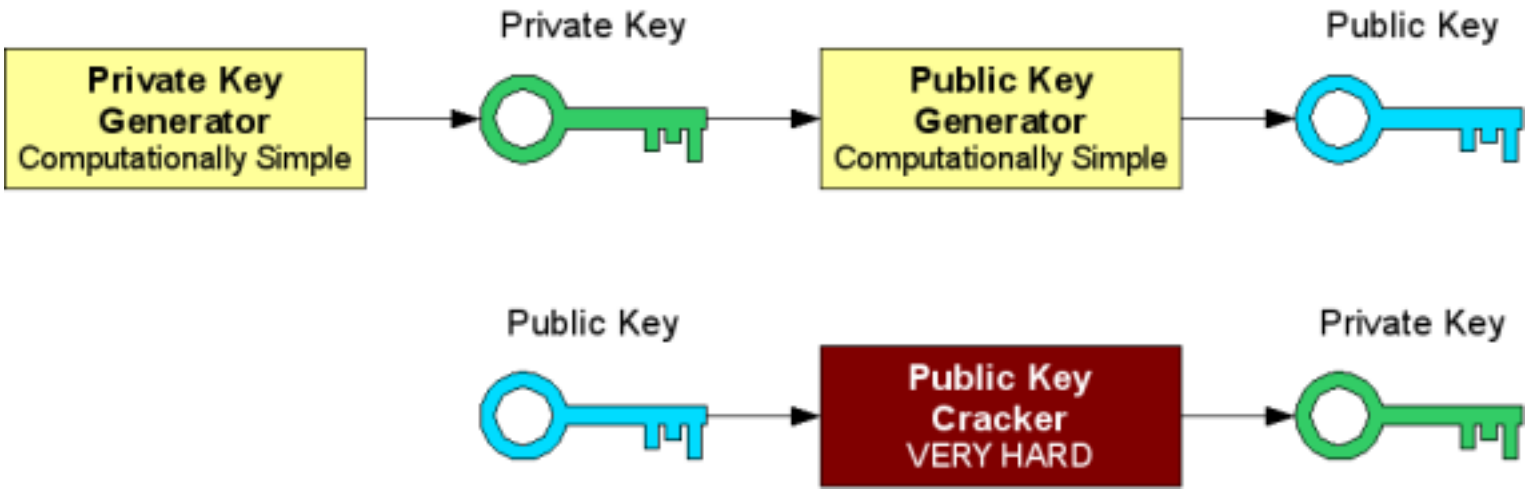
Asymmetric (Public) Key Cryptography

- Sender uses the public key to encrypt the data to send
- Receiver decrypts the cipher text with the private key
- Data encrypted by the public key can only be decrypted by the private key

- RSA (Rivest-Shamir-Adleman)
- https://www.devglan.com/online-tools/rsa-encryption-decryption

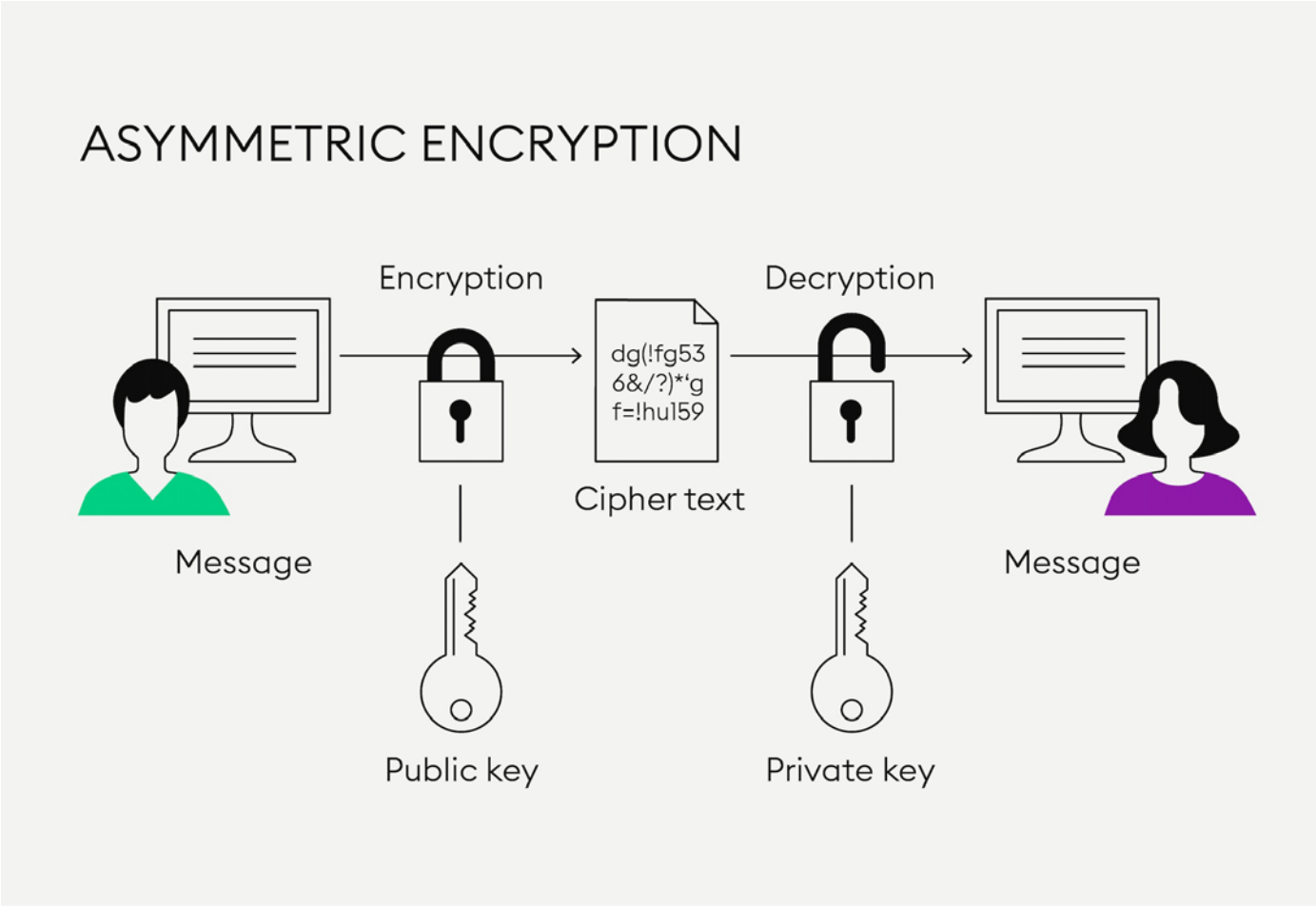


✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

How to Share a Key in Symmetric Cryptography

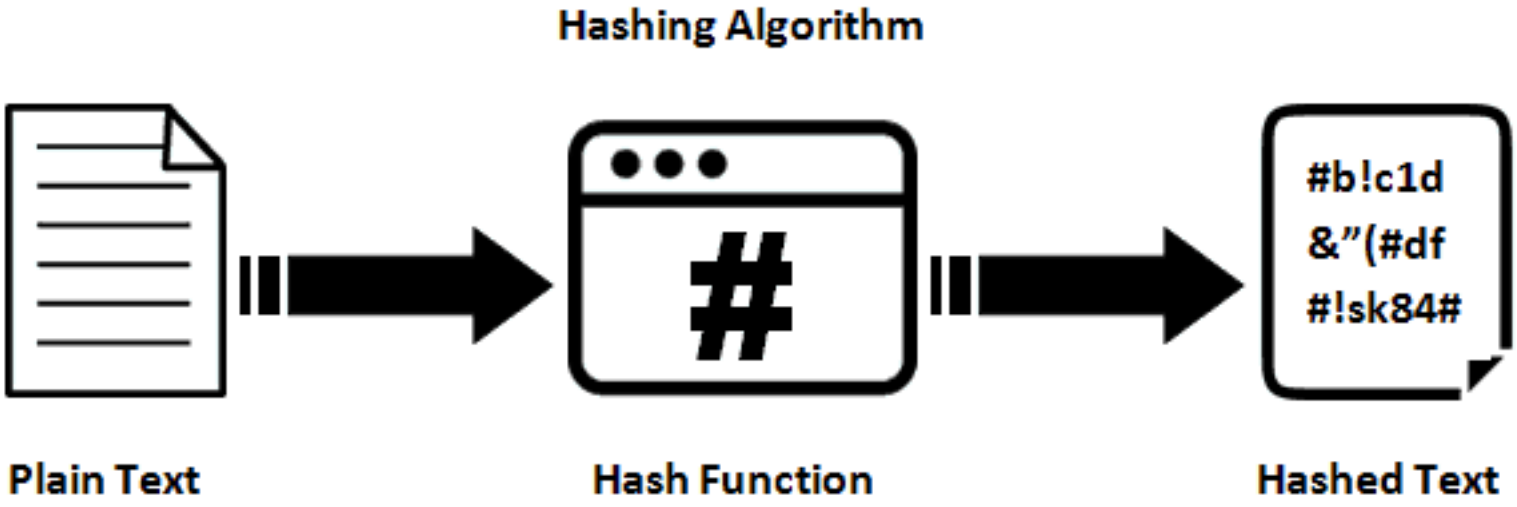


✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

Hash Function

- Deterministic
  - For the same input, a hash function will always produce the same hash value. This property is essential for consistent results and comparisons.
- Fixed Output Size
  - Regardless of the input size, a hash function produces a hash value of a fixed length. For example, SHA-256 produces a 256-bit hash value.
- Pre-image Resistance
  - It should be computationally infeasible to determine the original input from its hash value
- Collision Resistance
  - It should be difficult to find two different inputs that produce the same hash value.

https://andersbrownworth.com/blockchain/hash

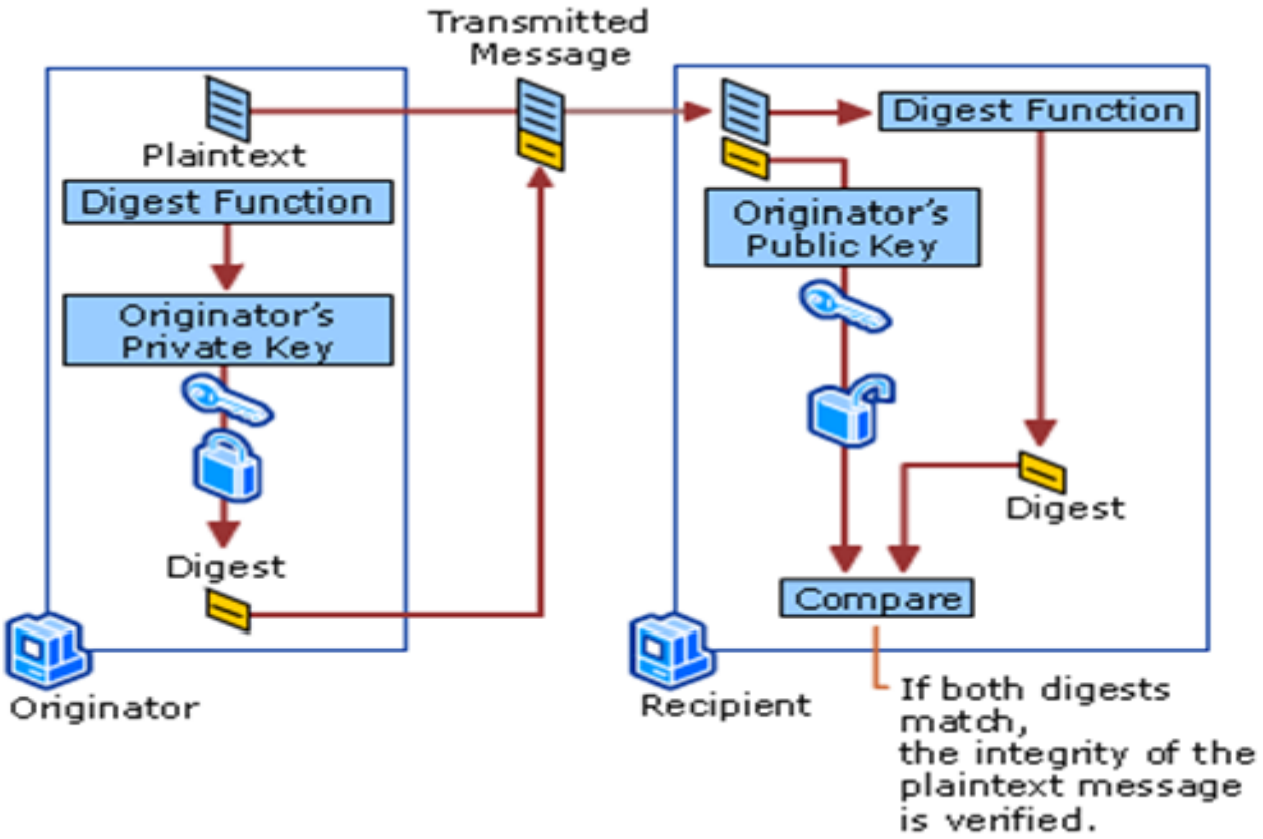


✝ Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)

Application of Hash Function

- Data Integrity
  - Hashing is used to verify the integrity of data during transmission or storage. If the hash value of the received data matches the expected hash value, it's highly likely that the data has not been altered.
- Password Storage
  - Hash functions are used to securely store passwords. Instead of storing actual passwords, systems store their hash values. When users provide passwords during login, the system hashes the provided password and compares it to the stored hash.
- Digital Signature
  - Hash functions are used in digital signature to create a hash of a message before signing it with a private key. The recipient can verify the signature by hashing the received message and comparing it with the decrypted hash value.
- Cryptographic Applications
  - Hash functions are fundamental in cryptographic protocols like HMAC (Hash-based Message Authentication Code), which provides message authentication and integrity.

Hashing and Public Key Cryptography



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

Digital Signature

- Digital signature is a cryptographic technique used to provide authenticity, integrity, and non-repudiation to digital documents or messages
- Digital signature assures the recipient that the document they received was indeed created and signed by the claimed sender, and that the document has not been altered since it was signed.

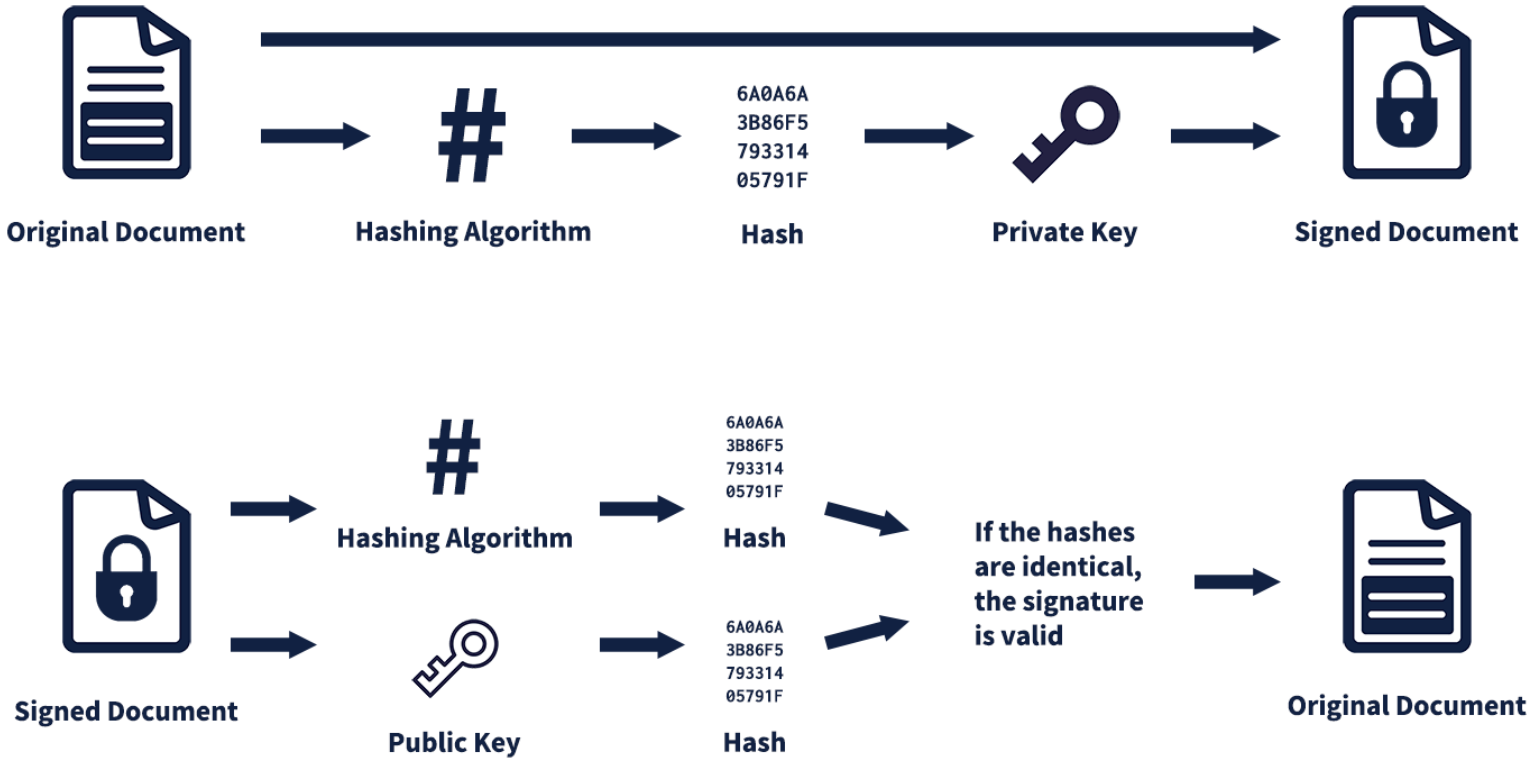
Signing Process

- Sender uses private key to create a unique digital signature for the document or message.
- The signature is created by applying a cryptographic algorithm to a hash (a fixed-size representation) of the document's content.
  - This produces a digital fingerprint of the data.
- Sender's private key ensures that only the sender could have produced that specific signature.

Verification Process

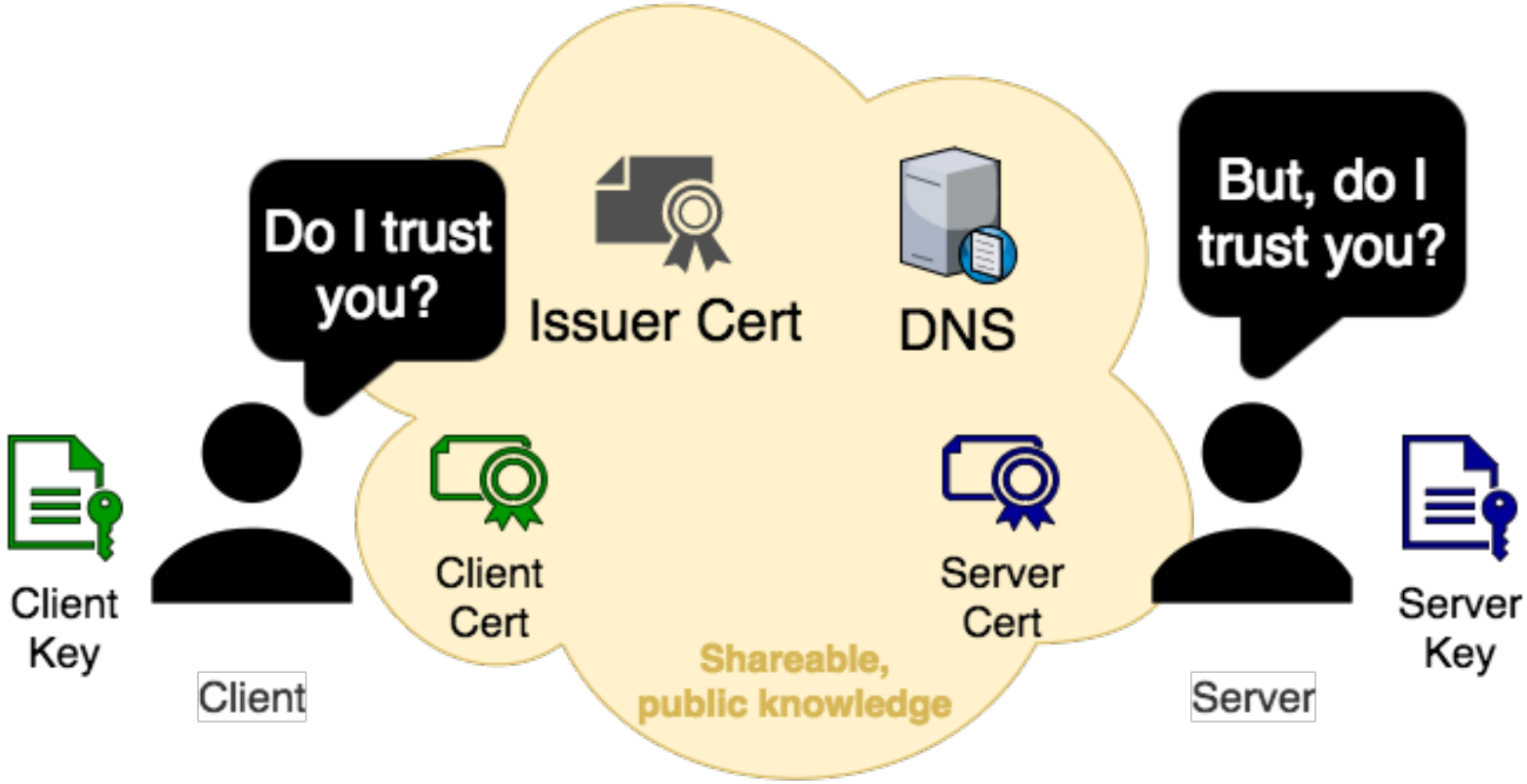
- Recipient uses the sender's public key (which is available to anyone) to verify the signature.
- The recipient applies the same cryptographic algorithm to the received document's content to generate a hash.
- The recipient then compares the computed hash with the decrypted digital signature.
- If the two hashes match, the recipient can be confident that the document is intact and that it was indeed signed by the sender.

Digital Signature



✝ Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)

Trust a Server ?



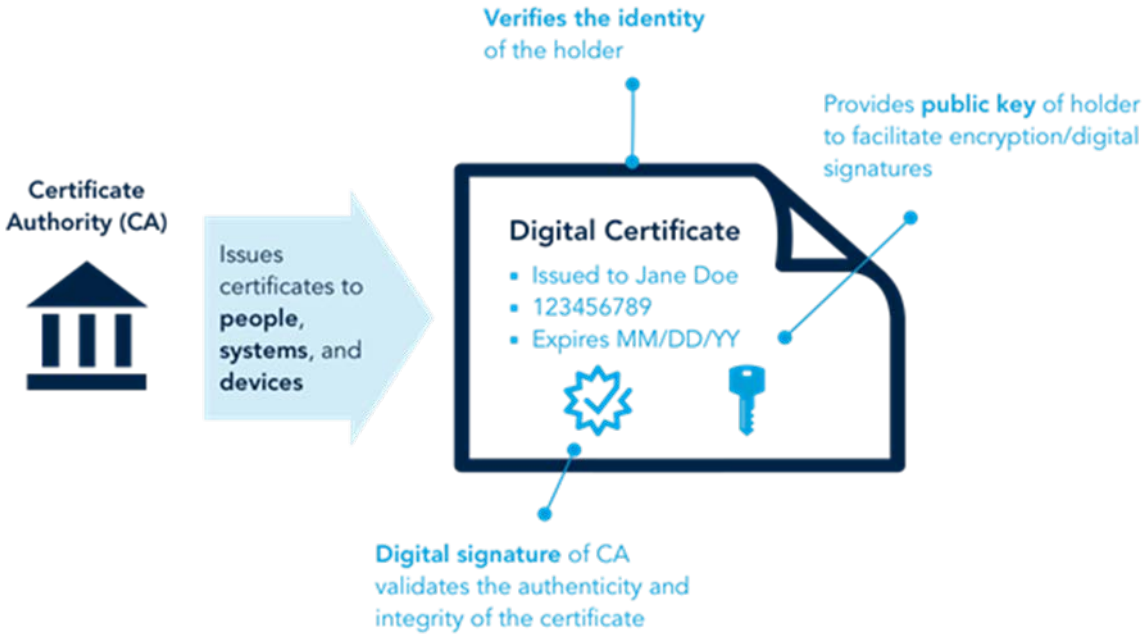
✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

Digital Certificate

Structure of Certificate

- Serial Number
  - Used to uniquely identify the certificate within a CA's systems.
- Subject
  - The entity a certificate belongs to: a machine, an individual, or an organization.
- Issuer
  - The entity that verified the information and signed the certificate.
- Public Key
  - A public key belonging to the certificate subject.
- Signature Algorithm
  - This contain a hashing algorithm and a digital signature algorithm. For example "sha256RSA" where sha256 is the hashing algorithm and RSA is the signature algorithm.
- Signature
  - The body of the certificate is hashed (hashing algorithm in "Signature Algorithm" field is used) and then the hash is signed (signature algorithm in the "Signature Algorithm" field is used) with the issuer's private key.

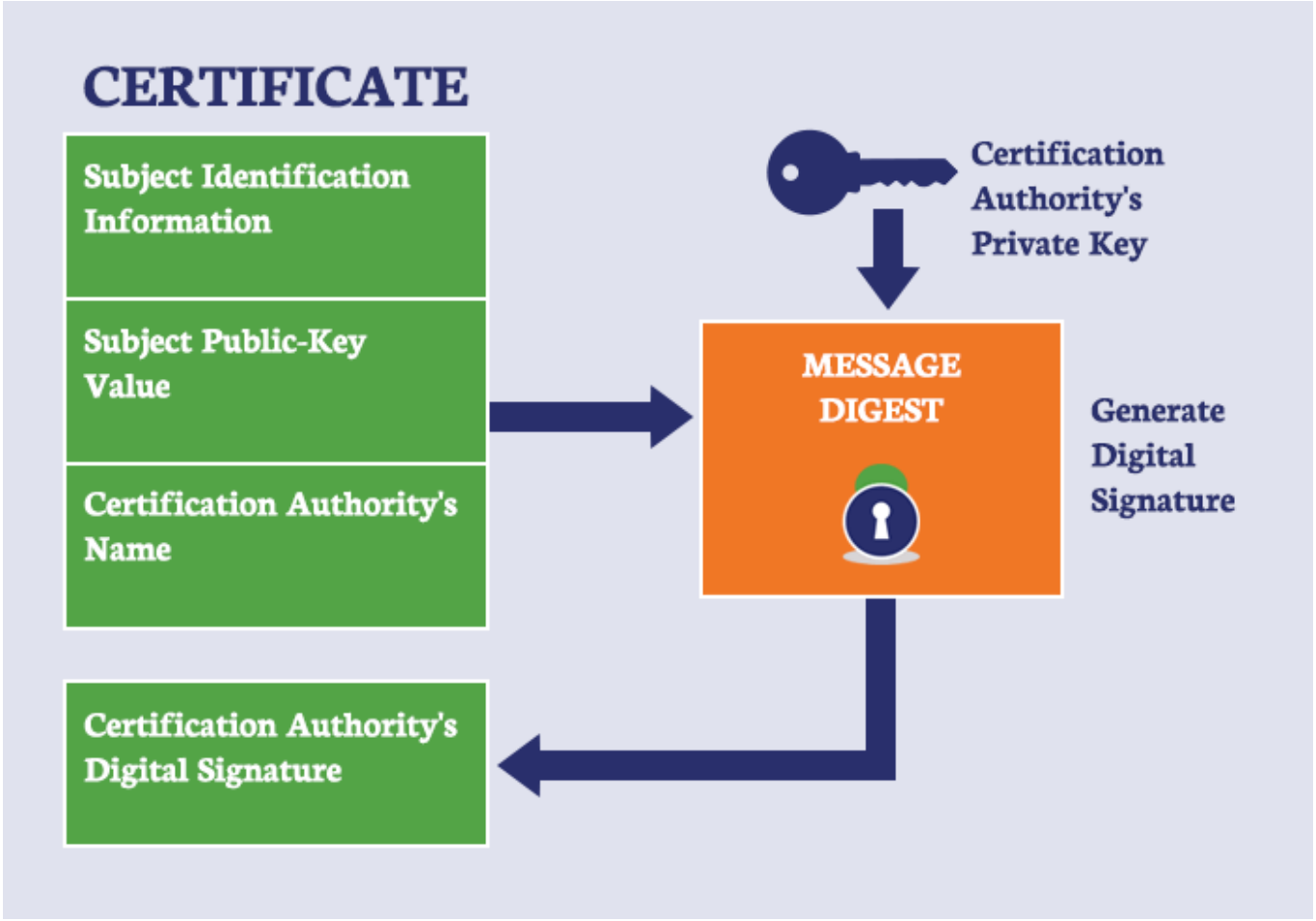
CA (Certification Authority)



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

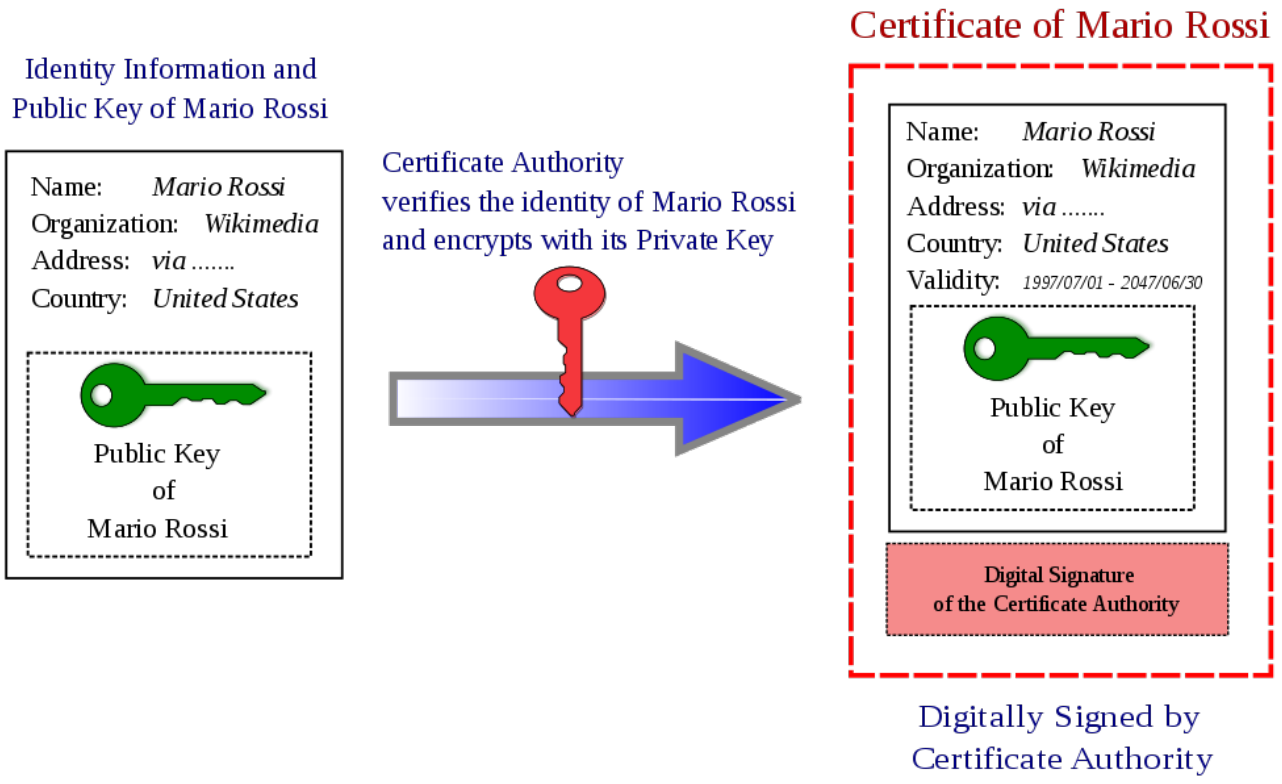
Types of Certificate

- CA (Certification Authority) signed Certificates
  - TLS/SSL server certificate
  - TLS/SSL client certificate
- Self signed certificates



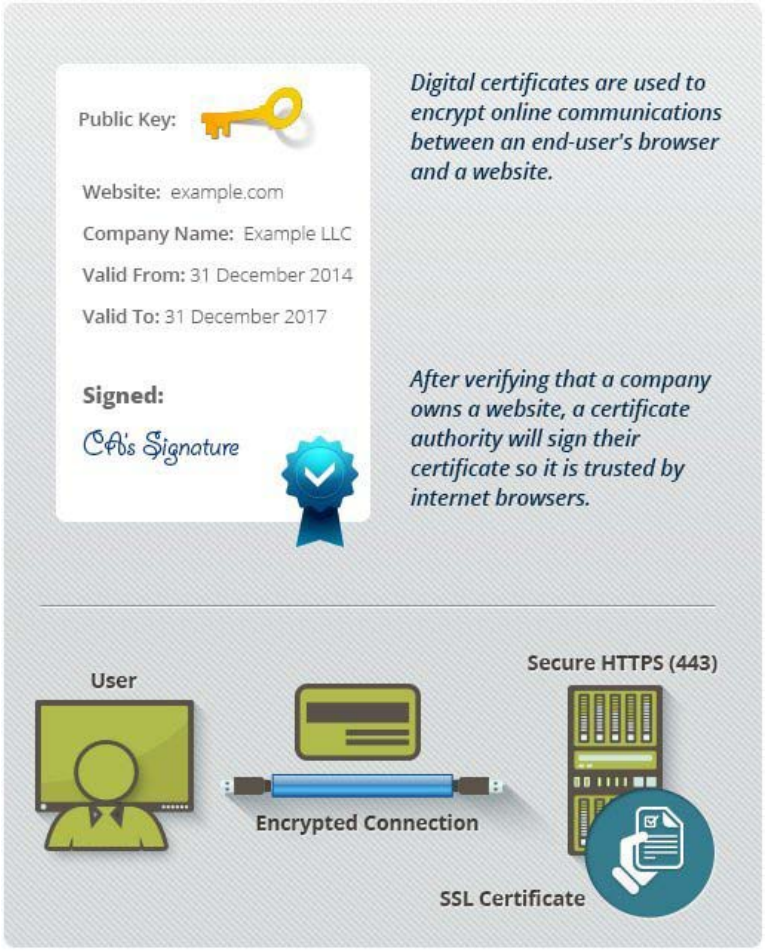
✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

Digital Certificate or Public Key Certificate

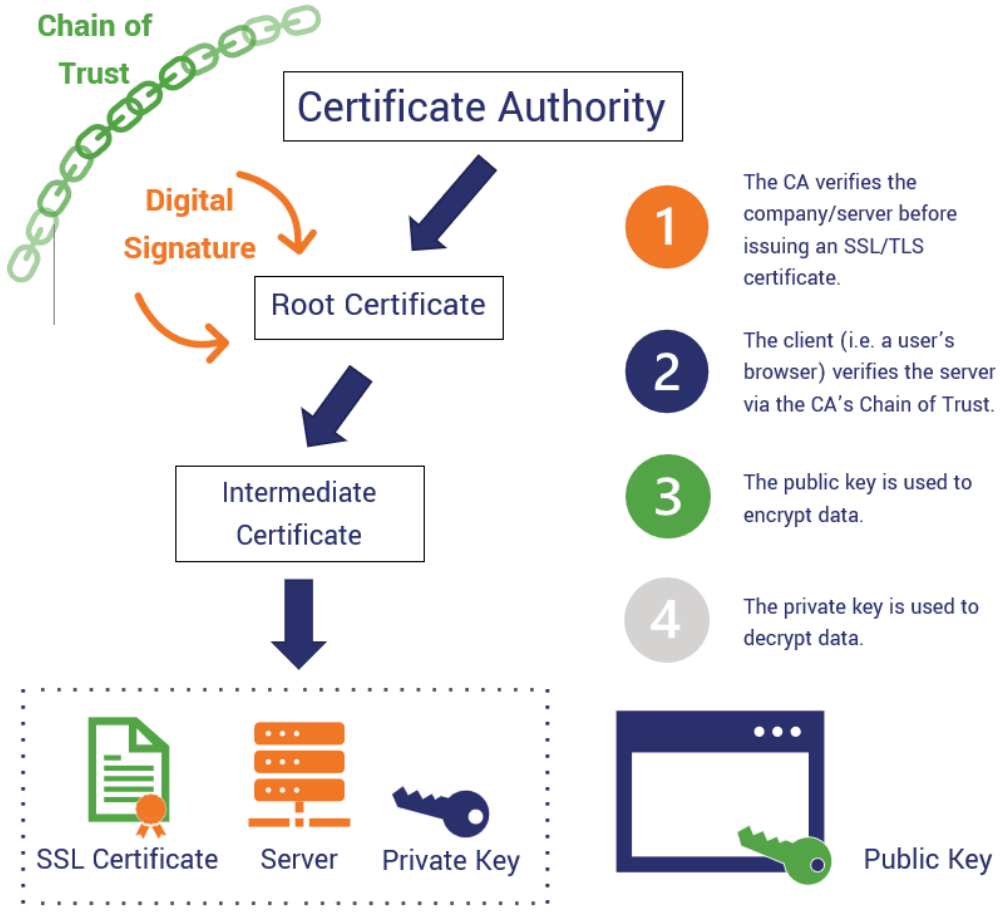


✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

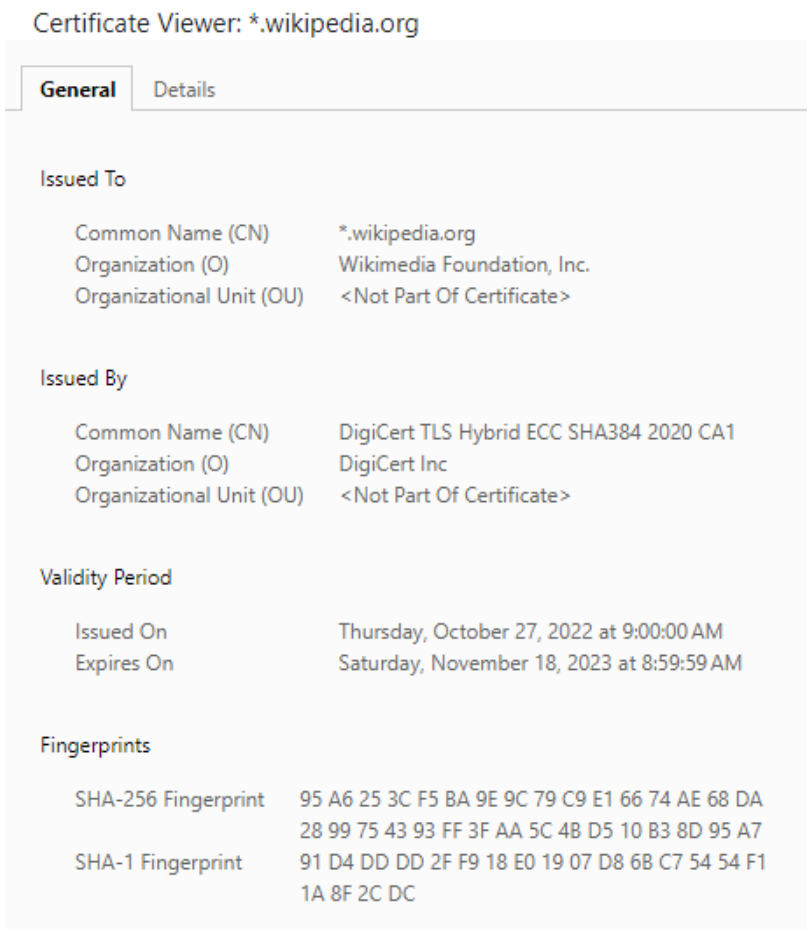
https://www.wikipedia.org/



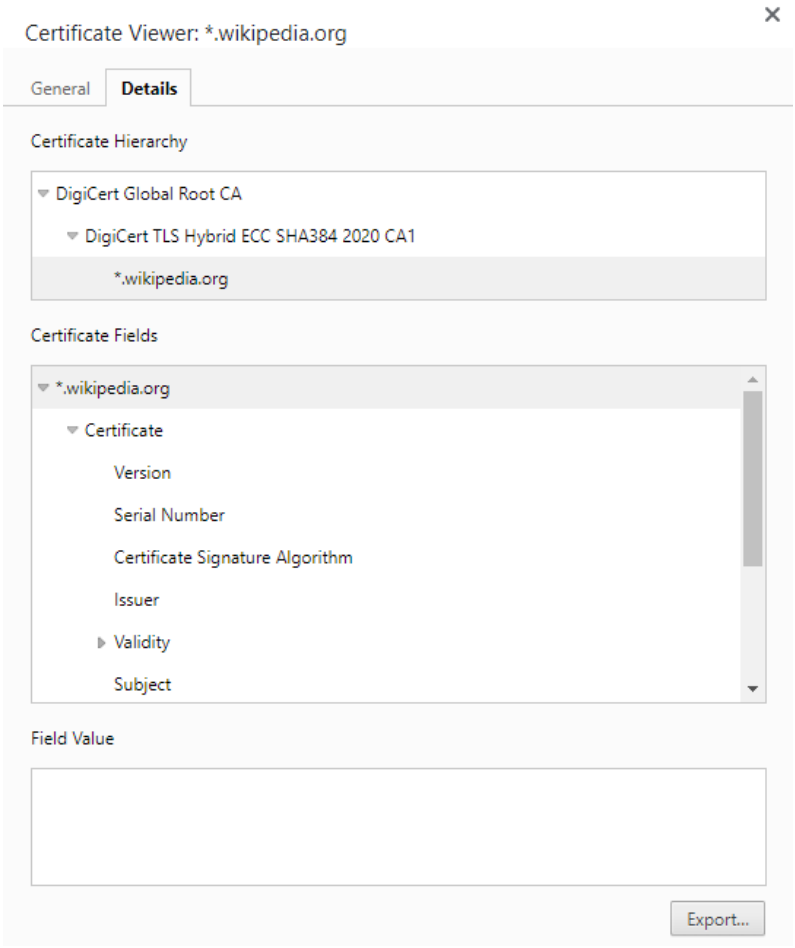
✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

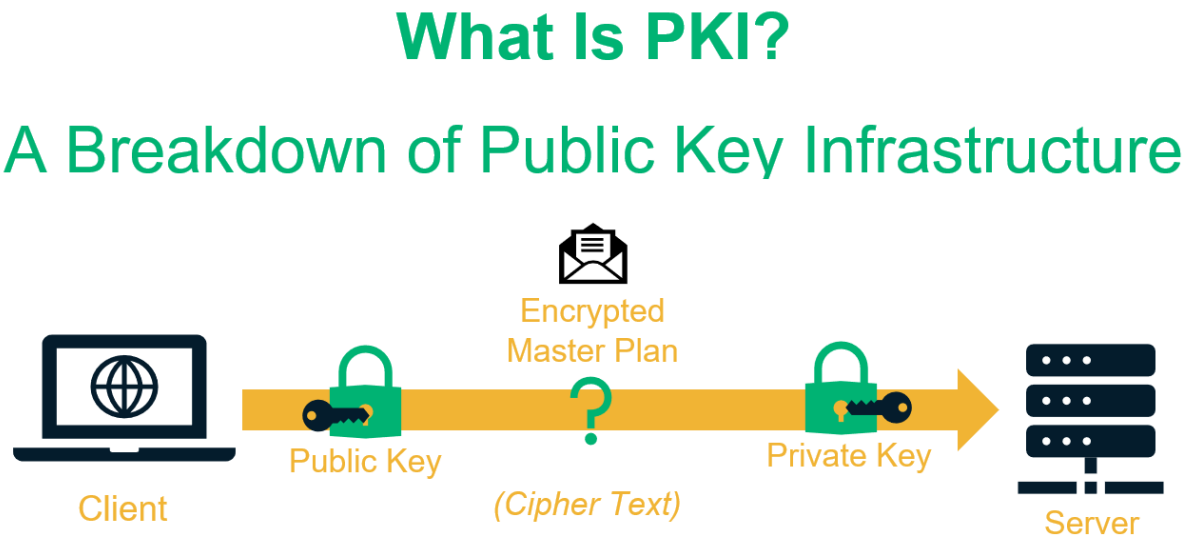


✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

PKI

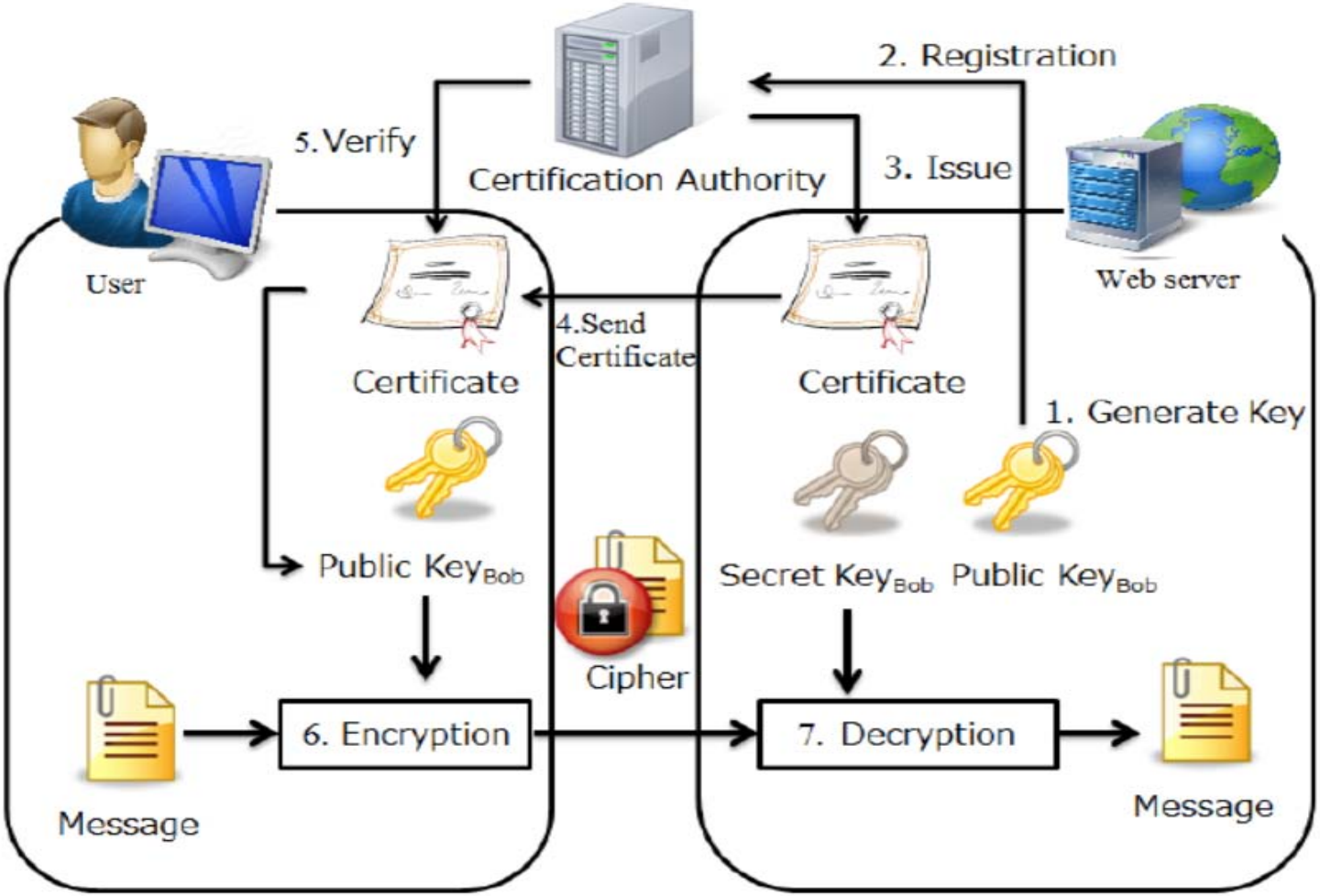


✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

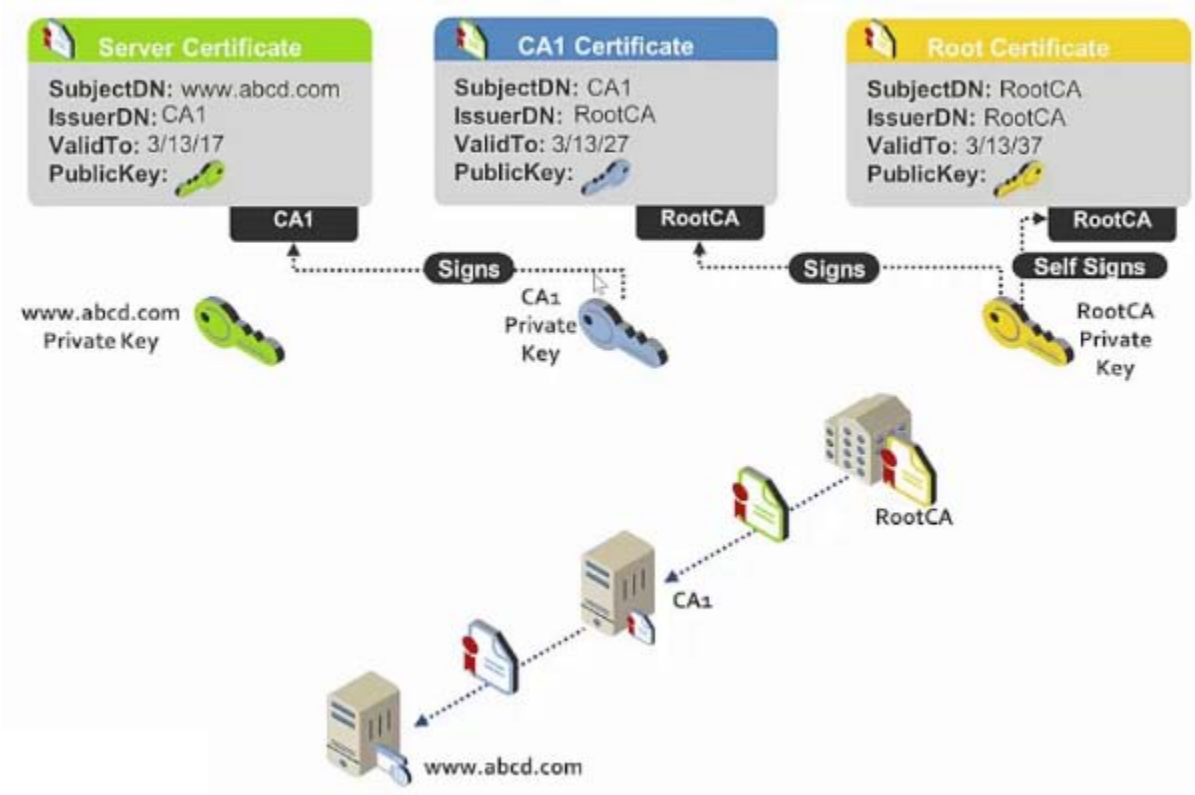
PKI (Public Key Infrastructure)

- A trusted certificate authority (CA) that can verify user identities and issue users digital certificates and public/private key pairs.
  - Sometimes the verification of user identities is performed by a separate registration authority (RA), but this service can also be integrated with the CA.
- A certificate store in which users can access the public keys of other users for encrypting messages or validating digital signatures. This store is usually based on the X.500 directory recommendations.
- A digital certificate and key management system for generating, storing, and securely transmitting certificates and key pairs to users who request them.

Working of Digital Certificates

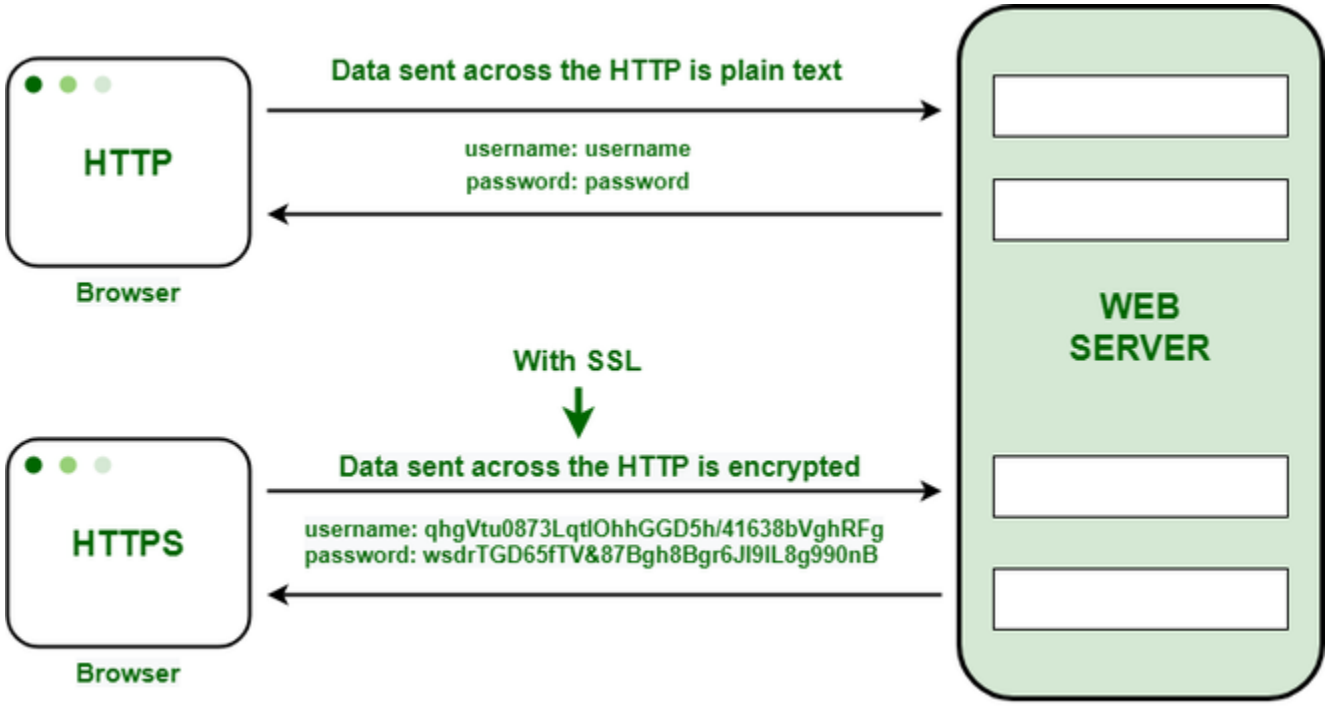


X.509 and PKI



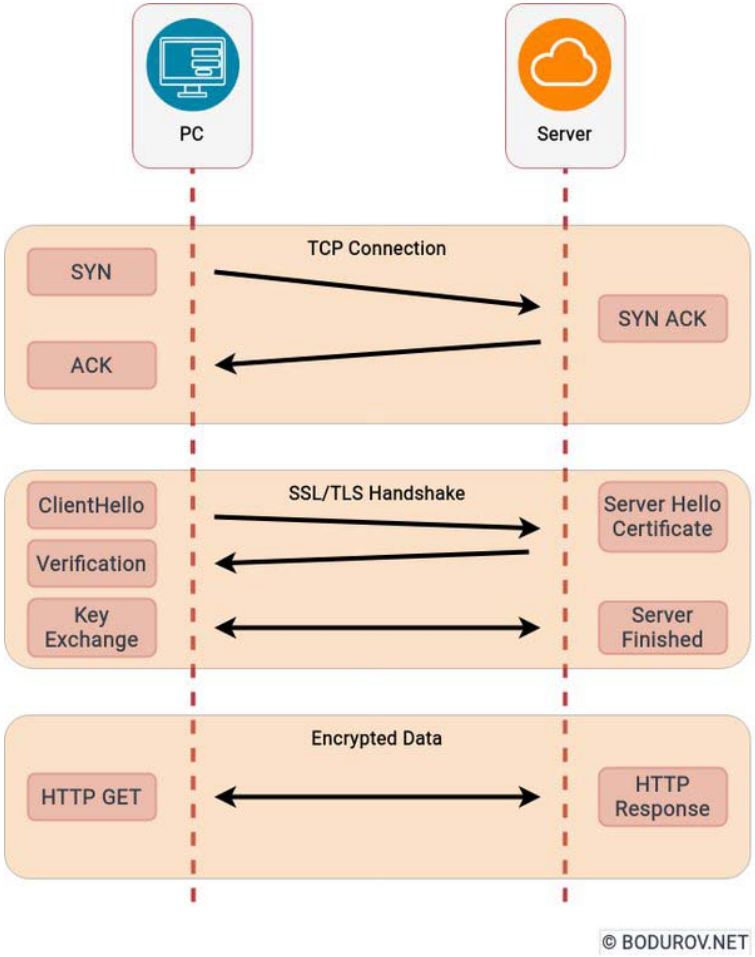
✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

HTTP vs HTTPS



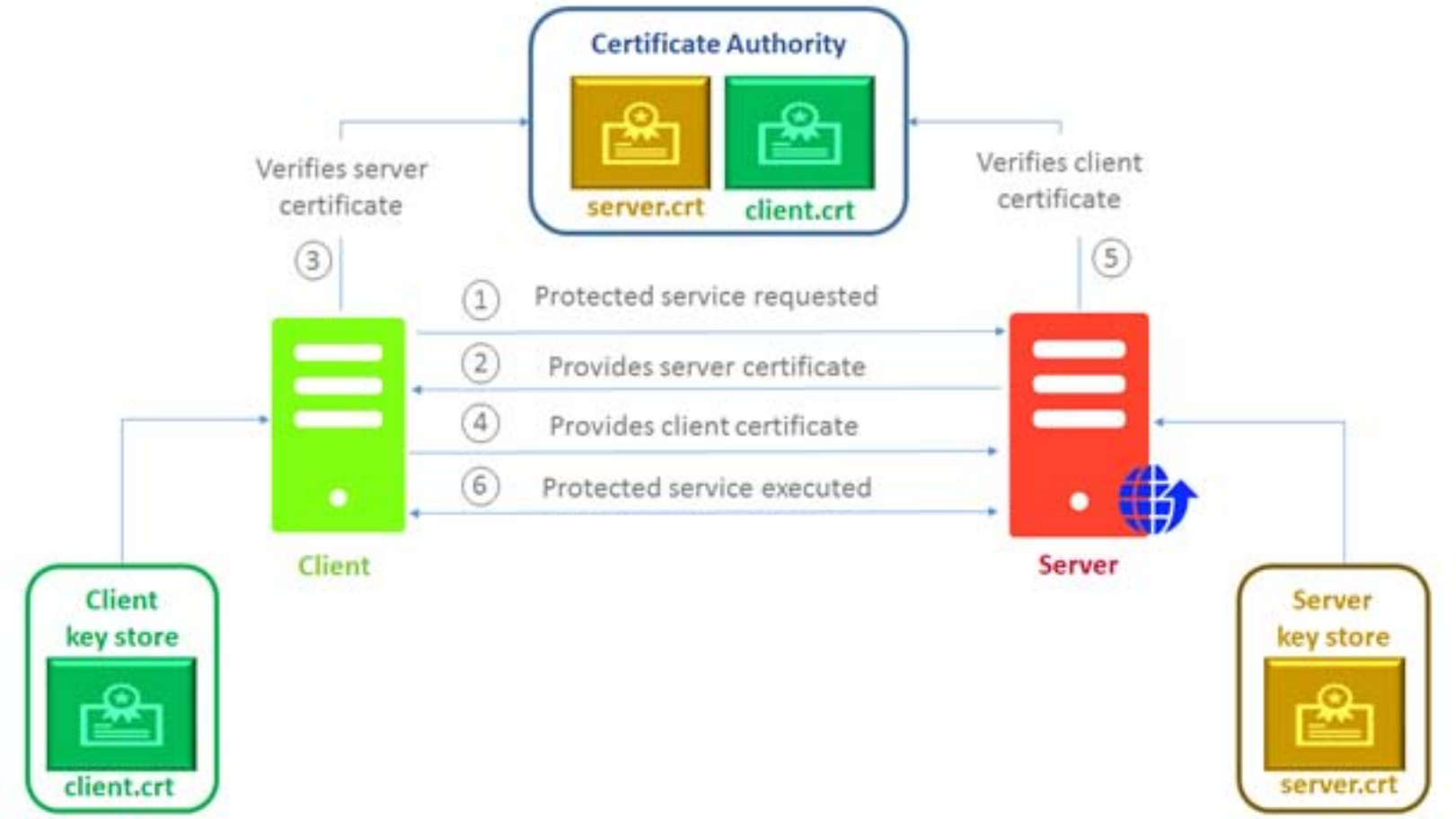
✝ Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)

HTTPS : SSL/TSL



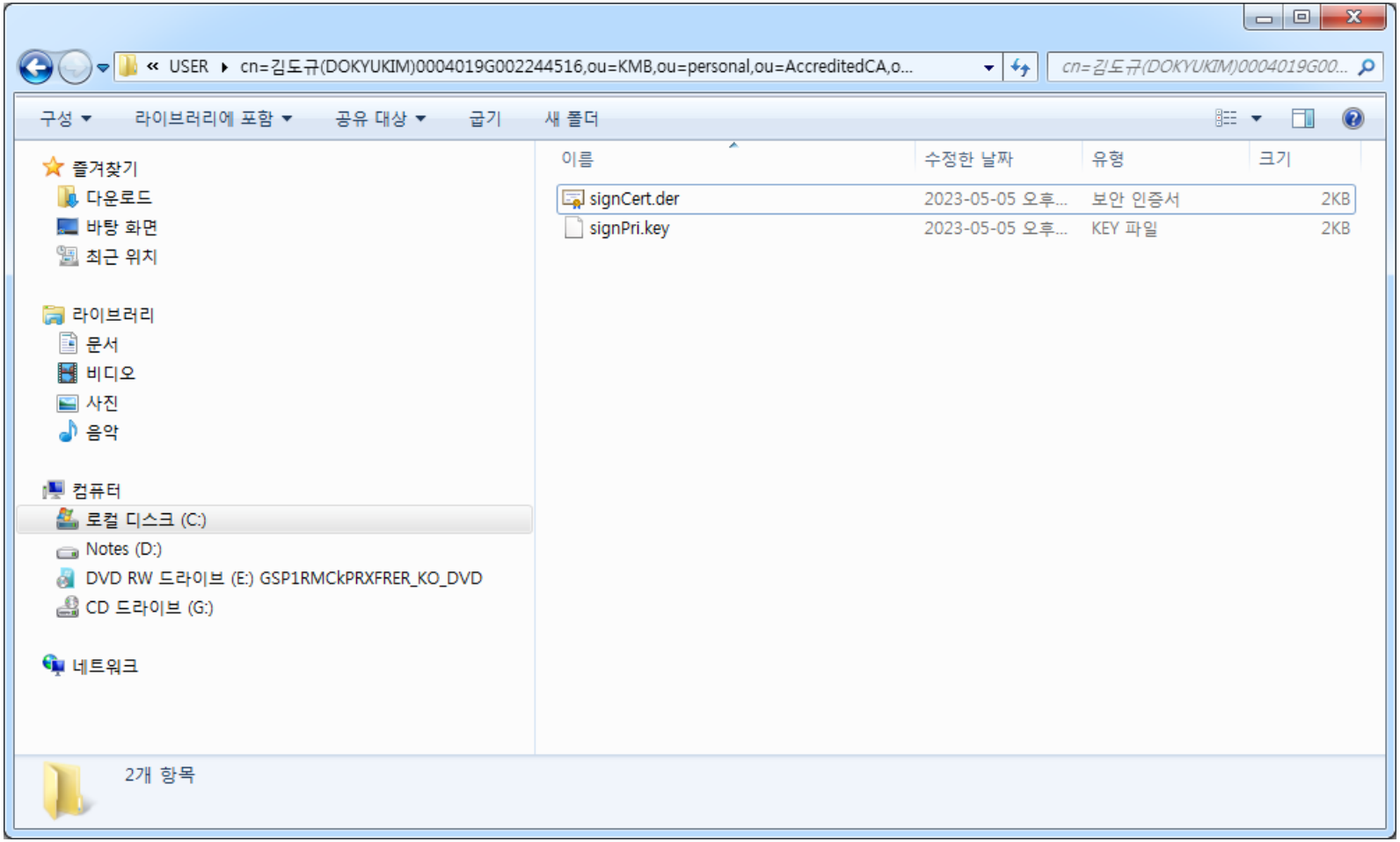
✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

Key Store



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**

c:\Users\dkkim\AppData\LocalLow\NPKI



✝ **Have reverence for God, and obey his commands, because this is all that man was created for (Ecclesiastes 12:13)**