Encryption

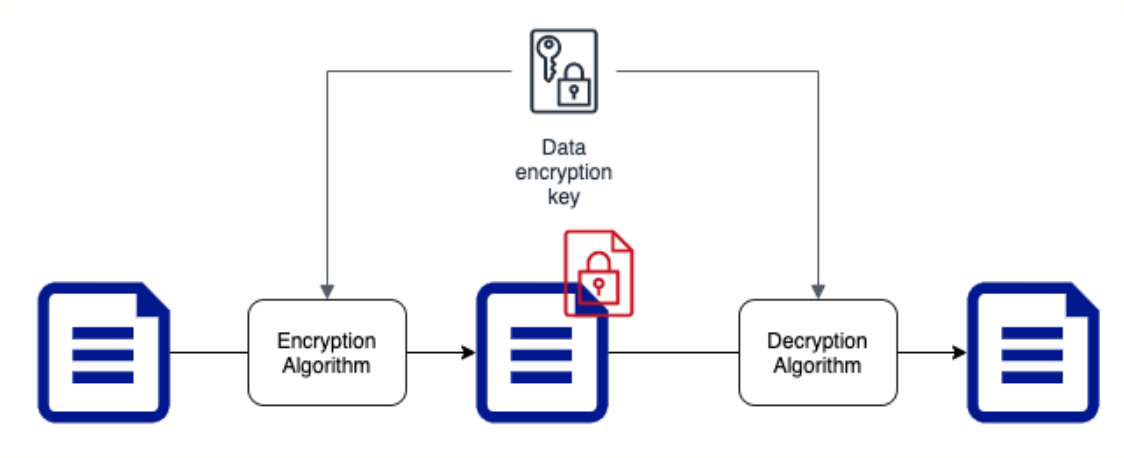
Data States

- **Data at rest**: Stored on a device or a backup
  - Examples : data on a hard disk, in a database, backups and archives
- **Data in motion:** Being transferred across a network
  - Also called **Data in transit**
  - **Examples** :
    - Data copied from on-premise to cloud storage
    - An application talking to a database
  - **Two Types**:
    - In and out of cloud (from internet)
    - Within cloud
- **Data in use**: Active data processed in a non-persistent state
  - Example: Data in your RAM

Encryption

- If you store data as is, what would happen if an **unauthorized entity gets access** to it?
  - Imagine losing an unencrypted hard disk
- **First law of security** : Defense in Depth
- Typically, enterprises encrypt all data
  - Data on your hard disks
  - Data in your databases
  - Data on your file servers
- Is it sufficient if you encrypt data at rest?
  - **No**. **Encrypt data in transit** - between application to database as well.

Symmetric Key Encryption

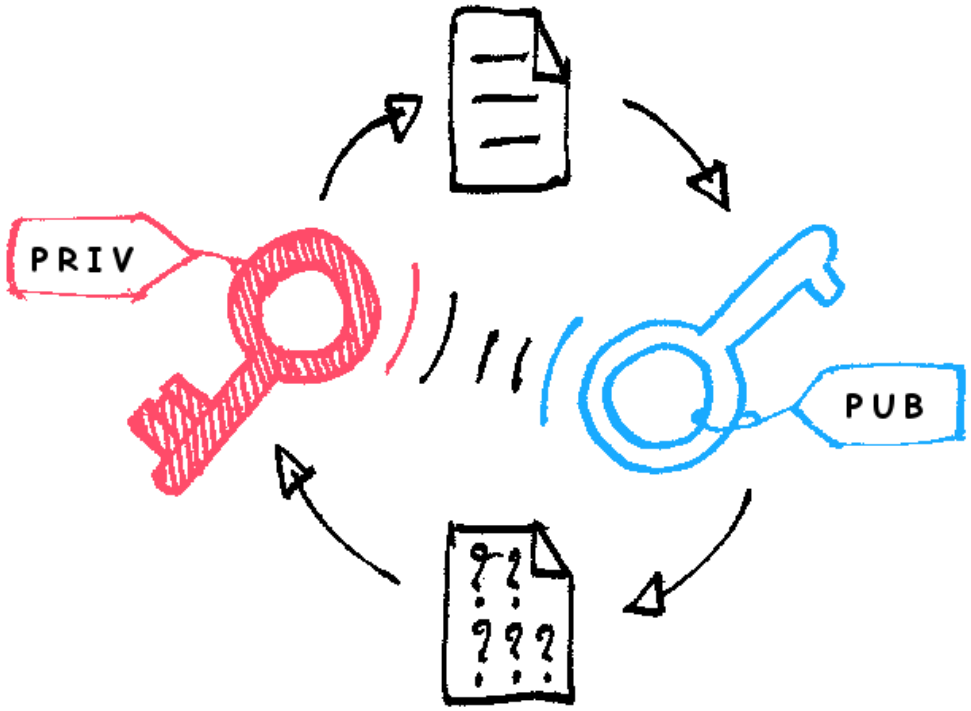


- Symmetric encryption algorithms use the **same key for encryption and decryption**
- Key Factor 1: Choose the **right encryption algorithm**
- Key Factor 2: How do we **secure the encryption key?**
- Key Factor 3: How do we **share the encryption key?**

Asymmetric Key Encryption

In 28 Minutes

- **Two Keys** : Public Key and Private Key
- Also called **Public Key Cyptography**
- Encrypt data with Public Key and decrypt with Private Key
- Share Public Key with everybody and keep the Private Key with you(YEAH, ITS PRIVATE!)
- No crazy questions:
  - Will somebody not figure out private key using the public key?
- How do you create Asymmetric Keys?



https://commons.wikimedia.org/wiki/File:Asymmetric_encryption_(colored).p

Cloud KMS

- Create and manage **cryptographic keys** (symmetric and asymmetric)
- **Control their use** in your applications and GCP Services
- Provides an API to encrypt, decrypt, or sign data
- Use existing cryptographic keys created on premises
- **Integrates with almost all GCP services** that need data encryption:
  - Google-managed key: No configuration required
  - Customer-managed key: Use key from KMS
  - Customer-supplied key: Provide your own key