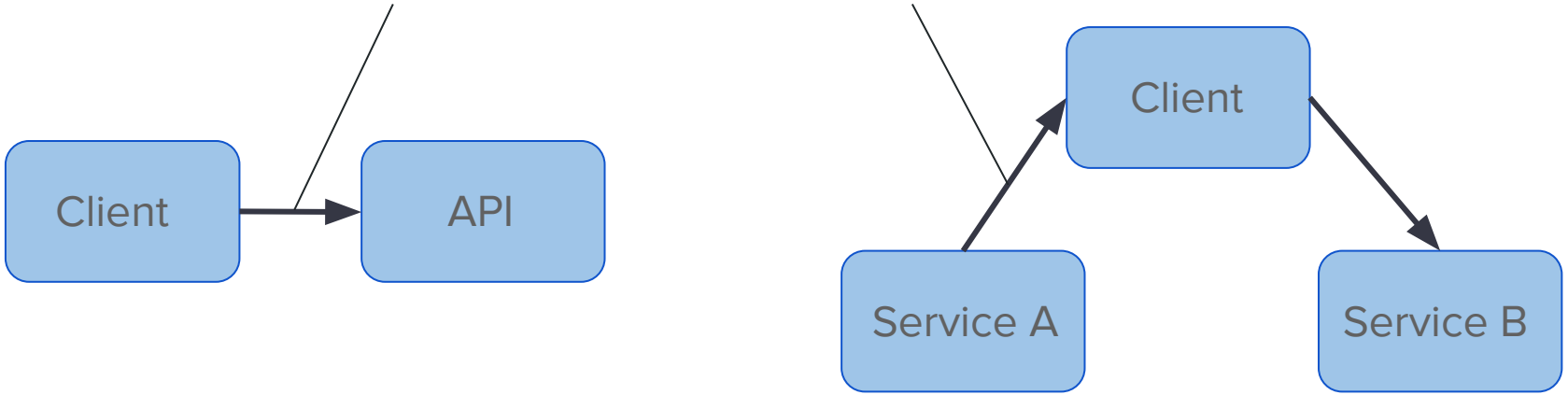
JSON Web Token

Problems with communicating

message = {“username”:”ryo”, ”action”:”change password”}

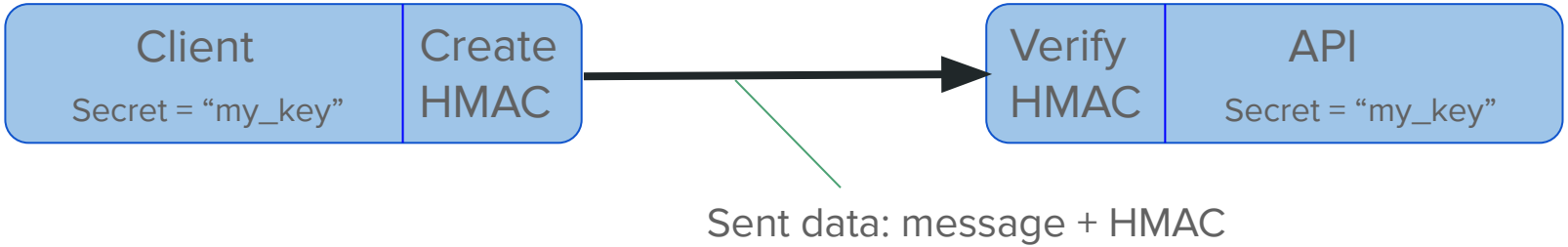


➔ Difficult to authenticate the sender and the data

Hash-based Message Authentication Code (HMAC)

➔ Messages are sent along with HMAC: `hash(shared_secret, message)`

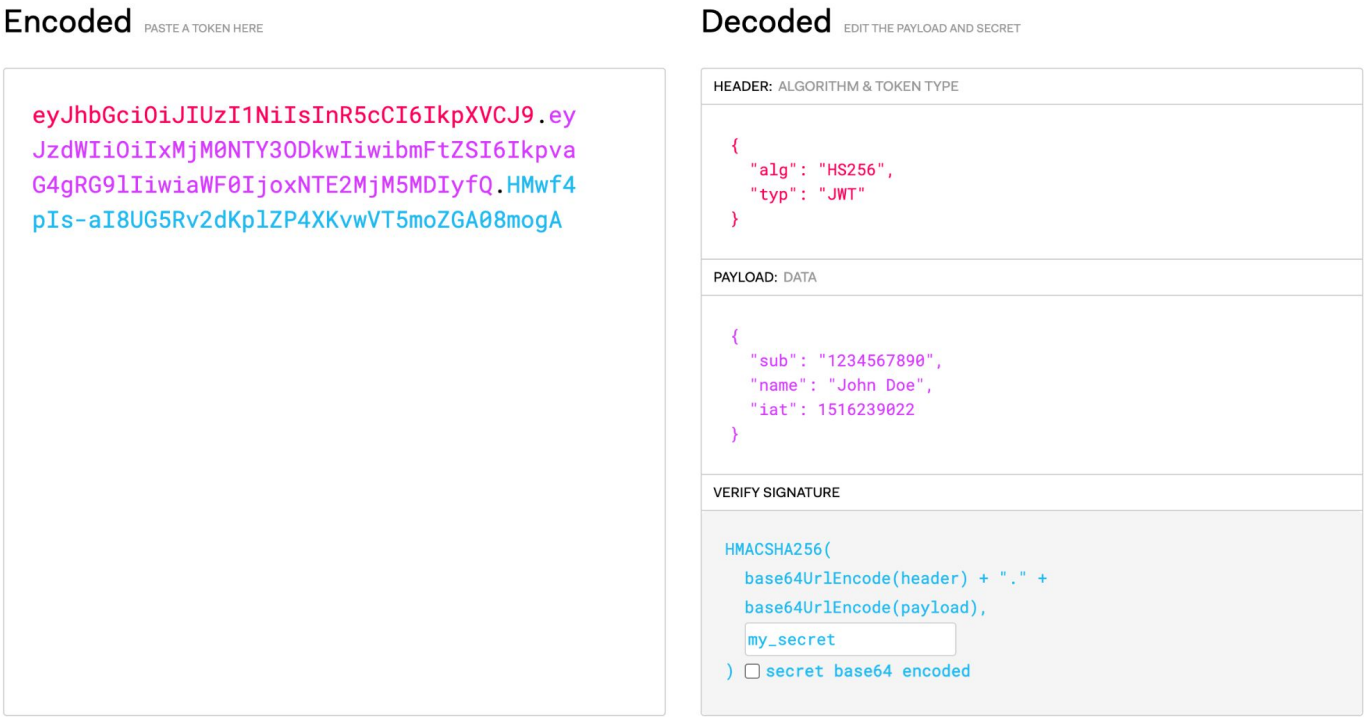
message = `{"username":"ryo", "action":"change password"}`
HMAC: `hash("my_key", message)`



➔ Verifies that:

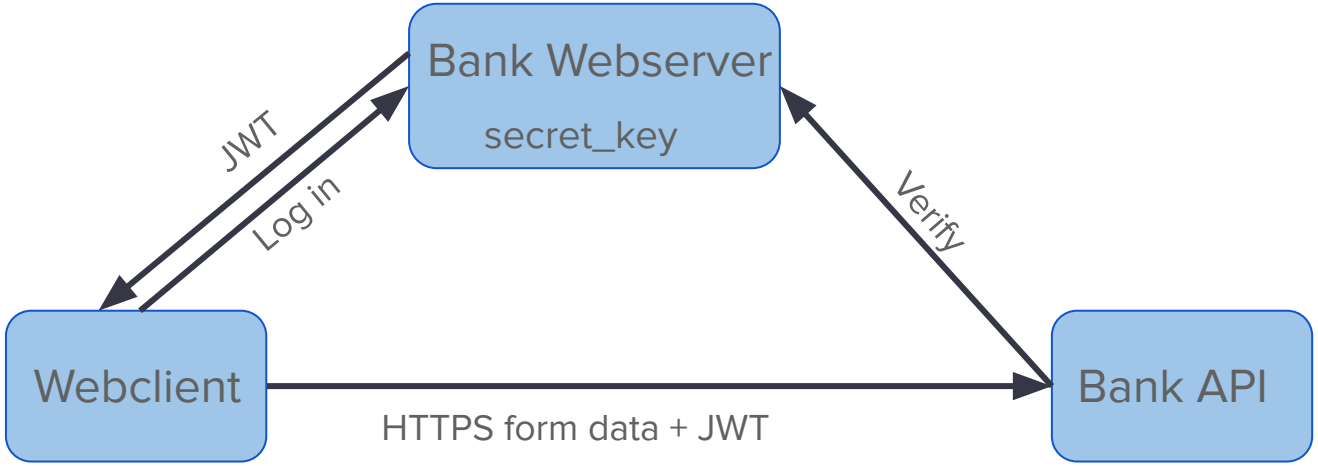
- message was sent by client with the shared secret,
- message was not tampered with in transit

JSON Web Token (JWT)



Source: https://jwt.io

Example use case: Bank website



➔ Some use cases
- APIs: Google, AWS, Colfax
- Auth Tokens: OAuth 2.0

Common mistakes

- Some common mistakes by developers.
  - Authentication vs Authorization
  - JWT info is readable w/o secret
  - Signed and unsigned data
  - JWT w/o expiration
  - Multi-use JWT

Conclusion

- ➔ JWT is a HMAC standard to secure communication.
  - ◆ Verifies that the message came from a sender with the key
  - ◆ Verifies that the message was not tampered with
- ➔ It is an industry standard tool for securing APIs and for Authentication.
- ➔ JWT is a secure tool, but like all tools is only as good as the developer.