

Module 3 Day 10

Authentication and Encryption

Authentication vs. Authorization

- Authentication

- Who you are
- Prove that you are who you claim to be
- “What you have and what you know” (2FA)
 - ATM card, cell phone, ID card, biometric (fingerprint, retina, face)
 - Password, PIN, Security Question
- Often involves a password

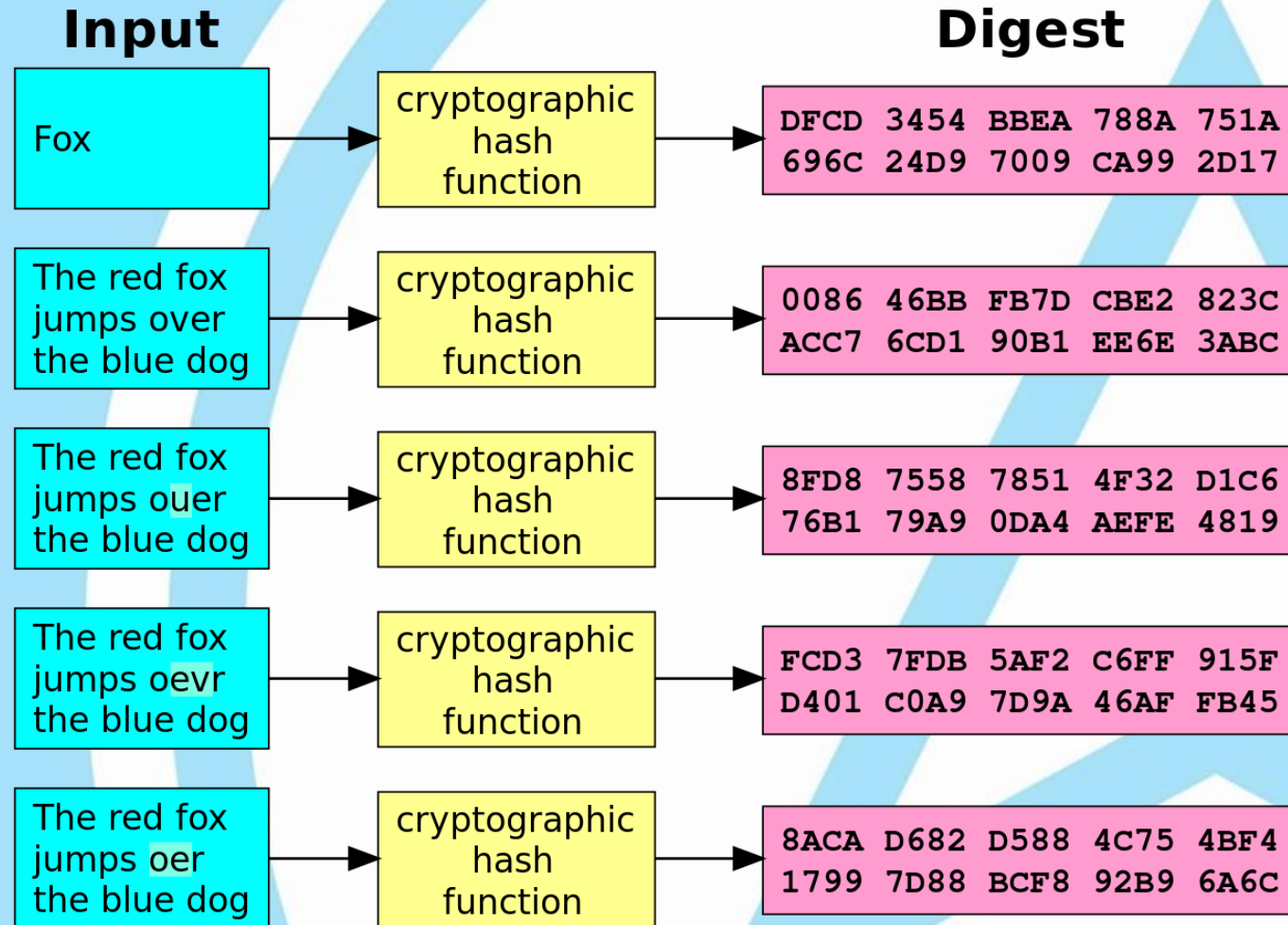
- Authorization

- What you can do
- Can you view data, or edit it? Delete it? Add users? Etc.
- Useless without Authentication

Hashing Data

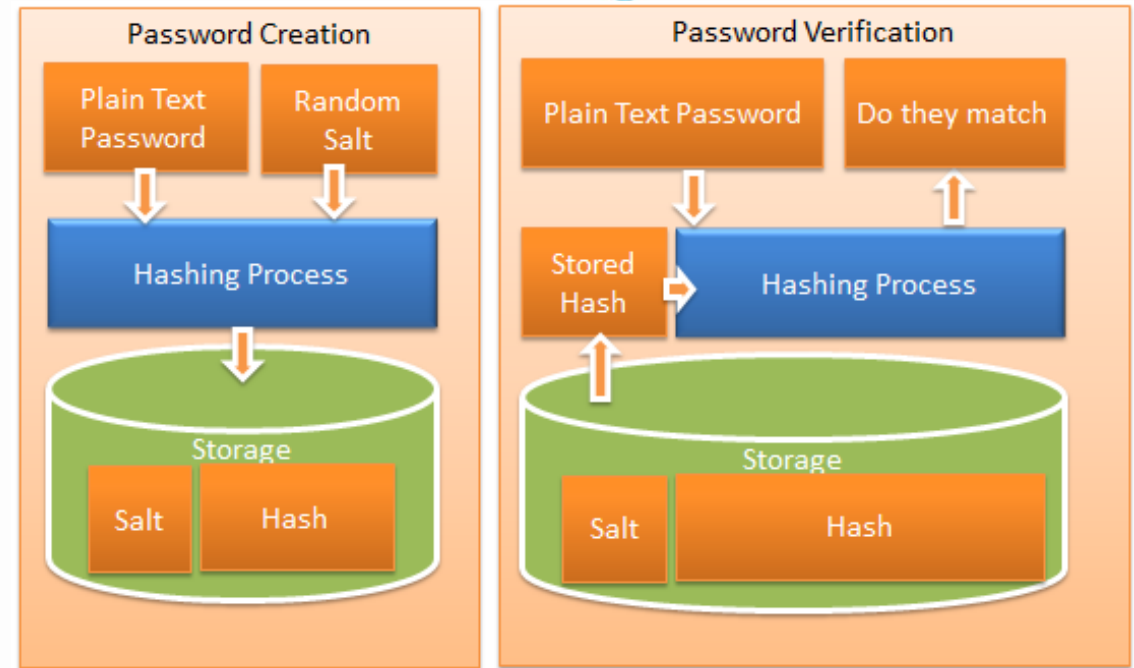
- One-way, repeatable algorithm to change data into a “hash value”
- One-way means there is no way to get to the original data, given only the hash
- Repeatable means if I run the same original data through the algorithm again, I’ll get the same result
- Used to verify data transmissions (aka, checksum)
- Used for storing passwords securely

Hashing Data



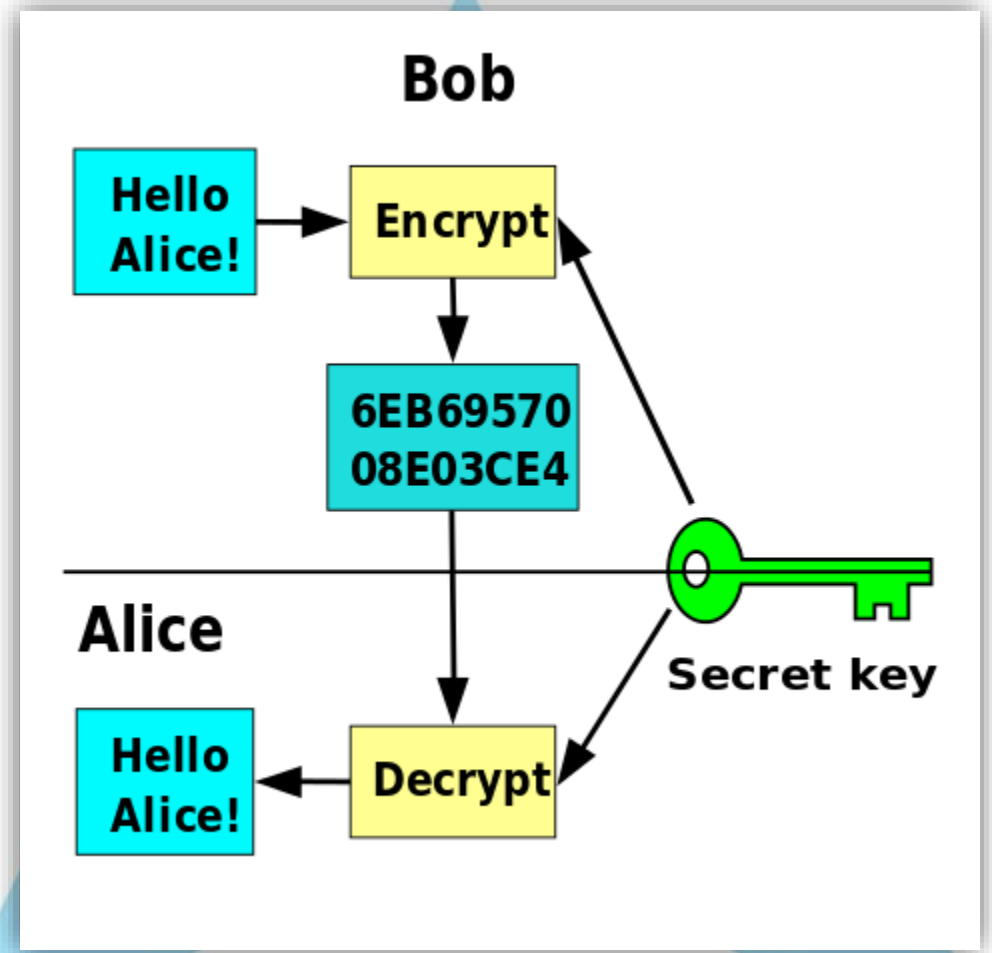
Hashing Passwords

- Password is hashed when created
 - Hash is stored in DB
- To login, password is hashed using the same algorithm
 - Hashes are compared.
- Adding a salt prevents dictionary attacks
 - Salt also stored in the DB
- Increasing work factor greatly increases security
 - Hash the hash



Encryption – Symmetric Key

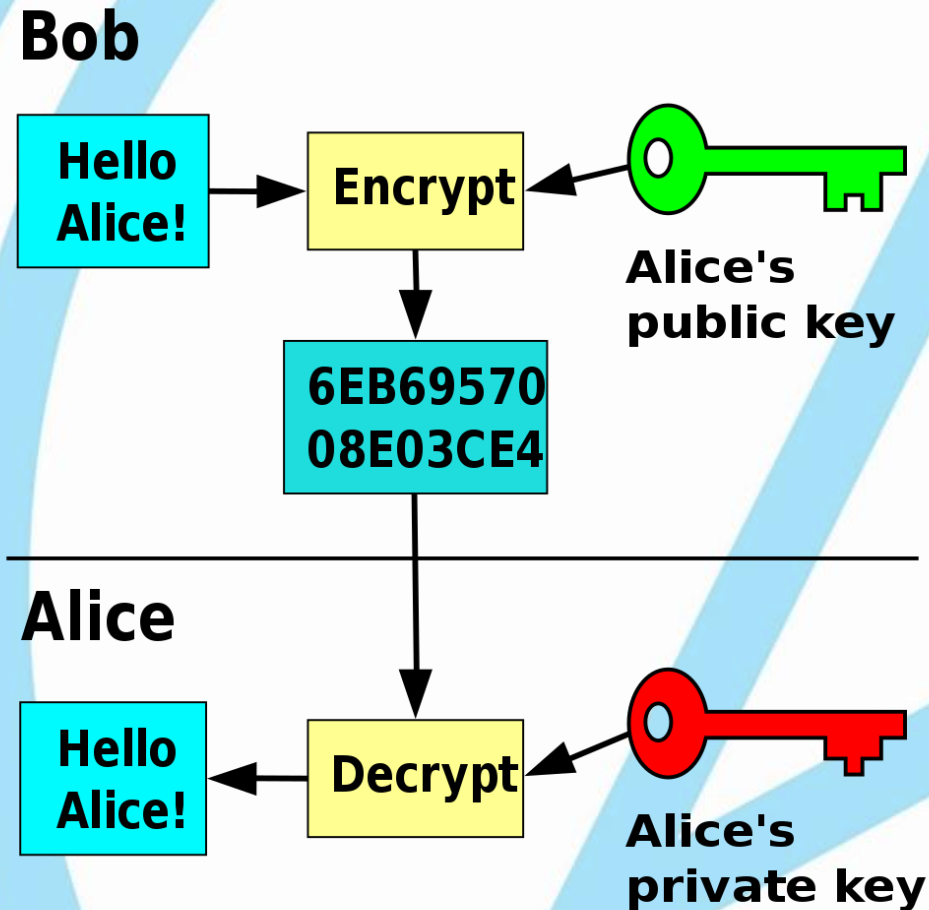
- Uses a single key to encrypt (lock) and decrypt (unlock) the data
- “Shared secret”
- Examples:
 - Password-protected files
 - Windows BitLocker



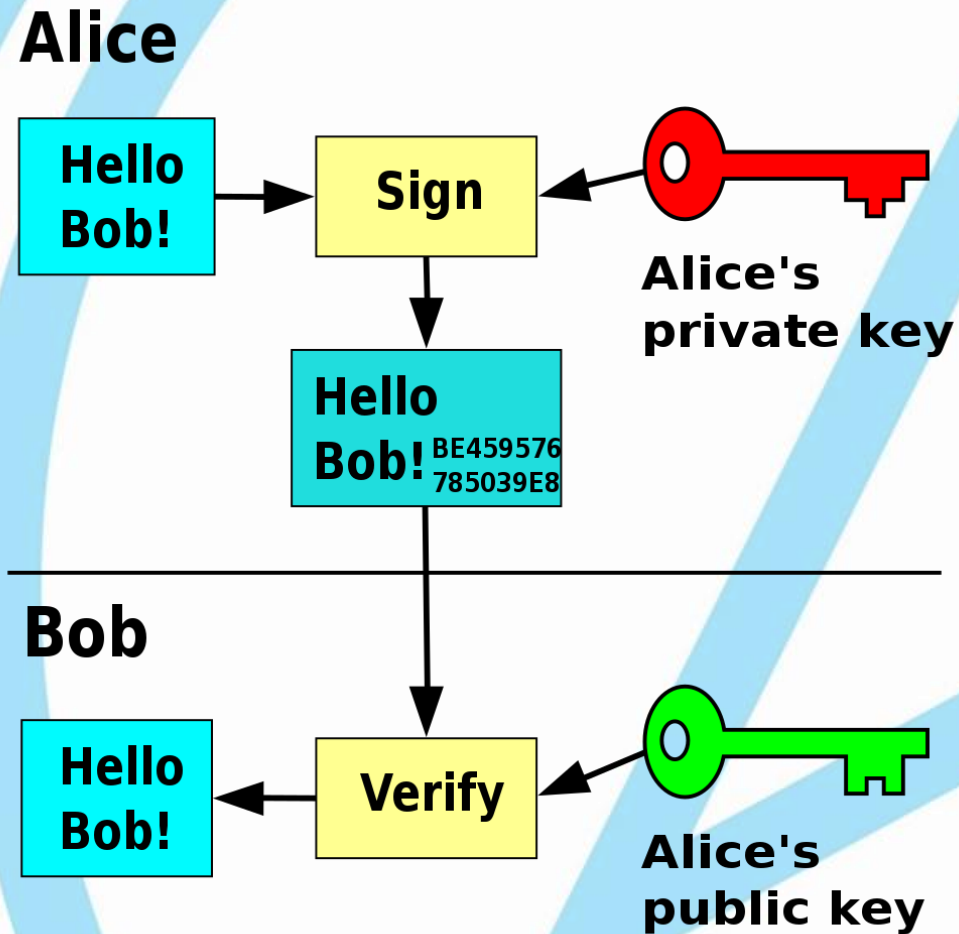
Encryption – Asymmetric Key

- Public key cryptography / Public Key Infrastructure (PKI)
- Two keys used: a “public” key and a “private” key
 - Messages encrypted using Public must be decrypted using Private
 - Message encrypted using Private must be decrypted using Public
- Can be used to
 - Securely send data to another user, or (encrypt public, decrypt private)
 - Guarantee the identity of the sender (encrypt private, decrypt public)

Bob securely sends message to Alice



Alice proves this message is from her



Authentication & Authorization in ASP.NET

- Parts of the app may be public, and parts may be protected to only authorized users
- Controllers or Actions can be protected with an Authorization attribute
- Some actions may require the user to be in a particular role
- There are different types of authentication
 - We are going to use Session-based today
 - In Module 4, we will use token-based



Let's
Code