Spring Security

Beginner to Guru

Password Encoding

Password Storage and Encoding

- When logging in, the application needs to verify the entered password matches the password value stored in the system
- Legacy systems sometimes store passwords in plain text
  - Obviously not ideal
- Other systems encrypt the password in the database, then decrypt to verify
  - Again not ideal - can be decrypted to original value

Password Hash

- A hash is a one-way mathematical algorithm applied to the password
  - One way meaning the hash value can be generated from a password
  - But the password cannot be generated from the hash value
- Example:
  - password: password1
  - hash value: 5f4dcc3b5aa765d61d8327deb882cf99
- In this theoretical example, the string ‘password1’ will always hash to 5f4dcc3b5aa765d61d8327deb882cf99

Password Salt

- Problem where hash functions generating known hash values
  - Became a dictionary attack to guess passwords from hash value
- Solution is to use a salt value
- A salt is additional data added to the value being hashed
- Example of password with salt: password1{ThisIsMyReallyLongPasswordSaltValue}
- Modern algorithms use random salt values
  - Thus hash value changes each time

Password Hash Functions

- The security area of Hash functions is effectively an arms race
  - As computational power increases, researchers find more vulnerabilities
- Spring Security supports plain text and older hash functions for compatibility with legacy systems
- These encoders are marked as deprecated to warn you they are not recommended for use

Delegating Password Encoder

- Spring Security 5 introduced a delegating password encoder
- Allows storage of password hashes in multiple formats
- Password hashes stored as - {encodername}<somepasswordhashvalue>
- Thus allows you to support multiple hash algorithms while migrating.

Password Encoder Recommendation

- The Spring Security team recommends using an adaptive one way encoding function such as:
  - BCrypt (Default)
  - Pbkdf2
  - SCrypt
- These are also considered 'slow', which are computationally expensive to guard against brute force attacks

SPRING FRAMEWORK

GURU