



Spring Security

Beginner to Guru

Authorization in Spring Security



Authorization in Spring Security

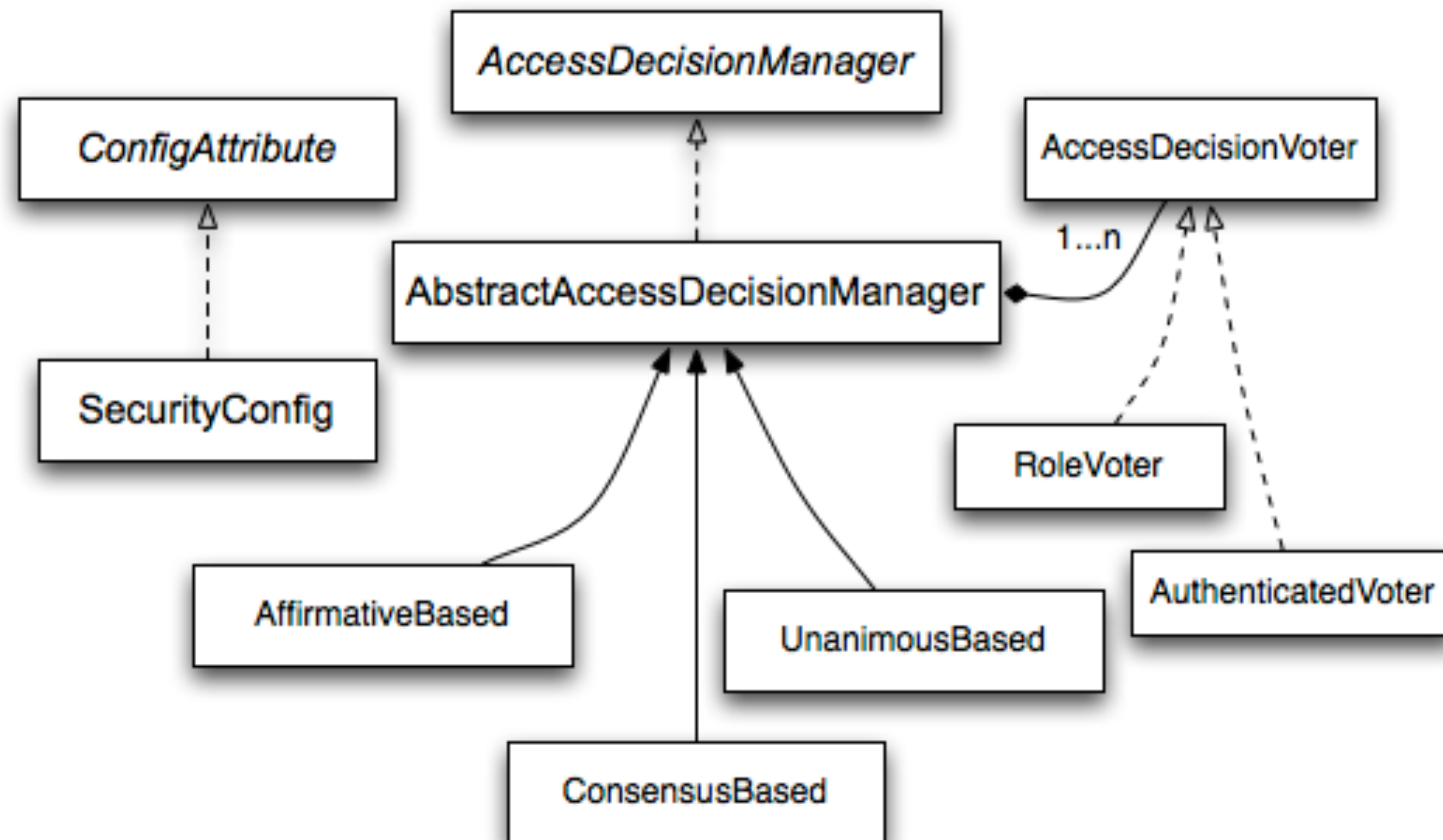
- Authorization is the approval to perform an action within the application
- Authorization can be as simple as allow all or is authenticated
- Specific actions can be limited to specific roles or authorities
- By default, Spring Security roles start with "ROLE_"
 - Example: ROLE_ADMIN
- Spring Security authorities may be any string value





Roles vs Authorities

- Typically a role is considered a group of one or more authorities
- In a Spring Security context:
 - Roles by default start with "ROLE_"
 - Configuration uses methods of `hasRole()` or `hasAnyRole()` - requires prefix
 - Authorities are any string
 - Configuration uses methods of `hasAuthority()` or `hasAnyAuthority()`





Access Decision Voters

- Access Decision Voters provide a vote on allowing access
 - **ACCESS_ABSTAIN** - Voter has no opinion
 - **ACCESS_DENIED** - Voter does not approve
 - **ACCESS_GRANTED** = Voter approves access



Role Voter

- Most commonly used voter in Spring Security
- Uses role names to grant access
- If Authenticated user has role, access is granted
 - If no authorities begin with prefix of ROLE_ this voter will abstain



Authenticated Voter

- Grants Access based on level of authentication
 - **Anonymously** - Not Authenticated
 - **Remembered** - Authenticated via Remember me cookie
 - **Fully** - Fully Authenticated



Consensus Voter

- Accepts list of Access Decision voters
- Polls each voter
- Access granted based on total of allowed vs denied responses



Role Hierarchy Voter

- Allows configuration of Role Hierarchies
- Example:
 - ROLE_USER
 - ROLE_ADMIN > ROLE_USER > ROLE_FOO
- ROLE_ADMIN will have all of its authorities, and those of ROLE_USER and ROLE_FOO



Security Expressions

- **permitAll** - Allows all access
- **denyAll** - Denies all access
- **isAnonymous** - Is Authenticated Anonymously
- **isAuthenticated** - Is Authenticated (Fully or Remembered)
- **isRememberMe** - Is Authenticated with Remember Me Cookie
- **isFullyAuthenticated** - Is Fully Authenticated



Security Expressions

- **hasRole** - Has authority with ROLE_***
- **hasAnyRole** - Accepts list of ROLE_*** strings
- **hasAuthority** - Has authority string value
- **hasAnyAuthority** - Accepts list of string authority values
- **hasIpAddress** - accepts IP Address or IP/Netmask





Http Filter Security Interceptor

- Securing specific URLs is done using Spring Security Filters
- Filters use configured voters to determine authorization
- Security expressions available for use in Java configuration of HttpSecurity



Method Security

- Spring Security also has method level security
- Enable using **@EnableGlobalMethodSecurity** configuration annotation
- **@Secured** - accepts list of roles, or IS_AUTHENTICATED_ANONYMOUSLY
- **@PreAuthorize** - accepts security expressions
- Under covers Spring Security is using AOP to intercept and use the AccessDecisionManager
 - Same technique as Filter

