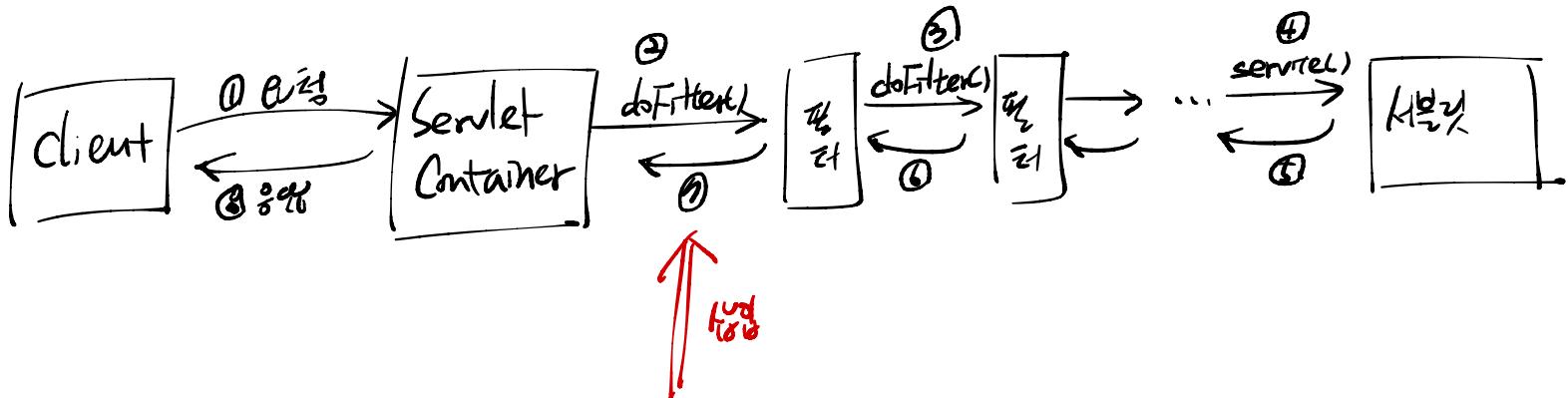


11. Spring Security 2회

* 31C



1. 컨테이너의
필터는 허용하지
않는다

⇒ **<<Filter>>**
DelegatingFilterProxy
org.springframework.web.filter

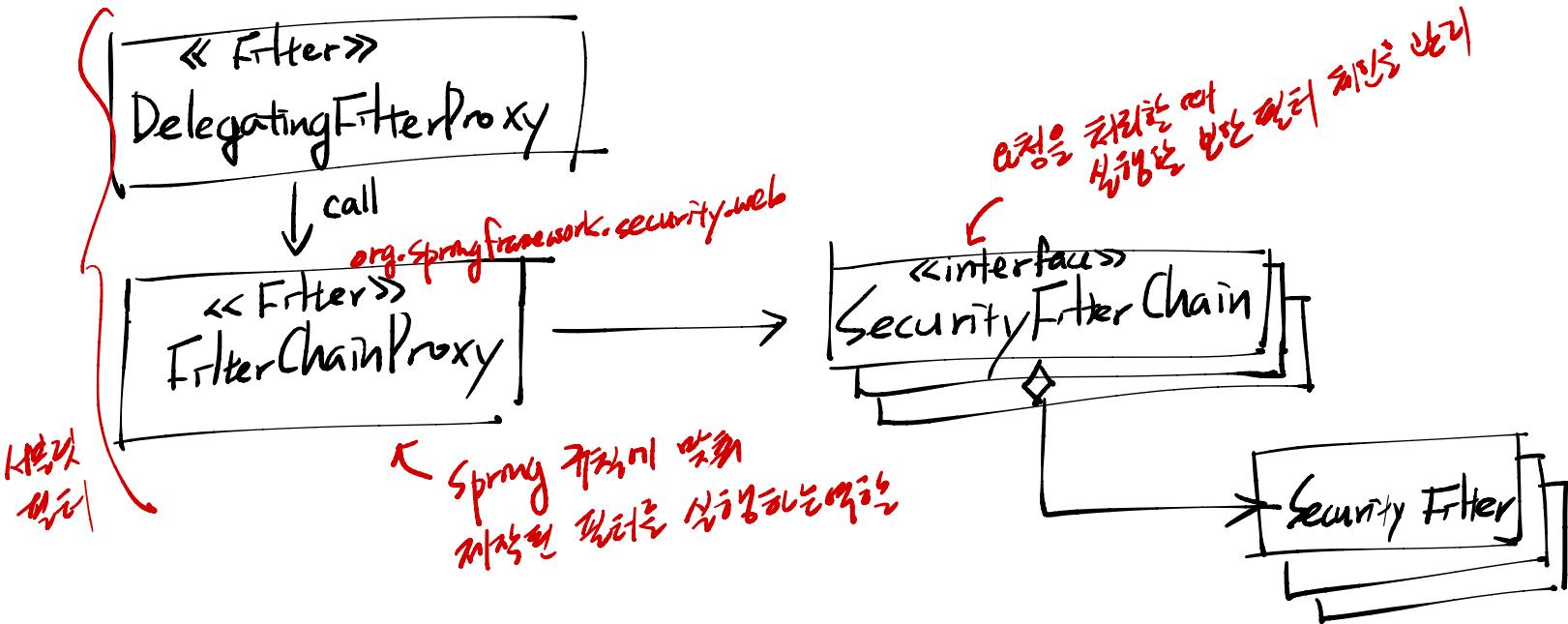
2. 컨테이너의
필터는 허용

**스프링
필터체인**

- security
- Batch
- Data
- cloud
- =

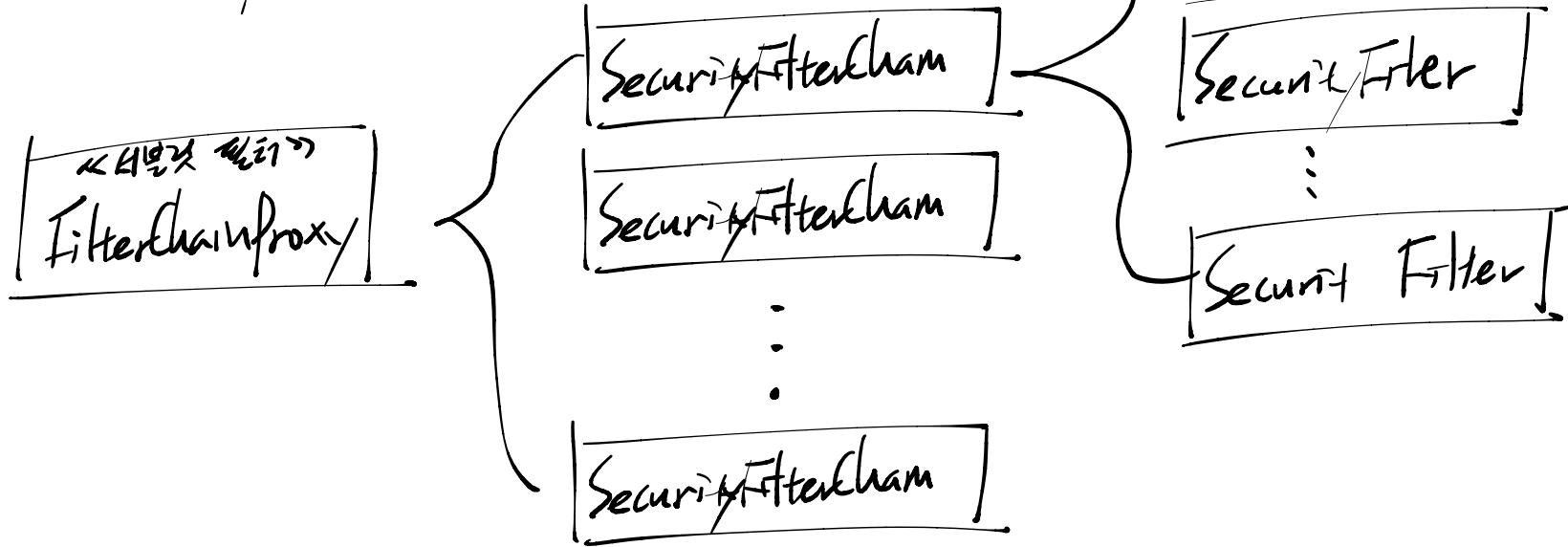
11. Spring Security 구조

* FilterChainProxy ← Spring Security 프로토콜의 핵심 구조.



11. Spring Security 介绍

- * SecurityFilterChain ← 逻辑过滤链



* HttpSecurity ← Security Filter 12.23 도입

httpSecurity.authorizeHttpRequest()

요청에 대해 실행 권한을
가지면 처리도록

수행 권한 설정

권한 설정

권한 설정하는 →
권한 있는 경우

(authorize) → ?

권한 있는 경우 12.23

* Nginx Mirroring

authorize.anyRequest().authenticated()

↑
Mirroring

↑
SSL passthrough

↑
SSL offloading by Nginx itself

* UserDetails et UserDetailsService

↑
32s! Ab82l284Uz
wz74xm

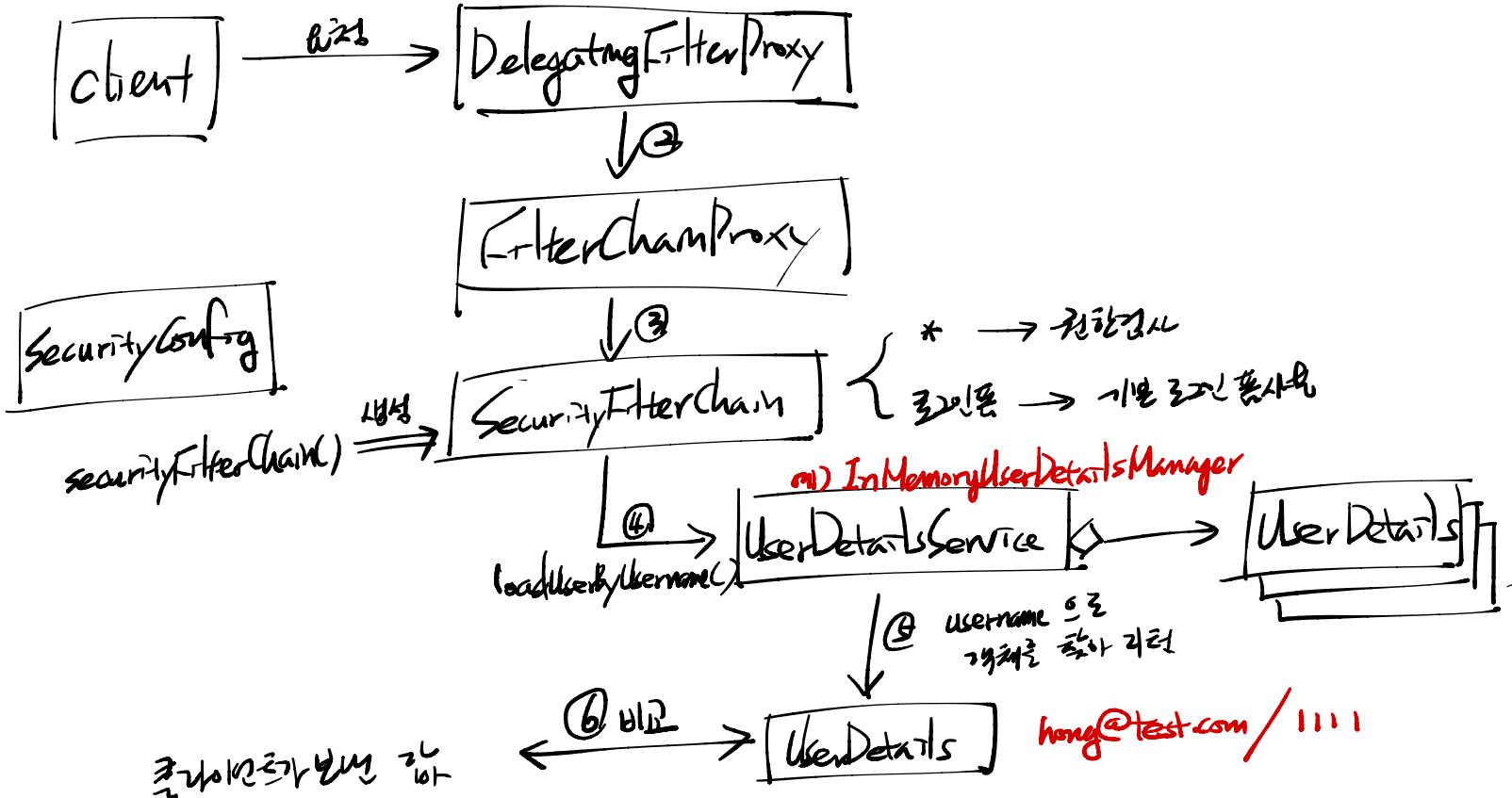
•||.

↑
ID/pwd z Ab82l284Uz
76133 74xm

-||.

User → Userservice
X
n/a erzeugt Ab82l284Uz

* 허깅 퍼시



* ↗ ↘ ↗ ↘ ↗

① classic way

→ ② modern way

```
class Car {  
    void setSunRoof();  
    void setNavigation();  
    void setBlackBox();  
    void coloring();  
    void printLabel();  
};
```

```
class Car {  
    void setSunRoof();  
    void setNavigation();  
    void setBlackBox();  
    void coloring();  
    void printLabel();  
};
```

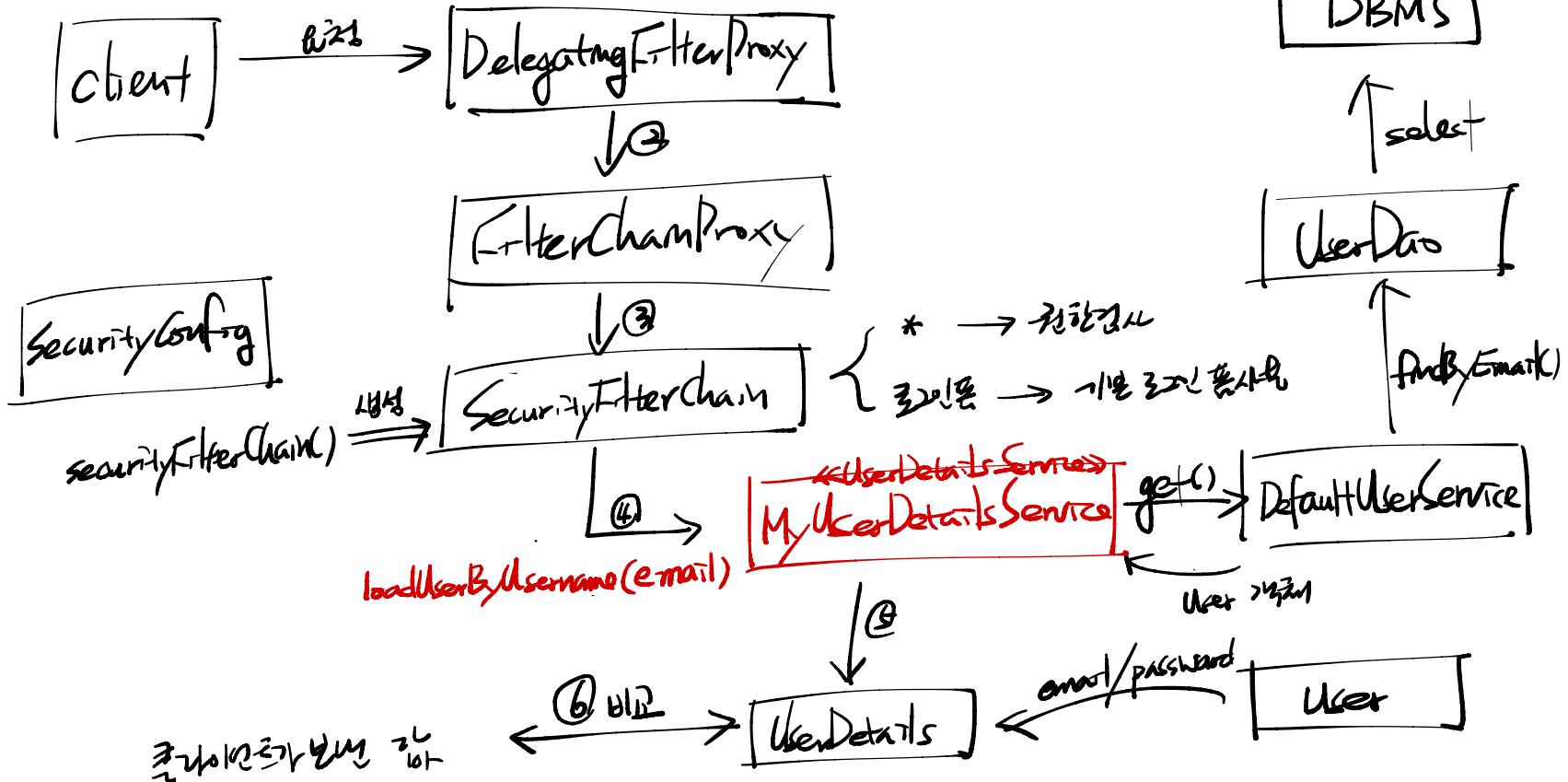
- setSunRoof()
- setNavigation()
- setBlackBox()
- coloring()
- printLabel();

return
this

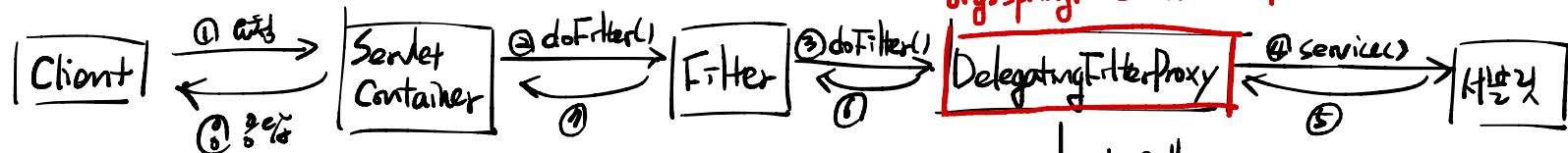


Method Chaining

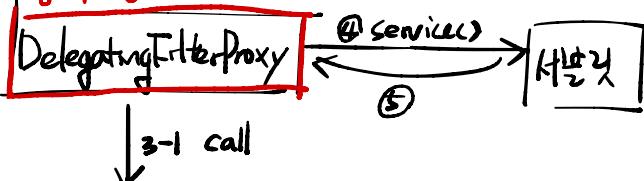
* သိမ်းသော



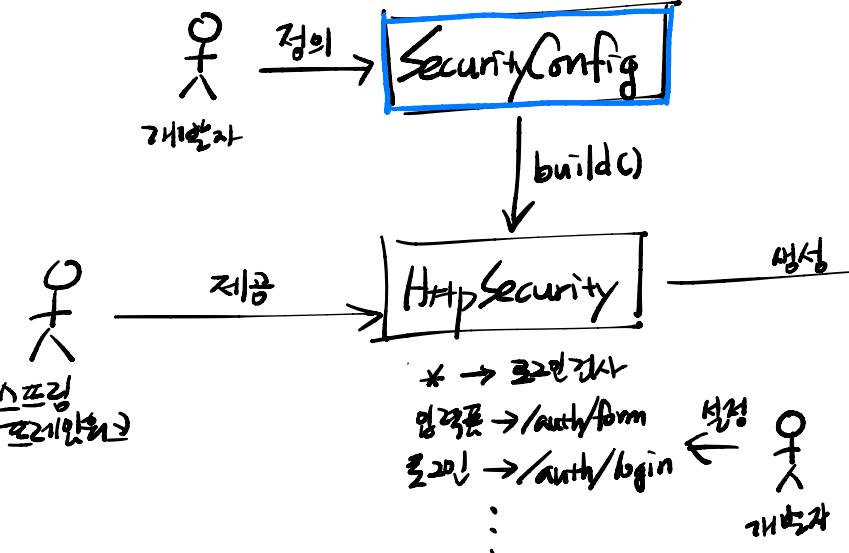
* 요청 처리 과정



org.springframework.web.filter



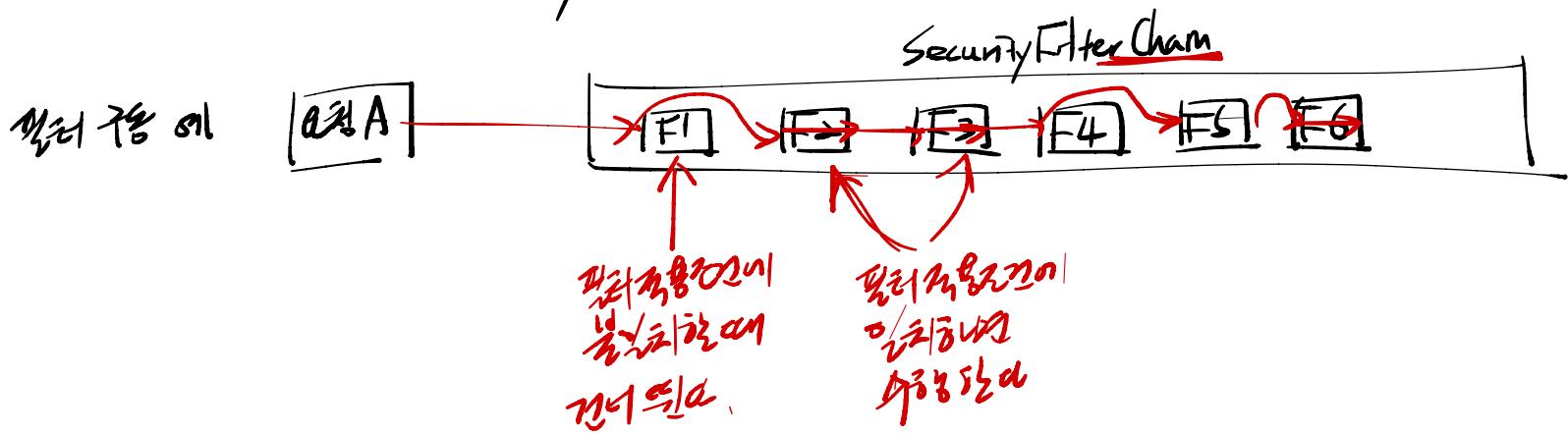
org.springframework.security.web



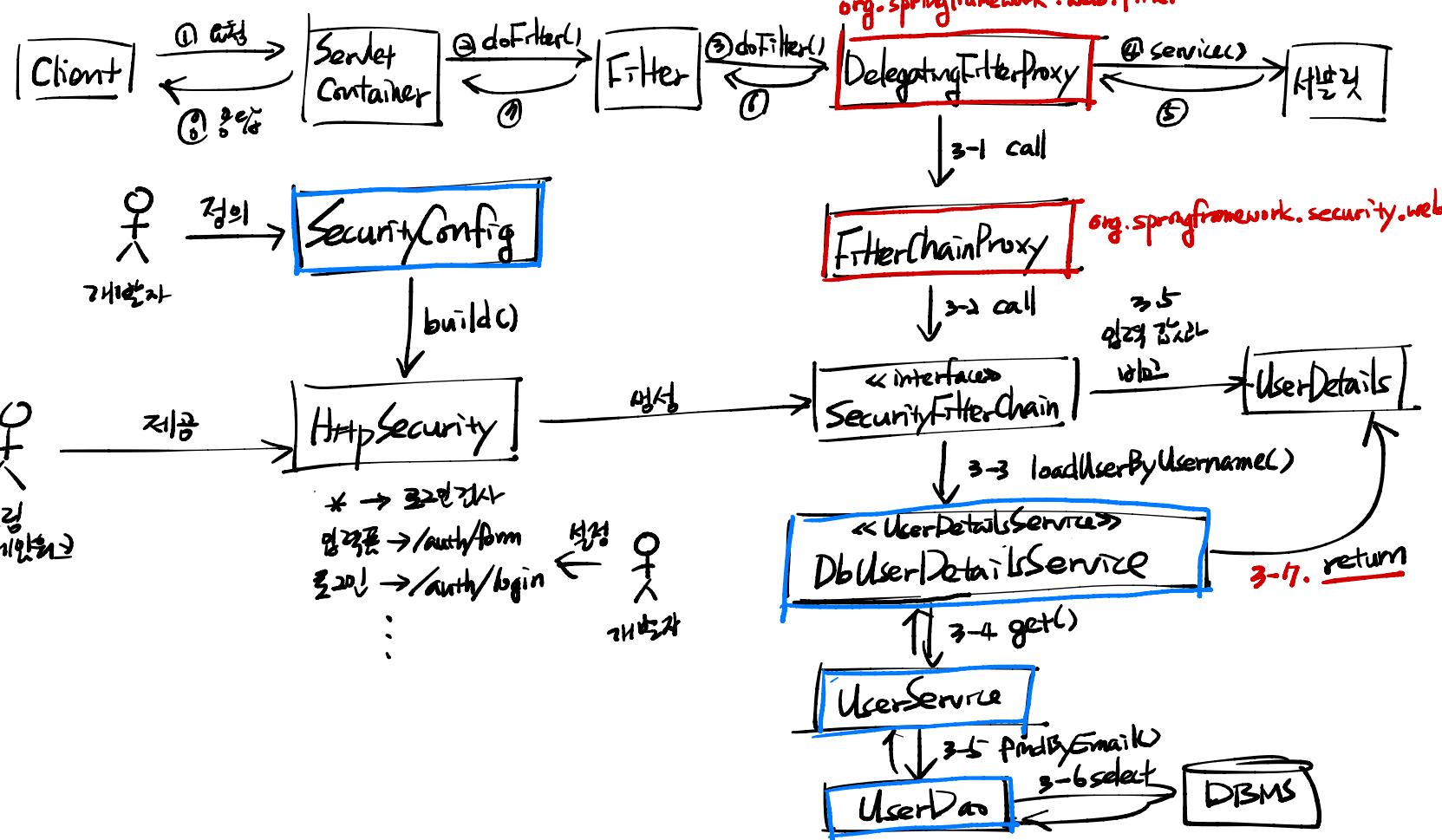
* filterSecurity ← 개발자가 설정한대로 필터체인을 구성



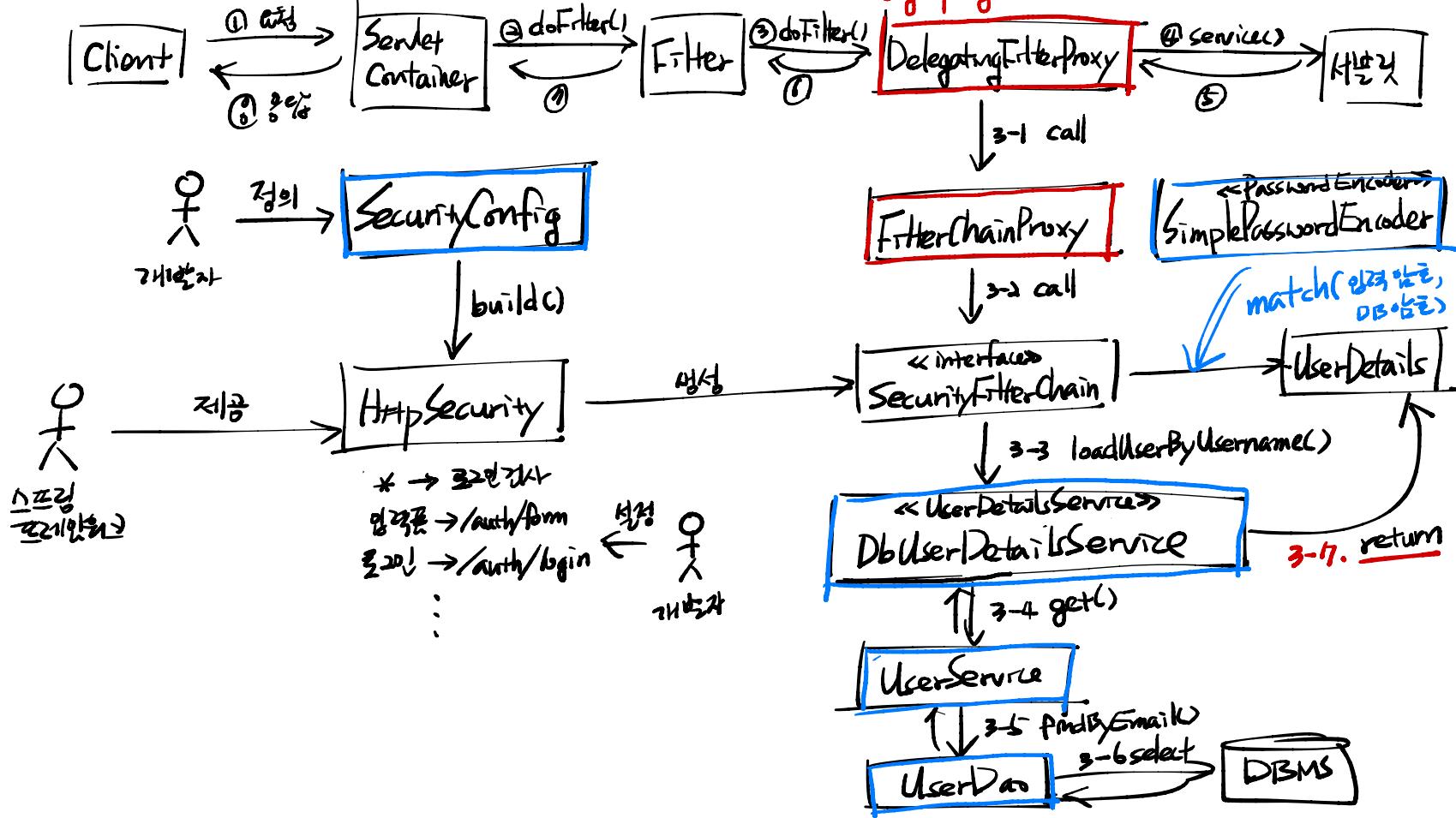
SecurityFilterChain



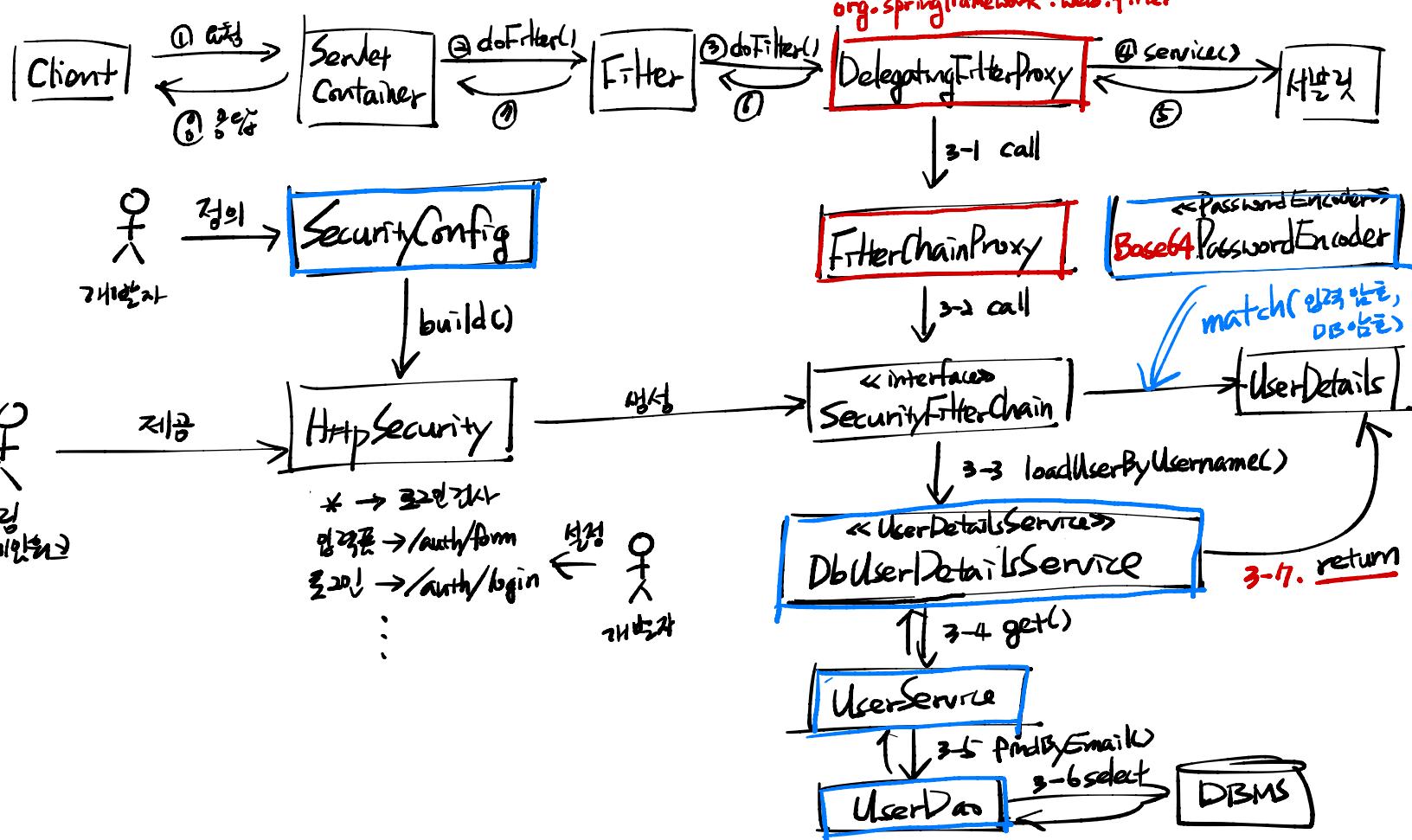
* 보안 처리 흐름 II - > UserDetailsService



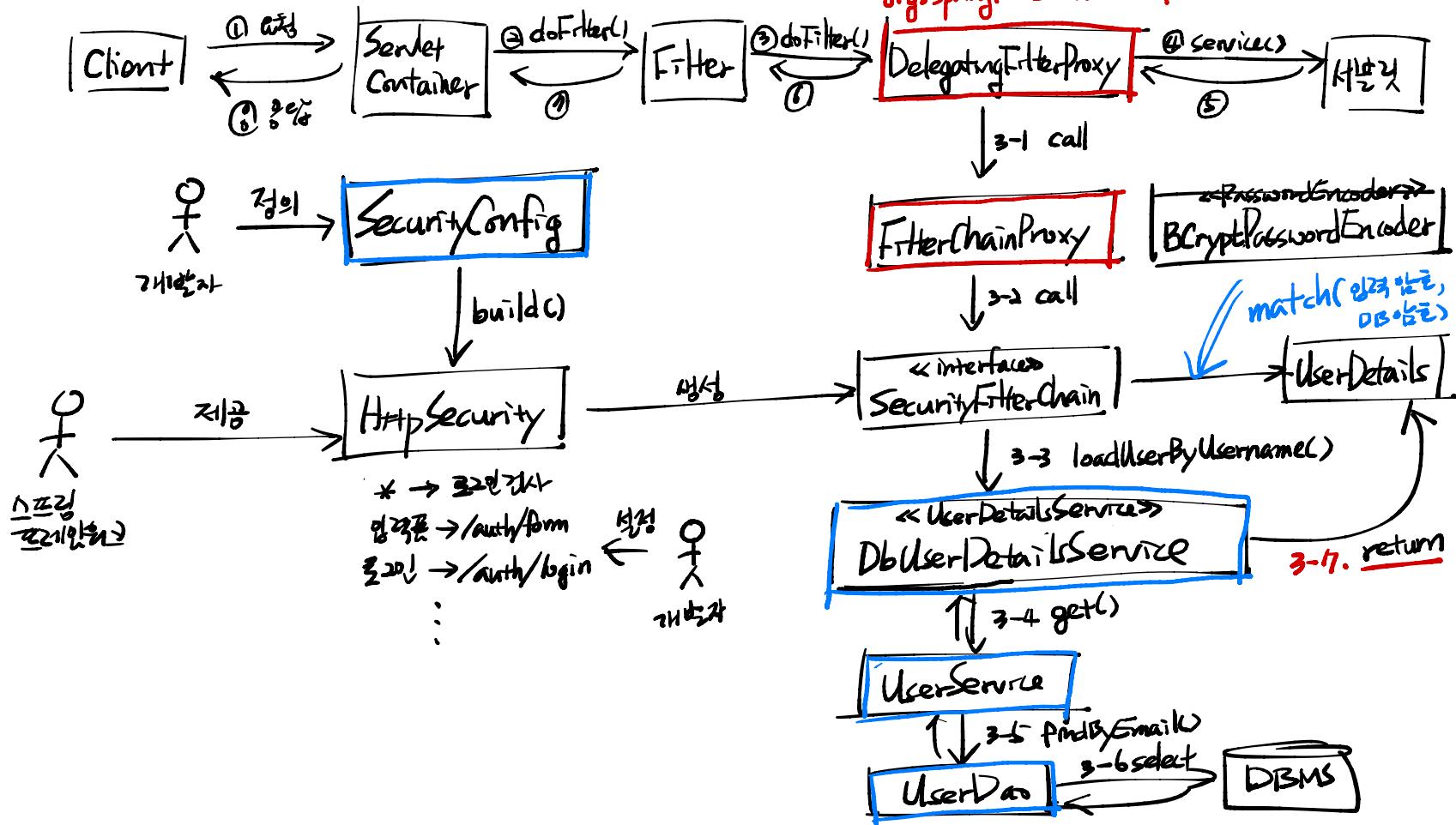
* 요청 처리 과정 III → 커스텀 PasswordEncoder



* 비정치권 고대 I - 키스럼 PasswordEncoder



* 보안 처리 과정 V - PasswordEncoder



* Log Level

FATAL - 시스템을永久로 침체시킬 위험성을 가진 경우가 발생할 때.

 a) `log.fatal("경우 메시지");` ↳ DB 연결 실패, 머신리 푸드

ERROR - 특별히 경고할 필요는 없지만, 그러나 시스템 실행에 대한 흔적을 남기고자.

 m) `log.error("경우 메시지");` ↳ 해당 디바이스의 데이터의
 전송 오류. 디버깅 목록

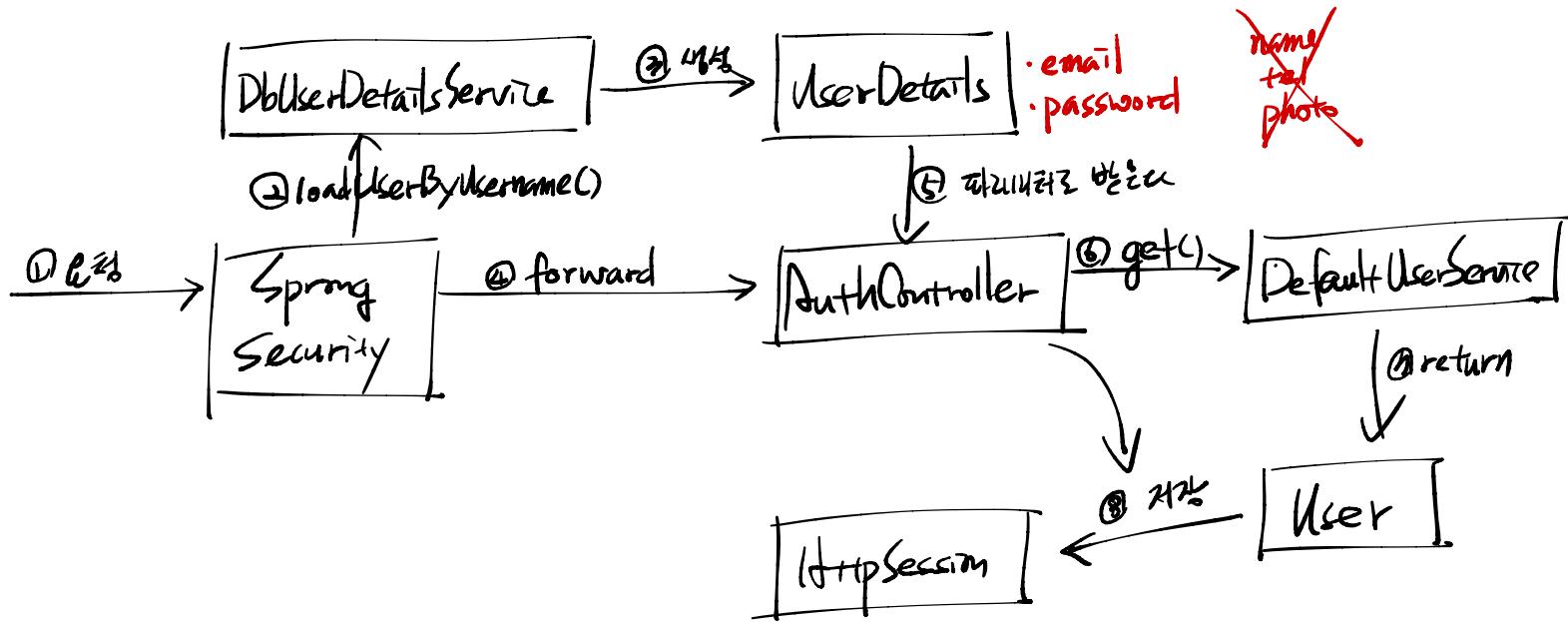
WARN - 정상적인 흔적을 남기는 경우,
 ↳ 비정상적인 경우(예: 브라우저나 API 사용)

INFO - 시스템 관리자가 솔루션 놓여온 핵심을 알 수 있도록, 정보를 확인하는 경우.
 ↳ 예: 사용자의 IP 주소, 요청 URL 등

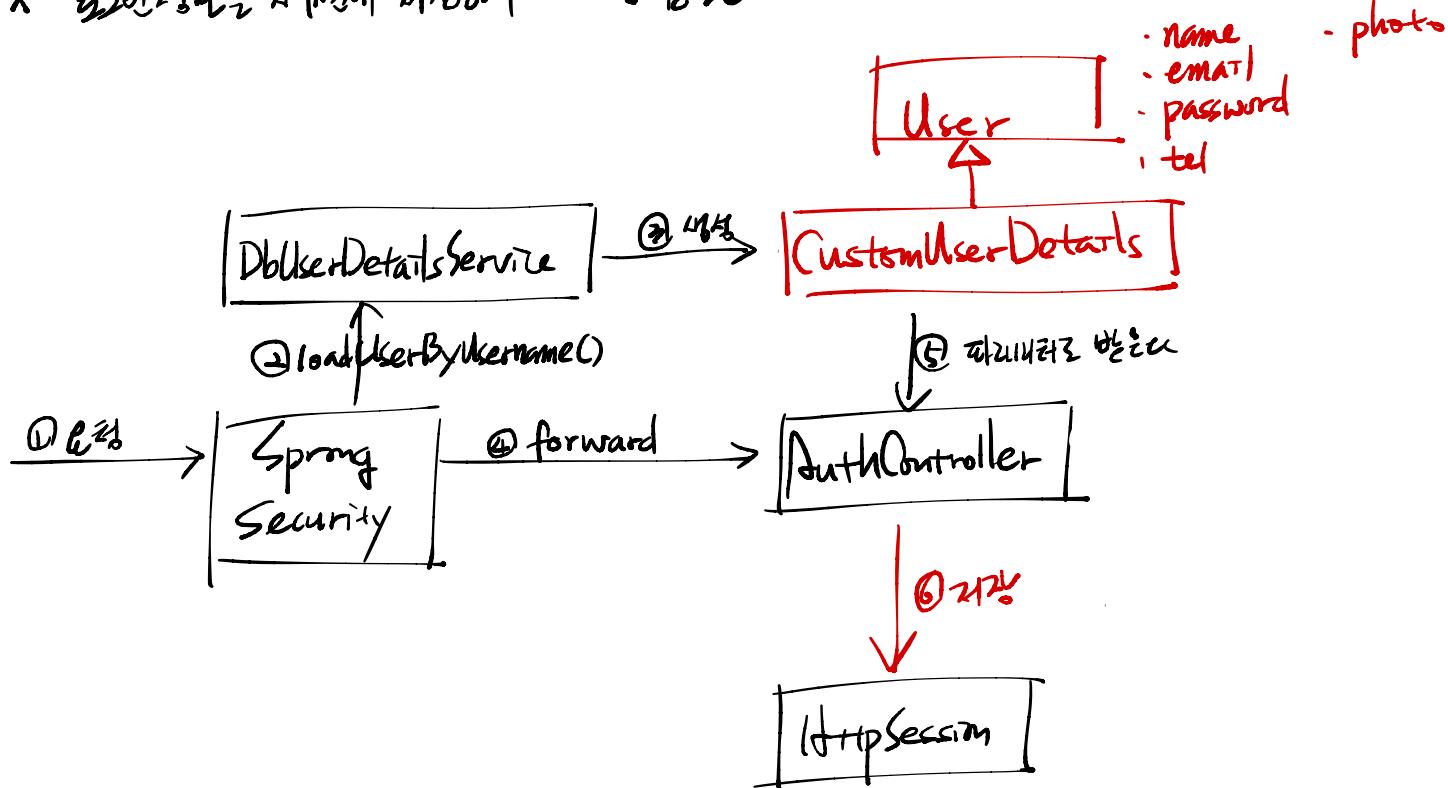
DEBUG - 개발자/운영자가 필요한 정보를 출력하는 경우
 ↳ 예: 로그, 흐름 디버깅 등

TRACE - 특정 실행점이나 솔루션 구조를 제작하는 경우 am 허용되는 범위로 출력

* 320123456789012345678901 - 10자



* 320123456789012345678901 - 10位数



* 320123456789012345678901 - 10자 3

