

<Day 2 Homework>

작성자 : 김영환

1. 액티브X가 무엇일까?

>

원래 뜻은 마이크로소프트가 개발한 소프트웨어 프레임워크지만, 보통은 인터넷 익스플로러(이하 IE)용 애드온인 ActiveX Control 을 뜻하는 말로 쓰인다.

웹 브라우저를 기반으로 한 프로그램임에도 불구하고 윈도우 내부 조작, 파일 생성, 삭제, 레지스트리 변경 등이 가능하다. 이 때문에 UAC (User Account Control, 뿌연화면에 alert 창 뜨면서 승인을 요청하는 그 상황.)를 획득하면 보안에 심각한 위험이 생길 수 있다. 다만 이러한 권한을 얻으면 액티브X 개발자 입장에서는 편리하고, UAC 권한을 되돌리면 프로그램이 동작하지 않을 수 있기 때문에 권한 반환을 잘 안하는 편이다.

2. 국내 웹에 액티브X가 정착되게 된 이유 및 배경

>

2000년까지 마이크로소프트는 IE에 40비트 수준의 보안 접속만 허용했다(수출용 기준, 미국 내에선 128비트 수준). 이런 이유로 정부는 1992년 2월, 128비트 및 256비트 대칭 키 블록 암호 알고리즘 SEED 를 채택하여, 이를 액티브X 기술을 바탕으로 적용한다.

1997년 당시에는 상당히 안전한 기술이었으나, 공인인증서 사용의 의무화, 사용자에게 책임회피를 유도하는 법안 등으로 인해 여러 문제가 제기된다. 게다가 미국 정부에서는 곧 '보안기술 수출제한'을 해제해서 액티브X는 그 효용성이 사라졌다. 액티브X 사용자제를 권고하던 마이크로소프트는 IE를 포기하고 Edge 브라우저를 채택했다.

그러나 정부는 액티브X로 구현한 공인인증서 제도를 계속 유지하면서, 웹 호환성 면에서 한국은 갈라파고스화되어있다.

<출처: 나는 프로그래머다, '액티브X와 EXE', 김호광, 임백준, 정도현>

3. 기술적 부채(Technical Debt)에 대해 알아보기

>

기술적 부채는 프로그래머로 인해 ‘문제가 발생할 가능성을 가진 모든 작업들’을 말한다. 이 말은 제안한 Ward Cunningham 은 이러한 제대로 다뤄지지 않은 문제들이 복리처럼 늘어나, 프로젝트 전체가 실패할 수 있음을 지적했다.

기술적 부채를 갚기 위해 프로그래머는 꾸준히 리팩토링과 테스트를 해야한다. 장기적으로 볼 때 기술적 부채가 적을 수록 프로젝트는 안정적으로 마무리될 수 있고, 차후의 유지보수 난이도도 감소하게 된다.

4. 위 내용을 조사하며 느낀 점

>

액티브X에 대해서는 막연하게만 알고 있었으나, 도입된 이유, 기술변화를 외면할 때 발생하는 문제 등을 조사하면서 여러 생각을 해볼 수 있었다. 개발자로서 액티브X는 타성에 젖은 개발자들의 비극인 것 같다.

사실 정부기관이나 큰 회사들은 변화가 느릴 수 밖에 없고, 검증된 기술에 머무를 수 밖에 없다는 한계가 있다. 그러한 점에서 결국 변화를 만들고, 그 흐름에 움직일 수 있는 것은 작지만 Solid 한 기업이나 스타트업이 될 수 밖에 없다는 생각을 했다.