공개소프트웨어! 얼마나 안전한가?

안철수연구소 전규현

1. 서 론

공개소프트웨어(Open source software)란? 소스코드를 공개한 상태로 제공되는 소프트웨어로서 누구나 자유롭게 개작하여 재배포 할 수 있는 소프트웨어를 말한다. 과거에 비하면 정말 많은 쓸 만한 공개소프트웨어들이 있다. 공개소프트웨어가 이렇게 활성화 된 데는 인터넷이 톡톡히 역할을 했다. 인터넷을 통해서 수많은 개발자들의 커뮤니티(http://sourceforge.net)가 생겨나고, 이를 통해 공개소프트웨어가 배포되면서 더욱 확산되었기 때문이다. 그리고 공개소프트웨어가 "돈"이 되기 시작했고, 새로운 비즈니스모델로도 각광을 받고 있는 상태이다.

공개소프트웨어를 사용하는 사용자(기업, 정부, 개인) 입장에서 보면, 여러 가지 매력적인 면이 있는 것은 확실하다. 대부분은 비용이 무료이고, 유지보수 비용이 적게 든다. 그럼에도 불구하고, 예상외로 공개소프트웨어의 도입은 기대보다 적은 것 같다. 공개소프트웨어의 도입을 저해하는 요소로는 다음과 같은 것들이 있다.

- 어렵다는 인식 및 교육/지원 부족
- **보안에 취약하다는 인식**
- 사용편의성 미흡
- 사용 Killer 소프트웨어 맹신
- 응용소프트웨어 미비
- 패치가 잦고 유지보수의 어려움

그 외에도 공개소프트웨어 확산을 위해서는 넘어야 할 여러 가지 장애요소가 있지만, 여기서는 특히 "보안"에 관련된 이슈를 중점적으로 알아보고자 한다. 본고를 전재하기 위해서 먼저 공개소프트웨어의 전반에 대해서 알아보고 "보안"적인 측면을 알아보기로 한다.

2. 공개소프트웨어의 경제학

우선, 공개소프트웨어가 어떠한 경제적인 장점이 있는지 알아본다. 공개소프트웨어로 어떻게 돈을 벌수 있을까 의문을 가지는 독자가 있을지는 모르겠지만, 실제로 수많은 사람들은 돈을 벌기 위해서 공개소프트웨어를 만들고 있다. 물론, 상당히 많은 사람들이 순순히 사명감으로 많은 사람들에게 혜택을 주기 위해서 공개소프트웨어를 만들고 있다. 하지만, 공개소프트웨어는 여러 가지 방법으로 돈이 되고 있고, 이로 인한 돈벌이가 꼭 부정적인 의미만을 가지고 있지 않다. 오히려, 공개소프트웨어의 사회공헌적인 모습과 경제적인 모습이 적당히 조화를 이룰 때 더 좋은 소프트웨어가 많이 나올 것으로 생각한다.

그럼, 공개소프트웨어가 어떻게 돈이 되는지 알아보려고 한다.

2.1 거시적인 경제효과

공개소프트웨어로 직접적인 수익을 거두기보다는 공개소프트웨어를 널리 쓰이게 함으로써 간접적인 수익을 거둘 수 있다. 간단한 예로 Netscape사가 웹브라우져의 소스를 공개하여 인터넷(웹)의 사용을 확대함으로써 Netscape사의 주력제품인 웹서버를 더 많이 팔려 하였습니다. 웹브라우저의 가격을 내리는 방법도 좋지만 소스를 공개하는 방법이 사용을 확대하는 가장 확실한 방법 중에 하나이다. 웹브라우져의 개발 기술 자체가 일반화 되어서 웹 자체의 사용이 많아지게 되어서 결국 웹서버 시장에서 거의 독점적인 지위를 가지고 있던 Netscape사의 매출이 오르게 된다. 이러한 현상은 우리 주변에 많이 볼 수 있다. 프린터를 싸게 팔면 "프린터 잉크"의 매출이 올라서 더 많은 수익을 올리는 것과 비슷하다. 소프트웨어 쪽을 보면 Linux가 공개 되어 있음으로써 더 많은 PC가 판매가 되고, CPU나 메모리 제조사가 더 많은 수익을 올리기 때문에 여기에 투자를 하는 것이다[1].

주로, 거대 회사나 특정 제품을 독점하고 있는 회사들이 많이 사용하는 방법이다.

2.2 소프트웨어는 공짜, 기술지원은 유료

Redhat은 Enterprise 버전을 내면서 이 전략으로

확실히 굳혀가고 있다. 공개소프트웨어의 세상에서는 자유로운 개작과 이의 공개와 재배포가 가능하므로 이 전략은 아주 효과적이다. 사용자 입장에서는 공개소프트웨어를 사용하면서 기술지원을 받을 수 있는 장점도 있다.

2.3 제품을 널리 알려서, 나중에 돈 받고 팔기

소프트웨어를 개발해서 많은 사람들이 기억할 만큼 널리 퍼지게 하는 일은 보통 어려운 일이 아니다. 엄청난 마케팅 비용이 들기도 하고, 아주 오랜 시간이 걸리는 일이다. 반면, 공개소프트웨어나 Freeware라면 종종 아주 짧은 시간에 많은 사람들이 기억할 만큼 널리 알려지기도 한다. 일단, 이렇게 널리 알려지면 차기 버전부터 유료화 정책을 펴던지, 별도의 유료버전을 출시하여 수익을 거두는 방법이다. MySQL 서버가 일부 비슷한 전략을 사용한 것으로 생각이 된다.

2.4 개인에게는 공짜, 회사는 유료

위와 아주 비슷하지만, 처음부터 개인적인 사용에 대해서는 무료정책, 회사에는 유료로 판매하는 방법이다.

공개소프트웨어를 포함하여 Freeware나 Shareware들이 이와 유사한 정책을 사용하고 있다.

공개소프트웨어의 주된 목적 중의 하나는 "널리 사용"하게 하는 것이다. 이를 사용하는 사용자는 여기에 일조를 하고 있다. 따라서 공개소프트웨어를 올바르게 사용하는 것도 의미 있는 일이라고 할 수 있습니다. 공개소프트웨어는 하나의 패러다임으로서 경제적으로도 충분히 가치가 있는 것이고, 앞으로 점점 더 확대되어 나갈 것이라는 것을 예측할 수 있습니다.

3. 공개 소프트웨어는 무료인가?

많은 사람들이 공개 소프트웨어가 무료로 생각하는 경우가 있다. 개인에게 있어서는 어느 정도 맞는 말이지만, 개인을 벗어나면 틀린 말이다. 심지어는 상용 소프트웨어보다 더 비싼 경우도 있다. 공개 소프트웨어를 도입하게 되면 이러한 비용들이 생각해야 한다.

- 더 전문적인 전산 관리자를 채용하는 비용
- 사용자 및 관리자 교육비용
- 공개 소프트웨어를 관리하는 비용
- 관련 소프트웨어 구매 비용

그림 1은 마이크로소프트사가 자주 활용하는 "사실은 Linux가 총 비용은 Windows보다 비싸요" 자료이다. 위와 같이 주장하는 사람들도 있지만, 그렇다고 꼭 Linux가 더 비싼 것은 아니다. 노력여하에 따라서 훨씬 저렴하게 사용할 수 있다. 하지만, Linux가 무료라는 생각은 버리는 것이 좋다.

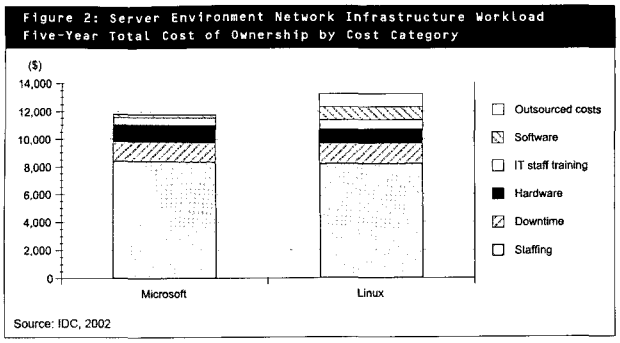

그림 1 Windows와 Linux의 TCO 비교[2]

4. Linux와 악성코드

공개소프트웨어의 대명사격인 Linux에서의 악성코드의 현황과 향후 전망을 보면 전체 공개소프트웨어와 악성코드의 관계를 짐작할 수 있지 않나 생각이 된다. 여기서 "바이러스"라고 지칭하지 않고 "악성코드"라고 지칭한 이유는 "악성코드"란 "바이러스"를 포함하여 웜, 트로이목마, 스파이웨어 등을 총칭하는 것으로서 **요즘은 순수한 바이러스는 숫자가 많이 줄면서 형태가 다양**해지고 있기 때문에, "바이러스"라는 용어보다는 "악성코드"로 지칭을 한다.

Linux를 타깃으로 하는 악성코드의 숫자는 정말로 적다. 매년 손에 꼽을 수 있을 정도이다. 그림 2는 국내에서 발견되는 Linux 악성코드의 수이다.

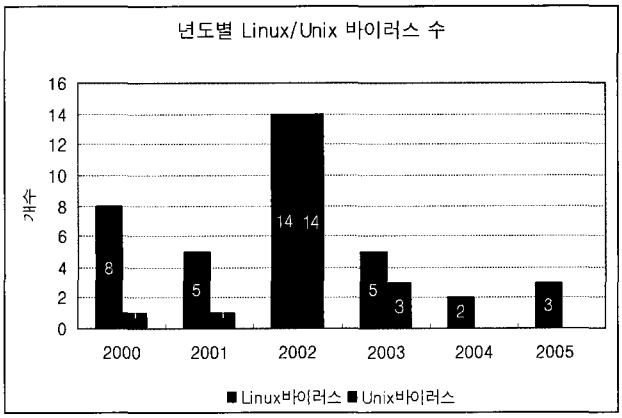

그림 2 년도별 Linux/Unix 바이러스 발생 빈도, 2000~2005, 안철수연구소

그리고 대표적으로 아래 악성코드들이 있다.

- Linux/Typot.116580 (http://info.ahnlab.com/smart2u/virus_detail_1176.html)
- Linux/Slapper.worm (http://info.ahnlab.com/smart2u/virus_detail_1030.html)
- Linux/Ramen

(http://info.ahnlab.com/smart2u/virus_detail_784.html)

Linux의 악성코드수가 적다는 것은 Linux가 원래부터 그만큼 더 안전하게 만들어 졌다는 것을 의미하지는 않는다. Linux를 Windows보다 적게 사용하는 것이 악성코드가 적은 가장 큰 이유이다. 하지만 사용비율로 보기에는 Linux의 악성코드는 너무나 적다. 이유가 하나 더 있다. 악성코드 제작자에게 있어서 Linux는 재미있는(돈이 되는) 시장이 아니기 때문이다. 요즘의 악성코드는 대부분 돈을 벌 목적으로 제작되기 때문이다. 그래서 마피아와 결탁하여 악성코드를 제작하는 경우도 드물지 않다.

현재 Linux의 대부분은 서버용으로 사용되고 있는데, 서버용 시스템에서는 광고효과가 적기 때문에 악성코드가 덜 제작된다. 돈을 벌 목적이라면 당연히 돈이 많이 되는 Windows를 더 많이 공격하는 것이 당연하다. 하지만, Linux도 Desktop 사용이 활성화 되어서, 많은 사람들의 책상 위를 점령한다면 Linux를 공격하는 것도 돈이 되기 때문에 얼마든지 많은 악성코드가 나올 수 있다. 또한, 그 종류도 Windows에서 사용했던 기법이 거의 그대로 대부분 사용되어, 다양한 형태를 띠게 될 것이다.

앞으로 Linux 사용의 확대를 가정해보고 예상을 해보면 다음과 같다.

• 밝은 면

Linux의 사용자(관리자)들은 일반사용자에 비하여 보안의식의 높고 Linux의 구조상 악성코드가 root권한을 가져야 하므로 Windows에 비하여 상대적으로 감염이나 확산의 우려가 적다.

• 어두운 면

취약점을 거래하는 암거래시장의 규모가 커질 것 이다. 정상적인 경우 취약점이 발견되면 해당 업체로 먼저 연락을 하여 패치를 하도록 한 후에 이를 공개한다. 하지만, 취약점을 발견 후 암거래시장을 통해서 광고업자에게 판매를 하고, 이 취약점을 이용하여 웜이나 스파이웨어를 제작하여 배포하는 경우가 있다. 최근의 WMF 취약점이 이러한 경로로 확산이 된 사례가 있다. Linux도 Windows만큼 활성화 된다면 Open source라는 특징 때문에 위와 같은 취약점의 거래 규모가 더 커질 수 있다. 왜냐하면, 취약점을 찾기가 더 쉬울 수 있기 때문이다. 물론 방어자도 취약점을 빨리 찾을 수 있지만, 많은 눈이 보고 있다고 꼭 빨리 찾아지는 것은 아니다.

결국 Linux 사용의 확대는 그 만큼의 보안에 대한 위협이 증가할 것이고, 그에 따른 대비책도 필요하다.

5. 공개 소프트웨어가 더 보안상 위험한가?

여기서 말하는 공개는 Open source를 말하는 것이다. 즉, "Source가 Open된 소프트웨어와 그렇지 않는 소프트웨어 중 누가 더 안전한가?" 비교이다. 소스코드가 Open되어 있으면 해커는 이를 분석하여 더욱더 쉽게 취약점을 찾아내고 이를 이용한 공격을 할 것으로 생각이 된다. 하지만, 역으로 생각하면 취약점을 찾기 쉬운 만큼 이를 방어하는 방법을 찾기도 쉽다.

표 1 공개소프트웨어와 비공개소프트웨어의 보안성 비교[3]

구분	방어자 (사용자)	공격자 (해커)
Open Source	**보안성 높음**	보안성 낮음
Closed Source	보안성 낮음	**보안성 높음**

공개 소프트웨어는 그 아키텍처가 만천하에 드러난 상태에서 개발이 되기 때문에 보안상 안전한 구조로 개발되는 경향이 있다. 하지만, 비공개 소프트웨어는 그렇지 않을 가능성이 높다. 비공개 소프트웨어는 아키텍처가 감춰져 있기 때문에, 취약한 구조로 만들어 지기 쉽다.

- 만들기 쉬운 보안에 취약한 방법으로 개발하는 경우
- 일정에 촉박하여 초기에 기획한 보안 기능을 제거하는 경우
- 기능과 편의성이 보안성보다 우선순위가 더 높게 책정되는 경우
- 개발사 또는 개발자가 의도적으로 백도어를 심어 놓는 경우
- 알고 있음에도 고치지 않는 수많은 Known-bug들

비공개 소프트웨어 보안 문제 예로는 다음과 같은 것들이 있다.[3]

- 마이크로소프트의 FrontPage Web은 백도어가 삽입되어 있다는 것이 4년 후에 드러났다.
- 볼랜드의 DB인 InterBase에는 프로그래머가 백도어를 숨겨 놓았었는데, 소스공개를 통해서 사실이 들어났었다.

반면 공개 소프트웨어는 보는 눈이 많기 때문에 처음부터 보안을 최우선으로 고려한 설계와 개발이 이루어지고 버그도 빨리 발견이 되고 빨리 고쳐지게 된다. 버그가 발견된 지 몇 시간 안에 패치가 이루어지기도 한다. 물론, 공개 소프트웨어가 모두가 항상 그렇게 되는 것은 아니다. 여러 사람이 보기 때문에 오히려 더

늦게 발견되는 경우도 있기는 하다. 이번에 [4] ISS의 X-포스(X-FORCE) 연구소에서 발표한 Sendmail의 취약점은 Sendmail 8.xx.xx의 모든 버전에서 존재하는 취약점이었지만, 그동안 찾지 못했었다. 하지만, 일반적으로 공개소프트웨어의 공개된 아키텍처에서 오는 보안성은 그 반대의 경우보다 더 안전한 것으로 생각된다.

"알고리즘의 보안을 알고리즘의 비밀성에 의존해서는 안 된다."는 암호 작성에 대한 금언이지만 오픈 소스 소프트웨어에도 충분히 적용될 수 있다.

"메쏘드를 개방하는 것이 보안에 더욱 유리하다-라는 말은 모순처럼 들리지만 진실입니다." Eric S. Raymond의 말이다.[5]

Raymond와 오픈 소스 지지자들은 오픈 소스는 운영 체계 보안을 위한 유일한 길이라는 것에 전적으로 공감하고 있다. 우선 폐쇄 소스 애플리케이션과 운영 체계는 보안 코딩을 시험하거나 입증할 수 없다. 이전의 비밀 코드가 드러나게 되면 반드시 추가적으로 보안 허점이 드러나기 마련이다. 게다가 폐쇄 코드는 허점(hole)이나 실수가 발견되었을 때 수정이 어려워진다.

오픈 소스 소프트웨어는 읽기, 재배포, 수정, 자유로운 소프트웨어 사용에 대한 권리가 보장되기 때문에 백도어는 감시에서 벗어나기 힘들었을 것이다. 결국 다른 누군가에 의해 쉽게 발견될 수 있는 개방적 소스에 백도어를 삽입할 사람은 없을 것이다.

6. 동료검토(peer review)의 중요성

위에서 계속 언급했듯이 공개소프트웨어의 뛰어남은 동료검토를 근간으로 한다. 공개된 소스코드는 수많은 사람들이 리뷰를 하게 된다. 동료검토의 범위는 소스코드에 그치지 않는다. 기획, 스펙, 코드, 테스트 관련 모든 문서가 리뷰 대상이다. 동료검토를 철저히 거친 소프트웨어는 아래와 같은 특징을 거친다.

- 설계가 보안성을 고려한 구조로 되어있다.
- 소스코드에 버그가 적다.
- 알고리즘의 선택이 합리적으로 되어 있다.
- 리뷰하기 쉽도록 소스코드가 정리가 잘 되어 있고, 주석이 풍부하다.

동료검토는 공개소프트에서만 그 중요성이 강조되는 것이 아니다. 모든 소프트웨어가 충분한 동료검토를 거쳐야만 뛰어난 제품이 탄생할 수 있다.

동료검토는 남의 코드에 문제가 없는지 감시하는 역할이 아니다. 코드를 가지고 Author와 Reviewer가 충분히 의논을 하면서 가장 좋은 방법을 찾아가는 것이다. 이 세상의 어느 누구도 Review가 필요 없을 만큼 모든 것을 다 잘 알지는 못한다.[6]

동료검토는 좋은 소프트웨어를 만들기 위한 한 방법이기도 하지만, 한 회사의 개발문화로서 꼭 정착이 되어야 할 방법론이다. 동료검토가 정착된 개발 회사는 이러한 장점을 가지게 된다.

- 좋은 소프트웨어를 개발한다.
- 개발 정보가 여러 사람에게 공유가 되어서 개발자 퇴사 시나 공석 시도 대처가 가능하다.
- 제품의 유지보수와 업그레이드에 비용이 적게 든다.
- 후배 엔지니어가 동료검토를 통해서 기존의 제품을 빨리 이해하고, 회사의 기술을 습득한다.
- 상호 동료검토를 통해서 개발자들의 수준이 전체적으로 올라간다.

동료검토는 수많은 장점을 가지고 있는 꼭 필요한 절차임에도 불구하고, 회사 내에서 정착시키기가 정말 어렵다. 귀찮기도 하고, 개발 스케쥴에 쫓기면 시간을 낼 수 없기 때문이다. 회사에서 이를 강제화 하고, 개발 스케쥴이 이미 동료검토 시간을 충분히 포함해야 할 것이다. 그렇지 않으면 동료검토를 생략함으로서 절약한 시간과 비용의 몇 배를 나중에 지불해야 한다.

7. 공개 소프트웨어 안전하게 사용하기

공개 소프트웨어를 사용하다가 보안 사고가 발생하면 누가 책임을 지게 될까? 대부분의 경우는 아무도 책임지지 않는다. 이는 상업용 소프트웨어를 쓰는 경우도 마찬가지이다. 1.25대란으로 국내에서만 수천억의 피해가 발생했지만 그 책임이 누구에게 있냐를 따지는 일은 대단히 어려운 일이다. 항상 그렇듯이 1차적인 책임은 웜 제작자에게 있는 것은 당연하다, 그렇지만 취약점을 만들어낸 마이크로소프트사에는 어느 정도의 책임이 있을까? 어쨌든 보안사고 시 누구에게 책임을 묻는 일은 쉽지 않다.

이러한 보안 사고를 미리 방지하기 위해서는 가장 중요한 것은 "보안 의식"이다다. 우선, 보안은 특정 보안담당자의 혼자만의 몫은 아니다. 모든 사람들의 적극적인 참여가 필요하다. 개인들은 항상 보안을 생활 습관화 하는 자세로 변화가 필요하고, 조직에서는 리더의 적극적인 관심이 필수이다. 이러한 보안의식이 널리 파급되어야 또 다른 대란을 미리 막을 수 있을 것이다. 각 개인이 할 수 있는 구체적인 방안으로는 많은 사람들에 의해서 검증된 공개소프트웨어를 사용하고, 항상 패치 등의 정보에 귀를 기울이며, 그 외에도 꼭 필요한

보안 제품들을 사용하는 것이다. 기업에서는 공개소프트웨어를 사용하면서 절약되는 비용의 일부를 "보안 제품" 구매에 투자를 하는 것도 좋을 것이다. 보안에 관심이 있다면, 백신만으로는 "보안"이 끝나는 것이 아니라는 것을 잘 알고 있을 것으로 생각이 된다. Desktop용 백신, 관리 솔루션, 방화벽, 개인 방화벽, IPS, Anti-Spam 솔루션 등 한 두개가 아니지만, 기업의 규모와 성격에 맞게 안전한 수준으로 구축을 하면 되겠다.

필자가 근무하고 있는 안철수연구소에서는 향후 Linux의 Desktop 사용이 늘 경우에 대비하여 Linux용 Desktop 백신 제품의 개발에 대해서 연구를 하고 있다. 가까운 미래에 많은 사용자가 Linux를 책상 위에 올려 사용하게 될 경우 이 제품을 볼 수 있을 것으로 기대한다. Windows는 엄청나게 많은 취약점이 있는 것처럼 보인다. 취약점도 많이 공개가 되고, 패치도 많이 나오므로 하지만 꼭 그런 것은 아니다. Windows는 많은 사람이 쓰기 때문에 관심도 많고, 결정적으로 돈이 되기 때문에 많은 공격자의 공격을 받고 있는 것이다. MS는 MS제품의 보안성을 확보하기 위해서 막대한 자금을 사용하고 있다. MS제품뿐만 아니라 Third party 제품도 소스코드 리뷰를 통해서 취약점을 찾는 노력을 기울이고 있다. 하지만 Linux와 OpenSource 제품이라면 얘기가 약간 달라진다.

OpenSource 제품의 취약점을 찾는 의무를 누구에게 지울 수 없기 때문이다. 어떻게 보면, 제작자와 사용자 모두의 책임처럼 보인다.m이러한 것이 기업에서 OpenSource 제품을 사용하는데, 꺼려지는 요소로 작용한다.

공개 소프트웨어에서는 MS처럼 역할을 해주는 것은 공공의 역할이라고 생각이 된다. 즉, 정부에서 담당을 해야 할 것 같다. 많이 사용되는 공개 소프트웨어의 취약점을 분석해서 발견되는 취약점은 제작자에게 전달을 하여 수정토록 하고, 일정 요구수준에 이르면 인증을 주는 방법도 한가지일 것으로 생각된다.

이처럼, 공개소프트웨어 확산을 위해서 정부가 할 수 있는 일을 몇 가지만 보자.

- 공개소프트웨어를 분석하여 취약점을 개발자에게 알려주기
- 안전하게 사용할 수 있는 공개소프트웨어를 찾아서 알려주기
- 공개소프트웨어를 개발하는 개발자에게 물질적으로 지원해주기
- 기업 및 정부에서 공개소프트웨어를 안전하게 사용할 수 있도록 정책을 펼치기

공개소프트웨어 사용의 확대는 "국가의 경쟁력"과도 어느 정도 연관이 있다. 특정 기업의 S/W가 한 국가의 소프트웨어 인프라를 독점하는 정도로 막강한 영향력을 발휘한다면 비용적인 부담뿐만 아니라, 국가의 안보에 영향을 줄 수 있다. 따라서 공개소프트웨어를 사용자들이 안전하게 사용하게 하는 일도 정부가 할 수 있는 일이다. 개발자, 사용자(개인, 기업) 그리고 정부에서 이 같은 노력을 기울이면, 공개소프트웨어를 더 안전하게 사용할 수 있을 것으로 기대한다.

8. 결 론

지금까지 공개소프트웨어가 얼마나 안전한지 알아보았다. 한마디로 단정 지어서 말할 수는 없지만 결론을 한마디로 요약하면 다음처럼 말할 수 있다.

"공개소프트웨어는 비공개소프트웨어보다 훨씬 더 안전한 방법으로 만들어진다."

설계, 소스코드 등 개발에 관련된 모든 내용이 공개되는 것이 소프트웨어를 가장 안전하게 만들 수 있는 방법이라는 것이 이 글의 결론이다.

공개소프트웨어의 가장 큰 특징은 수많은 사람이 Review를 한다는 것이고, 이를 통해서 견고하고, 안전한 구조로 설계가 되면, 더 안전한 소프트웨어로 개발이 된다.

비공개소프트웨어의 감춰진 아키텍처와 백도어 등의 위협은 지금도 여전히 사용자를 안심시키지 못하고 있다.

그럼에도 여전히 공개소프트웨어는 예상 외로 사용이 저조하다. 물론 과거에 비하여 비약적으로 발전을 하고는 있지만, 넘어야 할 산이 매우 많다. 과거부터 비공개 상용제품이 익숙해온 사용자들은 여전히 비공개소프트웨어의 사용을 꺼리고 있다.

세계 각국은 이러한 맥락에서 공개소프트웨어의 개발 및 사용을 정책적으로 적극 지원하고 있다. 특히 독일, 프랑스, 영국 등의 유럽과 중국, 남미 등에서 두드러지게 지원이 되고 있다. 물론, 우리나라도 이와 발을 맞춰 공개소프트웨어를 권장하는 쪽으로 정책을 추진중이다. 향후 국가 경쟁력 및 안보에도 연관이 되어 있는 중요한 사안이라서 공개소프트웨어의 사용 확대는 꼭 필요한 정책이 아닐 수 없다.

비공개소프트웨어의 반대하는 입장에서 말하는 것은 아니다. 비공개 상용소프트웨어도 앞으로 계속 쓰일 것이고, 공개소프트웨어와 상호 도움이 되는 형태의 발전이 가장 바람직해 보인다.

앞으로 공개소프트웨어는 더욱 많이 사용될 것이고,

더욱 안전하고 좋은 공개소프트웨어를 사용하기 위해서 정부를 비롯하여 이를 사용하는 모든 개인들의 동참하는 자세의 적극적인 노력이 필요하다. 공개소프트웨어는 소수의 개발자가 모두 만들어 가는 것이 아니고, 개발자 사용자 모두 같이 만들어 가는 것이다. 이러한 적극적인 참여와 관심이 현재 공개소프트웨어의 확산을 가로막는 수많은 걸림돌을 제거하게 될 것이다.

좋은 공개소프트웨어가 더 많이 사용되는 날을 기다리며, 글을 마친다.

참고문헌

[1] Joel Spousy, "조엘 온 소프트웨어", p.375, 에이콘출판, 2005

[2] Jean Bozman, Al Gillen, Charles Kolodgy, Dan Kusnetzky, Randy Perry, and David Shiang, "Windows 2000 Versus Linux in Enterprise Computing", p.11, IDC, October 2002.

[3] 박혁진, "오픈소스OS 도입확산과 정보보호", p.4, p.6, 리눅스코리아, 2004.

[4] Internet Security Systems Protection Advisory, "Sendmail Remote Signal Handling Vulnerability", http://xforce.iss.net/xforce/alerts/id/216, 22 March, 2006

[5] Natalie Whitlock, "Does open source mean an open door?", http://www-128.ibm.com/developerworks/library/l-oss.html, 01 March, 2001

[6] 김익환, "Peer Review의 중요성", 2006

전 규 현

1988. 3~1995. 2 연세대학교 화학공학과
1994. 5~1995. 1 한글과컴퓨터
1995. 2~1998. 1 아이소프트
1998. 2~1999. 6 컨케이터스코리아
1999. 7~2002. 9 아이메이트닷넷
2002. 10~2003. 2 한글로닷컴
2003.06~현재 안철수연구소
관심분야 : 네트워크보안장비, 웹보안, 메일보안, 스팸메일, 바이러스/웜
E-mail : gracegyu@ahnlab.com