

부요 리눅스 상에서 패치 업데이트를 위한 GUI 프로그램 구현

오정현*, 장혜영*, 최종천*, 조성제*
*단국대학교

warly0833@hotmail.com^o, {godofslp, chenil}@dankook.ac.kr, sjicho@dankook.ac.kr

Implementation of a GUI based Patch Management System under Booyo Linux

JungHunin Oh^o, Hye-Young Chang*, JongChun Choi*, Seongje Cho*
*Dankook University.

요 약

본 논문에서는 리눅스 환경하에서 원격 업데이트 서버에 있는 LSB(Linux Standard Base)를 기반으로 국내 표준 리눅스 플랫폼으로 개발된 부요(Booyo)리눅스에 RPM 패키지 또는 보안 패치들을 데스크 탑에 자동으로 설치하게 해주는 클라이언트용 자동 업데이트 프로그램으로 아버지를 구현하였다. 아버지는 부요 환경에서 SW패치 업데이트를 지원하는 한국형 GUI 프로그램으로 클라이언트가 패키지 관리 또는 보안 패치 작업을 쉽게 수행할 수 있도록 한다.

1. 서론

모든 소프트웨어에는 잘 모르는 수많은 오류/버그가 있다. 물론 오류/버그가 제작 당시에 발견만 된다면 보안이 되어 출시되었지만 일반적으로 "완벽한 소프트웨어는 존재 할 수 없다." 라는 말이 있듯이 소프트웨어는 지속적인 발전 및 안정화를 요구한다. 공개 소프트웨어는 온라인상에서 수많은 사람들이 커뮤니티를 통하여 하나의 프로젝트를 수행함으로써 비롯되는 프로그램 일관성(변조된 코드/데이터 구분)의 문제 및 공개 소프트웨어란 이름 자체가 의미 하듯이 소스가 오픈 되어 있다는 것이 빠른 안정화를 가져 올 수 있다는 장점에 반해 보안에 취약할 수 있다는 단점을 내포 하고 있다. 리눅스 배포판 또한 예외는 아니어서 오류/버그를 안고 출시된다. 이 중에는 그리 심각하지 않은 것도 있으나, 바이러스나 크래킹과 관련된 보안상 심각한 문제가 있는 오류/버그들도 포함되어 있을 수 있다. 또한, 이러한 오류/버그는 항상 최신의 패키지 및 보안등의 패치가 적용이 되어 있을 경우 안전한 동작환경을 갖출 수 있다.

본 논문에서는 Linux 국내 표준 플랫폼기반으로 개발된 부요(Booyo)에서 자동화된 공개 SW 패치 업데이트 GUI 프로그램 아버지(Abiji)를 구현하여 컴퓨터에 설치되어 있지 않은 보안패치 및 낮은 버전의 응용 프로그램을 자동으로 검색해주고 쉽게 설치를 도와주며, 최신 보안패치 및 최신 프로그램을 바로 얻어 시스템 크래쉬 및 크래킹을 원천적으로 막아낼 수 있게 되고 지속적으로 증가하는 크래킹에도 효과적으로 대처 할 수 있도록 한다.

2. 관련 연구

현재 리눅스 패키지들은 Redhat Package Manager(RPM)을 기반으로 하고 있다[5]. RPM은 레드햇 리눅스 상에서 소프트웨어의 설치, 추가, 삭제 등이 용이하도록 제공하는 패키지 관리 툴을 말한다.

RPM은 현재 가장 일반적인 패키지 매니저이며, 컴파일된 상태의 소프트웨어를 배포하면 사용자들은 간단한 명령어로 소프트웨어를 설치할 수 있다. 일반적으로 RPM 패키지파일은 다음과 같이 이루어져 있다.

[패키지 이름]-[버전]-[릴리스 번호].[아키텍처].rpm
RPM의 파일 설치시 rpm -Uvh filename.rpm 와 같은 옵션으로 처리되어 진다. RPM기반의 시스템을 위한 자동 업데이트 및 패키지 설치/삭제도구에는 YUM(Yellowdog Updater, Modified) 과 APT가 있다.

YUM은 RPM시스템을 사용하는 운영체제가 탑재된 워크스테이션 네트워크를 위한 패키지 관리 자동화 도구이며, 서버의 저장소를 구성하는데 사용된다. 일반적인 용도의 클라이언트도 저장소가 준비되면 어떤 클라이언트도 저장소의 RPM 기반의 패키지를 설치, 업데이트, 또는 제거할 수 있다. YUM 을 이용한 패키지 설치시 yum install [패키지 이름] 삭제는 yum remove [패키지 이름] 업데이트는 yum update [패키지 이름]이 있다. YUM이 제공하는 기능을 GUI로 보여주는 interface의 역할을 하는 프로그램인 YUMI(Yellowdog Updater, Modified Interface)가 있다[3]. YUMI는 Linux기반 기존의 패치 및 업데이트 프로그램을 이용한 GUI를 통한 자동업데이트 프로그램으로 YUMI 제공하는 기능을 GUI로 보여주는 interface의 역할을 하는 프로그램이다.

APT는 YUM과 같이 패키지나 패치를 관리하기 위한 데비안 도구이며, APT가 제공하는 기능을 GUI로 보여주는 interface의 역할을 하는 SYNAPTIC이 있다.

본 연구에서는 Linux 국내 표준 플랫폼기반으로 개발된 부요에 적합한 패치 업데이트를 위한 한국형 GUI에 대한 연구를 진행 하여, 그 결과물로 리눅스 최신 버전의 프로그램을 유지하고, 보안 / 프로그램 버그 패치를 사용자 편의성을 고려한 GUI 기반 패치 업데이트 프로그램인 아버지를 개발하였다.

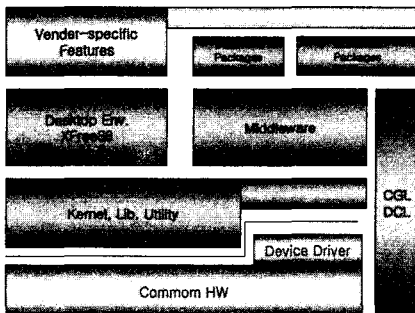
아버지는 리눅스 데스크탑에서 패키지를 최신버전으로 유지하고, 보안이 요구되는 긴급한 상황을 대처하기 위하여 GUI 기

반 패치 업데이트 프로그램이다. 이러한 리눅스 기반의 GUI(Graphic User Interface) 환경에서 개발하기 위해서는 GTK(General Image Manipulation Program Toolkit)가 필요하다. GTK란 비상업적인 공개 소프트웨어 개발을 위한 그래픽 유저 인터페이스를 제공하는 라이브러리로서 엑스윈도우(Xwindows)기반의 하위 레벨의 함수와 윈도우에 관련된 함수들을 제공하는 GDK(GIMP Drawing Kit)와 기존의 표준을 바탕으로 새로운 기능을 추가하고 리스트 구조와 같은 자료 구조를 추가하여 다양한 기능을 사용할 수 있도록 고안한 Glib로 구성되어 있다. GUI 개발언어로 파이썬과 오픈 소스 운영체제에서 널리 사용되는 그래픽 툴킷 라이브러리인 PyGTK로 구현한다 [2].

3. 시스템 구성

3.1 부요

부요는 국제산업 표준을 기반으로 최적의 성능 및 기능을 발휘할 수 있도록 개발된 커널 및 패키지 SW를 적용한 공개SW이다. 부요는 연구개발과 제품개발에 안정적이고 신뢰할 수 있는 기반 소프트웨어를 활용할 수 있으며, 소프트웨어의 독점에 대한 문제점들을 해결함으로써 시스템 구축 비용 절감 효과를 기대할 수 있다. 공개 SW를 사용할 경우 다음과 같은 이점이 있다. 원천 코드의 공개로 시스템 확장 및 유지보수의 호환성 확보가 유리하며, 국가 주요 정보 시스템의 안정성 확보 가능하다. 그리고 SW 개발기간의 단축과 개발 비용을 절감할 수 있다. 다양한 환경에 따른 사용자의 혼란을 제거하고 핵심SW 기술을 확보하여 안정적인 기술지원을 통한 활성화 장애 요인을 제거하기 위해서는 국내표준 리눅스 규격이 필요했다. 따라서 국제 표준을 근간으로 하는 최적의 성능 및 기능을 발휘할 수 있는 국내 표준 컴퓨팅 환경 규격을 만들고 커널 및 핵심 기반 기술을 개발하게 된 것이다. 그래서 부요 데스크탑 표준 규격 및 표준판이 만들어졌으며, 서버 표준규격 및 표준판을 만들게 된 것이다. (그림 1)의 표준 컴퓨팅의 환경규격의 범위는 XFree86, Kernel, Lib, Utility, Middleware, Common Packages of Desktop, Common Packages of Server 영역이 된다. 아바지는 이러한 영역에 적용되는 패키지에 대한 관리를 해주게 된다.



(그림 1) 국내 표준 규격 범위

3.2 파이썬

파이썬은 '객체지향 스크립트 언어'이다. 즉, 근본적으로

객체지향 언어이면서 스크립트 언어의 유용성을 도입하여 설계한 언어로써 사용하기 쉽게 되어 있다. 그리고 이식성이 강한 ANSI C로 작성되어 오늘날 사용 중인 실제적인 모든 주요 플랫폼에서 컴파일과 실행될 수 있는 장점을 가지고 있다. 사용할 GUI 프로그래밍도 쉽게 할 수 있도록 Tkinter라는 GUI API로 표준 객체지향 인터페이스를 지원하며, 수치연산 프로그래밍, 데이터베이스 프로그래밍, 이미지 처리, 인공지능, 분산 객체 등 다양한 영역에서 활용되고 있다. 파이썬은 C언어를 기본으로 하고 있기 때문에 문법이 거의 같고 다음의 내용을 기본으로 구현한다. 파이썬은 문자열, 리스트, 사전, 튜플, 파일과 같은 자료형이 내장되어 있어 프로그램 작성을 쉽게 한다. Pygtk는 파이썬이라는 프로그래밍 언어에서 이러한 GUI 개발 환경을 제공하기 위해 GTK를 기반으로 하여 파이썬의 특징에 적합하도록 클래스화하고 래퍼(wrapper) 함수들을 모아 놓은 API(Application Programming Interface)인 PyGTK를 지원하고 있다.

PyGTK는 파이썬 프로그래밍 언어의 프로그래밍 기법을 사용하여 윈도우를 생성하고 필요에 따라 다양한 컨트롤들을 생성하여 사용자로부터 특정 이벤트를 입력 받았을 때 그에 따른 작업을 수행하도록 프로그래밍 할 수 있다. PyGTK는 이벤트 처리 기반(Event driven)이라는 것과 컨테이너(Container)와 박스(Box)에 패키징(Packaging)을 통해 컨트롤의 위치와 크기를 조절한다는 것, 크게 두 가지 특징을 가지고 있다. 이벤트 처리 기반이란 gtk.main()은 초기화된 윈도우와 윈도우 내에 필요한 위젯(Widget)을 생성하고 무한 루프를 돌면서 사용자의 이벤트가 발생하기를 기다린다. 여기서 위젯(Widget)이란 PyGTK에서 사용하는 버튼, 레이블, 텍스트 박스, 스크롤 등과 같은 컨트롤들을 의미한다. 해당 위젯에서 이벤트가 발생하면 시그널이 발생하고 이 시그널은 위젯을 생성할 때 정의해 놓았던 이벤트 핸들러(Event handler)를 호출한다.

3.3 국제화

일반적으로 프로그램은 영문으로 쓰이고 기록된다. 따라서 대부분의 프로그램은 사용자에게 영문으로 사용자 인터페이스를 제공한다. 대부분의 오픈 소스 GNU 프로그램이 이러한 방식을 따른다. 한가지 언어(영어)를 사용하는 것은 전체의 개발자, 관리자들을 연결시켜 커뮤니케이션을 가능하게 하는 이점이 있지만, 대부분의 영어에 익숙하지 않은 사용자들은 이러한 프로그램을 사용하는 것이 익숙하지 않을 것이다. 반면, 사용자들은 자신의 언어로 실행되는 프로그램을 사용하는 것이 편리할 것이다. 이러한 사용자의 필요를 충족시키기 위하여 국제화 프로그램 도구로서 Gettext는 사용자의 locale에 따라 특정 message output에 대한 다국어 처리를 할 수 있는 GNU 유틸리티이다. Gettext는 다른 GNU 패키지가 다국어를 지원할 수 있는 프레임워크를 제공하는 툴의 집합이다.

4. 비교분석

[표 1] 아바지와 YUMI 비교

SYNAPTIC		YUMI		아바지	
기능	성능	기능	성능	기능	성능
*Reload	*편리한 패키지들을 cdrom에서 추출	*설치	*쉬운인터페이스	*Update	*한글지원
*Upgrade	*패키지들을 쉽게 구분하여 관리	*제거	*패키지관리 편리	*Install	*각 패키지들간의 의존성검사
*적용		*업데이트		*Removes	
*등록정보		*탐색		*Output	*지정소(서버)를 변경

[표 1]에서 보여주는 것과 같이 아비지는 YUMI에 비하여 사용법이 간단하고 설정이 필요 없이 패키지들을 분류된 상태에서 설치, 제거, 업데이트, 탐색을 적용할 수 있다. 아비지는 각 패키지들에 대한 의존성과 최신버전의 의존성까지 나와있어 오류가 날 수 있는 의존성 부분을 해결하기 용이하며, 클라이언트가 원하는 저장소(서버)를 변경할 수 있다. 하지만 SYNAPTIC과 비교하였을 경우 부족한 기능을 가지고 있음을 알 수 있다. 이는 구현상의 용이함을 위해 아비지가 YUMI를 기반으로 하여 구현되었기 때문에 한국형 GUI에 부족함이 없는 기능을 위하여 향후 보다 향상된 기능에 대한 연구 및 개발이 요구된다.

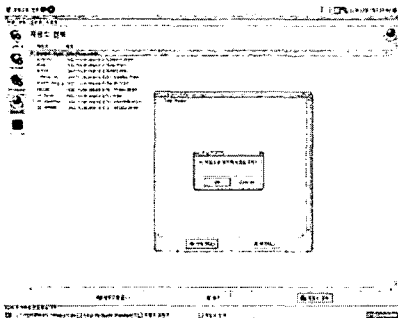
5. 구현내용

5.1 개발 환경 및 도구

서버 3대와 클라이언트 10대를 사용했으면 개발 소프트웨어로 운영체제 부요 32,64 bit와 vi, rpm, 파이썬, pygtk, 국제화 다국어 지원도구인 gettext, 각 파일들의 버전관리를 할 수 있도록 도와주는 툴인 CVS를 사용하였다.

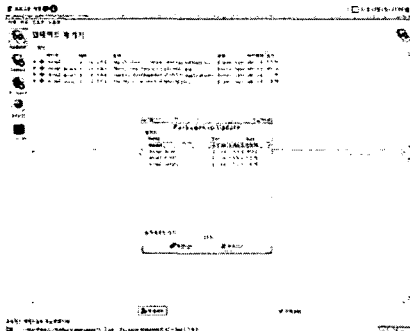
5.2 주요 기능

아비지에서는 [저장소 추가]가 있어 (그림 3)처럼 패키지가 저장되어 있는 서버의 설정파일을 추가하며 메뉴 중 좌측 중단에 Reops 아이콘인 해당 저장소 즉, 패키지가 저장되어 있는 서버의 설정 파일을 추가, 새로 고침을 할 수 있다



(그림 3)저장소 추가

(그림 4)는 해당 패키지를 선택한 후 패키지 다운로드를 하고 업데이트를 한다.



(그림 4) 패키지 업데이트

업데이트가 완료했을 때 [확인]버튼을 누르면 완료된다. 아비지는 시스템에 [제거] 할 전체 패키지 리스트를 나타내 제거할 수 있다. 패키지 선택 후 제거버튼을 누르면 다시 확인을 한 후 제거된다. [패키지 설치]는 설치할 패키지 리스트가 나오고 패키지 선택한 후 정말 선택할 것인지 재확인을 묻는다. [패키지 설치 확인]은 선택한 패키지를 정말 설치할 것인지 사용자에게 다시 확인하도록 한 후 해당 패키지들을 서버에서 다운로드 한 뒤에 패키지를 설치한다.

6. 결론 및 향후 과제

최근에는 패치 설치의 중요성을 깨닫고 적극적으로 패치 관리를 하려는 사용자가 증가하고 있지만 보안 취약점 보고가 해마다 크게 증가함에 따라 그 취약점을 보완하기 위한 보안패치 역시 증가하고 있다. 이러한 보안 패치나 최신 패키지들을 수동적으로 관리하는 것에 많은 비용과 시간이 요구되며, 수동적으로 관리하는 경우 관리 대상의 시스템들의 패치 설치 현황을 파악할 수 없기 때문에 효과적인 패치 관리가 이루어지도록 자동화된 패치 업데이트 GUI 프로그램이 필요로 한다. 본 연구에서는 리눅스 업데이트 서버에서 관리되고 있는 최신 패키지나 보안 패치들을 데스크 탑에 자동으로 설치해 주는 클라이언트용 자동 업데이트 프로그램 아비지를 국산 표준 리눅스 플랫폼인 부요상에서 한국형 GUI로 개발하였다.

아비지 프로그램은 지속적인 보안 / 프로그램 버그 패치 및 최신 버전의 응용 프로그램을 관리함으로써 리눅스를 사용하는 기업 및 일반 사용자들에 있어 보다 안전하고 편리하게 시스템을 유지·관리 할 수 있게 해준다. 이로써 보안적인 측면과 프로그램 버그로 인한 오 동작 및 취약점 공격의 재정적인 손실을 방지할 수 있다. 이러한 업데이트 프로그램 기반으로 보다 안정적인 리눅스 시스템을 구축할 수 있어 리눅스 확산에 기여를 할 것으로 기대되어 산업적·사회적 측면에서도 그 의미가 크다고 할 수 있다.

이러한 부요 데스크탑에서 패키지 패치 업데이트에 대한 향후 과제로서 SYNAPTIC과 같은 향상된 기능에 대한 보완은 물론, 보안에 따른 장치 드라이버 설치에 대하여 자동적으로 의존성을 관리하고, 패치 상황 정보(패치 버전 비교 정보 포함) 및 패치 파일의 이력을 통한 통합적인 관리와 같은 기능에 대한 연구가 필요하다.

참고문헌

- [1] M. Gagne, "Linux System Administration", Addison-Wesley
- [2] 마크 루츠 데이비스 애서 저 이상성 역 "파이썬 시작하기" 한빛미디어 1999
- [3] Cobind, "YumGUI 소스코드 및 관련 자료 모음" <http://cobind.com/yumgui.html>
- [4] gettextGNUproject "howto gettext", <http://www.gnu.org/software/gettext/>
- [5] Eric Foster-Johnson "Red Hat RPM Guide" redhat press 2003
- [6] W. Richard Stevens, "UNIX Network Programming", Prentice Hall PTR
- [7] www.software.or.kr
- [8] John Finlay, PyGTK 2.0 Tutorial, 2005/4/13
- [9] PYGTK, "Pygtk 관련 자료 모음", <http://www.pygtk.org>
- [10] Python Programming Language, Python document 2.4.
- [11] IBM, RPM, <http://www-128.ibm.com/developerworks/linux/library/l-rpm1/index.html>