

LSB 인증대상 발굴보고서 (GINUX 2.2 · SULinux 2.0)

공개SW 역량프라자 운영 사업

2009. 12. 15

주관연구기관 **정보통신산업진흥원**

지 식 경 제 부

목 차

1. 개요	1
가. 목적	1
나. 절차	1
2. LSB 인증	2
가. LSB 인증 개요	2
나. LSB 인증 효과	2
다. LSB 인증 구성요소	3
라. LSB Test Tools	4
마. LSB 인증 프로세스	6
4. LSB 인증 지원 대상	8
가. 선정 기준	8
나. 대상 솔루션 선정	8
다. LSB 인증 테스트 결과	17
라. 인증 지원 로드맵	26
5. 맺음말	27
6. 별첨	28

1. 개요

가. 목적

- ☐ 국내 공개SW의 경쟁력 강화
- ☐ 공개SW의 호환성을 기반으로 SW 개발 시 TCO 절감효과
- ☐ 공개SW의 규격 표준화를 통한 이식성 및 상호 운용성 확대
- ☐ 안정성과 신뢰성 있는 공개SW 활성화에 기여

나. 절차

- ☐ LSB 인증 개요
- ☐ LSB 인증 효과 정의
- ☐ LSB 인증 구성 요소
- ☐ LSB 인증 지원 대상 선정 기준
- ☐ LSB 인증 지원 대상 솔루션 선정 기준
- ☐ LSB 인증 테스트

2. LSB 인증

가. LSB 인증 개요

LSB 인증은 배포판(DTK: Distribution Test Kit)과 애플리케이션(ACK: Application ChecKer)의 두 가지 인증을 제공하고 있다. LSB 인증을 취득한 제품만이 LSB 인증 Trademark를 사용할 수 있다. 이 마크는 개발자와 사용자가 LSB 인증된 배포판에서 LSB 인증된 애플리케이션을 사용할 경우 문제가 없다는 것을 보증한다.

LSB(Linux Standard Base)는 수많은 리눅스 배포판들이 각각의 기능과 환경을 적용하여 개발됨에 따라 각 배포판 및 애플리케이션간 의존성 결여 등 많은 문제점이 나타나 공개SW 활성화의 저해요인으로 부각되자 이러한 저해요소를 해결하기 위한 방안으로 만들어 졌다.

- ☐ 1998년 5월경 Linux Torvalds, 하드웨어 및 상용 소프트웨어 업체, Linux International, 그리고 FreeBSD의 Jordan Hubbard가 지원하여 프로젝트로 구성
- ☐ Linux Foundation¹⁾의 WorkGroup으로 활동
- ☐ LSB Specification²⁾ 1.0을 2001년 6월 발표를 시작으로 현재 4.0까지 발표 됨

나. LSB 인증 효과

LSB는 독립 소프트웨어 벤더(ISV : Independent Software Vendor)와 개별 애플리케이션 개발자들만을 위하여 만들어진 솔루션이 아니며 리눅스 전체 환경을 위한 표준화 정책이다.

LSB의 인증 효과는 다음과 같다.

- ☐ ISV가 지원하는 리눅스의 라이브러리 버전, 파일의 위치, 파일시스템 구조 등의 복잡한 문제와 여러 배포판별로 어플리케이션을 추가로 개발해야 하는 비용 감소
- ☐ 하드웨어와 소프트웨어의 다른 특성으로 발생하는 소프트웨어 개발과 설치(Poting)의 복잡성을 감소시켜 Vendor들의 협력을 유도

1) Linux Foundation: 프리 스탠다드 그룹(FSG)과 오픈소스개발연구소(OSDL)가 2007년 1월 22일에 합병하여 탄생한 조직으로 분산된 리눅스 진영의 지원조직을 단일화 하고 리눅스 진영을 대표하고 있다.

2) Linux Specification: LSB를 준수하는 시스템에서 요구되는 구성 요소(Module)를 정의하는 명세서

- 저렴한 개발 비용과 감소된 지원으로 추가적인 리눅스 배포판 지원
- Linux Foundation과 Linux Developer Network(LDN)은 ISV의 개발 프로세스와 효율적 마케팅을 보다 쉽게 지원

다. LSB 인증 구성 요소

- LSB는 리눅스 시스템과 어플리케이션 간 상호운용성을 촉진하는 리눅스 운영체제를 위한 핵심 표준으로서 테스트를 목적으로 한 샘플 도구, 배포판과 어플리케이션에서 작성된 표준을 위한 테스트 도구 설정, 바이너리로 작성된 인터페이스 명세서를 포함하고 있다.
- LSB를 구성하는 핵심은 리눅스 어플리케이션 표준을 개발하는 방법과 어플리케이션 설정을 알려주는 바이너리로 작성된 인터페이스 명세서(Specification)이다.
- LSB 명세서 구성 요소
 - 공통 패키지와 설치 가이드
 - 공통 공유 라이브러리와 선정
 - 설정 파일
 - 파일 배치
 - 시스템 명령
 - 시스템 인터페이스를 위한 어플리케이션 바이너리 인터페이스(플랫폼과 어플리케이션 레벨 모두 해당)
- LSB 지원 플랫폼
 - Intel IA32
 - Intel IA64
 - x86-64/EM64T
 - IBM PPC 32
 - IBM PPC 64
 - IBM 31-bit S/390
 - IBM 64-bit zSeries

라. LSB Test Tools

LSB는 Linux Developer Network(LDN)에서 모든 정보를 얻을 수 있다. LSB가 오늘날의 독립 소프트웨어 벤더(ISV)에게 조금 더 쉬운 개발 환경을 제공한다. 어플리케이션 개발자들을 수용하기 위한 기본적인 방식은 다음과 같다.

☐ 최대한 많은 어플리케이션의 설치 구성

- 1) 어플리케이션의 설치구성(Porting) 여부를 LDN에서 제공하는 리눅스 Application Checker를 사용하여 확인
- 2) 설치 구성(Poting)을 방해하는 어플리케이션 주요 요소를 확인
- 3) 개발자가 더 많은 어플리케이션을 설치구성(Porting)하여 변경을 적용할 수 있도록 함
- 4) 리눅스 Application Checker는 어플리케이션 개발을 위한 최고실습의 개발자 가이드 경로를 추천

☐ Linux Developer Network Resource

☐ Linux Application Checker

어플리케이션 포팅을 점검하고 더 많은 포팅을 만들 수 있도록 권장사항을 확인

☐ LSB Database Navigator

포팅 및 이동에 필요한 솔루션 및 대안을 배울 수 있는 구성요소를 식별하지 못할 수 있으므로 데이터베이스 네비게이터는 C/C++ 프로그래머가 볼 수 있는 리눅스 프로그램의 풍부한 정보를 제공

☐ LSB Build Tools

LSB SDK는 개발자가 LSB를 준수하기 위해 어플리케이션을 빌드하는 동안 API를 사용하여 확인하고 바이너리와 RPM 패키지를 검증할 수 있도록 적합성을 보장

☐ LSB Sample Implementation

LSB Sample Implementation (LSB-si)은 최소한의 테스트를 목적으로 한 LSB를 따르는 실행 환경을 사용하며, LSB를 준수하는 어플리케이션은 변화가 많은 특정 배포판에서 잘 동작되는지 테스트를 수행 함. LSB 인증 프로그램은 LSB-si 기반에서 테스트 한 어플리케이션을 요구함

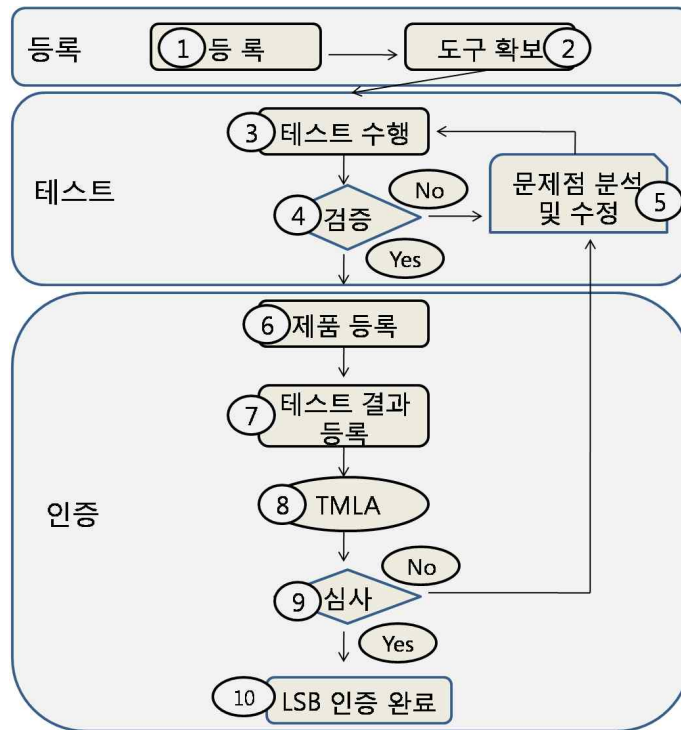
☐ Tutorials and Blogs

일반적인 리눅스 어플리케이션 개발의 최신 방법과 LSB 준수사항, 어플리케이션 포팅을 위한 정보와 최신 관련 문서를 제공

- Forums and Mailing Lists
많은 개발상의 문제점과 해결점 및 각종 팁을 확인
- LSB Certification and General Marketing Support
어떤 경로를 선택하더라도, LSB나 포팅, 리소스는 리눅스 환경에서 개발자가 어플리케이션의 홍보를 최대한 결정하도록 하며, 인증된 어플리케이션은 제품 디렉터리에 포함 가능
- LSB Test Tools
 - Linux Application Checker
LDN(Linux Developer Network)이 제공하고 있는 Linux上에서 동작하는 어플리케이션의 호환성 및 LSB 표준 근접성을 확인하는 테스트 도구로서 테스트뿐만 아니라 실제로 '제품등록/테스트결과 업로드' 등의 인증절차를 처리할 수 있다.
 - LSB Distribution Testkit (DTK)
리눅스 디스트리뷰션이 LSB에 준거하고 있는지를 테스트하는 도구이다.
 - LSB Software Development Kit (SDK)
LSB SDK를 사용하여 바이너리와 RPM 패키지가 LSB에 준거하고 있는지를 검증하고 빌드실행 중에 어플리케이션에 따른 API의 사용 상황을 모니터 하여 LSB 적합성을 보증할 수 있다.
LSB SDK를 사용하여 개발된 어플리케이션은 모든 LSB 준거 디스트리뷰션에서 동작 가능하게 된다.
 - Sample Implementation Tools (LSB SI)
LSB 디스트리뷰션의 핵심이 되는 최소한의 라이브러리와 인터페이스를 구현한 어플리케이션의 테스트를 목적으로 한 LSB 기반 실행 환경을 구성한다.

마. LSB 인증 프로세스

LSB 인증 절차는 Linux Foundation 웹페이지에서 회원가입 후 등록, 테스트, 인증 순서로 프로세스가 진행된다. 아래의 그림을 통하여 인증 프로세스를 한눈에 볼 수 있다.



[그림2-1. LSB 인증 프로세스]

1) 등록

- 인증을 취득하기 위하여 제일 먼저 Linux Foundation의 웹사이트에서 자신의 계정을 생성한다.
- 로그인 후, 사용자 정보가 맞는지 확인한다.
- 같은 회사 내에서 다른 누군가가 이미 회사명의 계정을 가지고 있는 경우는 등록된 사람에게 필요한 추가 절차를 요청한다. 그렇지 않을 경우 새로운 회사명으로 등록한다.

2) LSB 도구 확보

- 다운로드 페이지에서 플랫폼 아키텍처에 맞는 'LSB 툴/LSB 테스트 툴'을 다운로드한다.

3) 테스트 수행 및 검증

- 제품이 LSB 표준에 준거하고 있는 것을 확인하기 위해서, LSB Distribution Testkit(DTK) Manager/Linux Application Checker를 이용해 적정한 인증테스트를 실행한다.

4) 문제점 분석 및 수정

- 테스트에서 문제점이 발견되면 수정하여 다시 테스트한다.

5) 제품 등록

- 정식 인증을 받기 위해서 테스트 결과를 제출할 준비가 갖추어지면, 인증 시스템에 제품을 등록한다. 등록 절차는 온라인 (<https://www.linuxfoundation.org/lsb-cert/product.php?add>) 또는 DTK Manager/Linux Application Checker로 실시할 수 있다.

6) 테스트 결과 등록

- 테스트 결과는 제품의 등록이 완료한 다음에 업로드 한다.
- 가장 간단한 방법은 DTK Manager/Linux Application Checker를 사용하는 것이며, 인증 시스템내의 자신의 제품 페이지로부터도 업로드 할 수 있다.

7) TMAL(Trademark License Agreement)에 서명

- LSB 상표 라이선스 계약(Trademark License Agreement = TMLA)에 서명하여, 소정의 신청비용을 지불한다. (http://www.linuxfoundation.org/en/Fee_Schedule)
- TMLA는 인증 시스템내의 제품 페이지로부터 확보 가능하다.

8) 심사

- Linux Foundation에 의한 테스트 결과의 심사는 TMLA 체결과 비용의 지불이 확인된 다음에 진행된다.
(테스트 결과가 심사에 통과되지 못한 경우는 심사에서 밝혀진 문제점을 해결하고 재 테스트를 실시하여 그 결과를 다시 제출할 필요가 있다.)

9) LSB 인증 완료

- 심사에 합격한 제품은, LSB 인증 제품 리스트(register of LSB Certified products)에 기재된다(https://www.linuxfoundation.org/lsb-cert/productdir.php?by_prod).
- Linux Foundation는, 그 제품이 LSB 인증의 필요조건을 계속하여 만족하고 있는지, 간혹 관련 내용의 제출을 요구하는 일이 있다.

4. LSB 인증 지원 대상

가. 선정 기준

- ☐ 특정 SW에 의한 독과점 방지를 통한 공정한 경쟁 환경이 필요한 공개SW
- ☐ 공공기관, 교육기관, 기업 업무 환경의 전산 시스템으로 활용되는 공개SW
- ☐ 안정성과 신뢰성을 기반으로 활용도 높은 대표적인 공개SW
- ☐ 표준화를 통한 기술지원의 신뢰성을 확보하고자 하는 공개SW

나. 대상 솔루션 선정

(1) 인증 지원 대상 솔루션

[표4-1. 인증 지원 대상 솔루션]

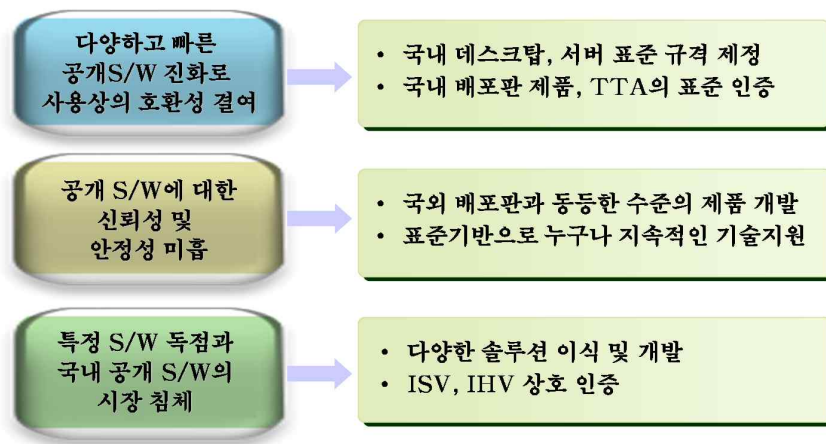
구 분	대상 솔루션
서버용 배포판	GINUX 2.2
서버용 배포판	SULinux 2.0

(2) 인증 지원 대상(OS 분야) 선정 근거

- ☐ LSB 설립 목표에 충실
 - LSB(Linux Standard Base)는 다양한 리눅스 배포판들이 각각의 기능과 환경을 적용하여 개발됨으로 인해 발생하는 어플리케이션 개발 및 활성화 저해 요소를 해결하기 위한 방안으로 설립
- ☐ 배포판과 어플리케이션의 두 가지 인증 중 배포판 인증 선 수행
 - LSB 인증의 기반이 되는 배포판 인증을 먼저 수행함으로써 향후 어플리케이션 인증 부분에 대한 기준 제시
 - 리눅스 전체 환경을 위한 표준화 정책으로 LSB 인증이 진행됨에 따라 다양한 어플리케이션에 대한 상호운용성을 위해서는 표준화된 배포판의 인증이 우선적으로 수행될 필요성 제기

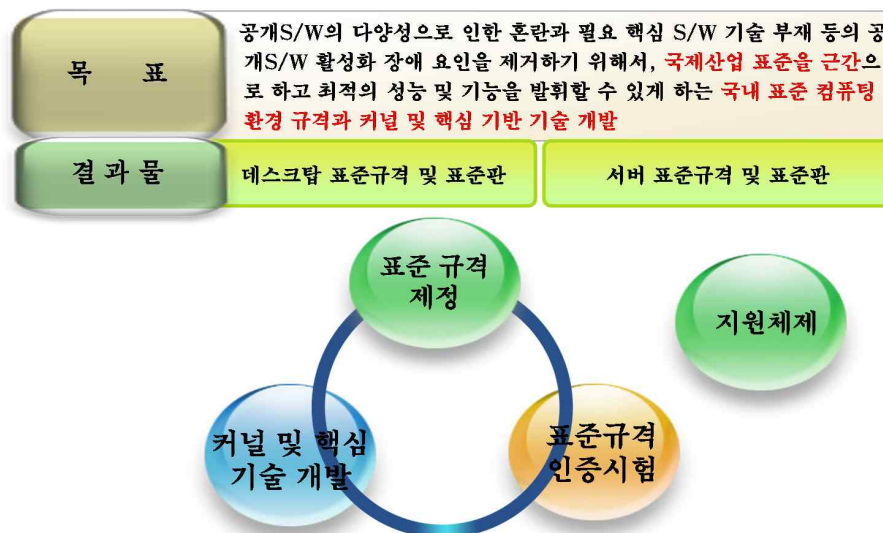
□ 국내 공개SW 기반 배포판의 표준화를 주도했던 부요의 사례

○ 공개SW 문제점 및 해결 전략



[그림4-1. 부요 기반 공개SW 문제점 및 해결 전략]

○ 부요의 목표



[그림4-2. 부요 개발 시 목표]

○ 부요의 LSB 인증 자료

Distributions:

Product	LSB Version	Architectures	Notes
SUSE Linux Enterprise 11	4.0	x86, x86-64, IA64, PPC32, PPC64, S390, S390X	
Oracle Enterprise Linux Server 5.3	4.0	x86, x86-64	The 4.0 LSB package needed for full 4.0 certification may be available separately; check the OEL yum repository or ULN (linux.oracle.com) to download and install that package.
Boss Server 2.0	3.2	x86	
Mandriva Enterprise Server 5	4.0	x86, x86-64	
Ubuntu 8.04 (hardy)	4.0	x86, x86-64	
Ubuntu 9.04 (jaunty)	4.0	x86, x86-64	
Oracle Enterprise Linux 5.1 (Carthage)	3.1	x86, x86-64	
Oracle Enterprise Linux 4 update 5	3.1	x86, x86-64	
MontaVista Linux Carrier Grade Edition 5.0	3.0	x86, x86-64, PPC32	
Bharat Operating System Solutions (BOSS) Linux 2.0	3.1	x86	
Linpus Linux 9.5	3.1	x86, x86-64	
Asianux Server 3	3.1	x86, x86-64	
Mandriva Linux Corporate Desktop 4.0	3.1	x86	
Booyo 2.5	3.1	x86, x86-64	
openSUSE 10.2	3.1	x86, x86-64, IA64, PPC32	

[그림4-3. 부요 LSB 인증]

(3) 대상 솔루션 개요

(가) GINUX 2.2

□ 개요

GINUX는 SK C&C 가 엔터프라이즈 서버용으로 개발한 리눅스 운영체제로 기업 비즈니스 제로에서 필요한 하드웨어의 성능 및 기존 소프트웨어와의 호환성과 범용성에 초점을 두어 개발되었다.

□ GINUX 2.2 중요 특징

○ 성능개선

- 개선된 프로세싱 파워
- 대용량 볼륨 활용
- 32/64비트 동시 지원 플랫폼

○ 가상 스토리지

- Distributed Parallel Fault Tolerant File System
- Locking Mechanism

- 대용량/고성능 파일처리
- 디스크 효율성 극대화

- 관리 효율성
 - 최신 패키지 유지
 - 손쉬운 패키지 업데이트
 - Bug Tracking System(BTS)
 - 효율적인 시스템 관리

- 시스템 보안
 - 강력한 암호화 및 인증 서비스
 - 한정적인 시스템 허가(SELinux)
 - 표준 보안 프로토콜

- 서비스 이중화
 - 정교한 서비스 자원 모니터링
 - 손쉽고 편리한 서비스 관리
 - 멀티노드 클러스터링
 - 최적 리소스 선점 기능
 - 서비스 운용 시간 최대화
 - 서비스 유지 수준 향상

- 신뢰성과 안정성
 - 안정적인 H/W 추상화, 쓰레드 제어
 - 정교한 멀티 쓰레딩
 - 멀티 코어 최적화
 - 완벽한 장애처리
 - 디스크 효율성 극대화
 - 고성능 & 안정적인 데이터 처리
 - 완벽한 서버, 네트워크, 데이터 보호

- 호환성
 - IHV / ISV 인증을 통한 HW/SW 호환성 확보
 - LSB, LTP 를 통한 표준 준수, OS 기능 및 성능 검증
 - 표준 UNIX 적합성

□ 적용 사례

[표4-2. GINUX 도입 및 적용 사례]

번호	발주처	사업명	공급시기	비고
1	교통안전공단	철도안전 종합 정보시스템 구축사업	2007년	
2	산업자원부	G4B 시스템	2007년	
3	정보통신부	우편업무용 단말장비 도입 및 설치사업	2007년	
4	제주국제자유도시 개발센터(JDC)	JDC 공개소프트웨어 도입사업	2007년	
5	통계청	통계청 노후서버 교체사업	2007년	
6	항공우주연구원	항공우주연구원 커뮤니티 서비스 구축사업	2007년	
7	SK 주식회사	DCS 백업서버 구축	2007년	
8	SK 텔레콤	SK 텔레콤 07년 서버 대개선 사업	2007년	
9	SK 텔레콤	SK 텔레콤 대리점경영관리 시스템 구축	2007년	
10	SK 텔레콤	Mobile RFID 확산 사업	2007년	
11	가림정보기술	스토리지 서버 O/S 공급	2008년	
12	대검찰청	영상물관리 시스템 구축 사업	2008년	
13	대검찰청	영상물 관리 시스템 확대 구축 및 ITA 수립	2008년	
14	한국전자통신연구원	Glory 2차 Infra 구축	2008년	
15	SK 에너지	D-World 포탈 구축	2008년	
16	SK 텔레콤	Rapid Prototype 시스템 구축	2008년	
17	SK 텔레콤	Incubation 시스템 구축	2008년	
18	SK 텔레콤	자동화 Tool 도입	2008년	
19	SK 텔레콤	Enclean Mall 구축	2008년	
20	SK 에너지	08년 서버 대개체 사업	2008년	
21	SK 에너지	CRM 분리구축 사업	2008년	
22	한국디지털대학교	DB 서버 구축	2009년	
23	SK C&C	텔넷 접속 시스템 구축	2009년	
24	SK C&C	네트워크 환경 개선 시스템 구축	2009년	
25	SK C&C	자동화 관리 시스템 구축	2009년	
26	SK C&C	통합 모니터링 시스템 구축	2009년	
27	SK C&C	E-Project 인프라 구축	2009년	
28	SK C&C	net Flow 분석 솔루션 구축	2009년	
29	SKT	Pilot 시스템 구축	2009년	

[※ 순서는 ABC(영문), 가나다(한글)로 정렬]

(나) SUIinux 2.0

□ 개요

SUIinux는 “보안 최적화된 서버 전용 리눅스 운영체제”이다. SUIinux는 커널 2.6기반의 보안이 강화된 안정된 버전이며, 서버로서 최고의 성능을 발휘 할 수 있게 최적화되어 탄생한 서버 전용 리눅스 OS이다.

□ SULinux 주요 특징

- 서버용으로 최고 성능 발휘
- GUI와 TUI 버전으로 구분되어 배포
- 서버용으로 최적화된 리눅스 OS
 - 설치 후 서버용으로 자동 최적화
 - 서버용으로 최적화 된 패키지 구성
 - 360여개의 관리자 전용 명령어에 대한 기본 보안 설정
 - 서버 관리 전용 명령어 내장(SSU)
 - 뛰어난 안정성, 향상된 보안성
 - 커널 2.6 기반의 향상된 성능 및 안정성
 - SELinux 기술 및 자체보안강화
 - 안정성이 검증된 패키지들만 포함
 - DDOS방어에 최적화된 리눅스
- 쉽고 빠른 설치와 빠른 서비스 구현
 - 3단계로 설치 완료되며, 누구나 쉽게 설치가능
 - APM 자동 설치를 제공(APM_Install-tools)
 - yum을 통한 빠르고 지속적인 업데이트 지원
- 향상된 호환성과 쉬운 인터페이스 제공
 - 다양한 최신 RAID 컨트롤러 지원
 - 8TB 이상의 대용량 파일 시스템 제공
 - 국제 표준 언어셋(UTF-8) 기본 채택, 국제 리눅스 표준(LSB) 준수
 - 64bit 지원으로 64bit 서버 성능 향상 및 대용량 메모리 사용
 - 통합 서버 상태 상세 모니터링 지원(SSU)
 - 편리한 GUI 환경 제공(GUI 버전)
 - 편리한 TUI 환경 제공
 - 엔터프라이즈급 가상화 지원(Xen)

□ 적용 사례

- Joara : 판타지, 무협, 로맨스, SF, 게임 등 모든 장르가 있는 국내 최대 문학/소설 사이트
- BABOMall : 믿을 수 있는 자동차용품 쇼핑몰
- 리오엘리 : 화장품 전문 몰
- 바이크마트 : 오토바이에 관한 세상의 모든 지식

- Mfile : 디지털콘텐츠 거래소, 최대 지식 거래소
- Gfile : 디지털콘텐츠 전문 거래소
- 제트파일 : 우리들의 즐거운 콘텐츠 세상
- 부산대학교
- 순천대학교
- 한양대학교
- 서울대학교

(4) 대상 선정 시 고려 사항

- ☐ 국내 대표적인 기업용 공개SW 기반 리눅스 배포판은 Asianux, Glinux, SULinux 등
- ☐ Asianux는 기 LSB 인증을 받았기에 대상에서 제외

Distributions:

Product	LSB Version	Architectures	Notes
SUSE Linux Enterprise 11	4.0	x86, x86-64, IA64, PPC32, PPC64, S390, S390X	
Oracle Enterprise Linux Server 5.3	4.0	x86, x86-64	The 4.0 LSB package needed for full 4.0 certification may be available separately; check the OEL yum repository or ULN (linux.oracle.com) to download and install that package.
Boss Server 2.0	3.2	x86	
Mandriva Enterprise Server 5	4.0	x86, x86-64	
Ubuntu 8.04 (hardy)	4.0	x86, x86-64	
Ubuntu 9.04 (jaunty)	4.0	x86, x86-64	
Oracle Enterprise Linux 5.1 (Carthage)	3.1	x86, x86-64	
Oracle Enterprise Linux 4 update 5	3.1	x86, x86-64	
MontaVista Linux Carrier Grade Edition 5.0	3.0	x86, x86-64, PPC32	
Bharat Operating System Solutions (BOSS) Linux 2.0	3.1	x86	
Linpus Linux 9.5	3.1	x86, x86-64	
Asianux Server 3	3.1	x86, x86-64	
Mandriva Linux Corporate Desktop 4.0	3.1	x86	
Booyo 2.5	3.1	x86, x86-64	
openSUSE 10.2	3.1	x86, x86-64, IA64, PPC32	

[그림4-4. Asianux LSB 인증]

- ☐ GINUX 선정 기준
 - 한국형 공개SW 표준 플랫폼 BOOYO(부요) 기반
 - 부요는 LSB 3.1 certification을 받음

Distributions:

Product	LSB Version	Architectures	Notes
SUSE Linux Enterprise 11	4.0	x86, x86-64, IA64, PPC32, PPC64, S390, S390X	
Oracle Enterprise Linux Server 5.3	4.0	x86, x86-64	The 4.0 LSB package needed for full 4.0 certification may be available separately; check the OEL yum repository or ULN (linux.oracle.com) to download and install that package.
Boss Server 2.0	3.2	x86	
Mandriva Enterprise Server 5	4.0	x86, x86-64	
Ubuntu 8.04 (hardy)	4.0	x86, x86-64	
Ubuntu 9.04 (jaunty)	4.0	x86, x86-64	
Oracle Enterprise Linux 5.1 (Carthage)	3.1	x86, x86-64	
Oracle Enterprise Linux 4 update 5	3.1	x86, x86-64	
MontaVista Linux Carrier Grade Edition 5.0	3.0	x86, x86-64, PPC32	
Bharat Operating System Solutions (BOSS) Linux 2.0	3.1	x86	
Linpus Linux 9.5	3.1	x86, x86-64	
Asianux Server 3	3.1	x86, x86-64	
Mandriva Linux Corporate Desktop 4.0	3.1	x86	
Booyo 2.5	3.1	x86, x86-64	
openSUSE 10.2	3.1	x86, x86-64, IA64, PPC32	

[그림4-5. 부요 LSB 인증]

○ GS인증 획득

오픈소스 운영체제로서의 기능성과 안정성, 편의성 등에서 높은 평가를 받으며 그 우수성을 객관적으로 인정받음

○ 정보통신산업진흥원(NIPA:舊KIPA)에서 우수성 인정

2006년 9월 두 차례에 걸쳐 진행된 정보통신산업진흥원(NIPA:舊KIPA)의 기존 리눅스 운영체제와의 웹서비스 기반 성능 비교 평가에서 두 차례에 걸친 1위로 성능의 우수성을 입증 받음

○ 강력한 모니터링 기능

CPU, 네트워크, 메모리 등 리눅스 서버에 대한 자원 모니터링은 및 특정 노드(접속 포인트)자원 사용 현황 모니터링도 가능

○ 업그레이드 및 패치를 위한 자동 업데이트 기능을 기본적으로 제공

○ 주기적인 자동 백업 및 신속한 장애에 대한 복구 기능을 제공

시스템 장애 발생 시 시스템을 재구성해 서비스를 클러스터의 다른 노드로 이전해 무정지 서비스 운영을 보장

□ SULinux 선정 기준

- 성능 및 보안이 우수함
- 인텔서버에 최적화된 운영체제
드라이버 등의 설치가 간편함으로 일선에서의 사용이 용이함
- 서버용으로 최적화된 패키지를 제공
기본보안설정과 서버관리용 자체 명령어를 내장하는 등 관리의 편의성
- 가장 안정화된 커널 2.6 기반으로 제작
- Security Enhanced Linux 기술을 채택하는 등 검증된 패키지들을 도입

다. LSB 인증 테스트 결과

(1) GINUX 2.2

☐ LSB 인증 테스트 결과 개요

테스트 결과는 /var/opt/lsb/test/manager/results/amd64-sv1-001-<date><time> 디렉터리에 생성되는

- sum_rep.html
- one_line_summary.html
- test_log.html
- autotest_info
- host_info
- test_journals
- test_logs
- lsb-autotest-amd64-sv1-001-2009-11-02-11h-09m-46s.tgz 의 산출물로 확인한다.

☐ Summary Report - 1

THE LINUX FOUNDATION **LSB DISTRIBUTION TESTKIT MANAGER**

Get Certified Custom Tests Execution **Results** Help About Administration

Summary Report for Test Run of amd64-sv1-001-2009-11-02-11h-09m-46s

Test Execution Status

Automatic Tests	FAILED	View detailed report
Manual Tests	NOT FINISHED YET	

The manual tests are not executed yet. You can [run them now](#).

Analyze the Failing Tests

Please analyze the [detailed report](#) to understand the reasons of each fail and classify them into the following groups:

- Confirmed FAILs**, which are due to the real inconsistencies of your system with the LSB standard. To continue with the LSB certification, you have to fix your implementation to remedy such FAILs and then rerun the tests again.
- False FAILs**, which you believe are due to the incorrect tests. Please report such FAILs to the lf_lsbcert@lists.linux-foundation.org list. If the problem reported is confirmed you will be granted a *waiver* so the test failure will not affect your ability to certify.
- Unknown FAILs**, which you have no idea about. Please ask help at the lf_lsbcert@lists.linux-foundation.org list.

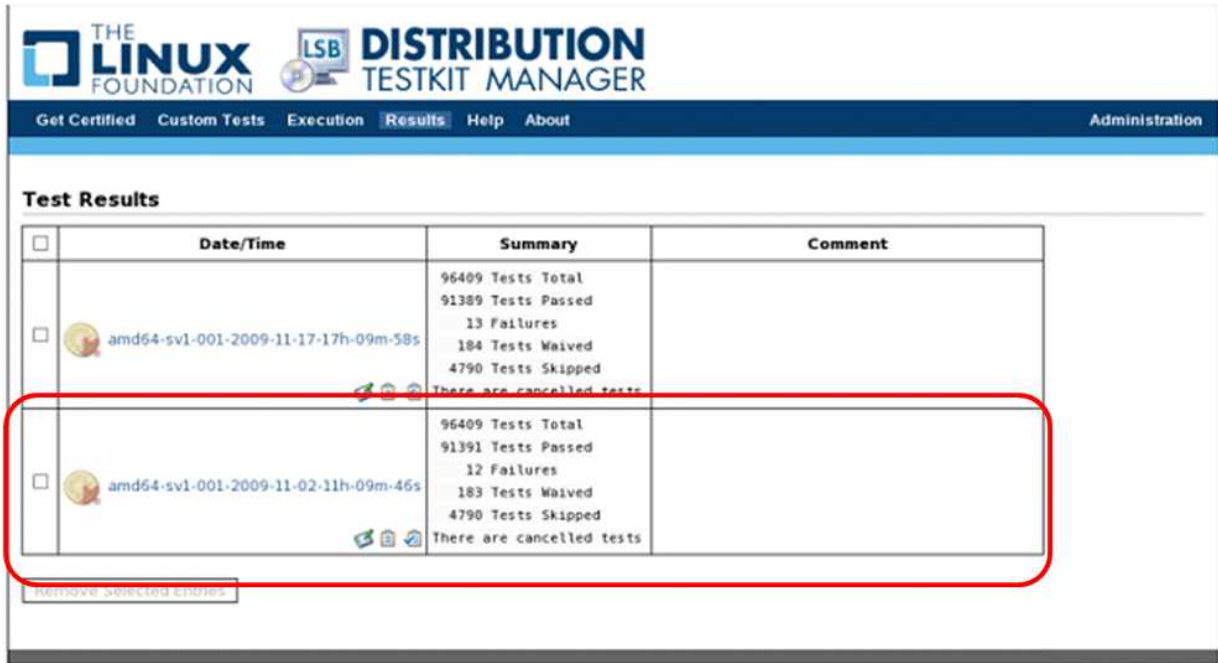
Note: only test results with all the FAILs waived are eligible for certification.

Locating Test Journals

You can find the consolidated test journals of this test run at the following path (click to download to another location):

- 테스트 방법으로 Automatic Tests와 Manual Tests가 있으며, Automatic Tests를 실시하였으나 결과가 FAILED 되었음을 확인할 수 있다.
- Failing Tests로는 Confirmed FAILs와 False FAILs, 그리고 Unknown FAILs 유형이 있으며, Confirmed FAILs는 LSB 표준 환경에서 시스템의 실제적인 비밀관성에 기인하는 오류를 의미하며, False FAILs는 부정확한 테스트에 기인하는 오류를 의미하며, 오류의 유형을 모를 경우에 Unknown FAILs를 출력한다.

□ Summary Report - 2



The screenshot shows the 'Test Results' page of the 'THE LINUX FOUNDATION DISTRIBUTION TESTKIT MANAGER'. The page has a navigation bar with links: 'Get Certified', 'Custom Tests', 'Execution', 'Results' (active), 'Help', 'About', and 'Administration'. Below the navigation bar, there's a table with the following data:

	Date/Time	Summary	Comment
☐	 amd64-sv1-001-2009-11-17-17h-09m-58s	96409 Tests Total 91389 Tests Passed 13 Failures 184 Tests Waived 4790 Tests Skipped There are cancelled tests	
☐	 amd64-sv1-001-2009-11-02-11h-09m-46s	96409 Tests Total 91391 Tests Passed 12 Failures 183 Tests Waived 4790 Tests Skipped There are cancelled tests	

The second row of the table is highlighted with a red rectangular box. Below the table, there is a small link that says 'Retrieve detailed results'.

- 테스트 결과에 대한 종합적인 정보를 출력한다. Tests Total, Tests Passed, Failures, Tests Waived, Tests Skipped 등이 있으며, 이 결과를 토대로 LSB 인증 테스트의 세부 항목에 대한 정보를 확인할 수 있다.

□ Detail Report



**DISTRIBUTION
TESTKIT MANAGER**

[Get Certified](#)
[Custom Tests](#)
[Execution](#)
[Results](#)
[Help](#)
[About](#)
Administration

LSB automatic test results for GINUX release 2.2 (mushroom) on amd64

The complete console output log is available [here](#).

Test Info

Tests Started At	11:09:46 02-11-2009
Tests Finished At	18:44:34 02-11-2009
Host Name	svl-001
Host OS	GINUX release 2.2 (mushroom)
Host Machine	x86_64
Host Platform	amd64
Host Kernel	2.6.18-92.GX2 #1 SMP Mon Jan 12 16:19:55 KST 2009

Tests executed

Certification Tests

Command Tests

Command Check	v 4.0.0-2	Success	Passed: 143
---------------	-----------	---------	-------------

Static Interface Tests

Command Check	v 4.0.0-2	Success	Passed: 143
Static Interface Tests			
Library Check	v 4.0.0-2	Warnings: 27	Passed: 49664
Runtime Interface Tests			
Core Tests	v 4.0.3-1	Failures: 2	Passed: 5823
Cpp-T2C Tests	v 4.0.0-3	Success	Passed: 1763
Desktop-T2C Tests	v 4.0.0-4	Success	Passed: 4899
Desktop Tests:	v 4.0.0-2		
Cairo Tests		No Result	Passed: 0
Fontconfig Tests		No Result	Passed: 0
FreeType Tests		No Result	Passed: 0
GTK Tests		Failures: 1	Passed: 0
Libpng Tests		No Result	Passed: 0
Libxml Tests		No Result	Passed: 0
Qt3 Tests		Warnings: 3	Passed: 15
Qt4 Tests		Success	Passed: 308
Libstdc++ Tests	v 4.1.0-4.lsb4	Failures: 5	Passed: 1794
OLVER Core Tests	v 4.0.0-5	Failures: 4	Passed: 4820
Perl Tests	v 4.0.5-1	Success	Passed: 871
Printing Tests	v 4.0.3-1	Success	Passed: 84
Python Runtime Tests	v 4.0.0-2	Success	Passed: 107
Qt3-Azov Tests	v 4.0.0-1	Success	Passed: 9791
Qt4-Azov Tests	v 4.0.0-1	Success	Passed: 11018
X11 Tests (XTS5)	v 5.1.5-16.lsb4	No Result	Passed: 0

○ Failures 내용 - 총 12건

- Core Test - 2건
- GTK Test - 1건
- Libstdc++ Test - 5건
- OLVER Core Test - 4건

□ Test log

```
PID: 16577, CHLD_PID: 16579 (cert)
[11:09] Script directory is /opt/lsb/test/manager/autotest-ext
[11:09] Using package directory: /var/opt/lsb/test/manager/lsb-packages
[11:09] LSB Test Execution Framework v4.0.0.2
[11:09] Cmd line: --webui -f /var/opt/lsb/test/manager/profiles/~session.127.0.0.1.1257125956 --results-dir=/var/opt/lsb/test/manager/results
[11:09] Host name: svl-001
[11:09] Host machine: x86_64
[11:09] Host platform: amd64
[11:09] Host kernel: 2.6.18-92.GX2 #1 SMP Mon Jan 12 16:19:55 KST 2009
[11:09] Host OS: LINUX release 2.2 (mushroom)
[11:09] Host triplet: x86_64-gnu-linux
[11:09] Host package manager: RPM
[11:09] LSB version: 4.0
[11:09] % ls /lib64/ld-lsb-x86-64.so.3 >/dev/null 2>&1
[11:09] cmdchk check...
[11:09] libchk check...
[11:09] libstdc++ check...
[11:09] core check...
[11:09] cpp-t2c check...
[11:09] olver check...
[11:09] perl check...
[11:09] python check...
[11:09] desktop check...
[11:09] xts5 check...
[11:09] printing check...
[11:09] desktop-t2c check...
[11:09] qt3-azov check...
[11:09] % ldconfig -p | grep "\<libqt-mt\so\3"
libqt-mt.so.3 (libc6,x86-64) => /usr/lib64/qt-3.3/lib/libqt-mt.so.3
libqt-mt.so.3 (libc6) => /usr/lib/qt-3.3/lib/libqt-mt.so.3
[11:09] qt4-azov check...
[11:09] % ldconfig -p | grep "\<libQtCore\so\4"
libQtCore.so.4 (libc6,x86-64) => /usr/lib64/qt4/lib64/libQtCore.so.4
libQtCore.so.4 (libc6) => /usr/lib/qt4/lib/libQtCore.so.4
[11:09] % ldconfig -p | grep "\<libQtGui\so\4"
```

□ 테스트 결과

[표4-3. LINUX LSB 테스트 결과]

Total	Passed	Failures	Waived	Skipped	비고
96409	91391	12	183	4790	LSB 인증 가능

○ Failures 내용

- Core Test : 2건
 - 표준 Core Libraries에서 오류 2건 발생
- GTK Test : 1건
 - GTK 라이브러리를 링크 과정에서 오류 1건 발생
- Libstdc++ Test : 5건
 - 표준 C++ 프로그램을 컴파일/링크할 때 필요한 headers and static library files에서 오류 5건 발생
- OLVER Core Test : 4건
 - Mail Linux System Libraries의 프로그래밍 인터페이스의 기능성과 적합성 테스트를 수행한 결과 4건의 오류 발생

○ 배포판 사용 시 문제 없음으로 결론

(2) SULinux

☐ LSB 인증 테스트 결과 개요

테스트 결과는 /var/opt/lsb/test/manager/results/amd64-sv2-001-<date><time> 디렉터리에 생성되는

- sum_rep.html
- one_line_summary.html
- test_log.html
- autotest_info
- host_info
- test_journals
- test_logs
- lsb-autotest-amd64-sv2-001-2009-11-17-21h-11m-40s.tgz 의 산출물로 확인한다.

☐ Summary Report - 1

THE LINUX FOUNDATION **LSB DISTRIBUTION TESTKIT MANAGER**

Get Certified Custom Tests Execution **Results** Help About Administration

Summary Report for Test Run of amd64-sv2-001-2009-11-17-21h-11m-40s

Test Execution Status

Automatic Tests	FAILED	View detailed report
Manual Tests	NOT REQUIRED	

Analyze the Failing Tests

Please analyze the [detailed report](#) to understand the reasons of each fail and classify them into the following groups:

- Confirmed FAILs**, which are due to the real inconsistencies of your system with the LSB standard. To continue with the LSB certification, you have to fix your implementation to remedy such FAILs and then rerun the tests again.
- False FAILs**, which you believe are due to the incorrect tests. Please report such FAILs to the lf_lsbcert@lists.linux-foundation.org list. If the problem reported is confirmed you will be granted a *waiver* so the test failure will not affect your ability to certify.
- Unknown FAILs**, which you have no idea about. Please ask help at the lf_lsbcert@lists.linux-foundation.org list.

Note: only test results with all the FAILs waived are eligible for certification.

Locating Test Journals

You can find the consolidated test journals of this test run at the following path (click to download to another location):
[/var/opt/lsb/test/manager/results/amd64-sv2-001-2009-11-17-21h-11m-40s/lsb-autotest-amd64-sv2-001-2009-11-17-21h-11m-40s.tgz](#)
 You need to attach them when applying for certification or when communicating with the support staff.

- 테스트 방법으로 Automatic Tests와 Manual Tests가 있으며, Automatic Tests를 실시하였으나 결과가 FAILED 되었음을 확인할 수 있다.

Failing Tests로는 Confirmed FAILs와 False FAILs, 그리고 Unknown FAILs 유형이 있으며, Confirmed FAILs는 LSB 표준 환경에서 시스템의 실제적인 비일관성에 기인하는 오류를 의미하며, False FAILs는 부정확한 테스트에 기인하는 오류를 의미하며, 오류의 유형을 모를 경우에 Unknown FAILs를 출력한다.

☐ Summary Report - 2



**DISTRIBUTION
TESTKIT MANAGER**

[Get Certified](#)
[Custom Tests](#)
[Execution](#)
[Results](#)
[Help](#)
[About](#)
[Administration](#)

Test Results

☐	Date/Time	Summary	Comment
☐	 amd64-sv2-001-2009-11-17-21h-11m-40s  	17849 Tests Total 13100 Tests Passed 2 Failures 56 Tests Waived 4684 Tests Skipped	

Remove Selected Entries

Copyright © 2007-2008 Linux Foundation. All rights reserved.
 LSB is a trademark of the Linux Foundation. Linux is a registered trademark of Linus Torvalds.

- 테스트 결과에 대한 종합적인 정보를 출력한다. Tests Total, Tests Passed, Failures, Tests Waived, Tests Skipped 등이 있으며, 이 결과를 토대로 LSB 인증 테스트의 세부 항목에 대한 정보를 확인할 수 있다.

□ Detail Report




[Get Certified](#)
[Custom Tests](#)
[Execution](#)
[Results](#)
[Help](#)
[About](#)
[Administration](#)

LSB automatic test results for SULinux release 2.0 on amd64

The complete console output log is available [here](#).

Test Info

Tests Started At	21:11:40 17-11-2009
Tests Finished At	01:47:14 18-11-2009
Host Name	sv2-001
Host OS	SULinux release 2.0
Host Machine	x86_64
Host Platform	amd64
Host Kernel	2.6.18-92.el5 #1 SMP Mon Feb 23 20:45:19 KST 2009

Tests executed

Certification Tests		
Command Tests		
Command Check	v 3.1.1-3	Success Passed: 133
Static Interface Tests		
Library Check	v 3.1.1-3	Success Passed: 5943

Tests executed

Certification Tests		
Command Tests		
Command Check	v 3.1.1-3	Success Passed: 133
Static Interface Tests		
Library Check	v 3.1.1-3	Success Passed: 5943
Runtime Interface Tests		
Core Tests	v 3.1.1-4	Failures: 1 Passed: 5288
Libstdc++ Tests	v 3.4.9-6.lsb3	Success Passed: 1272
Automated Application Battery Tests		
Apache Tests	v 2.2.0-2.lsb3	Success Passed: 1
Expect Tests	v 5.43.0-5.lsb3	Success Passed: 6
Groff Tests	v 1.19.2-2.lsb3	Success Passed: 5
Lynx Tests	v 2.8.5-4.lsb3	Success Passed: 20
Python Tests	v 2.4.3-3.lsb3	Failures: 1 Passed: 291
Rsync Tests	v 2.6.7-2.lsb3	Success Passed: 6
Samba Tests	v 3.0.22-2.lsb3	Success Passed: 6
Tcl Tests	v 8.4.11-4.lsb3	Success Passed: 129

Command Check v 3.1.1-3

The lsb-cmdchk test suite verifies that all the commands required by the LSB specification are present. It does not check the behaviour of those commands.

[Open the journal file](#)

Test suite information

Test Engine Version	tetj-1.0
Test Started At	21:11:48 20091117

○ Failures 내용 - 총 2건

- Core Test - 1건
 - 표준 Core Libraries에서 오류 1건 발생
- Python Test - 1건
 - Python 라이브러리 링크 과정에서 오류 1건 발생

□ Test log

```
PID: 6885, CHLD_PID: 6891 (cert)
[21:11] Script directory is /opt/lsb/test/manager/autotest-ext
[21:11] Using package directory: /var/opt/lsb/test/manager/lsb-packages
[21:11] LSB Test Execution Framework v4.0.0.2
[21:11] Cmd line: --webui -f /var/opt/lsb/test/manager/profiles/~session.127.0.0.1.1258458838 --results-dir=/var/opt/lsb/test/manager/results
[21:11] Host name: sv2-001
[21:11] Host machine: x86_64
[21:11] Host platform: amd64
[21:11] Host kernel: 2.6.18-92.el5 #1 SMP Mon Feb 23 20:45:19 KST 2009
[21:11] Host OS: SULinux release 2.0
[21:11] Host triplet: x86_64-gnu-linux
[21:11] Host package manager: RPM
[21:11] LSB version: 3.1
[21:11] % ls /lib64/ld-lsb-x86-64.so.3 >/dev/null 2>61
[21:11] cmdchk check...
[21:11] libchk check...
[21:11] libstdc++ check...
[21:11] core check...
[21:11] groff check...
[21:11] rsync check...
[21:11] lynx check...
[21:11] expect check...
[21:11] apache check...
[21:11] % netstat -an 2>/dev/null | grep -e ':80 \+[0-9:]'
[21:11] Warning: code 1(0) has been returned.
[21:11] % netstat -anp 2>/dev/null | grep -e ':80 \+[0-9:]'
[21:11] Warning: code 1(0) has been returned.
[21:11] python-app check...
[21:11] tcl check...
[21:11] samba check...
[21:11] Looking for available packages
[21:11] Max mode: a
[21:11] Max mode: a
[21:11] % rpm -q lsb-tcl lsb-rsync lsb-test-olver-core lsb-test-xts5 lsb-tet3 lsb-lynx lsb-test-python lsb-cmdchk lsb-expect lsb-test-printing
[21:11] Warning: code 15(0) has been returned.
[21:11] package lsb-tcl is not installed
```

□ 테스트 결과

[표4-4. SULinux LSB 테스트 결과]

Total	Passed	Failures	Waived	Skipped	비고
17849	13100	2	56	4684	LSB 인증 가능

○ Failures 내용

- Core Test : 1건
 - 표준 Core Libraries에서 오류 2건 발생
- Python Test : 1건
 - Python Libraries에서 오류 1건 발생

○ 배포판 사용 시 문제 없음으로 결론

(3) LSB 인증 테스트 결과

LSB 인증 지원 대상의 테스트 결과는 다음과 같다.

☐ GINUX

[표4-5. GINUX LSB 인증 테스트 결과]

Total	Passed	Failures	Waived	Skipped	비고
96409	91391	12	183	4790	LSB 인증 가능

○Failures 내용

- Core Test : 2건
 - 표준 Core Libraries에서 오류 2건 발생
- GTK Test : 1건
 - GTK 라이브러리를 링크 과정에서 오류 1건 발생
- Libstdc++ Test : 5건
 - 표준 C++ 프로그램을 컴파일/링크할 때 필요한 headers and static library files에서 오류 5건 발생
- OLVER Core Test : 4건
 - Mail Linux System Libraries의 프로그래밍 인터페이스의 기능성과 적합성 테스트를 수행한 결과 4건의 오류 발생

○ 배포판 사용 시 문제 없음으로 결론

☐ SULinux

[표4-6. SULinux LSB 인증 테스트 결과]

Total	Passed	Failures	Waived	Skipped	비고
17849	13100	2	56	4684	LSB 인증 가능

○Failures 내용

- Core Test : 1건
 - 표준 Core Libraries에서 오류 2건 발생
- Python Test : 1건
 - Python Libraries에서 오류 1건 발생

○ 배포판 사용 시 문제 없음으로 결론

라. 인증 지원 로드맵

- LSB 인증의 두 축인 DTK와 ACK를 활용한 인증 메카니즘에서 본 공개SW 역량프라자 LSB 인증 지원에서는 국내 공개SW 기반의 Distribution을 인증하는 DTK를 우선 수행
- LSB DTK 인증 지원 선 수행 근거
 - 국내 공개SW 배포판의 상호 호환성의 부재로 인해 사용자에게 일관성, 솔루션 개발자에게 호환성을 제공하지 못해 공개SW 사용 기피와 도입 및 발전 저해의 요인이 됨
 - 또한 공개SW 기술 지원에 대한 표준적인 환경을 제공하고자 하는 필요성 제기
- LSB ACK에 대한 인증 필요성
 - 공개SW의 도입 저해 요인인 공개SW의 솔루션 부족 문제와 안정성 및 신뢰성 확보를 위한 LSB ACK의 필요성
 - 공개SW의 수요처 확보를 위해 다양한 솔루션에 대한 인증을 통해 상호 호환성 및 상호 운용성 확보
 - 공개SW 기반의 솔루션에 대한 표준화 환경 조성을 통한 국내 공개SW 시장 활성화

5. 맺음말

LSB 인증은 리눅스 전체 환경을 위한 표준화 정책의 일환으로 채택되고 있으며, 개발 비용 감소, Vendor간 협력 유도, 효율적인 마케팅 지원 등의 효과 외에, LSB 인증으로 인해 표준화가 정착되고 공개SW의 호환성과 신뢰성이 향상되며, 이를 기반으로 국내 공개SW의 국제적인 경쟁력이 강화되는 등의 효과가 수반된다.

국내 대표적인 공개SW 중 **Linux**와 **SULinux**의 인증을 통해 국내 공개SW의 국제적인 경쟁력이 강화되고 이를 기반으로 다양한 하드웨어와 소프트웨어의 호환성을 증대시켜 애플리케이션의 개발 시 추가 개발 기간 및 추가 개발 비용이 소요되지 않고, 기술 지원에 대한 신뢰성이 약하기 때문에 표준화를 통한 지속적인 기술지원에 대한 신뢰성을 확보하는 등의 효과가 있으며, LSB 인증과 같은 표준화 정책을 기반으로 공개SW의 이식성, 상호 운용성을 확보하고, 공개SW의 활성화 및 시장의 확대를 기대해본다.

6. 별첨

[별첨 1] LSB 인증 가이드

[별첨 2] LSB 인증 툴(DTK) 활용 가이드

별첨 1.

LSB 인증 가이드

2009년 12월 15일

정보통신산업진흥원

목 차

1. LSB와 LSB 인증	1
가. LSB 소개	1
나. LSB 인증(Certification)	1
다. 배경 및 목적	1
라. LSB 구성	2
마. LSB 사용	2
바. 요약	4
2. LSB 테스트 툴 및 LSB 툴 소개	4
가. Linux Application Checker	4
나. LSB Distribution Testkit(DTK)	4
다. LSB Software Development Kit(SDK)	4
라. Sample Implementation Tools(LSB SI)	5
3. LSB 인증 프로세스	5
가. 등록	6
나. LSB 툴 및 LSB 테스트 툴 확보	6
다. 테스트	7
라. 제품 등록	7
마. 테스트 결과 업로드	7
바. TMLA(Trademark License Agreement)에 서명	8
사. 테스트 결과 심사	8
아. LSB 인증 제품 리스트에 등재	8
자. LSB 인증명과 로고 부착	8
부록	
부록1. LSB 준수 어플리케이션 개발 프로세스	9
부록2. 참조 문헌 및 사이트	10

1. LSB와 LSB 인증

가. LSB 소개

- LSB(Linux Standard Base)는 다양한 리눅스 배포판의 비표준화 문제로 야기되는 어플리케이션 개발 및 활성화 저해요소를 해결하기 위하여 설립되었다.
 - 1998년 5월경 Linus Torvalds, 하드웨어 및 상용 소프트웨어 업체, Linux International, 및 FreeBSD의 Jordan Hubbard가 지원하여 프로젝트로 구성
 - Linux Foundation의 WorkGroup으로 활동
 - 2001년 6월 LSB Specification 1.0 발표를 시작으로 현재 4.0까지 발표 됨

※ Linux Foundation

프리스탠다드그룹(FSG)과 오픈소스개발연구소(OSDL)가 2007년 1월 22일에 합병하여 탄생한 조직으로 분산된 리눅스 진영의 지원조직을 단일화하고 리눅스 진영을 대표하고 있음

※ LSB Specification

LSB를 준수하는 시스템에서 요구되는 구성 요소(Module)를 정의하는 명세서

나. LSB 인증(Certification)

- LSB 인증은 Linux 배포판 및 어플리케이션의 두 가지 인증을 제공하고 있으며, LSB 인증을 취득한 제품만이 LSB 인증 Trademark를 사용할 수 있다. LSB 인증 Trademark는 개발자와 사용자가 LSB 인증된 배포판에서 LSB 인증된 어플리케이션을 사용할 경우 문제가 없다는 것을 보증한다.

다. 배경 및 목적

- LSB는 독립 소프트웨어 벤더(ISV)와 개별 어플리케이션 개발자들만을 위하여 만들어진 솔루션이 아니며 리눅스 전체 환경을 위한 표준화 정책이다.
- LSB의 필요성
 - ISV가 지원하는 리눅스의 라이브러리 버전, 파일의 위치, 파일시스템 구조 등의 복잡한 문제 와 여러 배포판별로 어플리케이션을 추가로 개발해야 하는 비용 감소
 - 하드웨어와 소프트웨어의 다른 특성으로 발생하는 소프트웨어 개발과 설치(Porting)의 복잡성을 감소 시켜 Vendor들의 협력을 유도
 - Linux Foundation과 Linux Developer Network(LDN)은 ISV의 개발 프로세스와 효율적 마케팅을 만들어 보다 쉽게 지원

라. LSB 구성

- LSB는 리눅스 시스템과 어플리케이션 간 상호운용성을 촉진하는 리눅스 운영체제를 위한 핵심 표준으로서 테스트를 목적으로 한 샘플 도구, 배포판과 어플리케이션에서 작성된 표준을 위한 테스트 도구 설정, 바이너리로 작성된 인터페이스 명세서를 포함하고 있다.
- LSB를 구성하는 핵심은 리눅스 어플리케이션 표준을 개발하는 방법과 어플리케이션 설정을 알려주는 바이너리로 작성된 인터페이스 명세서(Specification)이다.
- LSB 명세서 구성 요소
 - 공통 패키지와 설치 가이드
 - 공통 공유 라이브러리와 선정
 - 설정 파일
 - 파일 배치
 - 시스템 명령
 - 시스템 인터페이스를 위한 어플리케이션 바이너리 인터페이스(플랫폼과 어플리케이션 레벨 모두 해당)
- LSB 지원 플랫폼
 - Intel IA32
 - Intel IA64
 - x86-64/EM64T
 - IBM PPC 32
 - IBM PPC 64
 - IBM 31-bit S/390
 - IBM 64-bit zSeries

마. LSB 사용

- LSB는 독립 소프트웨어 벤더(ISV)에게 조금 더 쉬운 개발 환경을 제공하며, Linux Developer Network(LDN)에서 모든 정보를 얻을 수 있다.
- 어플리케이션 개발자들을 수용하기 위한 기본적인 방식
 - 최대한 많은 어플리케이션의 설치 구성

- 1) 어플리케이션의 설치 구성(Poting) 여부를 LDN에서 제공하는 리눅스 Application Checker를 사용하여 확인
- 2) 설치 구성(Poting)을 방해하는 어플리케이션 주요 요소 확인
- 3) 개발자가 더 많은 어플리케이션을 설치 구성(Poting)하여 변경을 적용할 수 있도록 함
- 4) 리눅스 Application Checker는 어플리케이션 개발을 위한 개발자 가이드 경로를 추천

○ LSB 인증 툴킷

- 1) 개발자가 처음부터 어플리케이션에 대한 문제점을 제거하고 어플리케이션을 구축하는데 이용
- 2) LSB가 소프트웨어 개발 킷(LSB SDK)을 LSB 워킹 그룹에서 만들고 유지 함
- 3) LSB SDK와 함께 소프트웨어 어플리케이션 포팅을 위해 요구되는 툴과 구축 환경으로 구성

□ 리눅스 및 어플리케이션 개발의 이점을 알리기 위한 최선의 방법은 리눅스 개발자 네트워크 (LDN)를 이용하여 정보를 얻는 것이다. 리눅스 개발자 네트워크에서 제공하는 리소스는 다음과 같다.

○ Linux Application Checker

- 어플리케이션 포팅을 점검하고 더 많은 포팅을 만들 수 있도록 권장사항을 확인

○ LSB Database Navigator

- 포팅 및 이동에 필요한 솔루션 및 대안을 배울 수 있는 구성요소를 식별하지 못할 수 있으므로 데이터베이스 네비게이터는 C/C++ 프로그래머가 볼 수 있는 리눅스 프로그램의 다양한 정보를 제공

○ LSB Build Tools

- LSB SDK는 개발자가 LSB를 준수하기 위해 어플리케이션을 빌드하는 동안 API를 사용하여 확인하고 바이너리와 RPM 패키지를 검증할 수 있도록 적합성을 보장

○ LSB Sample Implementation

- LSB Sample Implementation (LSB-si)은 최소한의 테스트를 목적으로 한 LSB를 따르는 실행 환경을 사용하며, LSB를 준수하는 어플리케이션은 변화가 많은 특정 배포판에서 잘 동작되는지 테스트를 수행 함. LSB 인증 프로그램은 LSB-si 기반에서 테스트 한 어플리케이션을 요구함

○ Tutorials and Blogs

- 일반적인 리눅스 어플리케이션 개발의 최신 방법과 LSB 준수사항, 어플리케이션 포팅을 위한 정보와 최신 관련 문서를 제공

- Forums and Mailing Lists
 - 많은 개발상의 문제점과 해결점 및 각종 팁을 확인
- LSB Certification and General Marketing Support
 - LSB, 포팅, 리소스는 리눅스 환경에서 개발자가 어플리케이션의 홍보를 최대한 결정하도록 하며, 인증된 어플리케이션은 제품 디렉터리에 포함 가능

바. 요약

- LSB는 어플리케이션 개발자에게 틀을 제공하고 전 세계적으로 리눅스의 장점을 활용할 수 있는 표준화 된 기반을 제공한다. 리눅스 개발 네트워크(LDN)는 개발자가 LSB 인증을 획득한 리눅스 배포판에서 어플리케이션의 설치 구성(Poting)과 Linux Standard Base에서 어플리케이션 벤더를 도울 수 있도록 설계되었다.

2. LSB 인증 틀 및 LSB 테스트 틀 소개

가. Linux Application Checker

- LDN(Linux Developer Network)이 제공하고 있는 Linux 상에서 동작하는 어플리케이션의 호환성 및 LSB 표준 근접성을 확인하는 테스트 도구로서 테스트 뿐 아니라 실제로 제품등록/테스트 결과 업로드 등의 인증절차를 처리할 수 있음
 - 설치 및 사용방법은 별첨 [LSB 인증 틀(AC) 활용 가이드]를 참조

나. LSB Distribution Testkit(DTK)

- 리눅스 디스트리뷰션이 LSB에 준거하고 있는지를 테스트하는 도구
 - 설치 및 사용방법은 별첨 [LSB 인증 틀(DTK) 활용 가이드]를 참조

다. LSB Software Development Kit(SDK)

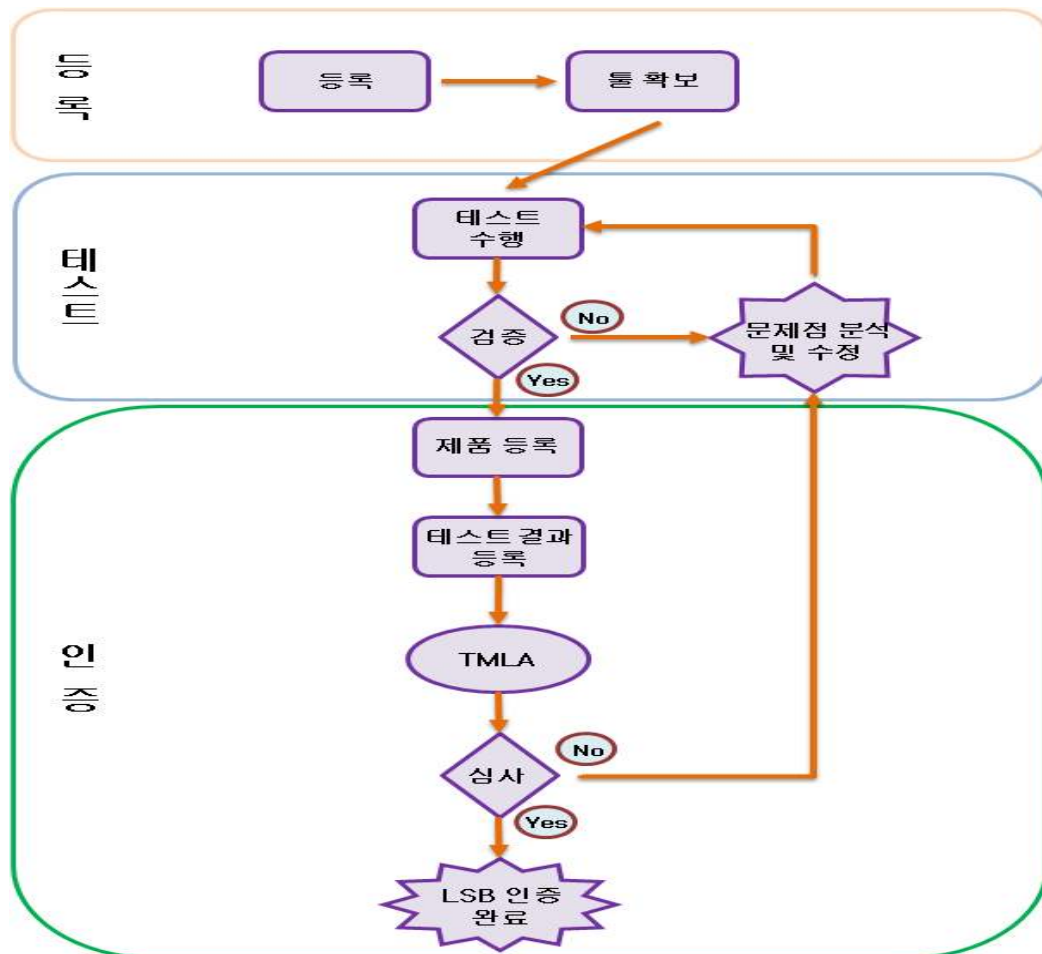
- LSB SDK를 사용하여 바이너리와 RPM 패키지가 LSB에 준거하고 있는지를 검증하고 빌드실행 중에 어플리케이션에 따른 API의 사용 상황을 모니터 하여 LSB 적합성을 보증할 수 있음
 - 설치 및 사용방법은 별첨 [LSB 인증 틀(SDK) 활용 가이드]를 참조

라. Sample Implementation Tools(LSB SI)

- LSB 디스트리뷰션의 핵심이 되는 최소한의 라이브러리와 어플리케이션의 테스트를 목적으로 한 LSB 기반 실행 환경을 구성할 수 있음
- 설치 및 사용방법은 별첨 [LSB 인증 툴(SI) 활용 가이드]를 참조

3. LSB 인증 프로세스

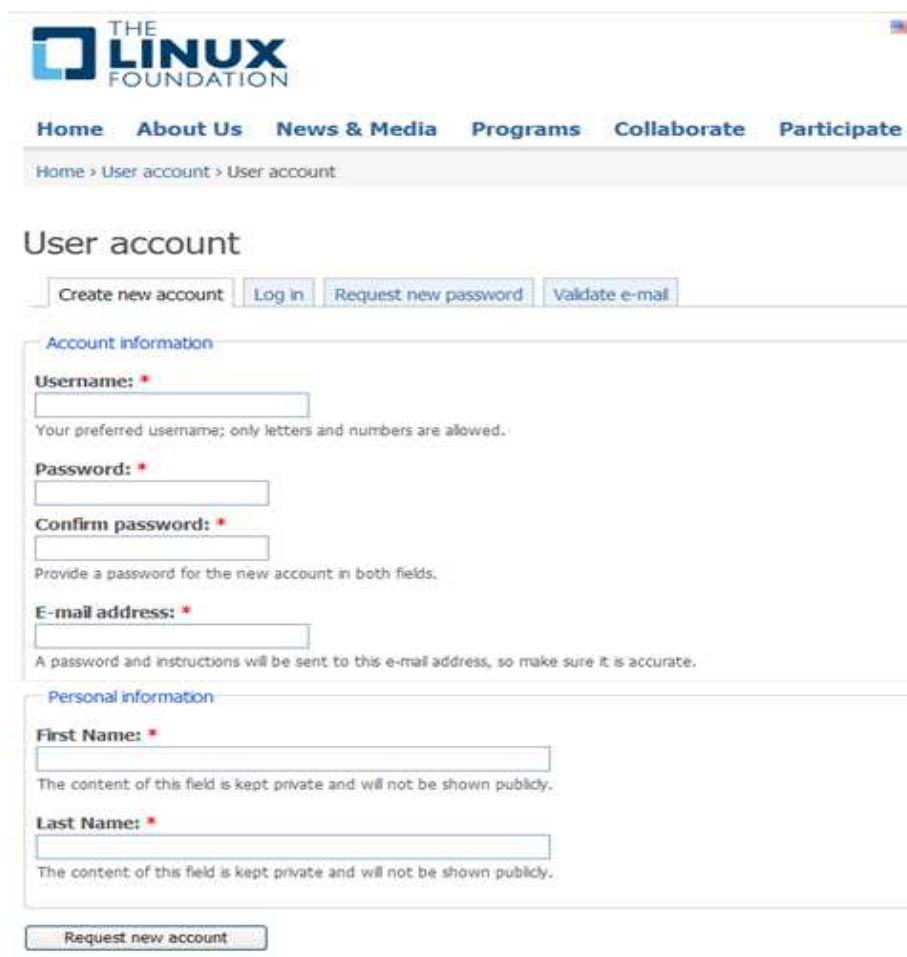
- LSB 인증 절차는 Linux Foundation 웹페이지에서 회원가입 후 등록, 테스트, 인증 순서로 프로세스가 진행된다.



[그림 3-1. LSB 인증 프로세스 절차]

가. 등록

- 1) 인증을 취득하기 위하여 제일 먼저 Linux Foundation의 웹사이트에서 자신의 계정을 생성한다.
- 2) 로그인후, 사용자 정보가 맞는지 확인한다.
 - 같은 회사 내에서 다른 누군가가 이미 회사명의 계정을 가지고 있는 경우에는 등록된 사람에게 필요한 추가 절차를 요청한다. 그렇지 않을 경우 새로운 회사명으로 등록



[그림 3-2. 계정 등록 화면]

나. LSB 툴 및 LSB 테스트 툴 확보

- 1) 다운로드 페이지에서 플랫폼 아키텍처에 맞는 LSB 툴 및 LSB 테스트 툴 다운로드
- 2) 별첨의 각종 활용가이드를 참조하여 설치

다. 테스트

- 1) 제품이 LSB 표준에 준거하고 있는 것을 확인하기 위해 LSB Distribution Testkit (DTK) Manager/Linux Application Checker를 이용해 적정한 인증테스트를 실행
 - 별첨. LSB 인증 툴(DTK) 활용가이드 참조
 - 별첨. LSB 인증 툴(AC) 활용가이드 참조
- 2) 테스트에서 문제점이 발견되면 수정 후 재 테스트 수행

라. 제품 등록

- 정식 인증을 받기 위해서 테스트 결과를 제출할 준비가 갖추어지면, 인증 시스템에 제품을 등록
 - 정식 등록 절차는 온라인 (<https://www.linuxfoundation.org/lsb-cert/product.php?add>) 또는 DTK Manager/Linux Application Checker로 수행 가능

The screenshot shows the 'New Product' registration page of the Linux Foundation's LSB Certification Management System. The page has a header with the Linux Foundation and LSB Certification logos, and a navigation bar with links like 'Certification', 'Product Directory', 'Problem Reports', 'Help', and 'About'. The main form area is titled 'New Product - BlogOfYang' and contains several input fields and checkboxes. The 'Choose the type of the product' section has radio buttons for 'Application' and 'Distribution'. Below this are text boxes for 'Product Name', 'Product Version', and 'Version Comment'. The 'Product Homepage' field has a 'http://' prefix. The 'Choose architectures' section has checkboxes for 'x86', 'x86-64', 'IA64', 'PPC32', 'PPC64', 'S390', and 'S390X'. A 'Submit' button is located at the bottom left of the form.

[그림 3-3. 제품 등록화면]

마. 테스트 결과 업로드

- 테스트 결과는 제품의 등록이 완료된 후 업로드 함
 - 가장 간단한 방법은 DTK Manager/Linux Application Checker를 사용하는 것이며, 인증 시스템내의 자신의 제품 페이지로부터도 업로드 할 수 있음

바. TMLA(Trademark License Agreement)에 서명

- LSB 상표 라이선스 계약(Trademark License Agreement = TMLA)에 서명한 후, 소정의 신청비용을 지불한다. (http://www.linuxfoundation.org/en/Fee_Schedule)
 - TMLA는 인증 시스템내의 제품 페이지로부터 확보 가능

Fee Schedule

LSB Certification Fee Schedule

Corporate members and non-individual affiliates of the Linux Foundation receive "member" level pricing for all registrations. The fee is due upon submission of test results.

LSB Certification includes a license for the use of the trademarked "LSB Certified" name and logo on certified products.

LSB Certified Distribution		
Certification Fee	Non-Member	Member
Distribution	\$5000	\$2500
Each additional architecture certified simultaneously	\$1000	\$500
Maintenance release or rebranded product	\$500	\$250

LSB Certified Application		
Certification Fee	Non-Member	Member
Application	\$0 \$1000	\$0 \$250
Each additional architecture certified simultaneously	\$0 \$250	\$0 \$100
Maintenance release or rebranded product	\$0 \$250	\$0 \$100

Special Announcement: Fees for application certification are being waived in 2009.

[그림 3-4. Fee Schedule]

사. 테스트 결과 심사

- Linux Foundation에 의한 테스트 결과의 심사는 TMLA 체결과 비용의 지불이 확인된 후 진행됨
 - 테스트 결과가 심사에 통과되지 못한 경우, 심사에서 밝혀진 문제점을 해결하고 재 테스트를 수행하여 그 결과를 다시 제출해야 함

아. LSB 인증 제품 리스트에 등재

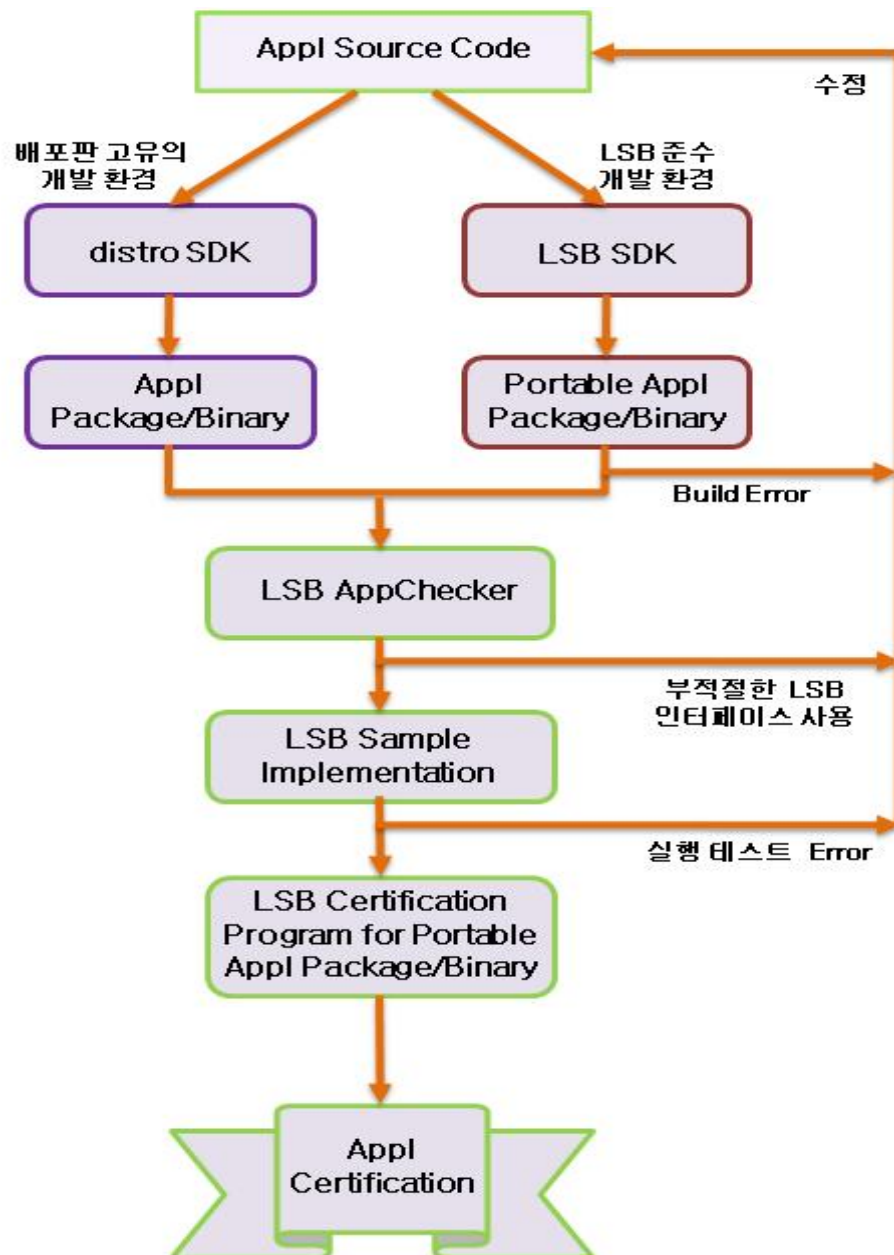
- 심사에 합격한 제품은, LSB 인증 제품 리스트 (register of LSB Certified products)에 등재됨(https://www.linuxfoundation.org/lsb-cert/productdir.php?by_prod)
 - Linux Foundation은, 해당 제품이 LSB 인증의 필요조건을 계속하여 만족하고 있는지, 간혹 관련 내용의 제출을 요구하는 경우도 있음

자. LSB 인증명과 로고 부착

- LSB 표준과의 호환성을 사용자에게 알리기 위해, 상표 등록된 LSB 인증명과 로고를 제품에 붙이는 것을 권고함
 - 자세한 사항은 Linux Foundation 발행의 Trademark Usage Guidelines (http://www.linuxfoundation.org/en/Trademark_Usage_Guidelines) 참조

부록1. LSB 준수 어플리케이션 개발 프로세스

LSB 인증을 위하여 개발되는 고유의 환경 및 어플리케이션의 개발 절차는 Source 단계부터 적용되어 수행된다.



[그림 3-5. LSB 준수 어플리케이션 개발 프로세스]

부록2. 참조 문헌 및 사이트

- LSB 사양서 다운로드
<http://www.linuxfoundation.org/en/Specifications>
- 배포판 및 어플리케이션 인증 정보 제공 사이트
<http://ldn.linuxfoundation.org/lsb>
- 인증 완료된 배포판 및 어플리케이션 리스트
https://www.linuxfoundation.org/lsb-cert/productdir.php?by_prod
- 주요 Linux 배포판의 인터페이스/LSB인증상황/채용하고 있는 커널
/gcc/glibc/GNOME/KDE 버전 등
<http://ldn.linuxfoundation.org/lsb/distro-component-matrix>
- 배포판 및 어플리케이션의 인증 비용, 등록상표 LSB 및 로고마크의 사용허가 비용
http://www.linuxfoundation.org/en/Fee_Schedule
- LSB포럼(LSB어플리케이션 개발, ApplChecker, 어플리케이션의 호환성, 어플리케이션의 인증 등에 대하여 LSB 커뮤니티가 질문에 회답)
<http://ldn.linuxfoundation.org/support>
- LSB Database Navigator
<http://dev.linuxfoundation.org/navigator/commons/welcome.php>
- Linux Application Checker, LSB Software Development Kit(SDK), LSB SI)등 다운로드(소스코드의 다운로드 사이트의 링크도 존재)
<http://ldn.linuxfoundation.org/support/downloads>

별첨 2.

LSB 인증 툴(DTK) 활용 가이드

2009년 12월 15일

정보통신산업진흥원

목 차

1. LSB Distribution Test Kit 설치 및 활용 가이드	1
가. Distribution Test Kit의 설치	1
2. Web-UI 활용 가이드	4
가. Web-UI 호출	4
나. LSB DTK Manager의 Get Certificated, Custom Tests	4
3. Command Line UI 활용 가이드	12
가. Command Line 인터페이스	12
4. 테스트 결과물 저장소(Test Results Repository)	14
5. Distirbution Test Kit(DTK)의 정지 및 삭제	14
가. Web-Server 정지	14
나. DTK Manager 삭제	14

1. LSB Distribution Test Kit 설치 및 활용 가이드

가. Distribution Test Kit(DTK)의 설치

- 1) LSB 다운로드 페이지에서 플랫폼 아키텍처에 맞는 LSB Distribution Test Kit Package를 다운로드 한다. 패키지 다운로드를 위해서는 로그인이 필요하다.

○ <http://ldn.linuxfoundation.org/support/downloads>

LSB Distribution Tools

LSB distribution tools help distribution and upstream developers in analysing their products with regard to LSB requirements. They can be useful both as just a part of QA cycle and as a part of LSB certification process.

The LSB Distribution Testkit (DTK) contains everything developers need to test a Linux distribution for LSB compliance. The front end of the Distribution Testkit is DTK Manager that helps visually manage the testing process. DTK Manager automatically downloads necessary tests from the Linux Foundation FTP server, so please be sure you have Internet connection on the testing system the first time you want to run the tests.

To install the LSB DTK Manager, please download the appropriate tar file, extract the tar file, and run the included install.sh script.

Please refer to [LSB Distro Testkit Getting Started](#) for instructions on how to use the tool.

LSB 4.0 Distribution Testkit	ia32	lsb-dist-testkit-4.0.0-4.ia32.tar.gz
LSB 4.0 Distribution Testkit	ia64	lsb-dist-testkit-4.0.0-4.ia64.tar.gz
LSB 4.0 Distribution Testkit	ppc32	lsb-dist-testkit-4.0.0-4.ppc32.tar.gz
LSB 4.0 Distribution Testkit	ppc64	lsb-dist-testkit-4.0.0-4.ppc64.tar.gz
LSB 4.0 Distribution Testkit	s390	lsb-dist-testkit-4.0.0-4.s390.tar.gz
LSB 4.0 Distribution Testkit	s390X	lsb-dist-testkit-4.0.0-4.s390x.tar.gz
LSB 4.0 Distribution Testkit	amd64	lsb-dist-testkit-4.0.0-4.x86_64.tar.gz

- 2) 압축을 해제한다.

○ `tar xvfz lsb-dist-testkit-<version>.<architecture>.tar.gz`

```
[root@localhost home]# tar xvfz lsb-dist-testkit-4.0.0-4.x86_64.tar.gz
./
./lsb-dist-testkit/
./lsb-dist-testkit/install.sh
./lsb-dist-testkit/post-install.sh
./lsb-dist-testkit/lsb-cmdchk-4.0.0-2.x86_64.rpm
./lsb-dist-testkit/lsb-libchk-4.0.0-2.x86_64.rpm
./lsb-dist-testkit/lsb-test-core-4.0.3-1.x86_64.rpm
./lsb-dist-testkit/lsb-test-desktop-4.0.0-2.x86_64.rpm
./lsb-dist-testkit/lsb-test-xts5-5.1.5-16.lsb4.x86_64.rpm
./lsb-dist-testkit/lsb-qm-2.2-8.lsb4.x86_64.rpm
./lsb-dist-testkit/lsb-test-libstdcpp-4.1.0-4.lsb4.x86_64.rpm
./lsb-dist-testkit/lsb-test-perl-4.0.5-1.noarch.rpm
./lsb-dist-testkit/lsb-test-python-4.0.0-2.x86_64.rpm
./lsb-dist-testkit/lsb-test-desktop-t2c-4.0.0-4.x86_64.rpm
./lsb-dist-testkit/lsb-test-printing-4.0.3-1.x86_64.rpm
./lsb-dist-testkit/lsb-test-olver-core-4.0.0-5.x86_64.rpm
./lsb-dist-testkit/lsb-test-cpp-t2c-4.0.0-3.x86_64.rpm
./lsb-dist-testkit/lsb-test-qt3-azov-4.0.0-1.x86_64.rpm
./lsb-dist-testkit/lsb-test-qt4-azov-4.0.0-1.x86_64.rpm
./lsb-dist-testkit/lsb-dtk-manager-4.0.0-2-5.x86_64.rpm
./lsb-dist-testkit/lsb-tet3-lite-3.7-15.lsb4.x86_64.rpm
./lsb-dist-testkit/lsb-tet3-lite-devel-3.7-15.lsb4.x86_64.rpm
./lsb-dist-testkit/lsb-xvfb-1.2.0-11.x86_64.rpm
./lsb-dist-testkit/lsb-python-2.4.5-2.lsb4.x86_64.rpm
./lsb-dist-testkit/lsb-setup-4.0.0-2.noarch.rpm
./lsb-dist-testkit/inst-config
```

- 3) 압축 해제 후 생성된 lsb-dist-testkit 디렉터리로 이동하여 설치 스크립트를 실행한다. 설치 중 root 암호를 입력해야 할 수도 있다.

○ cd lsb-dist-testkit

○ ./install.sh

```
[root@localhost home]# cd lsb-dist-testkit
[root@localhost lsb-dist-testkit]# ./install.sh

This system appears to be a RPM-based distribution, such as those from
Red Hat, SuSE/Novell, Mandriva, Asianux, etc.

Is this correct? yes

Installing packages...
Installed successfully.

Warning: Couldn't find the required tool: Xvfb (X Virtual Frame Buffer).
Xvfb is required for X11 tests for LSB 3.0 and 3.1. If you are going
to run these tests or to get certified for these LSB versions,
you should install Xvfb using the tools provided by your distribution.

ATTENTION!

This tool runs as the root user and installs packages on its own?
Therefore it can do pretty much harm if it gets out of control.
We have tried to make the script reasonably safe but there is no warranty.
Use it at YOUR OWN RISK only, and only if you know what you are doing.

Some setuid and setgid binaries may be generated and left behind after
execution of the test suites (especially if certain tests fail).
You should not install this package on production systems
or in situations where local users are not completely trusted.

PLEASE DO NOT RUN IT ON A MACHINE THAT HOLDS IMPORTANT DATA, OR PROVIDES
IMPORTANT SERVICES!

LSB Distribution Testkit Manager has been installed to
/opt/lsb/test/manager.

There were warnings.

Use /opt/lsb/test/manager/bin/lsb-dtk-start.sh to run LSB DTK Manager.

Read more info at
http://ispras.linux-foundation.org/index.php/About_LSB_Distribution_Testkit_Manager

localedef -c -f lang0.ascii -i lang0.src USX4L0
localedef -c -f lang1.ascii -i lang1.src USX4L1
localedef -c -f lang2.ascii -i lang2.src USX4L2
localedef -c -f lang3.ascii -i lang3.src USX4L3
localedef -f lang3.ascii -i lang4.src USX4L3@dict

Checking if there is a user named 'tester' on this system...
User 'tester' does not exist. Attempting to create it....
User 'tester' was created successfully.
If you later want to delete this user account, run the following command (as root):
    userdel -rf tester
Here 'userdel' is assumed to be somewhere in your path

Attempting to create group 'tester'

Installing custom locales. You may need root privileges to do this.
Locale "ru_RU_test" is now installed.
Locale "ru_RU_utf8_test" is now installed.

LSB cpp-t2c Test Suite has been installed to
/opt/lsb/test/cpp-t2c.

Checking if there is a user named 'tester' on this system...
Attempting to create group 'tester'

Installing "en_POSIX_test_dec" locale. You may need root privileges to do this.

Installing custom image loader. You may need root privileges to do this.
Updating the list of loaders in /etc/gtk-2.0/x86_64-redhat-linux-gnu/gdk-pixbuf.loaders
Uninstalling custom pixbuf loader...
Updating the list of loaders in /etc/gtk-2.0/x86_64-redhat-linux-gnu/gdk-pixbuf.loaders
Installing custom pixbuf loader...
LSB Desktop-I2C Test Suite has been installed to
/opt/lsb/test/desktop-t2c.
Preparing tester account...
Group 'olover' was successfully added
User 'olover_tester' was successfully added
Installing test data for locale.nlcat ... Done
Installing test data for locale.textdomain ... Done

Checking if there is a user named 'tester' on this system...
Attempting to create group 'tester'
LSB azov-qt3 Test Suite has been installed to
/opt/lsb/test/qt3-azov.

Checking if there is a user named 'tester' on this system...
Attempting to create group 'tester'
LSB azov-Qt4 Test Suite has been installed to
/opt/lsb/test/qt4-azov.
There may have been problems with the package installation. Check
error-log.txt for more information.
[root@localhost lsb-dist-testkit]#
```

4) LSB DTK Manager를 실행한다.

○ /opt/lsb/test/manager/bin/lsb-dtk-start.sh [포트번호]

- 포트 번호(기본 값은 8888)의 입력은 선택사항이다. 입력하지 않을 경우 기본 값이 적용된다.
- 이 스크립트는 LSB DTK Manager에 내장되어 있는 Simple Web Server를 실행시킨다.

```
[root@localhost lsb-dist-testkit]# /opt/lsb/test/manager/bin/lsb-dtk-start.sh
The port 8888 will be used by the LDTK Manager web-UI server.
If you want to change this, run /opt/lsb/test/manager/bin/lsb-dtk-start.sh <port>

Server started. Log file location:
/var/opt/lsb/test/manager/log/dtk-server.log.8888

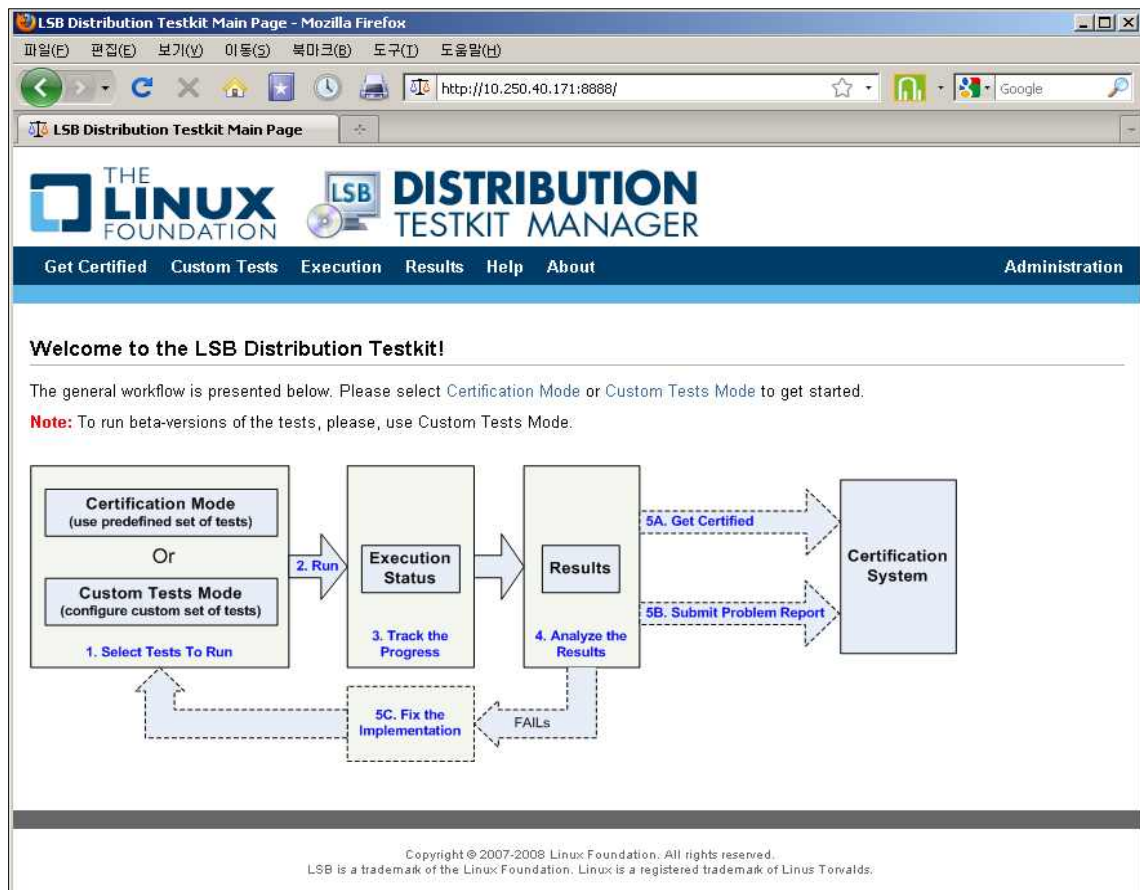
The start page should be opened in a browser shortly.
If it doesn't open, you can load it at http://localhost.localdomain:8888
[root@localhost lsb-dist-testkit]#
```

※ 참고 : 스크립트가 웹 브라우저를 찾지 못한다면 브라우저를 수동으로 시작하고 주소를 **http://localhost:8888**로 지정한다(포트번호는 이전 단계에서 설정한 것과 같아야 함)

2. Web-UI 활용 가이드

가. Web-UI 호출

- 1) 웹 브라우저 실행 후 Web-UI 호출
 - `http://<test-machine-name>:8888`



[그림 2-1. Web-UI]

나. LSB DTK Manager의 Get Certificated, Custom Tests

LSB DTK Manager에서는 Get Certified(Certification Mode : 인증 테스트), Custom Tests(Custom Tests Mode : 사용자 설정 테스트)를 수행할 수 있다.

1) Get Certified(인증 테스트) 수행

- 가) 이름, 조직, E-mail(계정이 있을 경우) 입력한다.
 - E-mail을 통해 보고서를 받으려면 체크박스에 체크
- 나) 아키텍처, LSB 버전, LSB 프로파일을 선택한다.
- 다) Run the Tests 버튼을 선택하여 테스트를 시작한다.

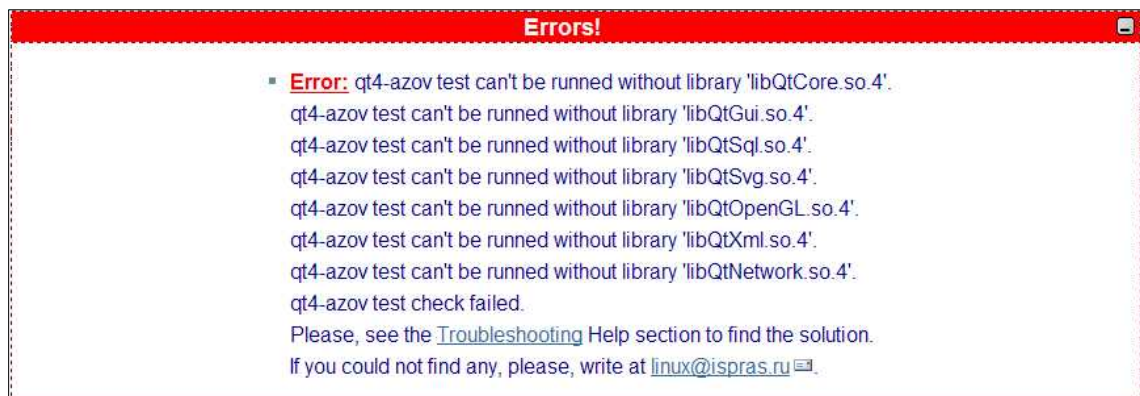
[그림 2-2. LSB인증 테스트 시작 페이지]

※ 참고 : 화면 우측의 User Profile Management를 사용하여 프로파일을 저장해 놓으면 이후 작업 시 프로파일을 활용할 수 있다.

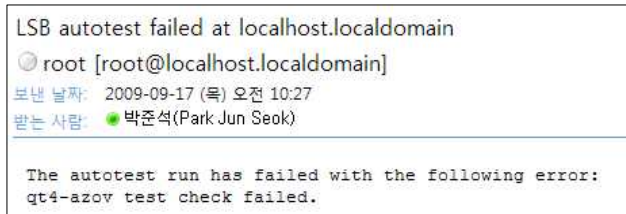
※ 참고 : Test Suite를 실행하기 위해서는 몇 가지 전제조건(<lsb-dist-test dir>/README 파일 참조)이 있다. 만약 전제조건이 충족되지 못한 상황에서 테스트를 수행하려고 한다면, 페이지 상단에 오류 메시지가 출력된다. 이러한 경우 문제를 제거하고 테스트를 다시 실행한다.

○ 예시 : LSB DTK Manager는 QT4 라이브러리를 사용한다. 다음과 같은 오류가 발생할 경우 QT4 패키지를 설치한다.

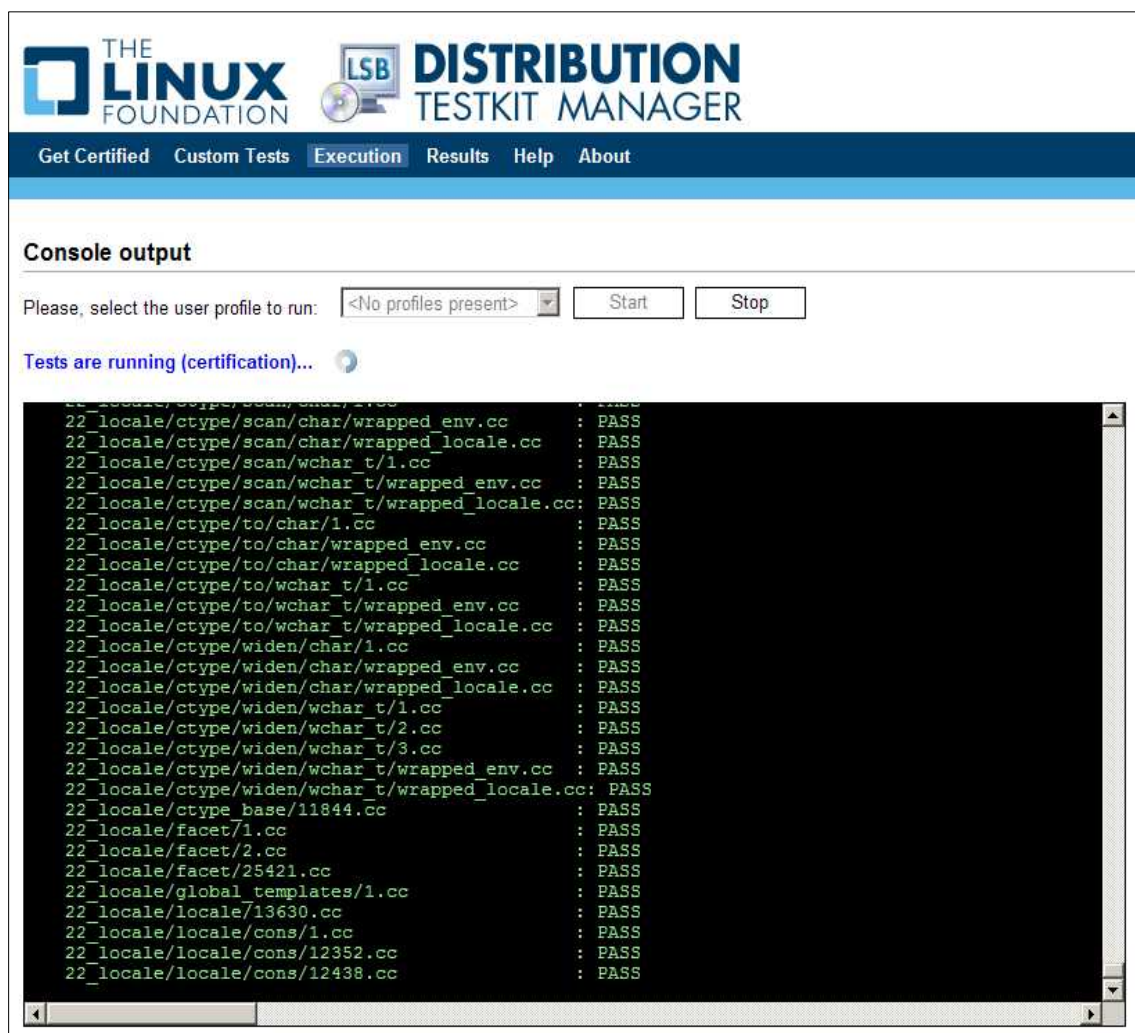
- yum install qt4



※ 참고 : Send report via E-mail 항목을 체크하였다면 이벤트 발생 시 실시간으로 메일이 전송된다.



라) 테스트 수행을 시작하면 Execution 페이지로 이동되며, 진행 상태를 보여주는 터미널 에뮬레이션이 활성화 된다.



[그림 2-3. Execution 페이지]

마) 테스트 수행이 종료되면 Result 페이지로 이동되며, Summary Report가 출력 된다.

THE LINUX FOUNDATION **LSB DISTRIBUTION TESTKIT MANAGER**

Get Certified Custom Tests Execution **Results** Help About

Summary Report for Test Run of 17.09.2009 16:25:53

Test Execution Status

Automatic Tests	FAILED	View detailed report
Manual Tests	NOT FINISHED YET	

The manual tests are not executed yet. You can [run them now](#).

Analyze the Failing Tests

Please analyze the [detailed report](#) to understand the reasons of each fail and classify them into the following groups:

- Confirmed FAILs**, which are due to the real inconsistencies of your system with the LSB standard. To continue with the LSB certification, you have to fix your implementation rerun the tests again.
- False FAILs**, which you believe are due to the incorrect tests. Please report such FAILs to the lf_lsbcert@lists.linux-foundation.org [mail](#) list. If the problem reported is confirmed the test failure will not affect your ability to certify.
- Unknown FAILs**, which you have no idea about. Please ask help at the lf_lsbcert@lists.linux-foundation.org [mail](#) list.

Note: only test results with all the FAILs waived are eligible for certification.

Locating Test Journals

You can find the consolidated test journals of this test run at the following path (click to download to another location):
[/var/opt/lsb/test/manager/results/amd64-localhost.localdomain-2009-09-17-16h-25m-53s/lsb-autotest-amd64-localhost.localdomain-2009-09-17-16h-25m-53s.tgz](#)
 You need to attach them when applying for certification or when communicating with the support staff.

You can easily upload these test results to the [Certification System](#).

[\[Upload the test results\]](#)

Viewing this Page Again

This page has been saved as a part of the test run results. You may view it at any time by clicking particular test result at the Results page.

[그림 2-4. LSB인증 테스트 결과 - Summary Report]

바) View detailed report를 선택하여 자세한 정보를 확인 할 수 있다.

THE LINUX FOUNDATION **LSB DISTRIBUTION TESTKIT MANAGER**

Get Certified Custom Tests Execution **Results** Help About

LSB automatic test results for CentOS release 5.3 (Final) on amd64

The complete console output log is available [here](#).

Test Info

Tests Started At	16:25:53 17-09-2009
Tests Finished At	00:15:36 18-09-2009
Host Name	localhost.localdomain
Host OS	CentOS release 5.3 (Final)
Host Machine	x86_64
Host Platform	amd64
Host Kernel	2.6.18-128.el5 #1 SMP Wed Jan 21 10:41:14 EST 2009

Tests executed

Certification Tests			
Command Tests			
Command Check	v 4.0.0-2	Success	Passed: 143
Static Interface Tests			
Library Check	v 4.0.0-2	Warnings: 27	Passed: 49664
Runtime Interface Tests			
Core Tests	v 4.0.3-1	Failures: 1	Passed: 5823
Cpp-T2C Tests	v 4.0.0-3	Success	Passed: 1763

[그림 2-5. LSB인증 테스트 결과 - Detailed Report]

사) 테스트 결과가 실패일 경우 Manual Tests 페이지로 이동하여 수동 테스트를 수행할 수 있다.

Manual Application Battery Tests

Please, perform the following Application Battery tests to finish with the certification:

lsb-alsaplayer	Status: Not finished
lsb-celestia	Status: Not finished
lsb-designer	Status: Not finished
lsb-ghostscript	Status: Not finished
lsb-leafpad	Status: Not finished
lsb-pidgin	Status: Not finished
lsb-scribus	Status: Not finished
lsb-transmission	Status: Not finished
lsb-xpaint	Status: Not finished
lsb-xpdf	Status: Not finished
lsb-xscreensaver	Status: Not finished

You need to finish all the tests to continue.

Functional Verification Test: lsb-alsaplayer

Expected result achieved?	Test case scenario
1. ☐ Yes ☐ No	Download lsb-alsaplayer and install the lsb-alsaplayer package for the LSB version chosen. Do not use <code>--no-deps</code> option, the dependency checks are part of the test. Application should be installed to <code>/opt/lsb/appbat</code> . If the <code>xdg-desktop-menu</code> binary is present in the system, and you are using a compliant window manager, lsb-alsaplayer should show up in the AudioVideo and/or Audio or Play categories, if present. This functional verification requires a working sound setup on the test machine, using one of the ALSA kernel drivers. - Was the installation successful?
2. ☐ Yes ☐ No	Run the command: <code>/opt/lsb/appbat/bin/alsaplayer</code> - Expected result: The alsaplayer application is launched with the "Alsaplayer" window. Additional windows may be present were left open in a previous run.
3. ☐ Yes ☐ No	Right click in the status area (where volume, etc appear) and select "Scopes" from the popup menu. - Expected result: The "Scopes" window appears.
4. ☐ Yes ☐ No	Double-click each of the seven "Scopes" entries to open the selected visualization plugin in it's own window. - Expected result: The seven visualization windows open.
5. ☐ Yes ☐ No	Download the sample wav file here: torvalds-says-linux.wav Click the "Play" icon and select the downloaded torvalds-says-linux.wav. Click the "Play" icon a second time. - Expected result: The audio file plays "Hello this is Linus Torvalds and I pronounce Linux as Linux". The visualization scopes should also show activity as the file plays.

Results

1. ☐ Yes ☐ No Were there any deviations in the install, setup, or test described above?
If yes, please describe:

[그림 2-6. Manual Tests 페이지]

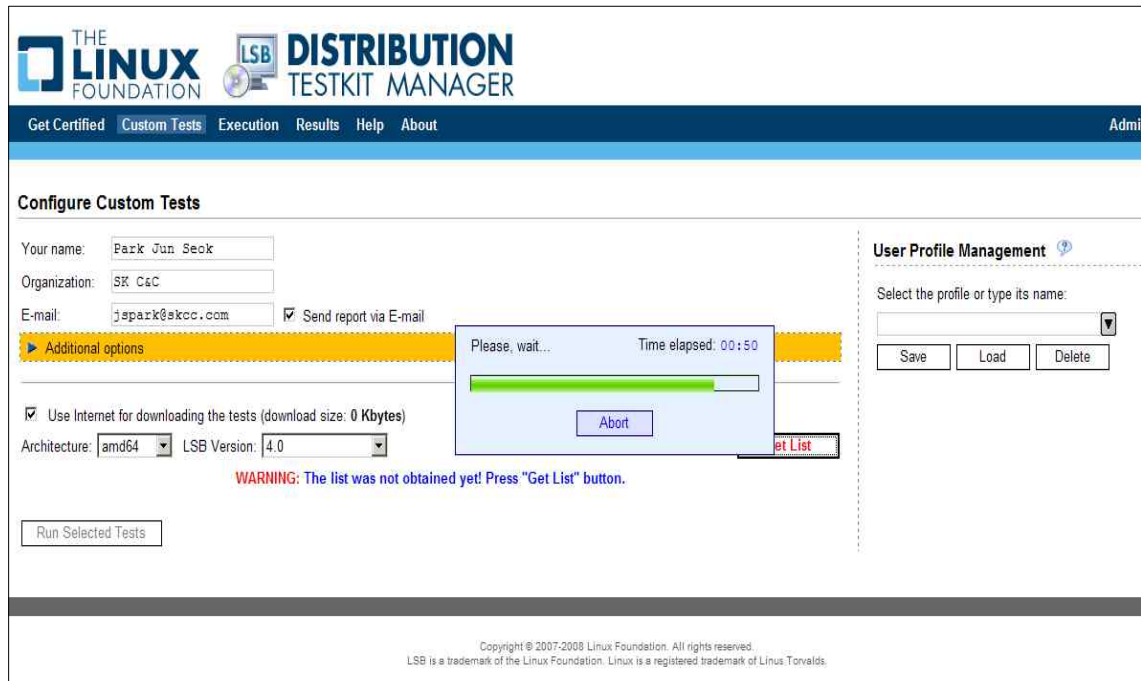
- Manual Tests에서는 자동으로 할 수 없는 테스트를 수행할 수 있다.
- 현재 상태는 좌측 테이블에 Passed, Failed 또는 Not finished로 표시된다.
- Finish 버튼을 클릭하면 업데이트된 Summary Report 페이지로 이동된다.

※ **참고** : 한 번에 모든 Manual Tests를 수행 할 필요는 없다. 일부의 테스트 수행 후 추후 다시 테스트를 이어서 수행할 수 있다. Manual Tests의 이어서 수행하기 링크는 Results 페이지에서 찾을 수 있다.

※ **참고** : Manual Tests 페이지에서는 선택된 테스트와 테스트 상태에 따라 몇몇 추가 정보(실패한 테스트, 인증신청 방법, 저널 파일들의 위치, 작업종료 후 이 페이지로 돌아오는 방법 등)들을 확인할 수 있다. 또한 해당 링크를 클릭하여 자세한 보고서를 열람할 수 있다. 인증 테스트 수행을 위하여 이 페이지에서는 배포판 인증 테스트 결과들을 사용할 수 있도록 Certification Management System(http://www.linuxfoundation.org/lsb-cert/welcome_ert.php)으로 이동할 수 있는 Apply for certification 링크가 제공된다.

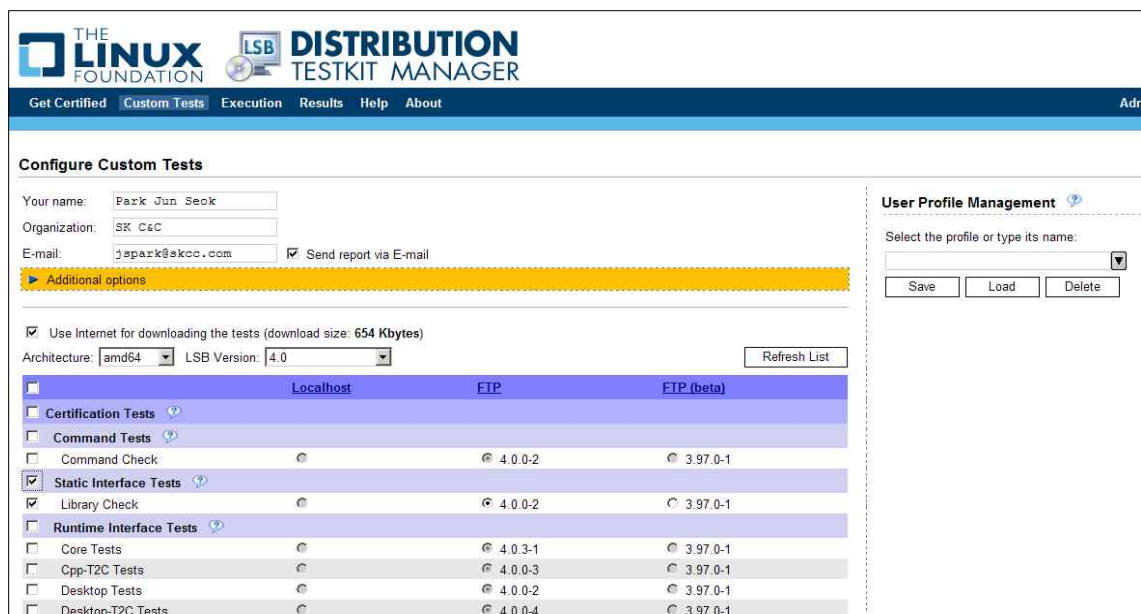
2) Custom Tests(사용자 설정 테스트) 수행

- 가) Get List를 클릭하여 Linux Foundation의 FTP 서버로부터 리스트를 업데이트 한다. Get List 메뉴는 최초 수행 후 Refresh List 메뉴로 변경되며, 이후에는 Refresh List 버튼을 클릭하여 이 작업을 수행 할 수 있다.



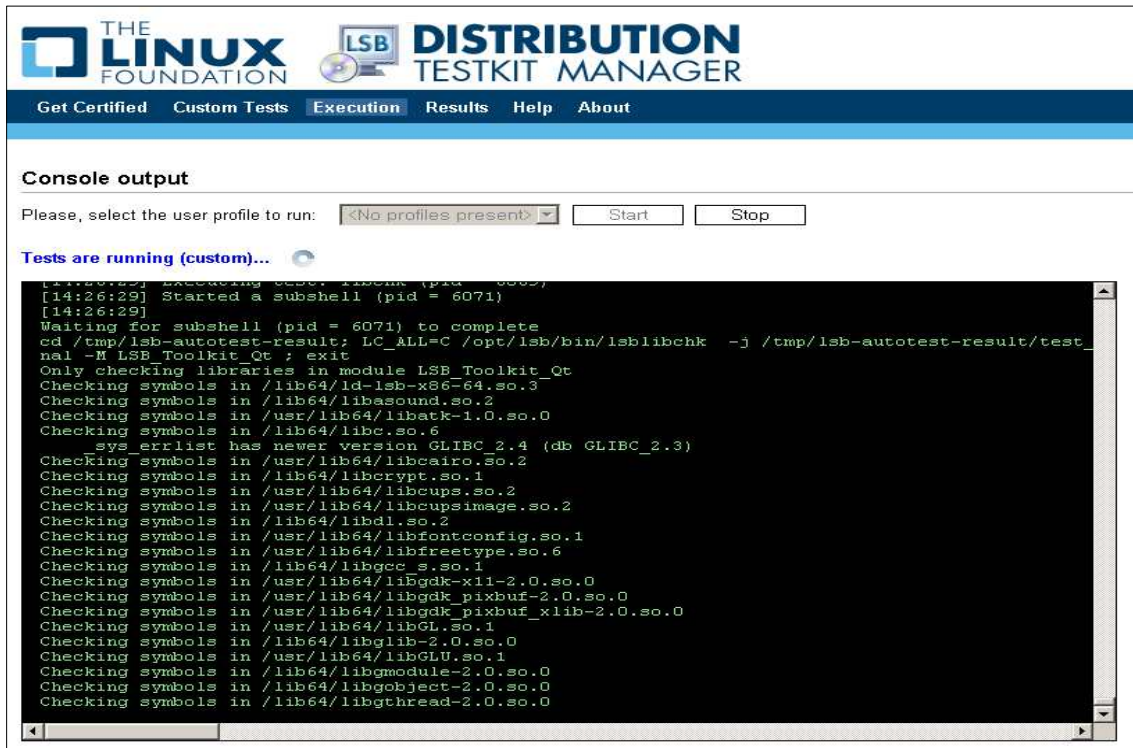
[그림 2-7. 리스트 업데이트 페이지]

- 나) 입력창에 필요한 정보를 입력하고 원하는 테스트 항목의 좌측 체크박스를 표시한 후 우측 리스트에서 버전을 선택한다.



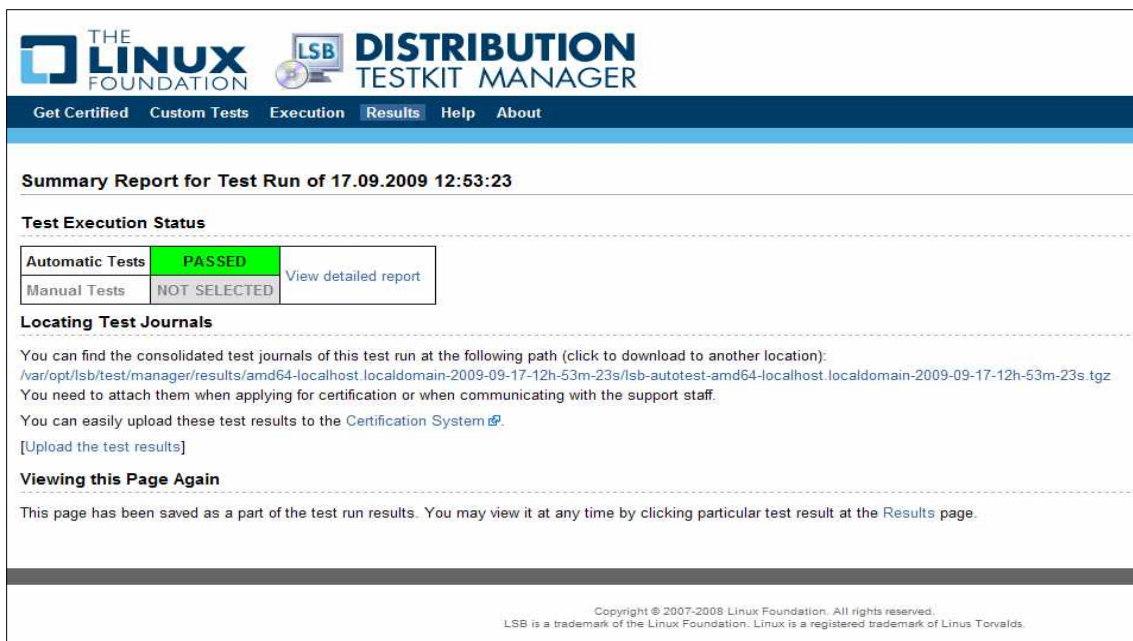
[그림 2-8. 테스트 항목 선택 페이지]

- 다) 페이지하단의 Run Selected Tests 버튼을 클릭하여 테스트를 시작한다.
 라) 테스트 수행을 시작하면 Execution 페이지로 이동되며, 진행 상태를 보여주는 터미널 에뮬레이션이 활성화된다.



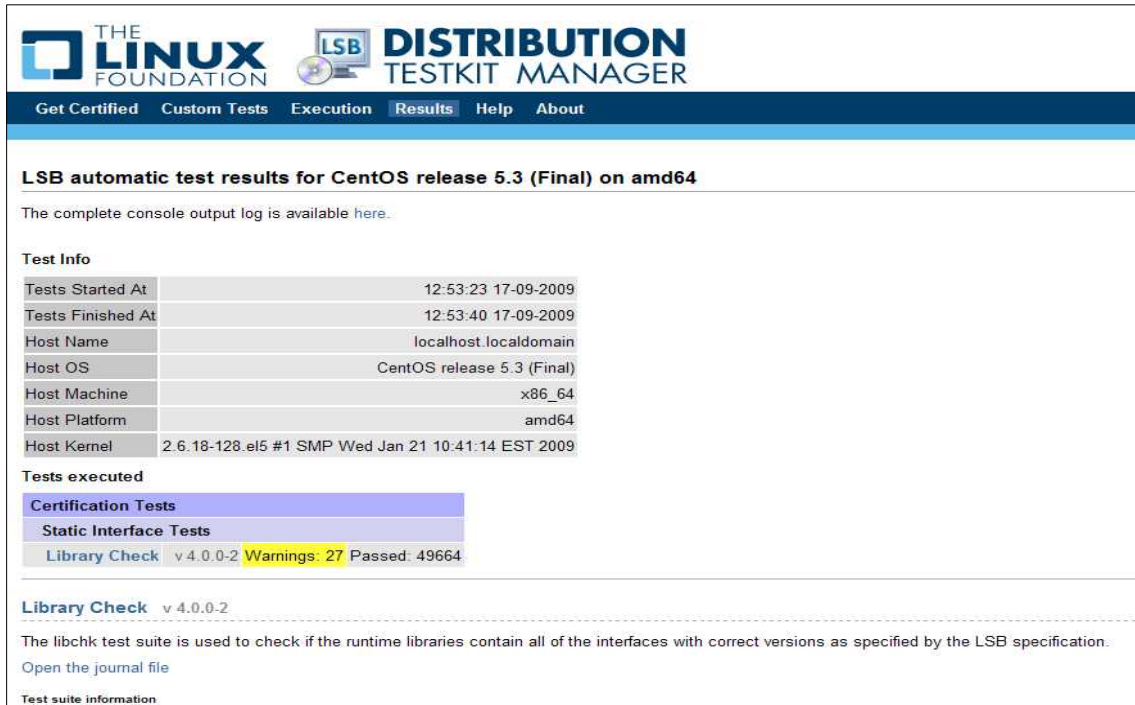
[그림 2-9. Execution 페이지]

- 마) 테스트 수행이 끝나면 Results 페이지로 이동되며, Summary Report가 출력된다.



[그림 2-10. LSB인증 테스트 결과 - Summary Report]

바) View detailed report를 선택하여 자세한 정보를 확인 할 수 있다.



THE LINUX FOUNDATION **LSB DISTRIBUTION TESTKIT MANAGER**

Get Certified Custom Tests Execution **Results** Help About

LSB automatic test results for CentOS release 5.3 (Final) on amd64

The complete console output log is available [here](#).

Test Info

Tests Started At	12:53:23 17-09-2009
Tests Finished At	12:53:40 17-09-2009
Host Name	localhost.localdomain
Host OS	CentOS release 5.3 (Final)
Host Machine	x86_64
Host Platform	amd64
Host Kernel	2.6.18-128.el5 #1 SMP Wed Jan 21 10:41:14 EST 2009

Tests executed

- Certification Tests
- Static Interface Tests
- Library Check v 4.0.0-2 Warnings: 27 Passed: 49664

Library Check v 4.0.0-2

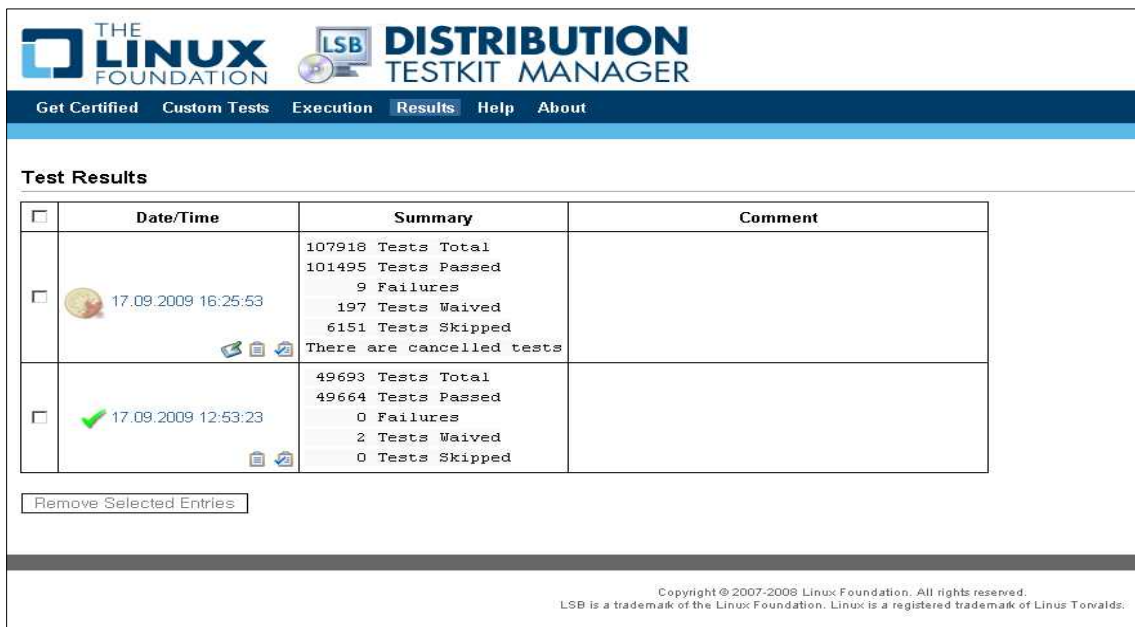
The libchk test suite is used to check if the runtime libraries contain all of the interfaces with correct versions as specified by the LSB specification.

[Open the journal file](#)

Test suite information

[그림 2-11. LSB인증 테스트 결과 - Detailed Report]

※ 참고 : 일반적으로 모든 보고서의 목록은 Results 페이지에서 확인할 수 있다. 특정 보고서를 보기 위해서는 Date/Time 칼럼에서 해당 링크를 클릭하면 된다. 또한, Date/Time 칼럼의 우측 하단 모서리에 있는 아이콘을 클릭하면 Manual Tests, Summary Report, Detailed Report로 바로 갈 수 있다.



THE LINUX FOUNDATION **LSB DISTRIBUTION TESTKIT MANAGER**

Get Certified Custom Tests Execution **Results** Help About

Test Results

☐	Date/Time	Summary	Comment
☐	17.09.2009 16:25:53	107918 Tests Total 101495 Tests Passed 9 Failures 197 Tests Waived 6151 Tests Skipped There are cancelled tests	
☐	17.09.2009 12:53:23	49693 Tests Total 49664 Tests Passed 0 Failures 2 Tests Waived 0 Tests Skipped	

[Remove Selected Entries](#)

Copyright © 2007-2008 Linux Foundation. All rights reserved.
LSB is a trademark of the Linux Foundation. Linux is a registered trademark of Linus Torvalds.

[그림 2-12. LSB인증 테스트 결과 페이지]

3. Command Line UI 활용 가이드

가. Command Line 인터페이스

- 1) 웹 기반 인터페이스 대신 lsb-autotest를 확장한 LSB DTK에서 제공하는 Command Line 기반 인터페이스를 사용할 수 있다.

- /opt/lsb/test/manager/autotest-ext 디렉터리로 이동한다.
- 인증 테스트 수행을 위해 root권한으로 명령어를 실행한다.

./lsb-tef.pl --cert

- 전체 콘솔 로그를 저장하기 위해 'tee' 유틸리티를 사용할 수 있다.

./lsb-tef.pl --cert | tee /tmp/tests_output.8888

- 만약 /tmp/tests_output.8888 파일이 저장을 위해 사용된다면, 이 로그 파일은 테스트가 끝난 후 자동으로 결과 디렉터리에 복사 된다.

- 모든 테스트를 수행하기 위해서는 다음의 명령을 수행한다.(root 권한)

./lsb-tef.pl all

- 사용자 설정 테스트 수행을 위해서는 다음의 명령을 수행한다. test1, test2, etc. 는 사용자가 수행하기를 원하는 특별한 Test Suite이다.

./lsb-tef.pl test1 test2 ...

- 지원되는 Test Suite의 리스트를 얻기 위해서는 다음 명령을 수행한다.

./lsb-tef.pl --help

- 스케줄링을 통한 심야 자동 테스트 수행을 위해서 커맨드라인 UI를 사용할 수 있다. 이 주제에 대한 자세한 소개는 다음의 URL을 참조한다.

<http://ldn.linuxfoundation.org/distro/lsb-dtk-manager-nightly-run-howto>

- 모든 옵션들은 다음의 URL에 자세하게 설명 되어 있다.

<http://ldn.linuxfoundation.org/distro/lsb-dtk-manager-command-line-options>

※ 참고 : 다음은 유용하게 사용되어지는 옵션들이다.

-D : 필요한 테스트 패키지들을 자동으로 다운로드

- lsb=<LSB-version> : 테스트에 대하여 LSB 버전의 명시적인 지정을 허용
- T<profile> : 테스트에 대하여 LSB 프로파일 지정을 허용. 가능한 값들은
core,c++ 과 core,c++,desktop (default)
- r <results-dir> : 결과 디렉터리 지정
- v[0-2] : 자세한 정보 열거
- help : 전체 옵션 리스트 확인

4. 테스트 결과물 저장소(Test Results Repository)

- 저장소 위치 : /var/opt/lsb/test/manager/results
- 파일 형식 : <arch>-<machine>-<date>-<time>
- ※ 참고 : Web-UI의 Results 페이지에 표시되는 결과는 /var/opt/lsb/test/manager/results/HISTORY 파일 안에 기록되며 HISTORY 파일은 자동으로 관리된다. 만약 다른 곳으로부터 테스트 결과를 복사한다면, 수동으로 이 파일 안에 디렉터리 이름을 써 주어야 한다.

5. Distribution Test Kit(DTK)의 정지 및 삭제

가. Web-Server 정지

- Command Line UI : /opt/lsb/test/manager/bin/lsb-dist-testkit-stop.sh
- Web-UI : 브라우저에서 Administration 페이지로 이동 후 Stop Server 버튼을 선택



[그림 5-1. Web-Server 정지 결과]

나. DTK Manager 삭제

- RPM 패키지 삭제 : /opt/lsb/test/manager/autotest-ext/lsb-tef.pl --cleanup all
- /var/opt/lsb/test/manager 디렉터리에서 패키지 캐시, 오래된 결과들과 기타 데이터 등을 삭제한다.